

# GSM/GPRS Quectel M95

## Table of contents

---

- GSM/GPRS Quectel M95
    - Documentation
    - Experimentally-obtained data
    - AT Commands for checking the module state
    - AT Commands to start using the GPRS
    - Creating a TCP/IP connection
- 

## Documentation

Can be found at Quectel's Website, [here](#) (needs registration).

## Experimentally-obtained data

- The unit worked fine with a 2A, 3.6V power supply, which also powered an [EPOSMote III](#).
- With 3.5V on the same power supply, the unit sends the URC (Unsolicited Result Code) "UNDER\_VOLTAGE WARNING", and turns off sporadically (usually when communicating).
- Between 3.3V and 3.4V the unit warns about the under voltage and turns on very quickly (about 10 seconds).

## AT Commands for checking the module state

The following commands can be used to check if everything is OK after powering on the device. Commands are shown with a `>>` prompt, the following lines represent the data received back from the modem.

```
>> AT
OK
>> AT+CPIN?
+CPIN: READY
OK
```

The command **AT** should always be responded with **OK**. The **AT+CPIN?** command checks if the SIM card needs unlocking, and will return **+CME ERROR: 10** in case the SIM card is not present. Refer to the AT Commands manual for other cases.

## AT Commands to start using the GPRS

The following commands are needed to attach to the GPRS network and transmit/receive data.

```
>> AT+CGDCONT=1,"IP","gprs.oi.com.br"
```

```
OK
```

```
>> AT+CGACT=1,1
```

```
OK
```

```
>> AT+CGATT=1
```

```
OK
```

This will:

1. Define a PDP (Packet Data Protocol) context for the connection, using IP, where the second string (in our case "gprs.oi.com.br" is the APN (Access Point Name) for your provider. You may need to search the internet for your case.
2. Activate the context we just created.
3. Attach to the GPRS network.

Refer to the AT Commands manual for more detail if needed.

## Creating a TCP/IP connection

You can easily open a TCP/IP connection with the (Quectel-specific) **AT+QIOPEN** command as follows.

```
>> AT+QIDNSIP=1
```

```
OK
```

```
>> AT+QIOPEN="TCP","tarcis.io","80"
```

```
OK
```

```
CONNECT OK
```

The first command, **AT+QIDNSIP** is used to signal the module whether you will use an IP address or a qualified domain name in the **AT+QIOPEN** command. If you are using an IP address instead of a domain name, omit it or explicitly send **AT+QIDNSIP=0**.

Now to send data over the open connection, use the **AT+QISEND** command.

```
>> AT+QIOPEN="TCP","tarcis.io","80"
```

```
OK
```

```
CONNECT OK
```

```
>> AT+QISEND
>
>> test
>> 0x1a (send as hex input, this is a CTRL+Z character)

SEND OK

HTTP/1.1 400 Bad Request
Server: nginx/1.6.2
Date: Fri, 26
  Jun 2015 20:16:54 GMT
Content-Type: text/html
Content-Length: 172
Connection: close

<html>
<head><title>400 Bad Request</title></head>
<body bgcolor="white">
<center><h1>400 Bad Request</h1></center>
<hr><center>nginx/1.6.2</center>
</body>
</html>

CLOSED
```

In this example, a test string was sent, and a response was received from the nginx server, which, obviously, didn't understand the request.