

## Application Note: Pass Through Data Mode

### Scope

AT220A, AT220B, AT240, AT210, AT110, AT200

### Overview

Our devices can be used to send and receive data between an RS232 serial port device (attached to the device serial port) and a remote TCP/IP application. This technique is commonly referred to as pass through data mode and is described in this application note.

### Related Documents

The following documents are recommended reading to accompany this document:

- AT220, AT240, AT210, AT110 & AT200 User Guides

These documents can be obtained from:

<https://gps-telematics.co.uk/downloads/>

### Operation of Pass Through Data Mode

Pass through data mode allows data to be sent and received from a second TCP socket and port number, as defined by parameters IPAD2 and PORT2 or using the reporting TCP socket defined by IPAD1/PORT1. Pass through data mode is enabled by setting PTDM <mode> to a value in the range 1 to 3 (default is zero).

In PTDM mode, the device will attempt to open a TCP socket and will listen for data on that socket. Any data received will be sent to the relevant device serial port (see below for device related aspects). Similarly, any data received on the serial port will be assembled into packets and sent to the TCP socket. Packets will be sent according to the PTDM parameters described in the table below. Note there is no packet queue for PTDM data.

### Device RS232 Port Aspects

The AT220, AT210 and AT200 have only one RS232 port. Enabling PTDM will suppress all other data on the serial port and further serial mode commands will not be accepted. Serial commands are accepted for 30 seconds after power up to allow PTDM mode to be disabled.

Other devices have 2 serial ports, so the second port is dedicated to pass through data mode (when enabled by PTDM <mode>).

## Configuration of Pass Through Data Mode

\$PTDM,<mode>,<baud\_rate>,<packet\_timeout\_ms>,<packet\_max\_size>,<packet\_terminator>,<device\_login>,<rs232\_port>,<tcp\_socket>,<add\_terminator>,<packet\_headers>,<ack\_enable>,<auto\_off\_time>

Field	Description	Range	
<mode>	PTDM mode	0 1 2 3	Disabled Enabled Garmin FMI Enabled with connect message
<baud_rate>	RS232 baud rate	4800 9600 19200 38400 57600 115200 230400 460800 921600	
<packet_timeout_ms>	Packet assembler timeout	0 – 65535 ms	
<packet_max_size>	Packet assembler maximum packet size	1 – 1024	including header & terminator
<packet_terminator>	Packet assembler termination character options	0 1 2 3	Send according to <packet_timeout_ms> or <packet_max_size> criteria Send when <CR> received Send when <LF> received Send when <CR><LF> received
<device_login>	Enable a device login (IMEI) when a TCP socket is opened	0 1	no login login enabled
<rs232_port>	Select RS232 Port	0 1	RS232 Port 1 RS232 Port 2
<tcp_socket>	Select TCP socket to use for PTDM	0 1	Use primary socket defined by IPAD1/PORT1 Use secondary socket defined by IPAD2/PORT2
<add_terminator>	specify additional termination characters to add to packets in both directions	0 1 2 3	Do not add any characters Add <CR> Add <LF> Add <CR><LF>
<packet_headers>	Enable packet headers	0 1	no packet headers packet headers enabled
<ack_enable>	Option to ACK packets sent in both directions	0 1	Acknowledgment disabled Acknowledgment character is sent in response to received packets and must be received after packets are sent
<auto_off_time>	Optional pass through data mode disable timer	0 1-65535	PTDM mode runs continuously Defines a limited time (minutes) to run PTDM mode, after which PTDM will automatically disable (<mode> will be set to 0)

Notes:

`<mode>`

This is provided for legacy purposes to support previous fixed PTDM modes. If PTDM options are being selected manually, `<mode>` should be set to 1.

`<baud_rate>`

Select any standard baud rate for use with the RS232 port selected for PTDM

`<packet_timeout_ms>`

Packet assembler timeout. If no bytes/characters are received on the port (in either direction) for `<packet_timeout_ms>`, the device will assemble the data into a packet and send it. See other packet assembler options below, which can work in conjunction with `<packet_timeout_ms>`

`<packet_max_size>`

Packet assembler maximum bytes. If `<packet_max_size>` bytes/characters are received on the port (in either direction), the device will assemble the data into a packet and send it. Note that the maximum packet size supported is 1024 bytes, including packet header and terminator.

`<packet_terminator>`

Packet assembly can be triggered by a `<CR>`, `<LF>` or `<CR><LF>` using this option. Packet terminators will be included in the data and sent to the host. Default setting is zero, no packet terminators.

`<packet_headers>`

Headers can be enabled on packets in both directions. When enabled, the following headers will be added:

`#PTDA:<data>`

Default setting is zero, packet headers disabled

`<device_login>`

Enables a device LOGIN packet each time a new TCP socket is opened. The format of the LOGIN packet is as below:

`<DEVICE>,CONNECT,<IMEI>,<FW_VER><CR><LF>`

Where:

<code>&lt;DEVICE&gt;</code>	is the device type / model, e.g. AT240V8
<code>&lt;IMEI&gt;</code>	is the 15 digit unique device serial number
<code>&lt;FW_VER&gt;</code>	is the device firmware version
<code>&lt;CR&gt;</code>	is a carriage return character
<code>&lt;LF&gt;</code>	is a line feed character

e.g. `AT240,CONNECT,352218030578118,4.0.17.0`

Default setting is 1, LOGIN enabled.

## <rs232\_port>

0 RS232 port 1

1 RS232 port 2

We recommend using port 2, leaving port 1 free for debug, diagnostics and configuration.

## <tcp\_socket>

Select the TCP socket to use for PTDM, either the primary socket defined by IPAD1/PORT1 that is also used for routine device reporting or IPAD2/PORT2 which is dedicated to PTDM use.

When <tcp\_socket> is set to 0 and the device MODE is set to 6 (TCP with login) then pass through data will only be sent on the primary socket after the device has received acknowledgment of the login packet.

The default setting is 1 for IPAD2/PORT2.

## <add\_terminator>

Enables the addition of a packet termination characters <CR>, <LF> or <CR><LF> on packets in both directions. Default setting is zero for no packet termination characters.

## <ack\_enable>

Enables packet acknowledgment in both directions using a single byte 0x05 (note: this is byte with numeric value 5, not ascii '5'). When enabled, an ACK must be sent by the host server in response to all packets received from the device. The device will respond to all packets received from the host server with an ACK. Default setting is zero, acking disabled.

## <auto\_off\_time>

PTDM mode can be enabled for a pre-specified time using this option. Default is zero, PTDM mode runs indefinitely.

## Garmin FMI Device Support

PTDM mode 2 is specifically designed for use with Garmin FMI mode. In this mode, the device will indicate to the host when the FMI appears to be online/offline based on a 90 second communication timeout. When a timeout occurs, the device will send an FMI Enable packet to the FMI. If an FMI ACK is received in return, the device will send an FMI online message to the host.

FMI offline is indicated when the device sends the following:

```
<DEVICE>,FMI_OFFLINE<\r>
```

And when the FMI comes back online:

```
<DEVICE>,FMI_ONLINE<\r>
```

Device identification (as described above) is also enabled in PTDM mode 2.

The baud rate will change to 9600 on entering PTDM mode 2.

The default FMI enable command does not enable/disable specific features. The following command can be used to do that.

```
$FMIC,<hex_encoded_command_bytes>
```

where <hex\_encoded\_command\_bytes> is a maximum of 64 bytes. This must be the complete Enable Fleet Management Protocol message from the DLE to ETX characters. This allows a maximum of 11 features to be specified in the command. Once set, the enable message will be stored in the device and used each time the device initiates communication in Garmin FMI mode.

## Timing

Data can be delayed somewhat due to normal reporting activity on the primary IPAD/PORT, but typically there is no more than a second or two delay between submitting the data to the serial port and transmission to the host (and vice versa).

## TCP Socket Status

If the TCP socket is inactive for several minutes, it is likely to be closed by the GPRS network operator (subject to individual network inactivity timeouts). The device will periodically check the TCP socket status and attempt to maintain an open connection. Application developers are advised to consider socket keep-alive solutions or to seek suitably long inactivity timeouts from your GPRS network service provider.

## **Access to GPS NMEA Data**

GPS NMEA output is suspended during PTDM mode operation for devices that have only one serial port. In some cases, it may be useful for the serial device to access GPS NMEA data from time to time. This can be done by sending \$GPRMC,REQUEST to the device, which will result in a single \$GPRMC sentence being sent to the PTDM serial port. The \$GPRMC,REQUEST data string will not be sent to the TCP host.

For devices with 2 serial ports the GPS NMEA output is sent to the first serial port as usual.