

# ***Getting Started***

***TPS Version 5.0 Network***

Getting Started

Issue: 1.01 16/03/97

TPS Version 5.0 Network

Copyright Micro SciTech Ltd. 1991-1998

Information in this document is subject to change without notice and does not represent a commitment on the part of Micro SciTech Ltd. The software described in this document is furnished under a licence agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the licence or nondisclosure agreement. No part of this manual may be reproduced or transmitted in any form or by means electronic or mechanical, including photocopying and recording, for any purpose without the express permission of Micro SciTech.

Copyright Micro SciTech Ltd., 1991-1998. All rights reserved.

TPS and TPS M4 is a trademark of Micro SciTech Ltd.

Windows, MS-DOS, Windows NT, Windows for Workgroups and Windows 95 are registered trademarks of Microsoft Corporation.

Pentium is a registered trademark of Intel Corporation.

This page is intentionally left blank.

1 About TPS .....	4
2 Getting Started.....	5
2.1 Starting TPS.....	5
2.2 Exit of TPS .....	7
2.3 TPS Configuration .....	7
2.4 Test Data Checkout.....	8
2.5 Transmitting Data using an Ethernet Network .....	10
2.6 Receiving Data using an Ethernet Network.....	11
2.7 Transmitting Data to a Serial Port .....	13
2.8 Receiving Data from a Serial Port.....	15
2.9 Transmit and Receive Summary.....	17
2.10 Configuring the Packet Format .....	18
2.11 Configuring your own Displays.....	21
2.12 Plotting Data .....	31
2.13 Recording and Replaying Packet Data .....	36
2.14 Recording Parameters.....	38
2.15 Printing the Display Window .....	40
2.16 Setting Alarms.....	41
2.17 Selective Packet Replay.....	44
2.18 Where to go next.....	45

## 1 About TPS

TPS is a real-time or offline packet telemetry software product for the Microsoft Windows operating system (all platforms).

It processes data packets either incoming to the PC in real-time or read from a disk file. TPS can read and process packet in many different formats, the only information it requires is the synchronisation bit pattern and the total packet length. Incoming packets must have a fixed synchronisation byte pattern of 1-4 bytes and a maximum fixed (1) length of 4224 bytes. This information is configured online and can be reconfigured to handle a different packet format at any time during use.

(1) Variable length packet processing is due in V5.02 to follow shortly.

TPS can run standalone or as part of a multi-user network, receiving data globally across the Internet using the ethernet TCPIP protocol or via a local area network using TCPIP or UDP. A serial RS232 network is also available. Networks can accommodate an unlimited number of users all monitoring the same or different data.

TPS can record data packets to disk and simultaneously retransmit them to another PC, all in real-time. It can also extract data from the packets for real-time numeric or graphical display, optionally recording parameters in the packet and also allowing alarm limits to be set. All parameters are displayed in 'view-ports', also completely configurable online. TPS can display numeric parameters embedded within the packet from 1 to 8 bytes in length as well as hexadecimal byte dumps and text strings. All parameters can be converted to engineering units using a variety of functions including polynomial and bitmask. A single display screen, termed a display 'page' in TPS jargon, houses up to 1024 parameters and TPS can display up to 10,000 different pages.

In addition to the display of data held within the packets, TPS also provides many statistical displays containing such statistics as the incoming and outgoing packet rate and the total number of bytes read. All these parameters are displayed on the default display page 10 to which TPS loads upon starting.

TPS also provides a basic telecommand system enabling user defined byte strings to be sent to the outside world for such purposes as remote commanding of external devices. The system has been developed for user expansion to allow for greater flexibility.

A more detailed summary of all the TPS features is given in the online help.

This Getting Started manual is intended for both first-time and existing TPS users and, together with the Release Notes, should be read before any other manuals.

## **2 Getting Started.**

Please see the Installation and Release Notes for details on how to install TPS on your computer.

### **2.1 Starting TPS**

Once TPS is installed, double click on the `TPS for Windows` icon to start the program. You should then see the standard display shown in Fig 2.1. Note, only the first 35 lines and 80 character columns are shown in the picture - the full screen is much larger (80 rows, 160 columns), use the scroll bars to see the whole display. When you first start TPS, it will try and size itself to the maximum size display possible.

```

LASTPKTR
LASTPKTW                      Operational_Data      Version   W32V5.01h
LASTBYTR                      Page                10      Loop-meas   12
LASTBYTW                      Loop-time     10

```

INNAME	NO INPUT SOURCE	INFMT	NONE	INTYPE	NONE
OUTNAME	NO OUTPUT SOURCE	OUTFMT	NONE	OUTTYPE	NONE
STATSTRD	0	PKTSYNC	00001234	> NO DATA <	Pkt-Selct'd
STATSTWR	0	PKTLENGTH	126		ON
PKTSRD	0	Hexadecimal Packet Dump			
PKTSWR	0	0	1	2	3
BYTESRD	0	0	00	00	00
BYTESWR	0	16	00	00	00
LOSTSYNC	0	32	00	00	00
		48	00	00	00
BITRTRI	0	64	00	00	00
BITRTRA	0.0	80	00	00	00
BITRTWI	0	96	00	00	00
BITRTWA	0.0	112	00	00	00

See display for plots

```

0.0 Hz  27.2 s    0.0 Hz  27.2 s    0.0 Hz  58 s    0.0 Hz  58 s
> .....
Page    Operational Data    0010 NO ERRORS. LAST ERROR AT Step 0..3..6..9..c..f
10      Thu Feb 12 13:26:53    OFF ↑
0000.00:00:58    TPS for Windows    (C) MST 1991-1998    Thu Feb 12 13:27:50 1998

```

**Fig 2.1 TPS Display upon starting. (Top left 80 cols and 35 rows only).**



TPS has a 'system' display (shown below) which appears on every page. The system display is the dark blue band approximately in the middle, left half of page 10. This displays, amongst other things, the elapsed time and current PC date/time. See the bottom left of the blue band for the elapsed time and bottom right for the PC time. Check these times increment every second. The top line of the system display, that starting with a '>' character on the far left, is the Message Bar. This displays, in red, diagnostic text message useful when editing view-ports and/or monitoring data.

```
> .....
Page   Operational Data      0010 NO ERRORS. LAST ERROR AT Step 0..3..6..9..c..f
  10                               Thu Feb 12 13:26:53      OFF ↑
0000.00:00:58    TPS for Windows      (C) MST 1991-1998   Thu Feb 12 13:27:50 1998
```

## 2.2 Exit of TPS

To exit from TPS, select the Exit option from the File menu as is standard for every Windows application. You will be prompted with the Exit dialog box before TPS finally exits.

You can also press ALT Q instead of using the menus to start an exit. This will pop-up the same Exit dialog box.

You are advised to exit and then re-enter just as a simple check on TPS before continuing.

## 2.3 TPS Configuration

The two most common questions when starting with TPS are:

- 1     How do I configure TPS to read my packet data ?
- 2     How do I configure TPS to display parameters embedded in my packet data ?

This guide starts with a general overview before answering the above questions. If you are already familiar with the basics of TPS then you can proceed straight to the relevant sections:

- 1     turn to section 2.10
- 2     turn to section 2.11

## 2.4 Test Data Checkout

TPS will not be configured to receive any data upon first starting. This is reported by the red NO DATA text blinking about a third of the way down the screen in the middle.

To simulate some incoming data, switch on the test data by selecting the *Transfer, Test data* menu item. You should then see several parameters updating and the red NO DATA text will change to DATA, blinking slowly in green. The display should look something like Fig 2.4.

TPS provides its own internal test data which effectively simulates incoming data from the outside world (or a previously stored disk file).

In the upper left of the screen there is a column of grey-background parameters showing key data statistics. The column starts with the parameter 'STATSTRD' which is the elapsed time in seconds since TPS first received some data. With the test data on, you will see it increment the seconds.

In the same column, is the parameter PKTSRD (packets read). This will increment every time a complete packet is received or generated by the test data. Verify PKTSRD increments and that the first plot at the foot of the page, namely the instantaneous packet receive rate (RDPKTRTI - an abbreviation of 'ReaD PacKeT RaTe - Instantaneous'), plots a value of about 7.0 to 8.0 Hz - it will take a few seconds to settle after switching on the test data.

There are many different parameters on the display, the full list is described in the online Help topic 'What you see when starting TPS'. TPS can have many different displays showing many different parameters, each display is termed a 'page' and assigned a number. TPS automatically displays page 10 upon starting. You can switch between pages by pressing PGUP or PGDN or selecting the *Open* item from the *Display* menu.

Display page 10 shows general TPS operational parameters, mainly statistics on the incoming and outgoing data. Most user pages you configure will usually contain many more packet specific parameters. A key feature on page 10 is the hexadecimal packet dump comprising 8 rows of 16 bytes - this is seen in the middle left in blue. This shows the first 128 bytes of the incoming packet data in hex. The first two bytes are the default packet synchronisation bytes 0x12 and 0x34 (the 0x prefix denotes that the digits following are in hexadecimal). The remaining 126 bytes (the default packet size is 128 bytes) are filled with the index 0 to 125 which makes test data easy to identify and useful for checkout.

```

LASTPKTR Thu Feb 12 13:29:52 1998
LASTPKTW Operational_Data Version W32V5.01h
LASTBYTR Thu Feb 12 13:29:52 1998 Page 10 Loop-meas 10
LASTBYTW Loop-time 10

```

```

INNAME          TESTDATA    INFMT    RAW    INTYPE    TEST
OUTNAME         NO OUTPUT SOURCE  OUTFMT  NONE    OUTTYPE  NONE

STATSTRD        23  PKTSYNC    00001234  >    DATA <  Pkt-Selct'd  ON
STATSTWR        0  PKTLENGTH  126
PKTSRD          136  Hexadecimal Packet Dump
PKTSWR          0      0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
BYTESRD        17136  0 12 34 00 01 02 03 04 05 : 06 07 08 09 0a 0b 0c 0d 15
BYTESWR         0   16 0e 0f 10 11 12 13 14 15 : 16 17 18 19 1a 1b 1c 1d 31
LOSTSYNC        0   32 1e 1f 20 21 22 23 24 25 : 26 27 28 29 2a 2b 2c 2d 47
               48 2e 2f 30 31 32 33 34 35 : 36 37 38 39 3a 3b 3c 3d 63
BITRTRI         6048  64 3e 3f 40 41 42 43 44 45 : 46 47 48 49 4a 4b 4c 4d 79
BITRTRA        5697.6 80 4e 4f 50 51 52 53 54 55 : 56 57 58 59 5a 5b 5c 5d 95
BITRTWI         0   96 5e 5f 60 61 62 63 64 65 : 66 67 68 69 6a 6b 6c 6d 111
BITRTWA         0.0 112 6e 6f 70 71 72 73 74 75 : 76 77 78 79 7a 7b 00 00 127

```

See display for plots

```

6.0 Hz  26.0 s  0.0 Hz  26.0 s  5.4 Hz  0 s  0.0 Hz  0 s
> .....
Page Operational Data 0010 NO ERRORS. LAST ERROR AT Step 0..3..6..9..c..f
10 Thu Feb 12 13:26:53 OFF ↑
0000.00:03:01 TPS for Windows (C) MST 1991-1998 Thu Feb 12 13:29:53 1998

```

**Fig 2.4. TPS Display with Test Data on. (Top left 80 cols and 35 rows only).**

## 2.5 Transmitting Data using an Ethernet Network

If your PC is networked, you can re-transmit incoming data to any other listening PCs. Usually they will be running TPS although they needn't be. Note, PCs don't have to be actively listening for TPS to transmit data. To transmit incoming data on to the ethernet network, your Windows 95 or NT must be configured for the TCPIP protocol ('UDP' will come with it). If you are confused as to the network terminology here, see also the online help and the printed TPS V5.0 Supplement. Your network administrator will be able to tell you what the network capability of your PC is.

If you do not have network access or are running TPS for Windows 3.1, skip this section and move to section 2.7 on serial data transmission.

Select the `Transfer, Network, UDP Transmit Open` menu item.

You will see the message bar display a time-stamped message:

```
UDP transmit service opened
```

Although TPS is now configured to broadcast data to anyone listening, no data will be transmitted until the packet `Echo` service is switched on.

Select the `Echo data` menu item from the `Transfer` menu and switch the echo on. This will enable all received data to be automatically broadcast to any listeners on the network..

Once the following are satisfied:

- data is incoming (in this case, test data)
- the UDP service is started
- the Echo data option is on

then you should see some more page 10 statistics come alive, in particular, the packets written parameter 'PKTSWR' (upper left below PKTSRD) will increment and the plot of the instantaneous 'Packet Rate Write' (PKTRTWRI) showing a non-zero value.

If you quit TPS now (don't do this if you are going on to the next section) you will be prompted as to whether you wish to save the 'System Changes', i.e. the current configuration. If you wish to finish now, reply `no` so that the settings are not saved. If you reply `Yes` to save, TPS will default to starting the UDP broadcast service when you next start. It will also start the test data automatically so that TPS starts broadcasting test packets immediately.

## 2.6 Receiving Data using an Ethernet Network

TPS can receive data either from another TPS on the network or from an external source, where 'external' is basically any non-TPS source transmitting data.

For the purposes of this example, we will treat the TPS broadcasting its own test data (see the prior section) as the external source.

If you have a multi-user, network system then a second PC can reside on the network and receive data from it. Any other TPS PC connected to the network can also receive data from it. In the previous section, TPS was set up to broadcast its test data. It could, of course, be receiving data from an external source to which it, alone, is connected. For this example, it is behaving as the front-end or 'server'.

The front-end/server PC is that receiving data from the outside world and broadcasting it to TPS clients. In this document, the external data source is simulated using the test data. The 'network client' is that receiving its data from the server.

With the TPS system running as left at the end of the previous section, perform the following steps.

On the network client PC, install TPS and start it running. Then set TPS to receive broadcast data as follows:

Select the `Transfer, Network, UDP Receive Open` menu item

You will see the message bar display a time-stamped message:

```
UDP receive service opened
```

If all is okay then, within a few seconds, the client PC should start receiving data from the front-end.

If all is not okay, check your ethernet adapter and cable. See the Online Help 'Network' topic for more diagnostics information.

The client PC, page 10, should show the PKTSRD statistic incrementing and the incoming packet rate PKTRTRDI showing about 7-8Hz. It should look something like that shown in Fig 2.6.

When satisfied with the network, quit both the front-end server and network client TPS systems ready to restart TPS in the next section. Respond No to all Save prompts.

```

LASTPKTR Thu Feb 12 16:07:54 1998
LASTPKTW Operational_Data Version W32V5.01h
LASTBYTR Thu Feb 12 16:07:54 1998 Page 10 Loop-meas 66
LASTBYTW Loop-time 10

INNAME Network Client INFMT RAW INTYPE UDP
OUTNAME NO OUTPUT SOURCE OUTFMT NONE OUTTYPE NONE

STATSTRD 13 PKTSYNC 00001234 > DATA < Pkt-Selct'd ON
STATSTWR 0 PKTLENGTH 128
PKTSRD 109 Hexadecimal Packet Dump
PKTSWR 0 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
BYTESRD 14208 0 12 34 00 01 02 03 04 05 : 06 07 08 09 0a 0b 0c 0d 15
BYTESWR 0 16 0e 0f 10 11 12 13 14 15 : 16 17 18 19 1a 1b 1c 1d 31
LOSTSYNC 0 32 1e 1f 20 21 22 23 24 25 : 26 27 28 29 2a 2b 2c 2d 47
48 2e 2f 30 31 32 33 34 35 : 36 37 38 39 3a 3b 3c 3d 63
BITRTRI 8192 64 3e 3f 40 41 42 43 44 45 : 46 47 48 49 4a 4b 4c 4d 79
BITRTRA 8507.2 80 4e 4f 50 51 52 53 54 55 : 56 57 58 59 5a 5b 5c 5d 95
BITRTWI 0 96 5e 5f 60 61 62 63 64 65 : 66 67 68 69 6a 6b 6c 6d 111
BITRTWA 0.0 112 6e 6f 70 71 72 73 74 75 : 76 77 78 79 7a 7b 7c 7d 127

```

See display for plots

```

8.0 Hz 15.3 s 0.0 Hz 15.3 s 8.3 Hz 15 s 0.0 Hz 15 s
> Thu Feb 12 16:07:40 1998 UDP receive service opened
Page Operational Data 0010 NO ERRORS. LAST ERROR AT Step 0..3..6..9..c..f
10 Thu Feb 12 16:07:40 OFF ↑
0000.00:00:15 TPS for Windows (C) MST 1991-1998 Thu Feb 12 16:07:54 1998

```

**Fig 2.6. Page 10 display when receiving UDP broadcast data.  
(Top left 80 cols and 35 rows only).**

## 2.7 Transmitting Data to a Serial Port

This section and the next section illustrate using serial RS232 transmit and receive for the purposes of exchanging data in preference to, or absence of, an ethernet network.

Firstly, start TPS with the test data running, as at the end of section 2.4, and scroll the display down to reveal the serial statistics - green background view-ports below the system display.

To transmit incoming data you must have a free serial port. Invariably, Windows has a serial mouse attached to one of the ports which usually leaves the second port free. If you do not have a free serial port, skip this section and move to section 2.10. Whichever port is free, call this port COM $n$  where  $n=1$  to 4, it must be configured for TPS data output, to do this:

Select the `Transfer, Serial, COM $n$`  menu item.  
Check the `Out` button and clear the `In` button.  
press `OK` to accept all other default settings.

If there are any other devices connected to the same port, TPS will not open the port and a dialog box will appear informing you of a possible device conflict. In such a case, try a different port.

Although COM $n$  is now configured for output, no data will be written until the packet `Echo` service is switched on.

Select the `Echo` data menu item from the `Transfer` menu and switch the echo on. This will enable all received data to automatically be transmitted to the serial port. Without the Echo enabled, the output serial port will remain open but will not transmit any bytes.

Once the following are satisfied:

data is incoming (in this case, test data)  
a serial port COM port is configured in the `Out` state  
the Echo data option is on

then you should see some more page 10 statistics come alive, in particular, the packets written parameter 'PKTSWR' (upper left below PKTSRD) will increment and the plot of the instantaneous 'Packet Rate Write' (PKTRTWRI) showing a non-zero value.

If you quit TPS now (don't do this if you are going on to the next section) you will be prompted as to whether you wish to save the 'System Changes', i.e. the new serial settings (COM $n$  switched to the Out state). If you wish to finish now, reply `No` so that the settings are not saved. If you reply `Yes` to save, TPS will default to the new serial settings when next started.



## 2.8 Receiving Data from a Serial Port

TPS can receive data via the serial ports either from another TPS on a serial network or from an external source, where 'external' is basically any non-TPS source transmitting data.

For the purposes of this example, we will treat the TPS transmitting its own test data (see the prior section) as the external source. This PC is termed the 'front-end' or 'server' PC. Normally, the front-end PC is that receiving data from the outside world. In this document, the external data source is simulated using the test data.

With the TPS system running as at the end of the previous section, perform the following steps.

Connect a second serial 'client' PC to the first PC using a null modem cable (search the online help for 'null modem' details). Naturally, you must connect the cable to the output port on the front-end server, and to the input port on the client PC.

On the client PC, install TPS and start it running. Then set TPS to receive incoming data on a free serial port (termed 'COMm' in the steps following) to which the null modem cable is connected. To do this:

- Select the `Transfer, Serial, COMm` menu item
- Check the `In` button, clear the `Out` button
- Accept the defaults by pressing OK

If Windows has no spare serial ports on the client PC, TPS will inform you of this. If this is a problem, try another port or skip to the next section.

If all is okay then, within a few seconds (it may be longer - up to a minute), the client PC should start receiving data from the front-end.

If all is not okay, check the null modem cable, the serial settings, and the ports to which the cable is connected. You could use Windows Terminal on the network PC to independently verify incoming data.

The client PC, page 10, should show the PKTSRD statistic incrementing and the incoming packet rate PKTRTRDI showing about 7-8Hz. It should look something like that shown in Fig 2.8.

When satisfied with the workings, quit both the front-end and client TPS systems ready to restart TPS in the next section. Respond No to all Save prompts.

```

LASTPKTR Mon May 19 17:01:58 1997
LASTPKTW Operational_Data Version W16V4.10d
LASTBYTR Mon May 19 17:01:58 1997 Page 10 Loop-meas 55
LASTBYTW Loop-time 10

```

```

INNAME          COM2      INFMT      RAW      INTYPE  SERIAL
OUTNAME         NO OUTPUT SOURCE  OUTFMT    NONE    OUTTYPE  NONE

STATSTRD        21  PKTSYNC    00001234  > NO DATA <  Pkt-Selct'd  ON
STATSTWR        0  PKTLENGTH    128
PKTSRD          27              Hexadecimal Packet Dump
PKTSWR          0           0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
BYTESRD        3456    0 12 34 00 01 02 03 04 05 : 06 07 08 09 0a 0b 0c 0d 15
BYTESWR          0    16 0e 0f 10 11 12 13 14 15 : 16 17 18 19 1a 1b 1c 1d 31
LOSTSYNC        0    32 1e 1f 20 21 22 23 24 25 : 26 27 28 29 2a 2b 2c 2d 47
              48 2e 2f 30 31 32 33 34 35 : 36 37 38 39 3a 3b 3c 3d 63
BITRTRI          0    64 3e 3f 40 41 42 43 44 45 : 46 47 48 49 4a 4b 4c 4d 79
BITRTRA        1316.8  80 4e 4f 50 51 52 53 54 55 : 56 57 58 59 5a 5b 5c 5d 95
BITRTWI          0    96 5e 5f 60 61 62 63 64 65 : 66 67 68 69 6a 6b 6c 6d 111
BITRTWA          0.0 112 6e 6f 70 71 72 73 74 75 : 76 77 78 79 7a 7b 7c 7d 127

```

See display for plots

```

0.0 Hz    3.5 s    0.0 Hz    3.5 s    1.3 Hz    65 s    0.0 Hz    65 s
> .....
Page      Operational Data      0010 NO ERRORS. LAST ERROR AT Step 0..3..6..9..c..f
10                                     Mon May 19 17:01:12      OFF      ↑
0000.00:01:05    TPS for Windows    (C) MST 1991-1997    Mon May 19 17:02:15 1997

```

**Fig 2.8. Page 10 display when receiving serial data.  
(Top left 80 cols and 35 rows only).**

## 2.9 Transmit and Receive Summary

As you have seen, TPS provides Ethernet and serial RS232 data communications for exchange of data with itself and the outside world, i.e. other devices and software.

UDP transmission and receive of data is the very simplest method of data exchange using TPS. It is relatively fast and can offer megabit/sec data rates, but, it is only suitable for local area networks. For global Internet communications between TPS's and, indeed, any other data transmitter, the ethernet TCPIP method is necessary. Both methods are discussed in greater detail in the online help, search for the 'Network' keyword.

Because all PCs usually have two serial ports, it is relatively easy to connect PCs together with a serial cable and use serial communications to exchange data. Better still, you can use both ethernet and serial. For example, you can receive data from a serial port and then broadcast it on a local area network using UDP. Alternatively, send it across the world via the Internet using TCPIP.

To summarise, TPS can have three external input sources (1) which can also be used for transmission namely, Ethernet UDP, Ethernet TCPIP and Serial RS232.

(1) TPS can, of course, also have other internal input sources such as test data and file replay.

A single TPS can receive and transmit in any combination of the three summarised in the below table:

Receive\Transmit	UDP	TCPIP	Serial
Serial	Yes	Yes	Yes
UDP	No (2)	Yes	Yes
TCPIP	Yes	Yes	Yes

(2) Note, a UDP client cannot also broadcast as a UDP server since that would imply there are two PCs on the network both broadcasting the same data.

Serial communications with TPS is good for at least 100kbits/sec and can go a lot higher with special serial communications boards. On the other hand, many data sources, such as flight computers, send data at a much steadier speed rate of 9600 or 19200 bits/sec. For these low rates, TPS for Windows 3.1 is adequate but, for higher speeds, especially 100 Kilobits/sec plus, only TPS for Windows NT offers good performance - TPS for Windows 95 lies in between.

## 2.10 Configuring the Packet Format

Start this section by restarting TPS. If you have continued from the previous section, quit TPS now (respond `No` to all 'Save' prompts) and then restart TPS.

To read any packet data, TPS must know the packet format. Only two numbers are required to explicitly define the packet format, these are:

Packet length

Synchronisation bit pattern

These numbers are configured online using the `Transfer, Pkt format` menu item. To select this option you must first unlock the display - this is explained shortly.

**Packet length**                      Packet length in bytes. This size includes the number of synchronisation bytes (see below). The maximum value can be 4224 bytes. The minimum value is just the number of synchronisation bytes (albeit a packet of this size is of little use).

**Synchronisation bytes**                      1 to 4 bytes always present at the start of the packet.

TPS defaults to a packet with a length of 128 bytes which includes the two default synchronisation bytes, arbitrarily set to 12 34 in hex.

Switch the test data on (menu item `Test data` from the `Transfer` menu) and the packet dump on page 10 will show the first two bytes as hex 12 34 and the remaining bytes filled sequentially from 0 to hex 7d (125 decimal).

It is instructive to switch the test data to a different format. You don't have to stop the test data to do this.

Turn to page 13, this shows a larger packet dump than that on page 10. In fact, it shows the first 256 bytes. Since the current test data is only 128 bytes, you will only see the first 128 bytes filled. The packet dump also has a side-by-side ASCII character translation of each byte. Although packets do not generally contain text readable bytes, text messages are sometimes embedded in data packets.

In fact, outside of the telemetry world, the ASCII translation is very useful as message packets (those uncompressed) invariably have embedded text. Furthermore, since TPS can, using the offline toolkit, convert any text file into packets, TPS can transmit text files over a TPS network and simultaneously show the contents.

Before you can change the format you must unlock the display. TPS starts with the display locked to protect users from inadvertently changing the display settings.

To unlock the display, select the `Display, Lock` menu item and switch the lock off. Pop-up the `Packet, Format` dialog box by selecting `Pkt Format` from the `Transfer` menu.

Enter the 32-bit `Synchronisation` pattern `0xaabbccdd` (hex) and set the `Packet Length` to 192 bytes (arbitrary example). Press OK and you will immediately see the test data change to a 192 byte packet as shown in figure 2.10. Notice, the synchronisation pattern is now the four bytes `0xaa 0xbb 0xcc 0xdd`.

If you continue to the next section, set the packet format back to its default with a synchronisation pattern `0x1234` and length 128 bytes. Otherwise, exit TPS and reply `No` to any 'Save' prompts so as not to save any changes.

```

> DATA <
  0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 4 8 C
0 aa bb cc dd 00 01 02 03 : 04 05 06 07 08 09 0a 0b 15 15
16 0c 0d 0e 0f 10 11 12 13 : 14 15 16 17 18 19 1a 1b 31 31
32 1c 1d 1e 1f 20 21 22 23 : 24 25 26 27 28 29 2a 2b 47 47
48 2c 2d 2e 2f 30 31 32 33 : 34 35 36 37 38 39 3a 3b 63 63
64 3c 3d 3e 3f 40 41 42 43 : 44 45 46 47 48 49 4a 4b 79 79
80 4c 4d 4e 4f 50 51 52 53 : 54 55 56 57 58 59 5a 5b 95 95
96 5c 5d 5e 5f 60 61 62 63 : 64 65 66 67 68 69 6a 6b 111 111
112 6c 6d 6e 6f 70 71 72 73 : 74 75 76 77 78 79 7a 7b 127 127
lmnopqrstuvwxyz{

128 7c 7d 7e 7f 80 81 82 83 : 84 85 86 87 88 89 8a 8b 143 143
144 8c 8d 8e 8f 90 91 92 93 : 94 95 96 97 98 99 9a 9b 159 159
160 9c 9d 9e 9f a0 a1 a2 a3 : a4 a5 a6 a7 a8 a9 aa ab 175 175
176 ac ad ae af b0 b1 b2 b3 : b4 b5 b6 b7 b8 b9 ba bb 191 191
192 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 207 207
208 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 223 223
224 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 239 239
240 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 255 255

256 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 271 271
272 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 287 287
288 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 303 303
304 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 319 319
320 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 335 335
336 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 351 351
352 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 367 367
368 00 00 00 00 00 00 00 00 : 00 00 00 00 00 00 00 00 383 383

> .....
Page      PACKET DUMP      0010 NO ERRORS. LAST ERROR AT Step 0..3..6..9..c..f
13                      Mon May 19 17:05:13      OFF      ↑
0000.00:01:38      TPS for Windows      (C) MST 1991-1997      Mon May 19 17:06:50 1997

```

**Fig 2.10. 192 byte test data packet with a 0xaabbccdd synchronisation pattern.**

## 2.11 Configuring your own Displays

Before starting this section, if TPS is not already running, start it and switch the test data on.

Whilst TPS displays standard, non-specific packet data, you will invariably wish to view parameters embedded in your own packets.

See the previous section for details on configuring TPS to accept your packet format, i.e. synchronisation byte pattern and packet length.

Every data item (a number or text string) is displayed in a view-port, it can also be plotted. The view-port is a rectangular box on the display displaying a number, usually extracted from the incoming packet. The packets read statistic on page 10 'PKTSRD' is one such view-port which displays a number internally generated by TPS and put in a statistics array which TPS accesses to display.

This section will concentrate on a user-relevant example view-port supplied on page 9998.

Turn to page 9998 (press ALT P and enter 9998 in the file dialog box or click file PAGE9998.BIN in the file list box).

Before you can change any view-port you must unlock the display - this may have already been done if you have proceeded from the last section.

To unlock the display, select the `Display, Lock` menu item and switch off.

Place the mouse cursor over the port titled PKT\_MNEMONIC and then double click. This will spring up the parameter options dialog box. Click the `Settings` button, the parameter settings dialog box will appear as shown in Fig 2.11.1

The parameter settings dialog box contains several different edit boxes, list boxes and switches. The parameter displayed in the view-port, the view-port layout, colors and all other settings are changed from this dialog box.

Fig 2.11.1 PKT\_MNEMONIC settings dialog box.

Parameter 'PKT\_MNEMONIC' Settings

Parameter: meanallfr\_usecs  
meanfr\_usecs  
mst\_files  
packetdata  
packetheader

Datatype: %6hd

Commutation

Byteindex: 2 Sub-com

Count: 1

Increment: 0

Text

Mnemonic: PKT\_MNEMONIC

Title: PKT\_TITLE

Units: PKT#UNITS

Calibration

Function: nyesnabit  
onoffbit  
openclosebit  
packetbar  
polynomial

Row: 10 Plot

Column: 3 X Value

Format: %6hd

Rate: every10msecs

Switches

☐ Border

☐ No draw

Colors

Background: black  
blue

Title: red  
white

Value: white  
yellow

Border: red  
white

OK

Cancel

Help



### How the view-port appears...

The visible view-port consists of the parameter `Mnemonic`, `Title`, `Units` and the value, itself, all arranged in a rectangular box with an optional border like so:

```
          Title
Mnemonic value units
```

or, more specifically, in this example:

```
          PKT_TITLE
PKT_MNEMONIC value PKT_UNITS
```

The box is automatically sized according to the character length of the displayed value, `Mnemonic`, `Title` and `Units` text. Its position on the display is specified by a `Row` and `Column` - although you will not generally configure this position from within the settings dialog box but rather position by dragging and dropping the port around the screen (more later).

The title, mnemonic and units can each be blank, however, at least one of the three must not be blank as TPS uses the non-blank entry to identify the port. For example, if you only supplied a mnemonic, `PKT_MNEMONIC` in this example, TPS thereafter would always refer to this view-port as `PKT_MNEMONIC`. You will notice that it appears in the title-bar of the settings dialog box.

The use of a mnemonic to uniquely identify each parameter/view-port is a pseudo convention. Generally, although not mandatory, it makes life simple if all parameters have a unique mnemonic. **YOU ARE STRONGLY ADVISED TO STICK TO THIS!**

The title is automatically centred, the mnemonic and units automatically justified. Use the hash character '#' to prefix or append blank spaces to any of the text to force TPS to layout the port differently. TPS substitutes each hash character with a space when you press OK to make permanent any dialog box changes. A space character can be used but is often hard to see in the dialog box.

The colors for the background, text, titles and border are all selected from list boxes within the `Colors` group box, each can have one of 16 values.

The border is switched on and off via the `Border` check box within the `Switches` group.

The `No draw` switch disables TPS from drawing the actual parameter value although the view-port, itself, is drawn. Whilst this may seem pointless, it can be used to good effect to draw text

without any value, i.e. you can insert comments on the display by writing the text in the mnemonic, title or units field without having any value displayed.

### **How does TPS know where to extract the parameter value from the packet ?**

With a simple packet format, every parameter would appear only once in every packet - this is termed a single sampled or 'normally commutated' parameter. However, most packet formats are much more complicated and a parameter can appear one or more times in a packet or, instead, only in every 2nd, 3rd or 4th packet. etc. TPS can handle the following cases.

1. single sampled in every packet (normally commutated).
2. single sampled once in every nth packet (sub-commutated).
3. multiply sampled in every packet (super-commutated).
4. multiply sampled in every nth packet (super-commutation of a sub-commutated parameter).

Thus, to uniquely identify a parameter, the view-port settings must specify in what packet it appears, its starting byte-offset and how many samples are present in a given packet. (The number of bits or bytes per parameter is detailed following this sub-section). The view-port settings dialog box contains the following edit items to provide this information.

Byteindex	Byte Position of the first sample in packet
Count	Number of samples per parameter
Increment	Byte gap between samples in the same packet
Sub-com	Define packets in which the parameter appears

For this example, we will only treat PKT\_MNEMONIC as a normally commutated parameter. For more details on using the Count, Increment and Sub-com edit items to configure super and sub-commutated parameters, see the online help topic 'Displaying Packet Parameters' for more information.

The Parameter's actual value is extracted from the packet by specification of its byte offset (Byteindex) from the packethheader (you should see the packethheader highlighted in the Parameter list box). The example shows that this parameter is offset 2 bytes from the start of the packetheader, i.e. starts at the first byte after the default TPS two byte sync. All byteindex values are zero biased - the first byte of the packet (the first sync byte, TPS default hex 12) has a byteindex of 0.

The parameter list box contains many different parameters which can be found in the online help - search on keyword 'Parameter List Box'.

### **How does TPS know how many bytes the parameter contains ?**

The Datatype list box contains a list of simplified data types, e.g. `Integer`, from which, TPS infers the byte length (an integer is 2 bytes on Windows 3.1 TPS models and 4 bytes on Windows NT and 95 models). However, you will see that the highlighted selection shows the more esoteric `%6hd` which is a C language printf format string denoting a short int data-type, i.e. 2 bytes on all TPS models. If you bring down the list box, select integer, then TPS will accept this and internally convert it to the `"%6hd"` C format. Hence, the next time the PKT\_MNEMONIC settings dialog box is viewed, you will once again see `"%6hd"` and not the integer selection you made.

TPS uses C language printf formats as they are far more powerful, albeit not intuitive. You can directly enter a C language format instead of using the simplified data types shown in the list box. This will give you greater control. For the Datatype, this is not really necessary as TPS only needs to know the number of bytes in the parameter which can be inferred from any simplified data type, e.g. doubleintegers are 4 bytes, reals are 4 bytes, doublereal is 8 bytes etc. (see the Online Help - search for the 'Datatype' keyword). For the display format (see the Format box), the C printf format can give you much greater control over the appearance of the displayed parameter, this is where it really comes into its own.

With the parameter specified as 2 bytes, starting at byteindex 2, TPS will retrieve from the test data packet, the two bytes 00 01 (see the packet dump on page 10). When it actually comes to displaying the value, this is the integer equivalent of 256 on a PC. The value derives from the fact that the first byte (00) is the least significant byte and the second byte (01) is the most significant. Hence, the integer value is computed as  $00 + 256 \times 01 = 256$ . When viewing page 9998 with test data, this is what you should see providing TPS is configured with its default packet format (as upon starting TPS and switching on the test data).

### How does TPS control the value's visual display ?

In this example, the display format (box `Format`) is an integer, i.e. exactly as extracted (see `Datatype`) which, with a format of `"%6hd"`, would space the value to 6 spaces, i.e. it would appear as `###256` where each hash is replaced by a space. You could display it in another way, for example, if you wanted the first three digits, use the C printf format `"%3hd"`. Alternatively, you could display the value in hexadecimal by selecting `hexinteger` from the `Format` list box. There are almost infinitely many ways of displaying and reformatting data, search the online help for keyword 'Format'.

### How do I convert the value to engineering units ?

This process is called calibration in TPS jargon. It allows for such operations as scaling, byte swapping, bit masking and bit shifting and applying the raw value to a polynomial before displaying.

**Note, you will need a polynomial calibration to do bit-mask and/or byte-swap, and conversion from integer data-type to real/float display format - this necessity is regardless of whether you need a full polynomial.**

In this example, you should see the `Function` list box has the `polynomial` calibration highlighted. Click on this box and the `Polynomial Calibration` dialog box will appear as shown in figure 2.11.2.

Note, if a view-port has no calibration function, or you wish to switch an existing calibration off, then select any one of the three functions `dummy_calibfn`, `dummy_calibfn1`, or `dummy_calibfn2` as listed in the `Function` list box.

The most important entries in this box (at least for this example) are the coefficients `a0`, `a1`, `a2`, `a3` and `a4`. You will see these set to 0, 1, 0, 0 and 0 respectively. TPS extracts the raw value (256 in this example) and then applies it to the following polynomial:

$$(((a4 \times \text{value} + a3) \times \text{value} + a2) \times \text{value} + a1) \times \text{value} + a0$$

With the numbers given, this polynomial will simply return the same value 256. Although such a linear set of coefficients does seem a rather pointless calibration, the polynomial calibration does, additionally, allow for the possibility of applying a bit-mask and/or byte-swap as well as conversion from integer to real which is the only way bit and byte manipulations can be performed.

Note, bit-shifts to the right can be achieved by use of the `Quotient` edit item or, indeed, by using the `a1` coefficient with a real value between zero and one. For instance, to shift right 4 bits, you would use a `Quotient` of 16 or set the `a1` coefficient to 0.0625 (1/16th). In the latter case, ensure the display `Format` is of real type, e.g. `%3.0f`. Bit-shifting to the left, can be achieved by using the `a1` coefficient with a value greater than one.

Full details on calibration are found under the 'Calibration, Engineering Units Conversion help topic, see the contents page.

Cancel the polynomial calibration dialog box and return to the main parameter settings dialog box.

**Fig 2.11.2 PKT\_MNEMONIC Polynomial Calibration dialog box.**

**PKT\_MNEMONIC polynomial calibration**

**Bitmask**

**Modulus**

**Quotient**

**Average Count**

☐ **Reciprocal**

☐ **Byteswap**

**Coefficients**

**a4**

**a3**

**a2**

**a1**

**a0**

**Help** **OK** **Cancel**

The frequency with which the value is extracted and processed is given by the entry in the Rate list box. This is highlighted with `every10msecs`. The Rate must always be at least twice as great as the incoming packet rate to ensure TPS will extract and process every value. In fact, unless your packet rate is greater than 100Hz, we strongly advise you always choose the fastest, `every10msecs`, rate.

The displayed Rate selections can change with the TPS 'Loop-time'. However, with TPS installed and configured as in this guide, the Loop-time is set at 10 ms (100 Hz packet processing). The Loop-time is considered beyond the scope of this guide, suffice to say, it controls the rate at which TPS processes packets and is used mainly to speed-up or slow down the replay of packet data files. For more details, search the online help for keyword 'looptime'.

Now press `Cancel` to close the PKT\_MNEMONIC settings dialog box.

### **How do I create my own view ports ?**

Most importantly, with TPS, you never actually create a view-port from scratch rather you always start with an existing view-port, make a copy and modify that.

**The samples disk, pages 25-30, have view-ports and plots to extract bytes and integers so that you can immediately view the numeric values of all byte and 2-byte integers. They also have polynomial calibrations included which you may find very useful and save you a lot of time! See the Samples disk README.TXT file for more details (not the same file as the TPS README icon).**

As an example of creating a view-port from an existing view-port, we will use PKT\_MNEMONIC as already discussed in this section.

To make a working copy of PKT\_MNEMONIC cancel the previous dialog box, if not already done so, and then double click on the same view-port again.

Click the `Copy To` button, TPS will respond by prompting you for a page to where you want to copy the view-port. Note, prior to this, a warning message will appear, it is okay to press OK to continue. For the sake of argument, let's make a new page 99 (don't use an existing page as yet).

Enter 99 in the `File_name` edit box.

Since page 99 doesn't already exist, you will be prompted as to whether you wish to create a new page, respond Yes.

Note, if you accidentally changed the original `PKT_MNEMONIC` view-port on page 9998, you will first be prompted as to whether you wish to save the current page, i.e. page 9998 - respond No so as not to save the page.

TPS will then create a new page 99 and change to it. You should now see the copied `PKT_MNEMONIC` view-port.

On Windows 3.1/95 systems, after creating the new page, TPS will actually return the Windows input focus to the Program Manager window, you will then have to click the TPS Window to continue using TPS.

To move the view-port to a new position, position the mouse cursor anywhere over the view-port, click and hold down the left mouse button, and then drag the port to the new desired position. Release the left mouse button at the desired position.

You will see a frame, bounding the view-port, move in small jerks across the screen. This jerky movement is because the frame only redraws when the position changes by one or more text columns (not pixels). This is preferable to continually redrawing the frame every time it moves one pixel as it doesn't disturb any incoming data (by heavily loading the CPU). It also has an added advantage that you can 'snap' the ports for very easy alignment with each other.

Note, we also suggest you copy the same view-port to another position on the current page - page 99. To do this, repeat the same operation to move the port BUT ADDITIONALLY hold down the shift key before pressing the left mouse button. This will then move a copy of the port leaving the original intact.

You can now experiment, ad nauseam, with either of the copies of the view ports.

When you double click over a view port, the dialog box that appears also has a `Delete` button, click this to delete the port - you will be prompted again before TPS actually deletes the port.

If you wish to create a new port, double click on an unused area of the screen, a `New Parameter` dialog box will appear from where you can either copy an existing port from another page (`Copy From` button), or create a new port (`New` button). However, the 'New' function only actually allows you to copy an existing port on either the current page or the defaults page 9998 and use that as a template for a new view-port.

If you have a very crowded display, you may lose sight of an existing port buried under another port (usually a newer port). Move a few visible ports around to uncover any hidden ports.

Lastly, you can save all changes by selecting the `Display, Save` menu item. Don't do this now although it is harmless as page 99 was not a pre-supplied page. If at any time you quit TPS, without having previously saved your changes, you will be prompted before TPS exits. This also occurs when you change to another page.

Global operations with entire pages, i.e. deleting, copying, merging etc., although not difficult, is an offline procedure using a text editor. It can save you a lot of time when, for instance, changing the background color (or any other attribute) of several view-ports. You can also block delete whole sections. See the 'PGCKW and Offline Configuration' Reference Manual for full details. This is supplied in electronic form only (not pre-printed) on the Documentation Disk.



## 2.12 Plotting Data

Start this section by re-starting TPS, don't save any changes when prompted. After re-starting, unlock the display.

Every parameter displayed can be plotted against any other displayed parameter on the same page. Plots are really just an extension of a view-port, i.e. the view-port of the value to be plotted (y-axis).

Turn to page 9998 - you may be asked to save any prior edits; reply `No`.

On page 9998 you will see a plot entitled `TIME_MOD_20`. This is a good example plot.

The plot should show a saw-tooth. It is actually the TPS elapsed time (in seconds) plotted against itself but with the y-axis value polynomially calibrated with a modulus function (modulo 20) such that it increments the elapsed time between 0 to 19 and then wraps back to zero. If the modulus function were not applied, the plot would just be a straight line as the time would increase without bound along both axes.

Copy the `Time mod 20` view-port to another page as follows:

double click EITHER any portion of the numeric value (just above the plot) or any region of the plot. This will always bring up the view-port options dialog for the Y value.

Select `Copy To` and enter 99. If this page already exists (you may have created it in the previous section), then TPS will copy the plot and turn straight to page 99, otherwise it will prompt you to create the page first, please reply `Yes` to create page 99. Note, prior to this, a warning message will appear, it is okay to press OK to continue.

Now make a second copy of `Time mod 20` on the same page and move it to another part of the display. To do this, hold down the shift key, put the mouse over either the y view-port or plotting area and press the left button whilst dragging and dropping the view-port to a new region on the page. All parts of the plot will be copied when you drop the copy at a new display position. The plot area, itself, may not be visible, if so, just click the area below `TIME_MOD_20` which will reveal the outline of the plot.

You can create a plot of a view-port's value by double-clicking the view-port and pressing the `Plot` button from within its `Settings` dialog box. However, like view-ports, we suggest you always copy an existing plot. In fact, as for a view port, the `New` button only presents you with a list of pre-made plot templates on either the current page or the defaults page 9998.

You can move the Y view-port (mnemonic `TIME_MOD_20`) and X view-port (mnemonic `TIME`) completely independently of the plot and each other simply by dragging and dropping them. However, no matter where you put them on the display, they are always associated with the plot.

### Editing the plot

All edits to a plot are done via the Y parameter, double click the `TIME_MOD_20` view-port and bring up its settings dialog box. The plot area and the X value are edited by clicking their respective buttons `Plot` and `X Value` in the top right of the dialog box.

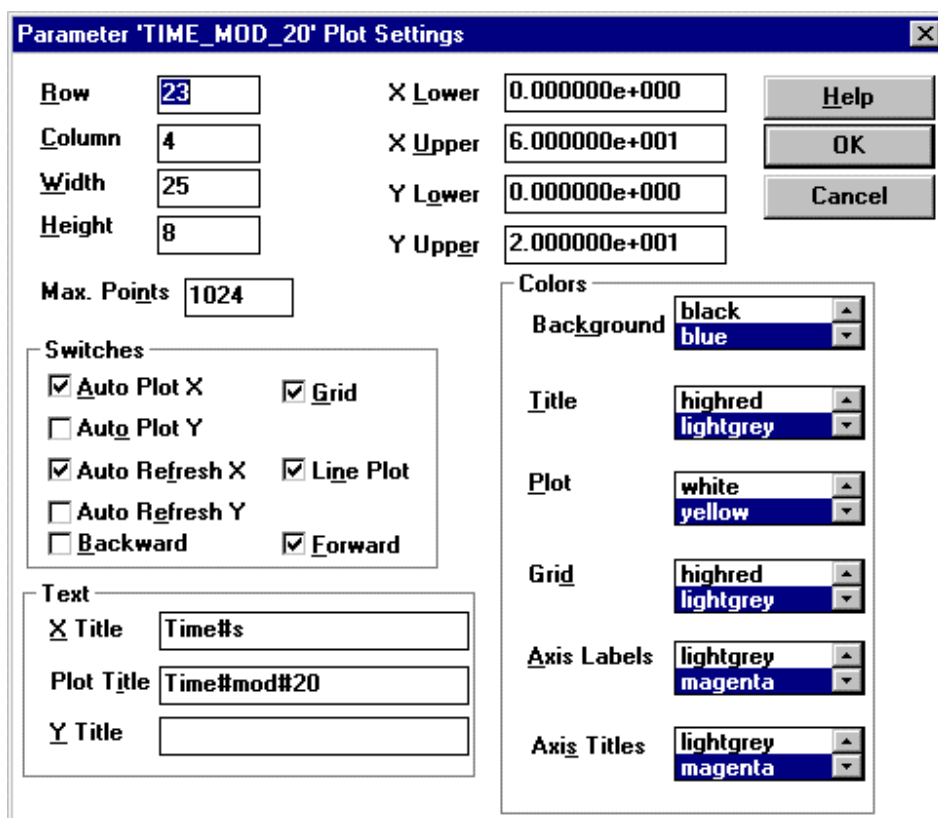
Now click the `Plot` button to bring up the plot settings dialog box. It should appear as in Fig 2.12.

The `Plot Settings` dialog box overlays the y-value's dialog box. Some of the edit and list boxes shown, such as the `Row`, `Column` and `Colors` group, should be familiar from the view-port settings dialog box, others are new such as the plot limits.

### Plot limits and auto plotting

On the plot, you should see limits `X Lower` and `X Upper` with values 0 to 60. However, when observing the plot, you will notice that the X axis (the TPS elapsed time in seconds) will probably have a value of a few thousand seconds (or at least 100). Normally, this would mean all plotted points would be off the X axis as its upper limit, `X Upper`, is 60 seconds. However, you will see in the `Switches` group box that the `Auto Plot X` switch has been checked. This means that TPS will re-scale its axis to the current scale and plot in a range `X + X Lower` to `X + X Upper`, hence, the time base is 60 seconds starting from the current X value. When the trace reaches the far right, the far right becomes the new origin and the new x limit becomes the origin + 60 seconds. This is a hybrid form of a strip chart - it does not continually scroll but only refreshes when the trace reaches the plot boundary. (Strip charts are very CPU intensive and TPS prioritises data loss avoidance over display refresh).

Fig.2.12 TIME\_MOD\_20 Plot Settings dialog box.



The dialog box is titled "Parameter 'TIME\_MOD\_20' Plot Settings". It contains several sections for configuring the plot:

- Row**: 23
- Column**: 4
- Width**: 25
- Height**: 8
- Max. Points**: 1024
- Switches**:
  - ☒ Auto Plot X
  - ☐ Auto Plot Y
  - ☒ Auto Refresh X
  - ☐ Auto Refresh Y
  - ☐ Backward
  - ☒ Grid
  - ☒ Line Plot
  - ☒ Forward
- Text**:
  - X Title**: Time#s
  - Plot Title**: Time#mod#20
  - Y Title**:
- X Lower**: 0.000000e+000
- X Upper**: 6.000000e+001
- Y Lower**: 0.000000e+000
- Y Upper**: 2.000000e+001
- Colors**:
  - Background**: black (selected), blue
  - Title**: highred (selected), lightgrey
  - Plot**: white (selected), yellow
  - Grid**: highred (selected), lightgrey
  - Axis Labels**: lightgrey (selected), magenta
  - Axis Titles**: lightgrey (selected), magenta

Buttons: Help, OK, Cancel

### Autoplots and autorefresh

Auto-refresh plots are used when you do not want the axis re-scaled, as for an auto-plot, but you do want the old trace deleted before it starts redrawing at the origin, having crossed the right hand edge. You will notice that the `Auto Refresh X` switch is checked, this is always true when `Auto Plot X` is on. The converse is not true. You can have an auto- refresh switch on (or off) without an auto-plot switch being on - the refresh controls the plot trace refresh, the auto-plot controls the automatic axis scaling.

An auto-plot will NOT automatically redraw itself, i.e. the plot trace will not be refreshed when the following conditions are true:

The Y view-port parameter is a `packetheader` parameter,  
i.e. when displaying a packet parameter,

AND

[There is no incoming data OR the incoming data is corrupt (incomplete packets)]

If plots are not auto-plots, then they are always displayed - although the trace won't be seen if the incoming data is corrupt as it will not redraw until a valid value is received. The incoming or corrupt data condition is stipulated because an autoplot would spend all its time trying to re-scale to corrupt data and so the plots would just seem to keep flashing as they are continually redrawn.

In this example, neither the `Auto Plot Y` or `Auto Refresh Y` switches are on. i.e. the `Y Lower` and `Y Upper` limits are always the absolute values 0 and 20 respectively. If you return to the Y value's view-port settings dialog box (click the Cancel button in the plot settings dialog box) and then click the `Calibration, Function polynomial` (should be highlighted), you will see in the polynomial calibration dialog box, a value `2.0e001`, i.e. 20, for the Modulus. The coefficients are set for a 1:1 linear scaling `a1 = 1`, all others are 0. Thus, the displayed value is identical to the raw value modulo 20. This keeps the Y value within the absolute range of the Y axis. The `Bitmask` is also set to `0x0000ffff` which means the raw value is masked to its lower 16-bits. In this example, the mask is actually redundant since the modulus function already clips the value to within 0 to 19.

As an experiment, delete the modulus value - a blank value switches off the modulus operation. Then keep pressing OK to clear down all dialog boxes and accept the change. Now the y-value will just be identical to the TPS elapsed time, i.e. the X value. You would now normally see a steadily rising straight line plot. However, since the `Auto Plot Y` switch is off, the plot points will be outside of the plot range and you will only see a trace along the upper boundary of the plot as the plot value is outside the absolute range of the Y axis. To remedy this, re-enter the plot

settings dialog box and click the `Auto Plot Y` check box. You may like to increase the Y Upper limit to match the X Upper limit, i.e. change it to 60. Then keep pressing OK to clear down all dialog boxes and accept the changes. Now observe the plot trace is a straight line.

The first thing you will probably notice is that the trace is slow to draw. If TPS has been running for a few thousand seconds, the elapsed time (Y value and X value) will be plotted with a  $\times 1000$  scaling. As a consequence, and because of the auto-scaling algorithms, the range 0 to 60 secs is small in comparison with the actual X value and, hence, the range has had to be adjusted internally so that the actual range is also around 1000 seconds. The range of the axis is usually internally re-computed to 20% of the absolute value. You can get around this by subtracting an offset - try setting both the X value and Y value `a0` polynomial coefficient to the approximate current value (rounded to 1000), e.g. if the elapsed time is currently 5330, set `a0` to -5000 and then watch the plot again (remember to press OK in all dialog boxes to register this value, pressing CANCEL in any will ignore the change). The plot trace should now draw a bit faster corresponding to a finer resolution x axis (smaller range).

To change the X Value of a plot, you must always start from the Y parameter settings dialog. Try clicking the X value view-port, it will popup a message informing you of this constraint.

## Lineplots

When this switch is on, TPS will draw a line trace between successive points on the plots giving it a solid feel. When off, TPS will only plot the pixels. When starting new plots, always switch the lineplot on as it may be hard to discern the plot points otherwise - they may be off the trace or just difficult to see, i.e. sparsely scattered.

## Forward and Backward plot switches

These switches are only valid for lineplots. If the `lineplot` switch is off, you will hear a beep when you try and switch the `Forward` or `Backward` options on.

When the `Forward` switch is on, TPS will only draw a line between two consecutive points if the trace moves to the right. The reverse is true for a `Backward` trace (rare if not non-existent). This avoids making a mess of the plot, particularly when a plot refreshes and, as sometimes happens (if the forward plot switch is off), that a trace gets redrawn from the right hand border to the origin (left hand border), i.e. in the reverse direction to the normal left to right trace evolution.

Quit TPS now and reply no to all save prompts. It is harmless if you accidentally save the changes as page 99 was not a pre-supplied page and can be modified at will.

## 2.13 Recording and Replaying Packet Data

Start this section by re-starting TPS, don't save any changes when prompted.

TPS can record incoming data and replay either previously recorded data or, indeed, any packet data generated by another application providing that the data conforms to the packet format you have configured.

Before continuing, configure TPS as follows:

Scroll down page 10 to reveal the green-background 'File I/O' statistics block (in amongst the green serial I/O statistics blocks).

Unlock the display.

Set the packet format, as shown in section 2.10, to a packet length of 192 bytes with a synchronisation bit pattern of 0xaabbccdd.

Switch the test data on

### To record data

Select the Record option from the TPS File menu item.

Select the Raw bin data format option from the pop-up Record menu, and then enter TEST.DAT as the filename to where the data will be stored.

Now wait for about a minute and, during this time, observe that the statistic BYTESWRF (Bytes written to file) is incrementing rapidly. This statistic shows the current recorded file size. After it has reached approximately 100Kbytes, switch off the recording by selecting the same menu items as used to start the record (File, Record, Raw bin).

The recorded file will be closed upon switch off and can then be replayed. Before replaying, switch off any incoming data, i.e. switch off the test data. TPS can only replay if there is no other incoming data. If you don't, you will be prompted to do this when you replay the pre-recorded file.

Now reset all statistics - press CTRL R and press OK. You should see all statistics return to 0.

### To replay data

Select the `File`, `Replay`, `Raw bin` menu item and then enter the filename of the previously recorded data, namely, `TEST.DAT`. Note, you will see the prior recorded file has been automatically appended to the `File` menu - you can just select this for quick replay. TPS considers the addition of file names to the `File` menu a system change - you will therefore be prompted to save the system settings, upon exit, whenever a file is recorded or replayed if the file-list changes in any way.

Upon entering the filename, you will see TPS rapidly update the statistics. In fact, TPS will be replaying one packet every 10 milliseconds.

To temporarily suspend the replay, press the Escape key 'ESC'. TPS enters 'Step mode' - this is accompanied by a periodic beeping tone (select `Options`, `Switches`, `Sound` to turn the sound off) . If you now press ENTER, TPS will replay a single packet, pressing ENTER again will read another packet and so forth. Press ESC a second time to toggle the step-mode state to off.

To slow down the replay, when not in step-mode, press the minus key '-' repeatedly until you see 'the little arrow at the bottom right of the screen' slow to an almost ticking pace. You can speed up the replay by pressing the plus key '+' again.

The plus and minus keys affect the TPS loop-time (loop-time parameter, top right on page 12). The default is set at 10 ms and so a single packet is replayed every 10 ms. Notice it halves and doubles when you press the plus and minus keys respectively. The loop-time controls the heart-beat of TPS. At its default of 10 ms (100 Hz), TPS ticks unnoticeably (like the flashing of a 50 Hz fluorescent light tube). When you slow it down to a value of 1000 ms, i.e 1 Hz, TPS runs noticeably slower - everything is affected, not just file replay and record. So, whilst it is useful to adjust the loop-time for replaying packets at a more sedate rate, don't set it too low (say, greater than 1 second) when up and running on a live system. TPS always loads with a default of 10 ms which you cannot change.

To gain experience of file replay and the other formats available, i.e. ASCII hex and compressed binary, we suggest you experiment.

## 2.14 Recording Parameters

This section does not require TPS to be running.

TPS can record individual parameters displayed on the current page. To start a parameter recording, double click over its view-port and then click the `Record` button in the pop-up dialog box that appears. A filename dialog box appears with the parameter mnemonic (first 8 characters) used as the root record filename. The file extension defaults to the standard 'TXT' as the recorded file is in plain text format that can be viewed with a text editor. You are free to enter a different filename. Pressing OK will start the recording.

Note, if the `Mnemonic` field of the view-port is blank, TPS will use the `Title`, if this is blank then TPS will use the `Units` field. Note, it is generally not possible to configure a view port with all three fields blank but view-ports on system pages 0 and 9999 can be configured offline in this way.

TPS can, in principle, record every parameter displayed. Practically, however, the number is limited by the host's operating system files limit - usually between 10 to 50 for Windows 3.1.

To see all the parameters currently being recorded, select the `Options`, `Info` menu item and press the `Record` button.

Any illegal filename characters embedded in the mnemonic are replaced with an underscore.

Unlike packet recording and playback, TPS cannot directly playback individual parameters because parameter record files are in plain text format and not stored as packets. This is so they can be easily imported into spreadsheet programs for more detailed and better graphical analysis. In fact, the TPS Toolkit utility COLTOPKT (Column to Packets) can convert the files to packets for replay on TPS. See the TPS Toolkit Manual on the Documentation disk.

### Parameter Record Files

Parameter record files are plain text files, readable with a text editor. The first line of the file is the standard TPS two-line file header giving the file open time and date, e.g.,

```
Filename = BYTESRD.TXT, Opentime = Wed Mar 18 13:45:01 1996  
Clockstart = 827143290000 ms
```

(taken from the recording of parameter BYTESRD on page 10). Following this are two columns with one or more lines of recorded data. The second column actually contains the parameter being recorded.



A typical record file may look something like:

```
Filename = VOLTS, Opentime = Fri Aug 21 10:50:20 1992  
Clockstart = nnnnnnnn0000 ms
```

130	0.310
131	0.311
132	0.310
133	0.310
134	0.315
135	0.310
136	0.314
137	0.313
138	0.311

The data is recorded in (x,y) data pairs, the first column is the x parameter which is termed the 'tag' parameter in TPS jargon. This tag parameter is the same for all parameter record files and is set, by default, to the parameter PKTSRD. This is the number of packets read since either TPS was loaded or since the last statistics reset - a copy of PKTSRD is also displayed on page 10. You can actually change this tag parameter to ANY parameter you wish. The tag parameter is defined in display page file 0, however, page 0 is a system page and can only be changed offline - search for the keyword 'tag' in the online help for full details on how to change the tag parameter.

In the above file extract it is seen that the recording was started on Friday, August 21st, 1992, at 10:50:20. A sample, with a value of 0.311, was recorded when PKTSRD was 130 and then the second sample, with a value of 0.311, was recorded in the next packet, PKTSRD = 131.

If you change a page whilst recording a parameter, without first switching the recording off, you will be prompted as to whether you wish to save the page. This is because recording information is stored with the page such that, when you load to it, TPS immediately resumes recording where it left off - identical action applies to alarms.

## 2.15 Printing the Display Window

This section does not require TPS to be running.

To print the screen, press `CTRL P` or select the `File, Print` menu item.

A first-level TPS dialog box appears offering the following options:

Text	Print only the text (plots will not be printed).
Graphics	(default) Print a bitmap.
Scale Factor	Reduce the image. A value of 100% denotes the normal size, 50% is half-size etc. Valid only with the <code>Graphics</code> option. Enlargement is not currently possible.
Print to File	Select if you wish the output to be redirected to a file. You will then be further prompted for a filename. Text output goes to a file readable with a text editor thus, it will not contain graphical plots. Graphics files are of the Windows Paintbrush bitmap format, i.e. those with a <code>'.BMP'</code> file extension.

### **! Important !**

**Graphics bitmaps can only be printed to a file using the TPS for Windows NT/95 model, it is not possible to print graphics to a file using TPS for Windows 3.1. You will be prompted when this is the case.**

Once the above options have been selected, TPS displays the standard Windows print dialog box. Thereafter, Windows takes control of everything.

We recommend you use landscape printing rather than portrait as the screen scaled to full size (100%) will not always fit on an A4 sheet. For output on a 9 pin dot matrix, anything less than 100% scale factor will not be very readable - some printers will not print anything but just scroll the paper. This is a printer limitation.

The default output is a color graphics bit-map sent to the printer. On most printers, this is reasonably fast (20 secs) and is suitable for presentation output.

For very quick printer output, when only the parameters are of interest, switch the display to monochrome prior to printing and then print with the `Text` option.

## 2.16 Setting Alarms

Start this section by re-starting TPS, don't save any changes when prompted.

Every parameter on the displayed page can be monitored such that if it falls outside (or inside) of a specified range, an optional alarm can be sounded, the display value blinked, the incoming data suspended and the defaulting value logged to disk.

To set an alarm on a parameter, double click a parameter view-port with the display lock off, and click the Alarm button on the parameter options dialog box. This will pop-up the parameter alarm dialog box as shown in figure 2.16.

For test purposes, select a numeric parameter on the current page that is NOT changing, We suggest you use the `Loop-time` parameter on page 10. This has a default of 10 ms.

Enter 20 in the `Lower Limit` edit box

Enter 30 in the `Upper Limit` edit box

Then click the `Blink` and `Beep` check boxes. Leave the others off for now.

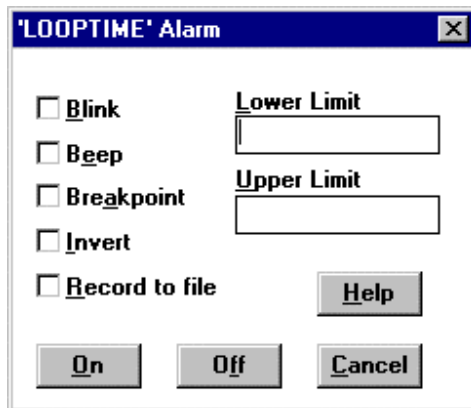
Click the `On` button to switch the alarm on.

You should now hear the alarm raised. Notice that the loop-time background color has switched to red, the value is blinking, and a beep tone can be heard approximately every second.

The alarm is raised because the loop-time parameter's 10 ms value is outside of the limits 20 to 30 (inclusive).

Now re-enter the alarm dialog box - double click on the view-port as before. This will show the previous settings. Without switching the alarm off, re-enter both upper and lower limits as 5 and 20. Now, after pressing ON, the value should be within the limits  $5 \leq \text{loop-time} \leq 20$  and, hence, the parameter display settles back to its previous colors.

Press the minus key '-' a couple of times and the loop-time should once again go out of limit with a value of 40. Press the plus key '+' twice to restore it to within limit.

**Fig 2.16 loop-time Alarm dialog box**

All limits should be entered in the format as specified by the parameter's `Format` identifier given in the view-port settings dialog box. For example, if you specify a `double` format then you must enter the limits in the `double` format. This requirement is not quite as strict as it sounds and entering an integer value such as 20, for a double value of 2.0e+01, is acceptable although the converse isn't.

It is acceptable to make the lower and upper limits identical, in which case, the alarm will be raised whenever the parameter does not equal the single limit value.

Note, you can also set alarms on all data types and not just numbers. For example, using the `Test Data` parameter on page 10, (currently displays the text `OFF`) you can set an alarm with the text `OFF` entered for both Lower and Upper limits. Whenever you start the Test data (see the `Transfer, Test data` menu item), the text will change to `ON` and, consequently, an alarm will be raised.

You can set alarms on several displayed parameters. To see all the alarms currently set, select the `Options, Info` menu item and press the `Alarms` button.

### Other Alarms options

If you wish the parameter to be recorded to disk whenever it exceeds its limits, click the `Record to File` check box in the alarm dialog box. You will be additionally prompted to enter a filename once the `On` button is pressed. Every time a value goes out of range it is recorded to disk. All recording aspects are identical to parameter recording as detailed in the previous section.

You can raise an alarm when the value falls inside a range, as opposed to the default outside range condition, by checking the `Invert` check box.

As soon as an alarm is raised, the incoming data can be temporarily halted by checking the `Breakpoint` dialog box. TPS actually enters step mode when an alarm is raised with the breakpoint option enabled.

Note, the beep tone is deliberately irritating so as to provoke immediate operator attention!

If you change a page whilst alarm monitoring a parameter, without first switching the alarm off, you will be prompted as to whether you wish to save the page. This is because the alarm settings are stored with the page such that, when you load to it, TPS immediately resumes monitoring where it left off - identical action applies to recording parameters.

## 2.17 Selective Packet Replay 'Filtering Packets'

This section does not require TPS to be running.

Lastly, similar to an alarm, TPS can selectively process ('filter') packets according to whether a displayed parameter in the packet has an alarm raised. The packets can be either incoming in real-time, or replayed from a disk file.

To filter incoming data, select the `Transfer, Filter pkts` menu item. A drop down dialog box appears from where you can select a parameter to be used as the filter. Once selected, a second dialog box appears - you will notice it is basically a cut-down version of the alarm dialog box with just two edit boxes where you can enter the lower and upper limits in exactly the same fashion as an alarm. The invert check box also performs the same function as that for an alarm.

When the filter limits are entered, every time a packet is received which has a parameter value outside of the range, the packet will be rejected with no further processing. If the invert option is on, the converse is true.

Unlike alarms, only one parameter on a page can be used to filter packets.

To see the current filter parameter, select the `Options, Info` menu item and press the `Filter` button.

The filter setting is saved with the page and restored immediately the page is viewed again. Thus, different pages can each have a different filter.

## 2.18 Where to go next

Firstly, if you want to quickly view your data, configure the packet format as in section 2.10 and view Samples disk pages 25-30 and pages 65-68 when reading your data. See the README.TXT on the Samples disk for more information on these and other sample pages. Use the sample pages as templates for your own view-ports.

We suggest you experiment with TPS and use the online help for more detailed information. The online help contains both reference and descriptive information.