Abstract/Purpose/Benefit

This document contains information about using T.38 fax relay implemented in Telogy Software to support fax call operation, analysis, and debugging. Its primary purpose is to provide detailed instructions for configuring T.38 fax relay, and for using built-in debug tools along with external devices and software to analyze fax relay calls, in order to help troubleshoot problems and resolve operational issues. It is intended for use by Texas Instruments applications support personnel, but may be utilized by customers as well.

This application note applies to:
- Systems using Telogy Software for T.38 fax relay operation
- High Density gateways in particular, but may be applied to other product verticals (example scripts are supplied for DIM test scaffold operation)
- Release 11 or later

Contents

1 Introduction ................................................................. 4
  1.1 Abbreviations ............................................................. 4
  1.2 Basic G3 FAX call ......................................................... 5
  1.3 FAX Call Modes ......................................................... 6
    1.3.1 V.VBD mode ......................................................... 6
    1.3.2 FAX relay mode ...................................................... 6
    1.3.3 V.VBD – T38 Interoperation over packet to packet ................. 7
  1.4 V.34 high-speed fax calls ........................................... 8
    1.4.1 V.VBD (pass-through with redundancy) ......................... 8
    1.4.2 T.38 fallback between two SG3 .................................. 8
  1.5 FAX call between SG3 and G3 ........................................ 9
    1.5.1 The calling fax device is V.34 capable and the answering fax device is not 9
    1.5.2 The answering device is V.34 capable and the calling fax device is not .... 9
  2 FAX Data For Analysis .................................................. 10
  3 FAX Configuration and Statistics ...................................... 11
    3.1 Fax Configuration ..................................................... 11
    3.2 Fax Statistics ........................................................ 13
      3.2.1 Fax Modem Statistics ........................................ 13
      3.2.2 Fax Relay Statistics ........................................... 14
      3.2.3 Fax Call Statistics ............................................ 16
  4 FAX Debug Trace Analysis ............................................. 18
    4.1 Debug Trace Generation .......................................... 18

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this document.
List of Figures

- Figure 1: Fax call diagram .......................................................... 5
- Figure 2: Fax In VBD Mode ......................................................... 6
- Figure 3: Fax In T.38 Relay Mode .............................................. 6
- Figure 4: T.38 - VBD Interoperation ....................................... 7
- Figure 5: T.38 - VBD Test Set-up ............................................ 8
- Figure 6: T.38 Decode Selection .............................................. 19
- Figure 7: T.38 Graphical Analysis .......................................... 20
- Figure 8: Wireshark PCM Extraction ................................... 23
- Figure 9: PCM Analysis .......................................................... 23
- Figure 10: T.38 In Wireshark ................................................. 23

List of Tables

- Table 1: Abbreviations Used In This Document .................. 4
- Table 2: Fax Modem Statistics Detail ................................. 13
- Table 3: Fax Relay Statistics Detail ................................. 14
- Table 4: Fax Call Statistics Detail ................................. 16
- Table 5: T.38 Call Graphical Analysis ............................ 20
1 Introduction

This section provides a brief description of the basic elements of a fax call, and discusses the different modes of operation for supporting fax calls over a packet network. For further information on fax operation, refer to Appendix A - References on page 24.

Since the HDGW (High Density Gateway) products emphasized here provide DSP access at the DIM (DSP Interface Module) API layer, higher layer application processes for call set-up and signaling will not be discussed. This document assumes that the reader has sufficient knowledge of fax requirements and the Telogy Software to implement fax detection during a voice call, followed by appropriate DSP channel configuration for the desired processing mode.

1.1 Abbreviations

Table 1 Abbreviations Used In This Document (Part 1 of 2)

<table>
<thead>
<tr>
<th>Abbreviations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CED</td>
<td>Called terminal identification</td>
</tr>
<tr>
<td>CI</td>
<td>Call Indicator (V.34)</td>
</tr>
<tr>
<td>CM</td>
<td>Call Menu (V.34)</td>
</tr>
<tr>
<td>CONF</td>
<td>Conferencing unit (DSP module)</td>
</tr>
<tr>
<td>CRP</td>
<td>Command repeat (T.30)</td>
</tr>
<tr>
<td>DCN</td>
<td>Disconnect (T.30)</td>
</tr>
<tr>
<td>DCS</td>
<td>Digital command signal (T.30)</td>
</tr>
<tr>
<td>DIS</td>
<td>Digital identification signal (T.30)</td>
</tr>
<tr>
<td>DTC</td>
<td>Digital transmit command (T.30)</td>
</tr>
<tr>
<td>ECM</td>
<td>Error Correction Mode</td>
</tr>
<tr>
<td>ECU</td>
<td>Echo cancellation unit (DSP module)</td>
</tr>
<tr>
<td>EOP</td>
<td>End of page (T.30)</td>
</tr>
<tr>
<td>FCD</td>
<td>Facsimile coded data</td>
</tr>
<tr>
<td>FCF</td>
<td>Facsimile control field</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>FIF</td>
<td>Facsimile information field</td>
</tr>
<tr>
<td>FIU</td>
<td>Fax interface unit (DSP module)</td>
</tr>
<tr>
<td>FM</td>
<td>Fax modem (DSP module)</td>
</tr>
<tr>
<td>FSK</td>
<td>Frequency shift keyed</td>
</tr>
<tr>
<td>FTE</td>
<td>Fax Terminal Equipment</td>
</tr>
<tr>
<td>IFP</td>
<td>Internet Facsimile Protocol</td>
</tr>
<tr>
<td>IFT</td>
<td>Internet Facsimile Transfer</td>
</tr>
<tr>
<td>MCF</td>
<td>Message confirmation (T.30)</td>
</tr>
<tr>
<td>MPS</td>
<td>Multi-page signal (T.30)</td>
</tr>
<tr>
<td>NEU</td>
<td>Network encapsulation unit (DSP module)</td>
</tr>
<tr>
<td>PIU</td>
<td>PCM interface unit (DSP module)</td>
</tr>
<tr>
<td>PPS</td>
<td>Partial page signal (T.30)</td>
</tr>
<tr>
<td>RTN</td>
<td>Retrain negative (T.30)</td>
</tr>
<tr>
<td>RTP</td>
<td>Retrain positive (T.30)</td>
</tr>
<tr>
<td>TCF</td>
<td>Training check (T.30)</td>
</tr>
</tbody>
</table>
1.2 Basic G3 FAX call

The detailed steps of a fax call following the T.30 protocol can have considerable variation, depending upon a variety of options. A basic fax call between two Group 3 (G3) fax machines is illustrated in Figure 1 “Fax call diagram” on page 4.

![Fax call diagram](image-url)
1.3 FAX Call Modes

A Fax call can be handled by a packet gateway in different modes:
1. V.VBD mode
2. FAX relay mode
3. V.VBD – T38 Interoperation over packet to packet

1.3.1 V.VBD mode

V.VBD (Voice Band Data) mode directly simulates the PSTN by passing all fax signals and data through a G.711 voice channel. The channel must be configured to avoid interference with this type of non-voice transmission. The essential configuration elements to be addressed are:

- Echo cancellation - disabled, or enabled with NLP disabled
- VAD (voice Activity Detection) - disabled
- DC offset removal - disabled
- Signal limiter - disabled
- Voice playout unit - fixed delay configuration, with packet loss recovery disabled

For more detailed information, please refer to the document VBD User’s Guide regarding how to set-up V.VBD mode.

The primary DSP processing elements involved in a V.VBD call are shown in Figure 2.

![Figure 2 Fax In VBD Mode](image)

1.3.2 FAX relay mode

T.38 fax relay mode may be used to provide a fax-over-packet method which conserves bandwidth and enhances reliability in comparison with V.VBD. It is shown in Figure 3.

![Figure 3 Fax In T.38 Relay Mode](image)

Fax relay mode in the DSP can be entered either from either idle mode or from a voice mode. The following list of fax relay parameters presents a typical set of values for T.38 fax relay operation:

Fax Relay Configuration Example:

Rate .................................................. 14400
Fax Playout Nominal Delay (in msec) ................. 300
Packet Size (in bytes) .................................. 30
TX Level (in dBm) ....................................... -10
Carrier Detect Threshold (in dBm) ..................... -33
No Activity Timeout (in sec) .......................... 20
Debug Level ........................................... 0
High-Speed Packet Rate (T.38 only) ................. 20
Low-Speed Redundancy ................................. 3
High-Speed Redundancy ................................. 0
TCF Handling Method for T.38 ......................... 1
Max Low Speed Data Packetization for T.38 (in bytes) .. 1
TX Network Timeout (in sec) .......................... 150
Eflag Start Timer (in msec) ........................... 2600
Eflag Stop Timer (in msec) ............................ 2300
Cleardown Timer (in msec) ............................. 1000
Valid Options ......................................... none
Options ............................................... none
T30 ECM ............................................... disable
T30 MR Page Compression ............................... disable
NSF Country Code ...................................... 0
NSF Vendor Code ....................................... 0
T.38 Version .......................................... 0

For a detailed description of these parameters, refer to section 3.1 “Fax Configuration” on page 10.

1.3.3 V.VBD – T38 Interoperation over packet to packet
Packet-to-packet operation between V.VBD and T.38 operation (Figure 4) is a relatively recent feature implemented in HDGW R11.2. Please check with your applications support engineer to verify this capability is supported in your DSP build. This capability is useful when T.38 operation is supported on only one leg of a call.

Figure 4  T.38 - VBD Interoperation

![Diagram showing VBD and T.38 interoperation]
A simplified configuration for testing this mode of operation is shown in Figure 5.

Figure 5  T.38 - VBD Test Set-up

A test script for use with the DIM Test scaffold, and the resulting DIM spy trace, are shown in section 7 “Appendix B - V.VBD/T.38 Interoperation” on page 25.

1.4 V.34 high-speed fax calls

Current High Density product releases do not support V.34 (Super G3, or SG3) fax relay through T.38. V.34 fax calls can be supported by either of two methods:

- V.VBD (pass-through with redundancy)
- T.38 fallback between two SG3

1.4.1 V.VBD (pass-through with redundancy)

V.VBD was implemented primarily to provide redundancy support (RFC2198 and RFC2733 FEC) for modem operations in voice pass-through mode. Thus you achieve redundancy that is as good as T.38, or even better (with FEC), but at the cost of a full 64Kbps bandwidth plus redundancy.

1.4.2 T.38 fallback between two SG3

The other option has been called SG3 force to V.34 fax spoofing. This processing method depends upon the ability for the DSP to detect and suppress CM and CI tones in the V.34 setup exchange. Essentially, CM and CI are used by V.34 faxes to recognize that they ARE V.34 - so if we detect and then suppress CM and CI, the V.34 fax machines will not realize both ends are V.34, and will relax to a lower speed that we can then handle with T.38 which we called T.38 force to G3 (spoofing).

1.5 FAX call between SG3 and G3

1.5.1 The calling fax device is V.34 capable and the answering fax device is not

In this case the answer fax device will answer the call as a legacy G3 with answer tone followed by V.21 preamble. The calling fax device will see the standard answer tone (not ANSsam) and assume the answering device is not V.34 capable and simply wait for the V.21 information to follow the answer tone.
1.5.2 The answering device is V.34 capable and the calling fax device is not

In this case, the answering fax device will send an ANSam answer tone and wait to see if the calling fax device responds with CM. Since the calling fax device is not V.34 capable, it will not send a CM and the answering device will timeout and fall back to legacy G3 start-up procedures – e.g., send V.21 control signals.
2  FAX Data For Analysis

There are several types of information available for analysis of a fax call:

- Channel configuration parameters from a spy trace ("FAX Configuration and Statistics" on page 10)
- Fax statistics reported by the DSP ("FAX Configuration and Statistics" on page 10)
- Fax debug trace data reported by the DSP ("FAX Debug Trace Analysis" on page 17)
- Network packet capture ("Using A Packet Capture" on page 18)

The following sections of the document will discuss how to use these data for fax call analysis.
3 FAX Configuration and Statistics

To analyze a fax relay call, a basic understanding may be gained by checking the channel configuration that was used, and reviewing the processing statistics provided by the DSP. These elements are defined in the following sections:

- 3.1 “Fax Configuration” below
- 3.2 “Fax Statistics” on page 12

3.1 Fax Configuration

The structure MGB_MDS_FAX_RELAY_PROTO_CONFIG defined in file mgmicdsp.h contains the following parameters to configure a channel for T.38 fax relay operation:

- **timeout_duration** - no activity timeout duration in sec. The valid range is from 10 to 32000 seconds. The recommended value is 20. A timeout is triggered when no activity is detected from either the network or the local FTE for the specified period of time.

- **debug_level** - fax relay debug level. The valid values are 0-2. The debug trace messages will be sent to the host if debug_level is set to non-zero. For more details, please look at Section Debug Trace Generation.

- **fax_nominal_delay** – fax nominal delay in ms. The valid range is from 0 to 500ms. The typical setting is 200-300. Delay is introduced in the receive path to compensate for network jitter.

- **fax_max_rate** – maximum fax data rate in unit of 100bps (i.e. 144 for 14400bps). This parameter applies only to G3 fax relay. To limit the data rate, we first try to modify supported modulations that are specified in the DIS and then corrupt TCF sequence by sending 0xA5 pattern (TCF sequence is supposed to be all 0’s) if necessary to achieve the desired data rate. For example, if the maximum fax data rate is set to 12kbps, since the DIS specifies only the modulation (12kbps is still one of the data rates in V.17), the data rate of 12kbps can only be achieved by corrupting the TCF sequence, forcing down train.

- **hs_pkt_rate** - size of high-speed data in primary IFP in ms. This parameter applies only to T.38 UDPTL and T.38 RTP protocols. The valid values are: 10, 20, 30, and 40. The recommended value is 20.

- **ls_redundancy** – level of low-speed data redundancy. This parameter applies only to T.38 UDPTL and T.38 RTP protocols. The valid range is from 0 to 8. The typical setting is 5.

- **hs_redundancy** – level of high-speed data redundancy. This parameter applies only to T.38 UDPTL and T.38 RTP protocols. The valid range is from 0-2 (maybe 3 in certain builds). The typical setting is 2.

- **TCF_method** – method of handling TCF for T.38. There are two possible values: 1 for locally checking (on the originating side) or generating (on terminating side) TCF, or 2 for sending TCF over packet network. Please note that this parameter needs to be negotiated via signaling or agreed upon between two gateways.
• **T38_Version** – T.38 version to be used. There are four possible values:
  - 0 for 1998 ASN.1 syntax, org doc: 06-1998
  - 1 for 1998 ASN.1 syntax, org doc: 11-2000
  - 2 for 2002 ASN.1 syntax, org doc: 03-2002
  - 3 for 2002 ASN.1 syntax extended, org doc: 04-2004

• **max_ls_packetization** – size of low-speed data in primary IFP in bytes. This parameter applies only to T.38 UDP and T.38 RTP protocols. The typical setting is 1.

• **tx_network_timeout** – transmit network timeout duration in sec. The valid range is from 10 to 32000. The recommended value is 150. A timeout is triggered if no data is received from the network for the specified period of time while the fax relay protocol is in a state expecting data from the remote.

• **eflag_start_timer** – The recommended value is 2600ms. The timer is used to determine when to start sending flags for the specified period of time while the fax relay protocol is in a state expecting data from the remote. Please note that to disable the eflag feature, the value can be set very high.

• **eflag_stop_timer** – The recommend value is 2300ms. After eflags are generated, if a network response has not been received by the duration of the Eflag stop timer, then the fax relay protocol will stop waiting and terminate the signal transmission by sending a valid CRP to the local FTE.

• **cleardown_timer** - the time duration (in ms) to wait after a DCN is received from the local FTE before reporting the protocol termination to the application. Please note that this timer does not apply to the situation when DCN is received from the remote FTE. The valid range is from 0 to 5000. The typical setting is 2000. The DSP timer resolution is 10 ms for G3 fax relay.

• **faxr_feature_selector** - fax relay feature selector

• **fiu_FEATURE_SLF_ENABLE** - Enable scan line fix-up feature

• **fiu_FEATURE_DIS_EFLAG_DISABLE** - Disable Eflags for first DIS. Some of the fax machines may require 2 DIS frames to be received.

• **fiu_FEATURE_TFOP_FC_REPEAT** - Enable repeating of Frame Complete packet over the packet network for improved performance.

• **fiu_FEATURE_NSF_OVERRIDE** - Allow user to specify their own NSF fields.

• **fiu_FEATURE_RTP_TS_PROC_DISABLE** - Disable the use of T.38 RTP timestamps for playout.

• **fiu_FEATURE_AAL2_CID_CHK_DISABLE** - Enable or disables checking of CID in AAL2 Rx.

• **fiu_FEATURE_MDM_TRN_FAIL_COMP** - Enable feature to compensate for modem training failures in high speed.

• **fiu_FEATURE_V34_FAX** - not currently supported.

• **fiu_FEATURE_T38_VOCALTEC_INTEROP** - Enable the workaround for interop with VocalTec T.38 gateways that cannot properly process reception of HS packets with t4-non-ecm-sig-end indications appended to them.

• **t30_feature_selector** - T.30 feature selectors
• **fiu_T30FEATURE_ECM_DISABLE**: Disable the ECM mode by modifying the DIS if the terminating fax machine supports ECM. Please note that if the feature is disabled, no action is taken by the fax relay processing (if one of the fax machines does not support ECM, the fax call is still going to be non-ECM).

• **fiu_T30FEATURE_MR_DISABLE**: Disable the MR compression by modifying the DIS if the terminating fax machine support MR compression. Please note that if the feature is disabled, no action is taken by the fax relay processing. Also, this feature only applies to G3 fax relay since ECM mode is required for V.34 fax as specified in the standard.

• **fiu_T30FEATURE_V34DISABLE**: Disable V.34 fax relay processing.

• **nsf_country_code** and **nsf_vendor_code** - In our current implementation, we always override the country code and vendor code. By default, the country code is overridden with U.S. country code and the vendor code is overridden with Telogy vendor code; however, if the option in the fax relay feature selector is enabled, we will override with the values of **nsf_country_code** and **nsf_vendor_code** specified.

### 3.2 Fax Statistics

There are three different sets of DSP statistics uniquely relevant to fax relay:

- 3.2.1 “Fax Modem Statistics” below
- 3.2.2 “Fax Relay Statistics” on page 13
- 3.2.3 “Fax Call Statistics” on page 15

#### 3.2.1 Fax Modem Statistics

These statistics are descriptive of the Fax Modem Unit (FM) which interfaces with the local fax machine, and represent a snapshot of conditions at the time they are requested.

**Table 2** Fax Modem Statistics Detail (Part 1 of 2)

<table>
<thead>
<tr>
<th>Pos (W)</th>
<th>Size (bits)</th>
<th>FM</th>
<th>FIU</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>0</td>
<td>16</td>
<td>x</td>
<td>Current fax modem status. The 8 MSBs contain the modulation and direction information. The 8 LSBs contain the current state in the state machine. This information is for internal debugging purposes.</td>
</tr>
<tr>
<td>rx_level</td>
<td>1</td>
<td>16</td>
<td>x</td>
<td>Receive power level in the unit of -dBm.</td>
</tr>
<tr>
<td>rx_rate</td>
<td>2</td>
<td>16</td>
<td>x</td>
<td>Receive data rate. The possible values are: 2400/4800/9600/12000/14400 bps and 2. In the case where the value is 2, the receive data rate is 300 bps for V.21 Channel 2 modulation which is used for sending control frames during T.30 handshake.</td>
</tr>
<tr>
<td>tx_level</td>
<td>3</td>
<td>16</td>
<td>x</td>
<td>Transmit power level in the unit of -dBm.</td>
</tr>
<tr>
<td>tx_rate</td>
<td>4</td>
<td>16</td>
<td>x</td>
<td>Transmit data rate. The possible values are: 2400/4800/9600/12000/14400 bps and 2. In the case where the value is 2, the transmit data rate is 300 bps for V.21 Channel 2 modulation which is used for sending control frames during T.30 handshake.</td>
</tr>
<tr>
<td>snr</td>
<td>5</td>
<td>16</td>
<td>x</td>
<td>Signal-to-noise ratio in the unit of dB. This is computed only in a high-speed receive data mode.</td>
</tr>
<tr>
<td>car_freq_offset</td>
<td>6</td>
<td>16</td>
<td>x</td>
<td>Carrier frequency offset in 0.1 Hz (i.e. if car_freq_offset is 100, the carrier frequency offset is 10 Hz). This is computed only in a high-speed receive data mode.</td>
</tr>
</tbody>
</table>
### 3.2.2 Fax Relay Statistics

These statistics describe detailed operation of the Fax Interface Unit (FIU) relative to the packet network.

**Table 2**  
Fax Modem Statistics Detail (Part 2 of 2)

<table>
<thead>
<tr>
<th>Pos (W)</th>
<th>Size (bits)</th>
<th>FM</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>timing_offset</td>
<td>7 16 x</td>
<td></td>
<td>Timing offset in 0.0001% (i.e. if timing_offset is 28, the timing offset is 28e-4%). This is computed only in a high-speed receive data mode.</td>
</tr>
<tr>
<td>HDLC_Tx_Underflow</td>
<td>8 16 x</td>
<td></td>
<td>Transmit buffer underflow counter during HDLC data transmission. This counter is incremented for every byte of 0’s that is filled. If there is an underflow, the HDLC frame in progress will be appended with a bad CRC.</td>
</tr>
<tr>
<td>Non_HDLC_Tx_Underflow</td>
<td>9 16 x</td>
<td></td>
<td>Transmit buffer underflow counter during non-HDLC data transmission. This counter is incremented for every byte of data that is filled.</td>
</tr>
<tr>
<td>Bad_CRC_From_Tele</td>
<td>10 16 x</td>
<td></td>
<td>Counter for the number of bad CRC frames received from telephony.</td>
</tr>
<tr>
<td>Bad_CRC_To_Tele</td>
<td>11 16 x</td>
<td></td>
<td>Counter for the number of bad CRC frames sent to telephony.</td>
</tr>
<tr>
<td>Good_CRC_From_Tele_MSW</td>
<td>12 16 x</td>
<td></td>
<td>Upper 16 bits of the 32-bit counter for the number of good CRC frames received from telephony.</td>
</tr>
<tr>
<td>Good_CRC_From_Tele_LSW</td>
<td>13 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of good CRC frames received from telephony.</td>
</tr>
<tr>
<td>Good_CRC_To_Tele_MSW</td>
<td>14 16 x</td>
<td></td>
<td>Upper 16 bits of the 32-bit counter for the number of good CRC frames sent to telephony.</td>
</tr>
<tr>
<td>Good_CRC_To_Tele_LSW</td>
<td>15 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of good CRC frames sent to telephony.</td>
</tr>
<tr>
<td>Non_HDLC_From_Tele_MSW</td>
<td>16 16 x</td>
<td></td>
<td>Upper 16 bits of the 32-bit counter for the number of non-HDLC data bytes received from telephony.</td>
</tr>
<tr>
<td>Non_HDLC_From_Tele_LSW</td>
<td>17 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of non-HDLC data bytes received from telephony.</td>
</tr>
<tr>
<td>Non_HDLC_To_Tele_MSW</td>
<td>18 16 x</td>
<td></td>
<td>Upper 16 bits of the 32-bit counter for the number of non-HDLC data bytes sent to telephony.</td>
</tr>
<tr>
<td>Non_HDLC_To_Tele_LSW</td>
<td>19 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of non-HDLC data bytes sent to telephony.</td>
</tr>
</tbody>
</table>

**Table 3**  
Fax Relay Statistics Detail (Part 1 of 2)

<table>
<thead>
<tr>
<th>Pos (W)</th>
<th>Size (bits)</th>
<th>FM</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>num_tx_pkts</td>
<td>0 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of packets sent to network.</td>
</tr>
<tr>
<td>num_rx_pkts</td>
<td>1 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of packets received from network.</td>
</tr>
<tr>
<td>num_lost_pkts</td>
<td>2 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of packets lost from network. This counter is incremented based on the sequence number gap. For T.38, this is tracked after redundancy error recovery.</td>
</tr>
<tr>
<td>num_drop_pkts</td>
<td>3 16 x</td>
<td></td>
<td>Lower 16 bits of the 32-bit counter for the number of invalid packets received from network.</td>
</tr>
<tr>
<td>out_of_seq_cnt_MSW</td>
<td>4 16 x</td>
<td></td>
<td>Upper 16 bits of the 32-bit counter for the number of out-of-sequence packets from network. This is applicable only to Telogy-2 protocol.</td>
</tr>
</tbody>
</table>
Table 3  Fax Relay Statistics Detail (Part 2 of 2)

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>out_of_seq_cnt_LSW</td>
</tr>
<tr>
<td>debug1</td>
</tr>
<tr>
<td>debug2</td>
</tr>
<tr>
<td>debug3_MSW</td>
</tr>
<tr>
<td>debug3_LSW</td>
</tr>
<tr>
<td>debug4_MSW</td>
</tr>
<tr>
<td>debug4_LSW</td>
</tr>
<tr>
<td>num_tx_packets_MSW</td>
</tr>
<tr>
<td>num_rx_packets_MSW</td>
</tr>
<tr>
<td>num_lost_pkts_MSW</td>
</tr>
<tr>
<td>num_drop_pkts_MSW</td>
</tr>
<tr>
<td>num_tx_drop_pkts_MSW</td>
</tr>
<tr>
<td>num_tx_drop_pkts_LSW</td>
</tr>
<tr>
<td>Bytes_From_Net_MSW</td>
</tr>
<tr>
<td>Bytes_From_Net_LSW</td>
</tr>
<tr>
<td>Bytes_To_Net_MSW</td>
</tr>
<tr>
<td>Bytes_To_Net_LSW</td>
</tr>
</tbody>
</table>

End of Table 3
### 3.2.3 Fax Call Statistics

These statistics describe overall characteristics of the fax call.

#### Table 4 Fax Call Statistics Detail (Part 1 of 2)

<table>
<thead>
<tr>
<th>Pos (W)</th>
<th>Size (bits)</th>
<th>FM</th>
<th>FIU</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>modulation</td>
<td>0</td>
<td>16</td>
<td>x</td>
<td>Most recent high-speed modulation used</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 for V.27 2400 bps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2 for V.27 4800 bps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3 for V.29 7200 bps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4 for V.29 9600 bps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5 for V.17 7200 bps - short training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6 for V.17 9600 bps - short training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7 for V.17 12000 bps - short training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8 for V.17 14400 bps - short training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9 for V.17 7200 bps - long training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10 for V.17 9600 bps - long training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>11 for V.17 12000 bps - long training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12 for V.17 14400 bps - long training</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>13 for V.33 12000 bps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>14 for V.33 14400 bps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-1 for unknown</td>
</tr>
<tr>
<td>page_cnt</td>
<td>1</td>
<td>16</td>
<td>x</td>
<td>Number of complete pages transferred. In non-ECM mode, the page counter is</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>incremented upon the reception of MCF/RTP/RTN after each high-speed data</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>transfer from either direction. In ECM mode, the page counter is updated</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>upon the reception of PPS which contains the page counter information.</td>
</tr>
<tr>
<td>debug1</td>
<td>2</td>
<td>16</td>
<td>x</td>
<td>Maximum depth of send in queue. This is for internal debugging purposes.</td>
</tr>
<tr>
<td>debug2</td>
<td>3</td>
<td>16</td>
<td>x</td>
<td>Maximum depth of receive in queue. This is for internal debugging purposes.</td>
</tr>
<tr>
<td>debug3</td>
<td>4</td>
<td>16</td>
<td>x</td>
<td>Maximum usage of memory pool buffers. This is for internal debugging purposes.</td>
</tr>
<tr>
<td>debug4_MSW</td>
<td>5</td>
<td>16</td>
<td>x</td>
<td>Upper 16 bits of the 32-bit counter for receive out queue overflow. This is for internal debugging purposes.</td>
</tr>
<tr>
<td>debug4_LSW</td>
<td>6</td>
<td>16</td>
<td>x</td>
<td>Lower 16 bits of the 32-bit counter for receive out queue overflow. This is for internal debugging purposes.</td>
</tr>
<tr>
<td>debug5_MSW</td>
<td>7</td>
<td>16</td>
<td></td>
<td>Reserved: should always be zero.</td>
</tr>
<tr>
<td>debug5_LSW</td>
<td>8</td>
<td>16</td>
<td></td>
<td>Reserved: should always be zero.</td>
</tr>
<tr>
<td>Pos (W)</td>
<td>Size (bits)</td>
<td>FM</td>
<td>FIU</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>----</td>
<td>-----</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| call_status | 9 | 16 | x | Call status bitfield:  
Bit 0: ECM status (0 for non-ECM and 1 for ECM)  
Bit 1: call direction (0 for originating and 1 for answering) |
| Call_Success_Bitfield | 10 | 16 | x | Call success bitfield:  
Bit 15: EOP or PPS-EOP was received from either telephony or network  
Bit 14: MCF was received from either telephony or network after EOP/EOM (or the PPS-equivalent)  
Bit 13: EOM or PPS-EOM was received from either telephony or network  
Bit 12: DCN was received from telephony  
Bit 11: DCN was received from network  
Bit 10: DIS was received from either telephony or network after EOM  
Bits 9-0: unused (should always be zero)  
The call success bit field is reset every time a DCS is received. For example, in a normal fax call where a DCN is received from telephony after EOP/MCF is exchanged, bits 15, 14 and 12 would be set to 1 in the call success bit field. In the case where DCN is not detected, only bits 15 and 14 would be set 1. Note that the interpretation of call success is left to the system integrator. |

End of Table 4
4 FAX Debug Trace Analysis

Telogy Software can be configured to provide a debug trace of T.38 fax relay processing in the DSP, which can then be decoded using a TI proprietary parser utility. The decoded output provides a detailed step-by-step accounting of operations in the fax processing modules, which can be used by applications support personnel to analyze the fax call.

4.1 Debug Trace Generation

In DIM, the fax debug trace level can be set (see the debug_level parameter in section 3.1 “Fax Configuration” above):

- Level 0 - reports only fatal errors (normal setting)
- Level 1 - reports high-level protocol state information
- Level 2 – used for internal debugging purposes (parser input)

**Example 1 Fax Debug Level Configuration**
```
dimt set template 0 fax details dbg_level 2
```
End of Example 1

A fax debug trace record is comprised of 4 16-bit words:

- 0: Source of the debug trace. The upper 8 bits represent the module ID (i.e. 0x0A for FIU and 0x22 for FM) and the lower 8 bits represent the DSP channel number which is 1-based.
- 1: The message ID.
- 2: The supporting data for the corresponding message code.
- 3: The 16-bit system timestamp applicable only to debug level 2.

The debug trace data is sent from the DSP to the host, where it may be displayed through the Spy debug message facility (Example 2):

**Example 2 Fax Relay Debug Trace**
```
0000066461 - DIM: 0:*, Rx='DEBUG_TRACE'(81) Len=25, trans_id=0x0
0000066461 -   DSP 0, Record 0: 0x0a01, 0x0080, 0x0005, 0x706f
0000066461 -   DSP 0, Record 1: 0x0a01, 0x0073, 0x0002, 0x706f
0000066461 -   DSP 0, Record 2: 0x0a01, 0x0040, 0x0005, 0x706f
0000066461 -   DSP 0, Record 3: 0x0a01, 0x0001, 0x000a, 0x706f
0000066461 -   DSP 0, Record 4: 0x0a01, 0x0095, 0x000a, 0x706f
0000066477 - DIM: 0:*, Rx='DEBUG_TRACE'(81) Len=61, trans_id=0x0
0000066477 -   DSP 0, Record 5: 0x2201, 0x0063, 0x0000, 0x708a
0000066477 -   DSP 0, Record 6: 0x2201, 0x00c2, 0x0000, 0x708a
0000066477 -   DSP 0, Record 7: 0x2201, 0x00c3, 0x0000, 0x708a
0000066477 -   DSP 0, Record 8: 0x2201, 0x00c1, 0x0002, 0x708a
```
End of Example 2

Save the spy trace in text form to a log file for decoding and analysis by TI support.
5 Using A Packet Capture

Wireshark (formerly Ethereal) is a very popular tool for packet analysis. It can be used to display a summary of a T.38 call, or to capture PCM samples from a V.VBD call:

- 5.1 “Wireshark T.38 Graph Analysis” below
- 5.2 “Wireshark PCM Sample Capture” on page 21

5.1 Wireshark T.38 Graph Analysis

The recent releases of Wireshark include built-in T.38 packet analysis. The Graph Analysis feature presents a clear timing picture of the command and data exchange between two gateways handshaking on the network side. Note that the example here uses a capture of T.38 UDPTL packets, and not of RTP encapsulated packets.

Procedure 1  Wireshark T.38 Graphical Analysis

Step – Action

1. Open the packet capture of a T.38 call in Wireshark.
2. Click on “Analyze” (or right click on a packet) and then select “Decode As…” In the pop-up window, select T.38 for the transport protocol (Figure 6).

![Figure 6 T.38 Decode Selection]

3. Click on “Telephony” and then select “VoIP Calls”. In the pop-up window, select one T.38 media stream, and then select “Graph”.
4. In the pop-up “Graph Analysis” the window will show a history of the fax call (Figure 7 on page 19).
See Table 5 “T.38 Call Graphical Analysis” for an example “Graph Analysis” window display:

Table 5  T.38 Call Graphical Analysis (Part 1 of 3)

<table>
<thead>
<tr>
<th>Time</th>
<th>Source</th>
<th>Destination</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>192.168.1.2</td>
<td>192.168.1.3</td>
<td>no-signal t38:t30 Ind:no-signal</td>
</tr>
<tr>
<td>0.005</td>
<td>(32768)</td>
<td></td>
<td>no-signal t38:t30 Ind:no-signal</td>
</tr>
<tr>
<td>8.576</td>
<td>(32768)</td>
<td></td>
<td>cng t38:t30 Ind:cng</td>
</tr>
<tr>
<td>16.991</td>
<td>(3268)</td>
<td></td>
<td>ced t38:t30 Ind:ced</td>
</tr>
<tr>
<td>17.046</td>
<td>(3268)</td>
<td></td>
<td>no-signal t38:t30 Ind:no-signal</td>
</tr>
<tr>
<td>20.852</td>
<td>(3268)</td>
<td></td>
<td>v21-preamble t38:t30 Ind:v21-preamble</td>
</tr>
<tr>
<td>22.282</td>
<td>(3268)</td>
<td></td>
<td>NSF t38:v21:HDLC:Non-Standard Facilities</td>
</tr>
<tr>
<td>22.602</td>
<td>(3268)</td>
<td></td>
<td>DIS DSRITU-T-V.27, V.29, and V.17 t38:v21:HDLC:Digital Identification Signal</td>
</tr>
<tr>
<td>22.632</td>
<td>(3268)</td>
<td></td>
<td>no-signal t38:t30 Ind:no-signal</td>
</tr>
</tbody>
</table>
Table 5  T.38 Call Graphical Analysis (Part 2 of 3)

<table>
<thead>
<tr>
<th>Time</th>
<th>192.168.1.2</th>
<th>192.168.1.3</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>23.267</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>24.467</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>24.587</td>
<td>v17-14400-long-training</td>
<td>t38:t30 Ind:v17-14400-long-training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>26.147</td>
<td>t4-non-ecm-data:v17-14400</td>
<td>t38:t4-non-ecm-data:v17-14400 Duration: 1.50s No packet lost</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>27.657</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>28.342</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>29.322</td>
<td>CFR</td>
<td>t38:v21:HDLC:Confirmation To Receive</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>29.352</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>30.297</td>
<td>v17-14400-short-training</td>
<td>t38:t30 Ind:v17-14400-short-training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>30.608</td>
<td>t4-non-ecm-data:v17-14400</td>
<td>t38:t4-non-ecm-data:v17-14400 Duration: 19.00s No packet lost</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>49.619</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>49.879</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>50.889</td>
<td>MPS</td>
<td>t38:v21:HDLC:MultiPage Signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>50.909</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>51.574</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>52.584</td>
<td>MCF</td>
<td>t38:v21:HDLC:Message Confirmation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>52.614</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>54.669</td>
<td>v17-14400-short-training</td>
<td>t38:t30 Ind:v17-14400-short-training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>54.980</td>
<td>t4-non-ecm-data:v17-14400</td>
<td>t38:t4-non-ecm-data:v17-14400 Duration: 18.72s No packet lost</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>73.711</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>73.991</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(32768)</td>
<td>(32770)</td>
<td></td>
</tr>
</tbody>
</table>
5.2 Wireshark PCM Sample Capture

A G.711 packet capture can be useful for analyzing the switchover between T.38 or VBD.

Procedure 2 Extracting PCM From A Packet Capture With Wireshark

Step – Action
1. Open the capture in Wireshark (refer to Figure 8 on page 22).
2. Click on “Statistics” and then select “RTP” to “Show All Stream”.
3. In the pop-up window, select one PCM steam, and then click on the button for “Analyze”.
4. In the next pop-up window, click on the button for “Save payload…”
5. In the last pop-up window, select the raw “Format” and a forward “Channel”
6. The above three steps will need to be done for each direction.

Table 5 T.38 Call Graphical Analysis (Part 3 of 3)

<table>
<thead>
<tr>
<th>Time</th>
<th>192.168.1.2</th>
<th>192.168.1.3</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>74.981</td>
<td>EOP</td>
<td>t38:v21:HDLC:End Of Procedure</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>75.001</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>75.676</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>76.666</td>
<td>MCF</td>
<td>t38:v21:HDLC:Message Confirmation</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>76.696</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>77.381</td>
<td>v21-preamble</td>
<td>t38:t30 Ind:v21-preamble</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>78.341</td>
<td>DCN</td>
<td>t38:v21:HDLC:Disconnect</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
<tr>
<td>78.371</td>
<td>no-signal</td>
<td>t38:t30 Ind:no-signal</td>
<td></td>
</tr>
<tr>
<td>(32768)</td>
<td>(32770)</td>
<td>(32770)</td>
<td></td>
</tr>
</tbody>
</table>

End of Table 5
7  Use Adobe Audition or a similar tool to analyze the PCM files to see the timing (Figure 9).

5.3 Retrieving A Fax Image From A T38 Capture

5.3.1 Using A Packet Capture

To get the fax image from a T38 capture, make sure that the T38 page data is completely captured. Then use Wireshark to view the captured file, decoded as T.38.
To extract the fax image from the capture file, follow this procedure:

**Procedure 3 Fax Image Extraction**

**Step – Action**

1. In the last a few T38 packets at the end of one page, there are a few packets with "t4xxxx_sig_end"; select the first one.

2. Browse the packet details to UDPTL->primary-ifp-packet->data-field->item 0->item->message fragments.

3. Single click the "message fragments" line, there is a jump out windows shows it tries to process packet details.

4. Right click the "message fragments" line, in the jump out menu, select "Export Selected Packets Bytes", save as a binary file.

**Figure 10 T.38 In Wireshark**

5. Install libtiff on your computer; it can be installed on Windows (http://gnuwin32.sourceforge.net/packages/tiff.htm), but more easily on Linux (it's a default lib for most Linux distributions).

6. Use utility fax2tiff to convert the binary file to a tiff image file. Syntax: fax2tiff -M input.bin -o output.tiff
6 Appendix A - References

6.1 Standards

6.1.1 Fax Terminal Modulations/Protocol/Coding

- ITU-T Recommendation V.29, “9600 Bits per Second Modem Standardized for Use on Point-to-Point 4-Wire Leased Telephone-Type Circuits.”
- ITU-T Recommendation V.34, “A Modem Operating at Data Signaling Rates of Up to 28800 bit/s for Use on the General Switched Telephone Network and on Leased Point-to-Point 2-Wire Telephone-Type Circuits.”
- ITU-T Recommendation V.8, “Procedures for Starting Sessions of Data Transmission Over the General Switched Telephone Network.”
- ITU-T Recommendation T.6, “Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus.”

6.1.2 Fax Relay Network Protocols

6.1.2.1 T.38


6.1.2.2 AAL2

- ITU-T Recommendation I.366.2, “AAL Type 2 Service Specific Convergence Sub-layer for Trunking.”

6.1.2.3 FRF.11

- Voice over Frame Relay Implementation Agreement (FRF.11).

6.2 Online Tutorials

- Fax Technology and Testing Issues (http://www.iec.org/online/tutorials/faxtech_test)
7 Appendix B - V.VBD/T.38 Interoperation

This appendix illustrates an implementation of V.VBD - T.38 packet-to-pack interoperation (applies to High Density releases 11.3 and later):

- DIM Test Command Script
- Spy Trace

7.1 DIM Test Command Script

**Example 3 V.VBD – T38 Interoperation script**

```bash
#----------------------------------------
# Optional: Fax Details updates
#----------------------------------------
dimt set template 0 fax details t38_version 2
dimt set template 0 fax details dbg_level 2

#------------------------------------------
# Protocol and parameters
#-----------------------------------------
cc xdp_cli_set_prot 0 ether ipv4 udp
cx xdp_cli_set_prot 1 ether ipv4udp
cx xdp_cli_set_prot 2 ether ipv4udp
cx xdp_cli_set_prot 3 ether ipv4udp
cx xdp_set 0 phy phy_id 24
cc xdp_set 0 ether loc_addr 2A:2B:2C:2D:2E:2F
cc xdp_set 0 ether rem_addr 1A:1B:1C:1D:1E:1F
cc xdp_set 0 ipv4 loc_addr 01.02.03.04
cc xdp_set 0 ipv4 rem_addr 127.0.0.1
cc xdp_set 0 udp loc_port 1036
cc xdp_set 0 udp rem_port 1034
cc xdp_set 1 phy phy_id 24
cc xdp_set 1 ether loc_addr 2A:2B:2C:2D:2E:2F
cc xdp_set 1 ether rem_addr 1A:1B:1C:1D:1E:1F
cc xdp_set 1 ipv4 loc_addr 01.02.03.04
cc xdp_set 1 ipv4 rem_addr 127.0.0.1
cc xdp_set 1 udp loc_port 1034
cc xdp_set 1 udp rem_port 1034
cc xdp_set 2 phy phy_id 24
cc xdp_set 2 ether loc_addr 2A:2B:2C:2D:2E:2F
cc xdp_set 2 ether rem_addr 1A:1B:1C:1D:1E:1F
cc xdp_set 2 ipv4 loc_addr 01.02.03.04
cc xdp_set 2 ipv4 rem_addr 127.0.0.1
cc xdp_set 2 udp loc_port 1026
cc xdp_set 2 udp rem_port 1024
cc xdp_set 3 phy phy_id 24
cc xdp_set 3 ether loc_addr 2A:2B:2C:2D:2E:2F
cc xdp_set 3 ether rem_addr 1A:1B:1C:1D:1E:1F
cc xdp_set 3 ipv4 loc_addr 01.02.03.04
cc xdp_set 3 ipv4 rem_addr 127.0.0.1
cc xdp_set 3 udp loc_port 1026
cc xdp_set 3 udp rem_port 1026

#----------------------------------------
# Open 4 channels
#----------------------------------------
dimt open 0 alloc 0 chan tdmcfg rxts 3 txts 3 encapcfg rtp txssrc 100 rxssrc 101
dimt open 1 alloc 0 chan tdmcfg rxts 1 txts 1 encapcfg rtp txssrc 100 rxssrc 101
dimt open 2 alloc 0 chan tdmcfg rxts 2 txts 2 encapcfg rtp txssrc 100 rxssrc 101
dimt open 3 alloc 0 chan tdmcfg rxts 4 txts 4 encapcfg rtp txssrc 100 rxssrc 101
cc xdp_cli_set_state 0 tx_enable rx_enable
cc xdp_cli_set_state 1 tx_enable rx_enable
cc xdp_cli_set_state 2 tx_enable rx_enable
cc xdp_cli_set_state 3 tx_enable rx_enable

#---------------------------------------------------------------------------------
# Go to the voice mode (refer VBD User's Guide for a VBD mode setting)
#---------------------------------------------------------------------------------
dimt voice 0 alloc 0
dimt voice 1 alloc 0
```
# Go to the fax relay mode

```plaintext
dimt fax 2 alloc 0
dimt fax 3 alloc 0
```

# Linear Domain cross-conferencing setting

```plaintext
dimt reset template 10
dimt set template 10 conn_req nelem 6
dimt set template 10 conn_req elem 0 req_type del ld_tele_pkt src 1 dst 1
dimt set template 10 conn_req elem 1 req_type del ld_pkt_tele src 1 dst 1
dimt set template 10 conn_req elem 2 req_type del ld_tele_pkt src 2 dst 2
dimt set template 10 conn_req elem 3 req_type del ld_pkt_tele src 2 dst 2
dimt set template 10 conn_req elem 4 req_type add ld_pkt_pkt src 1 dst 2
dimt set template 10 conn_req elem 5 req_type add ld_pkt_pkt src 2 dst 1
dimt conn_req 1 alloc 10
```

End of Example 3

### 7.2 Spy Trace

#### Example 4  V.VBD – T38 Interoperation DIM SPY trace

CH0 and CH1 go to the voice mode

```
0000062506 - DIM: 0:0, (TCID 0) Tx='VOICE_MODE2'(2050) Len=5, trans_id=0x8004, valid_params=0x7f, mode=0, dir='from tele, det_loc='pre config, pos_mask=0x0001, pos_val=0x0000
0000062506 - DIM: 0:0, ---- mf_params: power_lo=-30, power_hi=0, pos_twist=5, neg_twist=9, break=-32
0000062506 - DIM: 0:0, ---- action=0x3, filter_bfield=0x0, long_digit_bfield=0x0, long_digit_hangover=0x0, upspeed_profile_idx=0
0000062506 - DIM: 0:0, Tx='DIGIT_ACTIONS_CONFIG'(2063) Len=10, trans_id=0x8004, valid_params=0x1, vad_type=0
0000062506 - DIM: 0:0, ---- thresh_offset=0, hold_over=180
0000062506 - DIM: 0:0, (TCID 0) Tx='VP_CONFIG'(2054) Len=13, trans_id=0x8004, valid_params=0x8f, type=0
0000062506 - DIM: 0:0, --- MinDel=40, NomDel=60, MaxDel=80, buf_starve_timeout=00000000, ctrl=0x0, vp_offset_delay=0
0000062506 - DIM: 0:0, (TCID 0) Tx='COMMIT_STATE'(2053) Len=5, trans_id=0x0004
0000062507 - DIM: 0:1, (TCID 0) Rx='GENERAL ACK'(117) Len=8, trans_id=0x0004, gen_param=0, req_msg='VOICE_MODE2'(2050), ak_type=0x3, seq_num=13, priority=1
0000062507 - SC: 0, Called sc_report_transaction2(), trans_id = 0x8004, trans_state = 3(ACK DONE)
0000062507 - DIM: 0:1, Tx='PLC_CONFIG'(2081) Len=6, trans_id=0x8005, plc_type=0
0000062507 - DIM: 0:1, Tx='RX_NMU_CONFIG'(2080) Len=9, trans_id=0x8005, valid_params=0x1
0000062507 - DIM: 0:1, ---- mode=0, lp_order=0, level=0
0000062507 - DIM: 0:1, (TCID 1) Tx='VOICE_MODE2'(2050) Len=5, trans_id=0x8005, valid_params=0x7f, mode=0, dir='from tele, det_loc='pre config, pos_mask=0x0001, pos_val=0x0000
0000062507 - DIM: 0:1, ---- mf_params: power_lo=-30, power_hi=0, pos_twist=5, neg_twist=9, break=-32
0000062507 - DIM: 0:1, (TCID 1) Tx='DIGIT_ACTIONS_CONFIG'(2063) Len=10, trans_id=0x8005, valid_params=0x1, vad_type=0
0000062507 - DIM: 0:1, ---- action=0x3, filter_bfield=0x0, long_digit_bfield=0x0, long_digit_hangover=0x0, upspeed_profile_idx=0
0000062507 - DIM: 0:1, (TCID 1) Tx='VP_CONFIG'(2054) Len=13, trans_id=0x8005, valid_params=0x8f, type=0
0000062507 - DIM: 0:1, ---- thresh_offset=0, hold_over=180
0000062507 - DIM: 0:1, (TCID 1)Tx='COMMIT_STATE'(2053) Len=5, trans_id=0x0005
0000062508 - DIM: 0:1, (TCID 1) Rx='GENERAL ACK'(117) Len=8, trans_id=0x0005, gen_param=0, req_msg='VOICE_MODE2'(2050), ak_type=0x3, seq_num=14, priority=1
0000062508 - SC: 1, Called sc_report_transaction2(), trans_id = 0x8005, trans_state = 3(ACK DONE)
```

CH2 and CH3 go to the FAX relay mode
Linear Domain cross-conferencing:

After above set-up, the fax debug trace level 2 will show as below. We will discuss how to use this message to debug fax relate issue in the next chapter:

End of Example 4
8 Appendix C - Debug Trace Channel Separation

The following commands may be used in a .bat file to separate a multi-channel fax debug trace into multiple traces, each containing only one channel. It can be extended for any number of channels.

Example 5 Fax Debug Trace Separation

```bash
@echo off
set PATH=%PATH%;t:\gen\gnu\99-11-01\cygwin-b20\H-i586-cygwin32\bin

if "%2"=="" goto ERROR

grep -U ": 0x0a01\|: 0x2201" %2.log > %2_ch1.log
faxDebugParser f0 %2_ch1.log > %2_ch1.txt
if "%1"=="1" goto END

grep -U ": 0x0a02\|: 0x2202" %2.log > %2_ch2.log
faxDebugParser f0 %2_ch2.log > %2_ch2.txt
if "%1"=="2" goto END

grep -U ": 0x0a03\|: 0x2203" %2.log > %2_ch3.log
faxDebugParser f0 %2_ch3.log > %2_ch3.txt
if "%1"=="3" goto END

grep -U ": 0x0a04\|: 0x2204" %2.log > %2_ch4.log
faxDebugParser f0 %2_ch4.log > %2_ch4.txt
if "%1"=="4" goto END

:ERROR

echo

==================================================================================

echo USAGE: faxParser_multich [num of channels] [log debug trace filename w/o extension]
echo

==================================================================================

:END

End of Example 5
```
9 Appendix D - Fax Channel Setup

Fax channel setup is shown below in three examples:

1. Fax Template Setup below
2. “DSP Download And Channel Setup” on page 31
3. “Open Channel” on page 31

9.1 Fax Template Setup

Example 6 DIMT Template Setup Script

```bash
#---------------------------------------------
# Script to demonstrate boot time setup for telogy high density system
#---------------------------------------------

# Set shell to low priority
sys_set_shell_low
dbgcmd "wait 100"

#---------------------------------------------
# Enter MXP Shell.
# Telogy "Dimtest" commands can be executed under MXP shell.
#---------------------------------------------
dbgcmd
  echo on

#---------------------------------------------
# Setup templates
# RTP Template 0
#---------------------------------------------

dimt set template 0 chan procfg rtp vp numprof 7
dimt set template 0 chan procfg rtp vp tx_idx 0
dimt set template 0 chan procfg rtp vp rx_idx 0
dimt set template 0 chan procfg rtp vp prof 0 codectype g711_mu
dimt set template 0 chan procfg rtp vp prof 0 ploadtype 100
dimt set template 0 chan procfg rtp vp prof 0 vif 640
dimt set template 0 chan procfg rtp vp prof 1 codectype gen_sid
dimt set template 0 chan procfg rtp vp prof 1 ploadtype 13
dimt set template 0 chan procfg rtp vp prof 1 vif 8
dimt set template 0 chan procfg rtp vp prof 2 codectype g729ab
dimt set template 0 chan procfg rtp vp prof 2 ploadtype 18
dimt set template 0 chan procfg rtp vp prof 2 vif 80
dimt set template 0 chan procfg rtp vp prof 3 codectype g723_53
dimt set template 0 chan procfg rtp vp prof 3 ploadtype 110
dimt set template 0 chan procfg rtp vp prof 3 vif 160
dimt set template 0 chan procfg rtp vp prof 4 codectype g723_63
dimt set template 0 chan procfg rtp vp prof 4 ploadtype 120
dimt set template 0 chan procfg rtp vp prof 4 vif 192
dimt set template 0 chan procfg rtp vp prof 5 codectype g726_32
dimt set template 0 chan procfg rtp vp prof 5 ploadtype 2

dimt set template 0 chan procfg rtp vp prof 6 codectype g726_32m
dimt set template 0 chan procfg rtp vp prof 6 ploadtype 20

dimt set template 0 chan procfg rtp vp prof 6 vif 192

dimt set template 0 chan procfg rtp pp numprof 3
dimt set template 0 chan procfg rtp pp prof 0 flt nse
dimt set template 0 chan procfg rtp pp prof 0 firstxpt 104
dimt set template 0 chan procfg rtp pp prof 0 numtype 0
dimt set template 0 chan procfg rtp pp prof 0 firstsub 144
dimt set template 0 chan procfg rtp pp prof 0 numbub 16
dimt set template 0 chan procfg rtp pp prof 0 route can
dimt set template 0 chan procfg rtp pp prof 1 flt nse
dimt set template 0 chan procfg rtp pp prof 1 firstxpt 103
dimt set template 0 chan procfg rtp pp prof 1 numtype 0
dimt set template 0 chan procfg rtp pp prof 1 firstsub 0
dimt set template 0 chan procfg rtp pp prof 1 numbub 16
dimt set template 0 chan procfg rtp pp prof 1 route dtmf
dimt set template 0 chan procfg rtp pp prof 2 flt nse
dimt set template 0 chan procfg rtp pp prof 2 firstxpt 101
dimt set template 0 chan procfg rtp pp prof 2 numtype 0
dimt set template 0 chan procfg rtp pp prof 2 firstsub 0
dimt set template 0 chan procfg rtp pp prof 2 numbub 0
dimt set template 0 chan procfg rtp pp prof 2 route general
```
dimt set template 0 chan p2p_from encap rtp
dimt set template 0 chan p2p_from general route host
dimt set template 0 chan p2p_from general timeout 0
dimt set template 0 chan p2p_from general rtp lock_ssrc off
dimt set template 0 chan p2p_from general rtp ssrc 101
dimt set template 0 chan p2p_from alarm route host
dimt set template 0 chan p2p_from alarm timeout 0
dimt set template 0 chan p2p_from alarm rtp lock_ssrc off
dimt set template 0 chan p2p_from alarm rtp ssrc 102
dimt set template 0 chan p2p_from dtmf route dsp host
dimt set template 0 chan p2p_from dtmf timeout 150
dimt set template 0 chan p2p_from dtmf rtp lock_ssrc off
dimt set template 0 chan p2p_from dtmf rtp ssrc 103
dimt set template 0 chan p2p_from cas route dsp
dimt set template 0 chan p2p_from cas timeout 1000
dimt set template 0 chan p2p_from cas rtp lock_ssrc off
dimt set template 0 chan p2p_from cas rtp ssrc 104

dimt set template 0 chan p2p_to general init_pkt_cnt 3
dimt set template 0 chan p2p_to general init_rep_interval 5
dimt set template 0 chan p2p_to general keep_alive_interval 0
dimt set template 0 chan p2p_to general rtp ssr1c 101
dimt set template 0 chan p2p_to general rtp ploadtype 101
dimt set template 0 chan p2p_to alarm init_pkt_cnt 3
dimt set template 0 chan p2p_to alarm init_rep_interval 5
dimt set template 0 chan p2p_to alarm keep_alive_interval 0
dimt set template 0 chan p2p_to alarm rtp ssrc 102
dimt set template 0 chan p2p_to alarm rtp ploadtype 102
dimt set template 0 chan p2p_to dtmf init_pkt_cnt 3
dimt set template 0 chan p2p_to dtmf init_rep_interval 5
dimt set template 0 chan p2p_to dtmf keep_alive_interval 50
dimt set template 0 chan p2p_to dtmf rtp ssrc 103
dimt set template 0 chan p2p_to dtmf rtp ploadtype 103
dimt set template 0 chan p2p_to cas init_pkt_cnt 3
dimt set template 0 chan p2p_to cas init_rep_interval 5
dimt set template 0 chan p2p_to cas keep_alive_interval 0
dimt set template 0 chan p2p_to cas rtp ssrc 104
dimt set template 0 chan p2p_to cas rtp ploadtype 104

#dimt set template 0 chan tdmcfg companding alaw

dimt set template 0 voice vpcfg playout_type fixed
dimt set template 0 voice vadcfg type none

dimt reset template 0 voice eccfg

#dimt set template 0 voice digitcfg type dtmf

# dimt set template 0 voice digitcfg actions relay report squelch

dimt set template 0 voice digitcfg dir tele

# dimt set template 0 voice digitcfg info power_low -30

# dimt set template 0 voice digitcfg info power_high 0

# dimt set template 0 voice digitcfg info ptwist 5

# dimt set template 0 voice digitcfg info ntwist 9

# dimt set template 0 voice digitcfg info break -32

# dimt set template 0 fax details t38_version 0

# dimt set template 0 fax details dbgl_level 2

# dimt set template 0 fax encapcfg encapsulation none

# dimt set template 0 fax details rate 14400

# dimt set template 0 fax details nom_delay 300

# dimt set template 0 fax details pkt_size 30

# dimt set template 0 fax details ts_level -10

# dimt set template 0 fax details cd_thresh -33

# dimt set template 0 fax details nat 20

# dimt set template 0 fax details hs_pkt_rate 20

# dimt set template 0 fax details ls_redun 1

# dimt set template 0 fax details hs_redun 0

# dimt set template 0 fax details tcf_method 2

# dimt set template 0 fax details max_ls_data_pkt 1

# dimt set template 0 fax details tx_network_timeout 150

# dimt set template 0 fax details eflag_start_tmr 2600

# dimt set template 0 fax details eflag_stop_tmr 2300

# dimt set template 0 fax details options_valid none

# dimt set template 0 fax details options none

# dimt set template 0 fax details t30_eecm disable

# dimt set template 0 fax details t30_mr_compress disable

# dimt set template 0 fax details nsf_country 0

# dimt set template 0 fax details nsf_vendor 0

# Ignore error if the parameter is not supported by this Micro code

# dimt set template 0 fax details cleardown_tmr 0
9.2 DSP Download And Channel Setup

Example 7  DSP Download and Channel Setup Script

```bash
# Download DSP
# Dsp Core = 0
# Image Id = 0
cc dnld 0 0
wait 3000

# Associate Voice TCID
# TCID = 0
# DSP Core =0
# Channel =0
cc assoc 0 0 0

# Associate Voice TCID
# TCID = 1
# DSP Core =0
# Channel =1
cc assoc 1 0 1

# Associate Signaling TCID '0'
# TCID = 0
# DSP Core =0
# Channel =0
cc sig_assoc 0 0 0

# Associate Signaling TCID '1'
# TCID = 1
# DSP Core =0
# Channel =1
cc sig_assoc 1 0 1

# Register TCID '0' with XDP
cc xdp_cli_reg 0

# Register TCID '1' with XDP
cc xdp_cli_reg 1
```

End of Example 7

9.3 Open Channel

Example 8  Open Channel Script

```bash
# Script to setup an T.38 fax call
dbgcmd
dbgecho on
```
dimt open 0 alloc 0 chan tdmcfg rxts 3 txts 3 encapcfg rtp txssrc 100 rxssrc 101
dimt open 1 alloc 0 chan tdmcfg rxts 4 txts 4 encapcfg rtp txssrc 101 rxssrc 100
wait 500

spy dim 2
#------------------------------------------
# Protocol and parameters
#-----------------------------------------
c xdp_cli_set_prot 0 ether ipv4 udp
c xdp_set 0 phy phy_id 24
c xdp_set 0 ether loc_addr 2A:2B:2C:2D:2E:2F
c xdp_set 0 ether rem_addr 1A:1B:1C:1D:1E:1F
c xdp_set 0 ipv4 loc_addr 01.02.03.04
c xdp_set 0 ipv4 rem_addr 127.0.0.1
c xdp_set 0 udp loc_port 1024
c xdp_set 0 udp rem_port 1026

c xdp_cli_set_prot 1 ether ipv4 udp
c xdp_set 1 phy phy_id 24
c xdp_set 1 ether loc_addr 2A:2B:2C:2D:2E:2F
c xdp_set 1 ether rem_addr 1A:1B:1C:1D:1E:1F
c xdp_set 1 ipv4 loc_addr 01.02.03.04
c xdp_set 1 ipv4 rem_addr 127.0.0.1
c xdp_set 1 udp loc_port 1026
c xdp_set 1 udp rem_port 1024

cc xdp_cli_set_state 0 tx_enable rx_enable
c xdp_cli_set_state 1 tx_enable rx_enable

#----------------------------------------------
# fax  mode
#----------------------------------------------
dimt fax 0 alloc 0
dimt fax 1 alloc 0

echo off
dbgexit

End of Example 8