



RS232 Serial to GSM Network Gateway Python Software

The Gateway Python software allows the user to operate and control formerly RS232 serial only capable devices remotely via the GSM (GPRS) network.

It supports 2 primary modes:

- Client Mode – The software acts as a client, remotely connecting to a server at a pre specified IP address and port. It can be either always connected, or connect only when data is received on the serial port.
- Server Mode – The software acts as a server, listening for connections from a remote client on a specified port (additionally a source IP can be specified, and only connections from this IP will be accepted).

In both modes any data received via serial will be forwarded over the GPRS network to the IP / port specified, and any data received from the server (via the GPRS network) will be forwarded to the serial port.

The Gateway software is controlled via the use of SMS Command messages, these are standard text messages send via a mobile phone however they are formatted in a specific way that configure the Gateway software (i.e. change GPRS settings, serial port baud rate etc).

The Gateway Python software is designed to operate on a range of embedded Python supporting hardware modules from Telit. For further details see <http://www.telit.com/en/products.php> It can be used either directly by those designing their own hardware incorporating the Telit modules (GM862-QUAD-PY, GE863-PY, GC864-PY etc) or those who purchase off the shelf pre boxed hardware such as the GT863-PY, GT864-PY, Gatetel EZ10 PY or EZ863 PY Terminals.





SMS Commands

Default password: connect

gprs

Format: `[smspassword] gprs [apn] [username] [password]`

Example: `connect gprs orangeinternet user pass`

Allows settings of the GPRS apn, username, and password for the SIM card. These can be obtained from the SIM card provider.

baud

Format: `[smspassword] baud [baud]`

Example: `connect baud 115200`

Allows setting of the serial port speed used to communicate with the attached serial device. Accepted values are: 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.

charformat

Format: `[smspassword] charformat [charformat]`

Example: `connect charformat 8N1`

| | |
|-------|------------------------|
| 7 / 8 | Data bits |
| N/E/O | No / Even / Odd Parity |
| 0 / 1 | Stop bits |

Allows setting of the serial port data bits, parity, stop bits used to communicate with the attached serial device. Accepted values are: 8N2, 8E1, 8O1, 8N1, 7E1, 7O1.

mode server

Format: `[smspassword] mode server [permitted ip] [localport]`

Example 1: `connect mode server any 1024`

Example 2: `connect mode server 192.168.0.1 1024`





Allows setting of the software into server mode whereby it listens for connections from a remote client on a specified port (additionally a source IP can be specified, and only connections from this IP will be accepted).

mode client

Format: `[smspassword] mode client [server] [server port]`

Example 1: `connect mode client 192.168.0.100 1024`

Example 2: `connect mode client 192.168.0.100 1024 maintain`

Allows setting of the software into client mode where by it remotely connects to a server at a pre specified IP address and port. It can either connect only when data is received on the serial port, or permanently maintain a connection to the specified server reconnecting if it is disconnected.

socksetting

Format: `[smspassword] socksetting [socket timeout] [send timeout] [packet size]`

Example: `connect socksetting 300 10 0`

Allows setting of timeout values, and packet size used.

Socket Timeout – Time after which the socket will be closed if no data is sent or received. Valid values are 0 - 65535 (0 = no timeout, it is specified in seconds)

Send Timeout – Time after which a packet is sent whose size is less than the default. Valid values are 0 - 255 (0 = no timeout, wait for packets to reach max size, it is specified in 100ms units, i.e. a value of "10" is 1 second)

Packet Size – Size of packets used by the TCP stack when sending data. Valid values are 0 - 512 (0 = chosen automatically by device, it is specified in bytes)

The customer is advised to set a socket timeout value which balances their need for fast detection of a network condition causing a TCP connection to improperly close (ie availability of the RS232 Gateway for reconnection) against the increased GPRS traffic overhead incurred as the results of a lower timeout limit meaning the server must send arbitrary data at an increased frequency to keep the connection open.





smspassword

Format: `[smspassword] smspassword [new sms password]`

Example: `connect smspassword connect2`

Allows setting of a new password. The password is required in all SMS command messages sent to the unit. The default password is "connect". Valid values are 1-12 characters of 0-9, a-z only.

dialin

Format: `[smspassword] dialin [status]`

Example: `connect dialin on`

Allows turning on or off support for remote dial in via a CSD (Circuit Switch Data) connection. When this setting is turned on, and the unit is in server mode, inbound connections can be accepted via either GPRS or CSD. By default this setting is turned off. Regardless of the baud rate established by the CSD connection, the baud rate of the serial port will be the value configured by the "baud" SMS command message.

config

Format: `[smspassword] config`

Example: `connect config`

On receipt of this message the software will return the current active configuration settings, when in "server" mode this includes the local IP address.

reset

Format: `[smspassword] reset [reset interval]`

Example: `connect reset 120`

Allows enabling of a periodic unit reset, that will automatically reset internal software. Valid values are 0 – 1440 i.e. 24hr (0 = reset disabled). Note: This value is specified in minutes. This setting is not designed to be used in normal operation, it is designed when other avenues of fault finding the cause of a reliability issue have failed.





ping

Format: `[smspassword] ping [ping interval] [server(s)]`

Example 1: `connect ping 300 google.com`

Example 2: `connect ping 300 google.com bbc.co.uk`

Allows enabling of a periodic ping made from the gateway to a remote server. If specified the server / servers will be checked on the interval (assuming no connection is currently in progress), and if unreachable (due to e.g. GSM / software failure condition preventing access to the Internet) the unit will be rebooted. It will not be rebooted more frequently than once every 15 minutes. Interval specified in seconds. Valid values are 60 - 86400 (i.e. 24 hours). This setting is not designed to be used in normal operation, it is designed when other avenues of fault finding the cause of a reliability issue have failed.





General Notes

- Software will acknowledge receipt / action of correctly formatted command messages with a message in the format "Confirmed: [command confirmed]". Incorrectly formatted messages will be responded to with "Error: [error message]". Replies are sent to the sender of the original request. The only exception is that command messages sent with an invalid password will not be responded to.
- In order to use the server mode whereby the device accepts incoming connections, a special fixed public IP SIM card may be required.
- To accept new commands via SMS the software must not have any active socket connections. Active socket connections can be disconnected via closing them on the remote side or hard rebooting the unit.
- If the SMS password is forgotten unit software must be reloaded from the loader utility.

Ping Reboot

The ping is a standard ICMP echo request.

Multiple servers can be specified on the ping command, they will be tried one at a time in the order specified (having more servers specified means more potential delay e.g. when the unit is carrying out a ping rather than listening for a socket connection). If any server responds to the ping it stops checking further servers and takes the ping as successful (does not trigger a reboot). Only if all servers specified fail to respond to a ping will a reboot happen.

It is important to use both reliable servers, and servers you have control over. If an unreliable server is used this will only result in more downtime (while the unit constantly reboots). If a 3rd party server is used and that 3rd party decides at some point in the future to disable ping replies that will cause units to constantly reboot.

Network Timeouts

"It's a very common issue, when you are behind a NAT proxy or a firewall, to be disconnected without a reason. This behaviour is caused by the connection tracking procedures implemented in proxies and firewalls, which keep track of all connections that pass through them." <http://tldp.org/HOWTO/TCP-Keepalive-HOWTO/overview.html>

If required, it is the servers responsibility to keep alive the TCP connection to the Gateway software. i.e. if in the normal course of usage a situation may arise where no data is sent or received within 5 minutes the server should initiate sending of arbitrary data. If the server does not do this the RS232 Gateway will close the connection and start listening for a new connection.

"Server" = Computer talking to the Gateway

