

**University of Michigan  
Space Physics Research Laboratory**

<b>TIDI Real Time Command &amp; Display System Design and Maintenance</b>	CAGE No. 0TK63 Drawing No. 055-3957 Project TIDI Contract No. NASW-5-5049 Page 1 of 46
---	--

REVISION RECORD

Rev	Description	Date	Approval
	Initial Release		

APPROVAL RECORD

Function	Name	Signature	Date
Originator	D. Gell		
Flight Software	S. Musko		
Instrument Scientist	W. Skinner		
Program Manager	C. Edmonson		
Systems Engineer			
R&QA	J. Eder		

## Contents

<b>1.</b>	<b>References .....</b>	<b>5</b>
<b>2.</b>	<b>Introduction.....</b>	<b>5</b>
2.1	Purpose and Scope.....	5
2.2	Document Conventions .....	6
2.2.1	<i>Numbers .....</i>	<i>6</i>
2.2.2	<i>Typographic Conventions.....</i>	<i>6</i>
<b>3.</b>	<b>System Description .....</b>	<b>6</b>
3.1	Overview.....	6
3.2	Data Communications.....	10
3.3	Telemetry Decomposition.....	11
3.4	Displays and Graphical Interface .....	12
3.5	Commands.....	15
3.5.1	<i>TICL Command Formation .....</i>	<i>15</i>
3.5.2	<i>CSTOL Commands .....</i>	<i>17</i>
<b>4.</b>	<b>Telemetry Decomposition.....</b>	<b>19</b>
4.1	Spacecraft Packets.....	20
4.2	Instrument Packets .....	22
4.2.1	<i>The TIDI Instrument Housekeeping Packet Definition .....</i>	<i>24</i>
4.2.2	<i>Other TIDI Telemetry Packet Definitions.....</i>	<i>24</i>
4.3	GSE Packets .....	25
<b>5.</b>	<b>Displays.....</b>	<b>26</b>
5.1	CSTOL Pages .....	28
5.2	Modifying Fixed Format TAE+ Panels .....	28
5.3	Adding Fixed Format TAE+ Panels .....	28
5.4	Wildcard Display TAE+ Panels .....	29
5.5	Strip Chart TAE+ Panels.....	30
5.5.1	<i>Strip Chart Operation .....</i>	<i>30</i>
5.5.2	<i>Adding an additional strip chart .....</i>	<i>32</i>
5.6	Spectral Display TAE+ Panel .....	33
5.7	Command Panels.....	33
<b>6.</b>	<b>Commands .....</b>	<b>35</b>
6.1	Telemetry Server Commands.....	35
6.2	Real-Time TICL Commands.....	35
6.3	GSE Commands .....	35

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 3 of 46
--	---

- 7. Command Packaging and Transmission .....35**
- 8. Updating the OASIS Database Tables .....35**
- 9. Building the OASIS Application .....37**
- 10. Supporting Utilities .....37**
  - 10.1 makeDefineMacros ..... 37
  - 10.2 makeDefineProcs ..... 37
  - 10.3 make\_rcds ..... 37
- 11. Global Variables .....37**

## Figures

<b>Figure 1</b>	<b>OASIS Subsystems .....</b>	<b>7</b>
<b>Figure 2</b>	<b>Control Flow for TICL Command Generation Example .....</b>	<b>17</b>
<b>Figure 3</b>	<b>Example Strip Chart.....</b>	<b>30</b>

## Tables

<b>Table 1</b>	<b>OASIS External Communications Subsystem Tables .....</b>	<b>7</b>
<b>Table 2</b>	<b>OASIS Data Handling Subsystem Tables .....</b>	<b>8</b>
<b>Table 3</b>	<b>OASIS Display Subsystem Tables.....</b>	<b>8</b>
<b>Table 4</b>	<b>OASIS Command Subsystem Tables .....</b>	<b>9</b>
<b>Table 5</b>	<b>OASIS CSTOL Subsystem Tables .....</b>	<b>9</b>
<b>Table 6</b>	<b>OASIS Utility Tables .....</b>	<b>10</b>
<b>Table 7</b>	<b>Links Used by the TIDI RCDS .....</b>	<b>10</b>
<b>Table 8</b>	<b>Primary Streams Used by the TIDI RCDS.....</b>	<b>11</b>
<b>Table 9</b>	<b>Secondary Streams .....</b>	<b>12</b>
<b>Table 10</b>	<b>TIDI RCDS Displays.....</b>	<b>13</b>
<b>Table 11</b>	<b>TIDI RCDS Command Interface Panels.....</b>	<b>14</b>
<b>Table 12</b>	<b>TICL Command Macros.....</b>	<b>15</b>
<b>Table 13</b>	<b>TM Server Commands .....</b>	<b>17</b>
<b>Table 14</b>	<b>GSE Commands .....</b>	<b>18</b>
<b>Table 15</b>	<b>CSTOL Macros for Telemetry Decomposition.....</b>	<b>19</b>
<b>Table 16</b>	<b>Spacecraft TM Decomposition Definitions .....</b>	<b>20</b>
<b>Table 17</b>	<b>Telemetry Definition Files Naming Convention.....</b>	<b>21</b>
<b>Table 18</b>	<b>Instrument Packet Definitions .....</b>	<b>23</b>
<b>Table 19</b>	<b>GSE Packet Definitions .....</b>	<b>26</b>
<b>Table 20</b>	<b>Required Decomposition Table Fields for GSE Packets .....</b>	<b>26</b>
<b>Table 21</b>	<b>Data Display Definitions .....</b>	<b>27</b>
<b>Table 22</b>	<b>CSTOL Macros for Strip Charts .....</b>	<b>31</b>
<b>Table 23</b>	<b>Command Interface Panels .....</b>	<b>34</b>
<b>Table 24</b>	<b>OASIS Global Variables .....</b>	<b>37</b>

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 5 of 46
--	---

## 1. References

1. OASIS\_CC, *System Manager's Guide*, November 1994
2. OASIS-CC, *CSTOL Reference Manual*, November 1994
3. OASIS-CC, *Database Guide*, November 1994.
4. OASIS-CC, *Quick Reference Manual*, November 1994
5. Century Computing, *TAE+ User Interface Developer's Guide, Version 5.3*, September 1993, TAE-5.3-D02.
6. Musko, Stephen, "TIDI Flight Software Requirements Specification", 6 October 1999, SPRL File 055-3320
7. Musko, Stephen, "TIDI Flight Software Instrument Parameter Dictionary", 18 November 1999, SPRL File 055-3903
8. Miller, Ryan, "GSECOMM Software Interface Definition", 20 August 1999, SPRL File 055-3821
9. Musko, Stephen, "Instrument Parameter Definitions", 24 September 1999, SPRL File 055-3519
10. Grunberger, Paul, "Telemetry and Command Data Structures for TIMED", 26 Jan 1998
11. "General Instrument Interface Specification"

## 2. Introduction

### 2.1 Purpose and Scope

The TIDI Real Time Command and Display Systems (TIDI RCDS) is composed of an OASIS application, a GSE control application, and a collection of shell scripts. The OASIS application performs telemetry decomposition, data display, and provides the user command interface. The GSE control application (GSEcomm) operates the calibration sources, polling them for status information, packaging the status information into telemetry packets, and converting commands from the RCDS into the serial messages required by the various calibration subsystems. The shell scripts provide an interface between the RCDS and the TIMED mission control center, compiling TICL commands, packaging command messages, encrypting them messages and transmitting them to either the TIMED MOC or to the TIMED spacecraft emulator.

This document describes the project unique customizations applied to OASIS to create the TIDI RCDS application. The interfaces between the OASIS application and the external applications and scripts are described.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 6 of 46
--	---

## **2.2 Document Conventions**

### 2.2.1 Numbers

All numbers are decimal unless otherwise indicated. Hexadecimal numbers are indicated by the "H" suffix. Binary numbers are indicated by the "B" suffix.

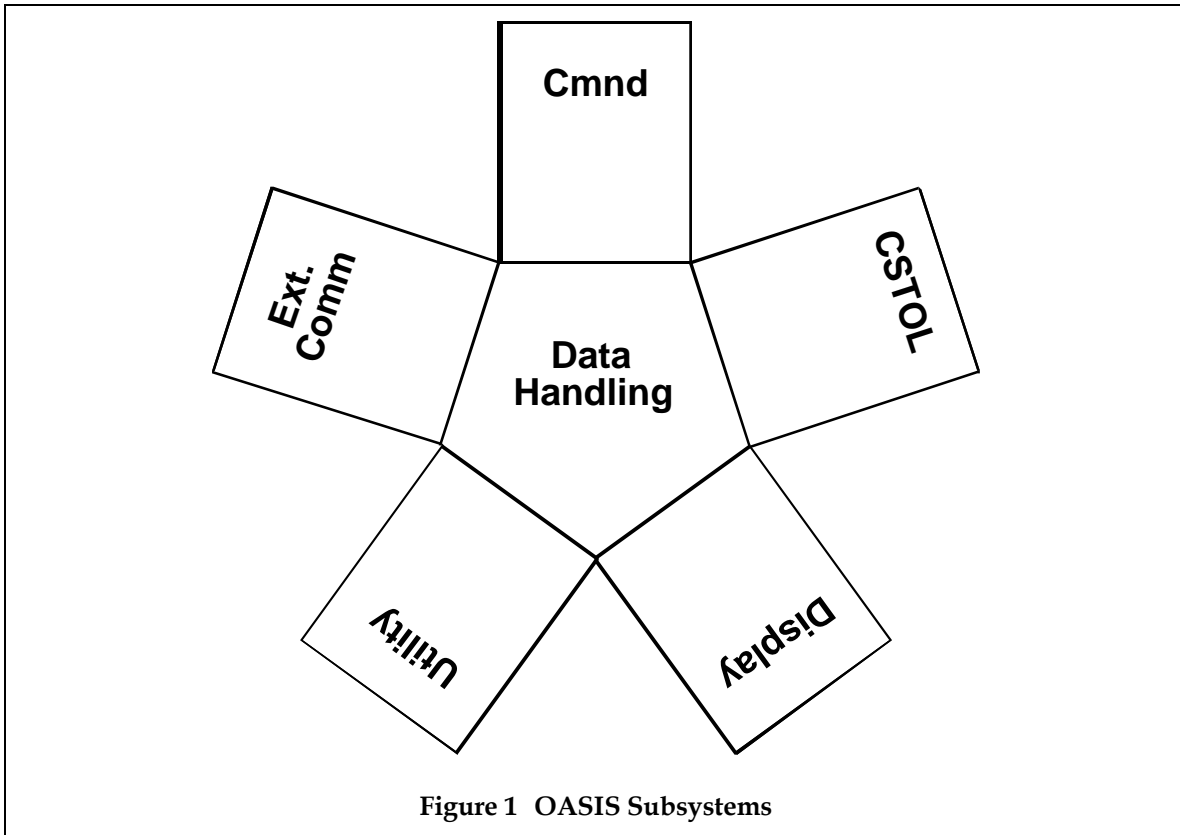
### 2.2.2 Typographic Conventions

In this document computer output is set in the **bold courier** typeface. User inputs are set in the `courier` typeface. Templates that are to be replaced in a command are set in the *italic courier* typeface.

## **3. System Description**

### **3.1 Overview**

The OASIS system is comprised of a number of subsystems, illustrated in Figure 1. The subsystems are the data handling subsystem, the command subsystem (Cmnd), the external communications system (Ext. Comm.), the display subsystem (Display), the CSTOL interpreter (CSTOL) and the utility subsystem. Each of these systems is configured by populating one or more tables. The population of these tables is performed by a series of CSTOL procedures which use data table manipulation commands to specify values for particular entries. Once the tables are populated, a CSTOL command, CHECKPOINT, may be used to save the current contents of the tables in the form used by OASIS when it starts up.



The external communications subsystem is defined by the contents of the links, streams and decomposition tables (Table 1). Streams define the characteristics of contiguous aggregations of bits received by the external communications subsystem. Streams may be acquired, decomposed or routed. The links table defines the physical characteristics of a communications link. Links may be either forward (from the OASIS application to an external element) or return (from an external element to OASIS). They may not be bi-directional. The decomposition table describes the manner in which items may be extracted from a stream. A stream may be decomposed into other streams, called secondary streams and substreams, into subcommutated streams (subcoms) or into atoms. Atoms are individual data items that are forwarded to the data handling system for further processing.

<b>Table 1</b> <b>OASIS External Communications Subsystem Tables</b>
links
streams
decomposition

The data handling subsystem is defined by the contents of the latest\_data, analog\_conversions, state\_conversions, limits, equations, triggers, and unit\_conversions tables (Table 2). The latest\_data table contains the attributes for all telemetry, ancillary and internal status data items used by the application. It stores the latest value for each received data item. Values of data items in the latest\_data table are accessible by name to CSTOL procedures and may be linked to display items. The analog\_conversions table specifies how to convert a telemetry atom from a digital number to an engineering unit value. The state\_conversions table specifies how to convert digital number representation of a discrete telemetry value to a named state. The limits table defines the alarm values for telemetry points. Alarms may be specified as red-low, yellow-low, yellow-high and red-high. Alarms may also be specified on the rate of change of a data item. The equations table allows the creation of derived data items by specifying a CSTOL statement or a C routine to be executed whenever a latest\_data item changes value. The triggers table is used to automate responses to limits excursions. The table contains a CSTOL statement to be executed when a specified limit is exceeded. The unit\_conversion table specifies the conversions to be applied from one engineering unit to another.

<b>Table 2</b> <b>OASIS Data Handling Subsystem Tables</b>
latest_data
analog_conversions
state_conversions
limits
equations
triggers
unit_conversions

The display subsystem is defined by the contents of the display\_definitions table and the display\_descriptions table (Table 3). The display panels are produced with the TAE+ interface builder called "Actions Workbench", which produces a c language function which is compiled and linked with the OASIS application. The display\_definitions table provides a mapping between the names of items in the latest\_data table and the name of a TAE+ display item. The display\_descriptions table provides a way to connect the name of a TAE+ panel with an identifier that may be used in CSTOL commands.

<b>Table 3</b> <b>OASIS Display Subsystem Tables</b>
display_definitions
display_descriptions



The command subsystem is defined by a number of tables, listed in Table 4, below. The command maps tables define where variable fields are inserted into the binary or ASCII commands generated by OASIS in response to a CSTOL command verb such as SET, OPEN, CLOSE, etc. The commands table defines the syntax of a command, specifying the command verb, the target element, and qualifying clauses. In the TIDI RCDS, this command definition mechanism is used for the commands which control the interface with the telemetry server and the commands that control the GSE Calibration stimuli. Further details on these tables can be found in the OASIS Database Guide and the OASIS System Manager's Guide

<b>Table 4</b> <b>OASIS Command Subsystem Tables</b>
ascii_command_maps
ascii_command_values
commands
command_maps
command_messages
command_state_conversions
command_value_conversions
command_values
element_characteristics
element_hierarchies
time_location_maps

The CSTOL subsystem is defined by the macros, procedures and utility\_tasks tables (Table 5). The macros table connects macro names to files containing sequences of CSTOL statements which are executed when the macro name is encountered in a CSTOL statement. The procedures table connects names to CSTOL procedures, collections of statements that execute as a subroutine. The utility\_tasks table permits the execution of UNIX shell scripts from the CSTOL prompt or from within procedures. In the TIDI RCDS, shell scripts are used to compile, package and transfer TIDI Command Language statements and programs .

<b>Table 5</b> <b>OASIS CSTOL Subsystem Tables</b>
macros
procedures
utility_tasks

The Utility Subsystem is a catch all providing other functionality. It consists of the bridge utility which allows the transfer of selected data items to files for storage or to external devices, including sockets. The user\_privileges table specifies different levels of access to OASIS for various classes of users.

<b>Table 6</b> <b>OASIS Utility Tables</b>
bridge_definitions
user_privileges

### 3.2 Data Communications

OASIS external communication is configured using “links” and “primary streams”. A “link” is a physical communications link. A “primary stream” defines the characteristics of the stream of data bytes transmitted over the physical communications link. The TIDI RCDS implements six links and seven primary streams. Primary streams are, in turn, composed of secondary streams and substreams. In the TIDI RCDS, secondary streams are defined for each of the different packet types.

The links used by the TIDI RCDS are listed in Table 7, and the primary streams in Table 8. The data\_cmds\_link and the mdc\_return link form the two way communications path the either the TIMED telemetry server or to the TIMED provided spacecraft emulator. The data\_cmds stream is used to transmit the protocol commands required to configure the telemetry packets returned by the server. The gse\_forward and gse\_return link form the two way communications path to the GSEcomm interface paths. The gse\_cmds stream is used to transmit commands to the gse equipment and the gse\_data stream is used to obtain the gse status data. The gse\_record stream is used as the source of data to be recorded by the OASIS bridge system.

TIDI telemetry consists of variable length packets which are blocked into the fixed length TIMED telemetry packets. The tidi\_fwd\_link and tidi\_ret\_link links and the tidi\_fwd and tidi\_ret streams are used to extract the TIDI packets from the spacecraft packets. The data portion of spacecraft packets bearing the TIDI application ID are placed in the tidi\_fwd stream and routed out the tidi\_fwd\_link. The tidi\_fwd\_link communicates with the tidi\_ret\_link, which fills the tidi\_ret stream. In this way, the OASIS external communications system performs all the necessary extraction and synchronization required to extract the variable length TIDI packets from the fixed length spacecraft packets.

<b>Table 7</b> <b>Links Used by the TIDI RCDS</b>	
name	function
data_cmds_link	link to telemetry server

<b>Table 7</b> <b>Links Used by the TIDI RCDS</b>	
name	function
gse_forward	link to GSEcomm interface task
gse_return	link from GSEcomm interface task
mdc_return	link from telemetry server
tidi_fwd_link	extracted tidi data stream source
tidi_ret_link	extracted tidi data stream input to OASIS

<b>Table 8</b> <b>Primary Streams Used by the TIDI RCDS</b>		
name	direction	function
data_cmds	forward	transmits commands to the TIMED telemetry server
gse_cmds	forward	transmits commands to the GSEcomm interface task
gse_data	return	transmits gse status data from the GSEcomm interface task
gse_record	n/a	buffers gse status data for the bridge subsystem for recording
ptp_ret_stream	return	receives TIMED POCC telemetry packets from the TIMED telemetry server
tidi_fwd	forward	transmits the TIDI telemetry stream
tidi_ret	return	receives the TIDI telemetry packets

### **3.3 Telemetry Decomposition**

Telemetry decomposition is the process of extracting particular elements from a primary stream for use by the data handling subsystem. The decomposition makes use of a hierarchical organization of the data, in which a primary stream may be composed of secondary streams, substreams or atoms. Secondary streams may be composed of substreams and atoms. Substreams, in turn, are composed of atoms, the lowest element in the hierarchy. Atoms identify a particular measurement point in the telemetry and are specified by the stream from which it is extracted, its position in the stream, and the number of bits that comprise it.

The TIDI RCDS defines a secondary stream for each spacecraft packet and for each TIDI telemetry packet that it processes. The packets are listed in Table 9, below. These streams are composed of substreams and atoms. In particular, each of the TIDI packets is composed of a substream containing the packet header and the data elements, one atom for each data element.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 12 of 46
--	--

<b>Table 9 Secondary Streams</b>		
Secondary Stream Name	Parent Stream	Description
gse_f8	gse_data	Gse data from hp data acquisition switch
gse_f9	gse_data	Diode laser status information
gse_fa	gse_data	Dye laser status information
gse_fb	gse_data	Gse data from lakeshore temperature monitor
sc_apid4	ptp_ret_stream	High priority spacecraft housekeeping
sc_apid6	ptp_ret_stream	spacecraft housekeeping
sc_apid7	ptp_ret_stream	spacecraft housekeeping
tidi_00	tidi_ret	CCD spectral data packet
tidi_02	tidi_ret	CCD image data packet
tidi_04	tidi_ret	TIDI instrument housekeeping
tidi_05	tidi_ret	Command confirmation packets
tidi_06	tidi_ret	Memory dump packet
tidi_07	tidi_ret	CRC result packet
tidi_08	tidi_ret	Error report packets
tidi_09	tidi_ret	TIDI null fill packets
tidi_10	tidi_ret	Control program global dump packet
tidi_extr	ptp_ret_stream	TIDI packets extracted from timed packets for routing

The contents of the TIDI instrument housekeeping packet are defined in a spreadsheet maintained as a TIDI document, Ref. 6. Two scripts have been written to automate the production of a STOL routine to populate the tables required to decompose, convert to engineering units and limit check these items. The first, `get_ipt`, copies the text file version of the spreadsheet from the TIDI document server to the TIDI POCC workstation. The second, `parse_ipt`, invokes an awk script, `parse_ipt.awk`, to create the CSTOL procedure `3519XX_instparam.prc` which contains all of the table loading directives required for the definition of the TIDI instrument parameters.

### **3.4 Displays and Graphical Interface**

OASIS permits two types of displays, CSTOL windows and TAE+ panels. The CSTOL windows are used for text information and are defined in the `display_descriptions` table. The TIDI RCDS defines a couple of windows in addition to those that the OASIS system requires. TAE+ panels are created using the Action Workbench (AWB), which is a graphical interface development tool. Once an interface is created, or modified, c code is generated which must be compiled and linked with the OASIS system to create the TIDI RCDS application. The data displays developed for the TIDI RCDS are listed in Table 10, below.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 13 of 46
--	--

<b>Table 10</b> <b>TIDI RCDS Displays</b>		
Window Name	Window Type	Description
ASK_WINDOW†	CSTOL WINDOW	Provides place for operator response to ASK Directive.
COMMAND_WINDOW†	CSTOL WINDOW	Display of CSTOL command buffer
CSTOL_ERROR†	CSTOL WINDOW	Display of CSTOL error messages
CSTOL_PROCS†	CSTOL WINDOW	Display of currently executing CSTOL Procedure
CSTOL_PROMPT†	CSTOL WINDOW	Display of CSTOL operator prompt. Provides place for operator CSTOL directive input
FS_EVENTS	CSTOL WINDOW	CSTOL message window containing TIDI flight software messages.
LIMIT_LOG	CSTOL WINDOW	CSTOL message window containing limit messages
MESSAGE_WINDOW†	CSTOL WINDOW	Display of all CSTOL event messages, error messages and WRITE directive text. This window is the session log.
REPORTS†	CSTOL WINDOW	Contents of OASIS tables produced using the REPORT directive.
SCHART1	TAE+ PANEL	Stripchart display of up to 4 parameters
SCHART2	TAE+ PANEL	Stripchart display of up to 4 parameters
SCHART3	TAE+ PANEL	Stripchart display of up to 4 parameters
SCHART4	TAE+ PANEL	Stripchart display of up to 4 parameters
SCHART5	TAE+ PANEL	Stripchart display of up to 4 parameters
SPECTRA1	TAE+ PANEL	Histogram display of up to 50 spectral bins
SPECTRA2	TAE+ PANEL	Histogram display of up to 50 spectral bins
SPECTRA3	TAE+ PANEL	Histogram display of up to 50 spectral bins
SPECTRA4	TAE+ PANEL	Histogram display of up to 50 spectral bins
SPECTRA5	TAE+ PANEL	Histogram display of up to 50 spectral bins
TIDI_CAL	TAE+ PANEL	GSE calibration stimuli status
TIDI_CCD	TAE+ PANEL	CCD status display
TIDI_CP	TAE+ PANEL	Control program status and globals display
TIDI_GSE	TAE+ PANEL	Display of GSE status

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 14 of 46
--	--

<b>Table 10</b> <b>TIDI RCDS Displays</b>		
Window Name	Window Type	Description
TIDI_HS	TAE+ PANEL	Health and Safety display
TIDI_MEM	TAE+ PANEL	Memory dump display
TIDI_PRO	TAE+ PANEL	Profile status display
TIDI_PWR	TAE+ PANEL	Power supply status display
TIDI_TEL	TAE+ PANEL	Telescope status display
TIDI_TMP	TAE+ PANEL	Thermal control system status display
TIMED_STAT	TAE+ PANEL	Spacecraft telemetry items of interest
† required CSTOL window.		

In addition to the display panels, a series of command panels have been created for easy interactive access to TIDI instrument command language statements. The command panels are hierarchical. The main panel named TIDI\_rcds has buttons that are connected to panels which either are menus to select subordinate panels, or are connected to panels which control the configuration of the OASIS application or the interface with the TIMED MOC.

<b>Table 11</b> <b>TIDI RCDS Command Interface Panels</b>	
Panel Name	Description
TIDI_rcds	Main user interface panel, containing buttons which select other interface items
sysview	selects CSTOL pages to view
links	configuration of forward and return links
pg_sel	menu of data display panels
graphics	menu of spectra and strip chart displays
commands	menu of command interface panels
send_cm	select and send a pre-compiled TICL command block file`
mech	mechanism positioning commands
mem_cmds	memory management commands
heater	heater duty cycle commands
boot_cmd	software initialization and reset commands
cntl_pgm	TICL control program commands

<b>Table 11</b> <b>TIDI RCDS Command Interface Panels</b>	
Panel Name	Description
scan_bin	scanning table and binning table management commands

### 3.5 Commands

The TIDI Real-time Command and Display System (RCDS) uses two mechanisms to create commands. The first is external to the OASIS application and is performed by shell scripts. This mechanism is used to generate TIDI Instrument Command Language command sequences and programs. The second mechanism is the built in OASIS command formation system mediated by the data base tables listed Table 4 (page 9).

#### 3.5.1 TICL Command Formation

TICL commands are formed in a multi-step process that may invoke a macro, a utility process, and a shell script. The process is illustrated in the following example (Figure 2). The macro cp\_start is used to command the TIDI instrument to start a previously loaded control program. The macro contains two CSTOL directives, a write directive to log the action and a run directive which runs a utility task, cpstart, that does the work. The cpstart utility task is a shell script that creates a temporary TICL source code file containing the TICL statements required to perform the desired action. In this case the source code contains an .immediate TICL compiler directive, a .purpose TICL compiler directive and the start\_cp TICL statement. The cpstart script next invokes the TICL compiler to create a command block file and either the sendto\_emu or sendto\_moc script to transmit the command message to the instrument via the spacecraft emulator or the TIDI mission operations center, respectively. The sendto scripts define environment variables required by the final script, sendCM, which invokes the package program to form one or more command messages from the command block file, uses PGP to encrypt them and then ftp to transmit them to the spacecraft ground system. The choice of sendto\_emu or sendto\_moc is made when the OASIS session is configured to communicate with the emulator or the MOC by updating the utility task table so that the utility task cpstart invokes either the cpstart\_emu or cpstart\_moc shell script.

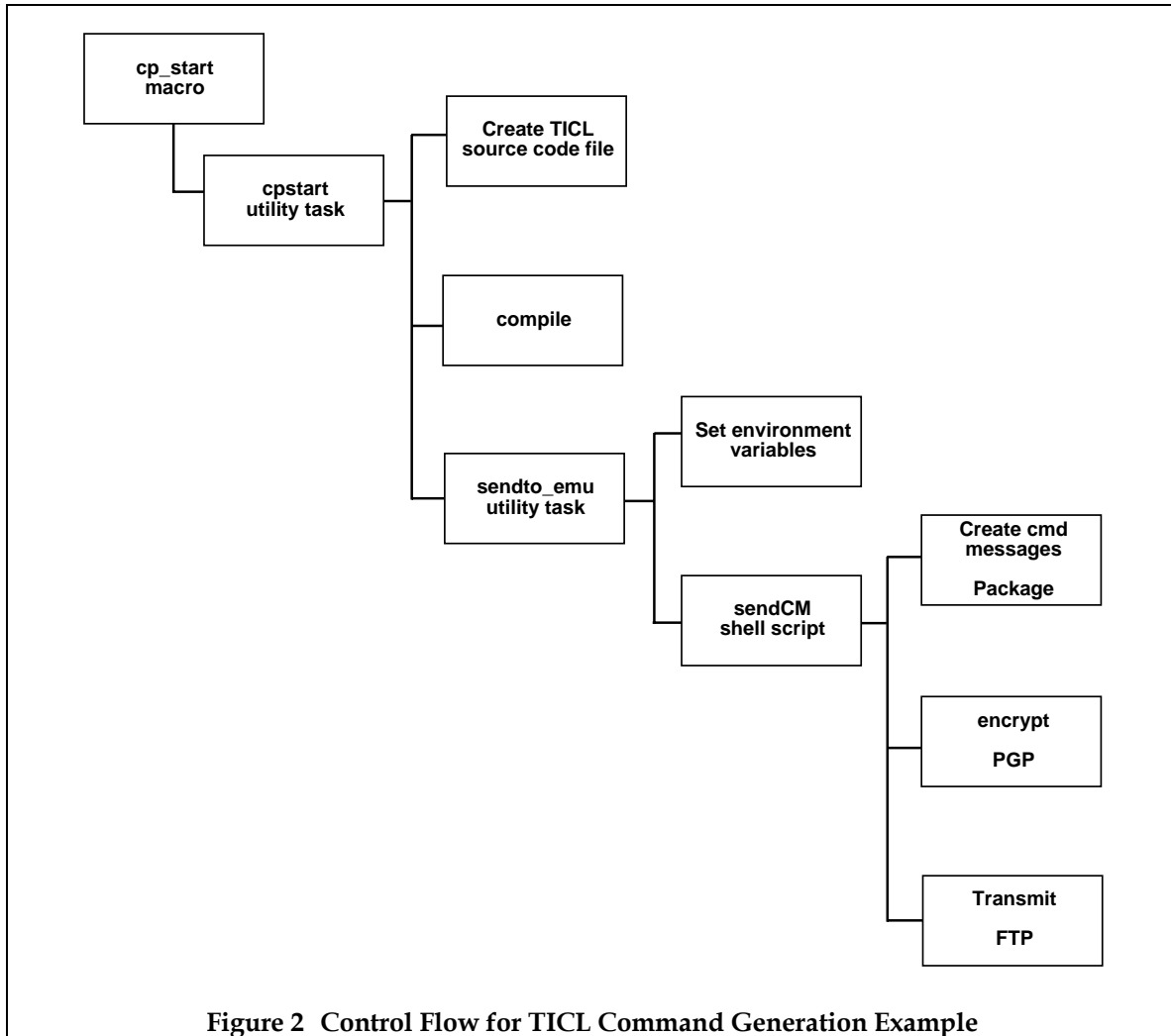
Other TICL commands are implemented in a similar fashion. The macros used for each of the commands are listed in Table 12, below. In each case the macro invokes a utility task which compiles one or more TICL statements, then invokes the sendto\_emu or then sendto\_moc utility task.

<b>Table 12</b> <b>TICL Command Macros</b>		
TICL Macro	Arguments	Descriptions
bootnow	0	boots instrument software
calc_crc	2	calculates the crc over a region of memory
cp_start	0	starts the stored control program
cp_stop	0	stops the active control program

**Table 12**  
**TICL Command Macros**

TICL Macro	Arguments	Descriptions
cp_valid	0	validates the control program holding buffer
dump_mem	2	dumps a region of memory
expire_wd	0	allows instrument watchdog timer rundown
load_bin	2	loads a binning table
load_mem	1	loads memory from an Intel HEX file
load_scan	1	loads a scan table from file
noboot	0	inhibits instrument software autoboot
poke_mem	4	writes a byte, word or double word to a memory address
pos_cal	1	Switches calibration lamp state
pos_fw	2	Positions a filter wheel
pos_shut	2	Positions a telescope shutter
pos_tel	1	Positions a telescope number elevation
report_globals	0	Reports control program globals
send_imm	1	Sends an arbitrary TICL command
set_ccd_dc	1	Sets the CCD heater duty cycle
set_fpa_dc	1	Sets sensor housing heater duty cycle
set_fw_dc	1	Sets filter wheel housing heater duty cycle
set_global	2	Sets the value of a control program global
set_pro_dc	2	Sets etalon \$1 heater duty cycle
set_tel_dc	2	Sets telescope \$1 heater duty cycle
start_scan	1	starts scan table execution
stop_scan	1	stops scan table execution





### 3.5.2 CSTOL Commands

Commands required to configure the MOC or spacecraft emulator telemetry server are implemented using the OASIS facilities. The commands are ASCII commands, which include a required parameter. The commands are listed in Table 13.

<b>Table 13 TM Server Commands</b>	
CSTOL Command	Function
SET APID TO XXXX	Selects the application ID to be returned by the TM server
TURN ON REALTIME_DATA	Starts the data flow
ENABLE DIRTY_DATA	Data marked bad is included in data returned by the TM server.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 18 of 46
--	--

<b>Table 13</b> <b>TM Server Commands</b>	
CSTOL Command	Function
SET FRONT_END TO XXXX	Selects the front end system from which data will be received
SET FRAME_SOURCE_ID TO XXXX	Selects data from an emulator, simulator, gse or spacecraft
SET SUBSYSTEM_ID TO XXXX	Selects a subsystem from which data is to be returned by the TM server (A range of APIDs)
SET PACKET_TYPE TO XXXX	Selects the type of packet returned: tp, stp, ptp, tf, stf
SET VIRT_CHAN_ID TO XXXX	Selects virtual channel, 6 for real-time, 7 for recorder playback
SET TLM_PORT TO XXXX	Specifies the port on the RCDS system to which the TM server will send data

The commands required to support the GSE system are implanted as serial commands. Both the generic serial command "send string to element" and specific commands are used. The generic serial command is used to control particular GSE sources that use commands compliant with Standard Commands for Programmable Instruments (SCPI) protocol. Specific commands are established for the diode laser to permit parameters to be calculated within a CSTOL procedure and passed to the instrument.

<b>Table 14</b> <b>GSE Commands</b>	
CSTOL Command	Function
ACTIVATE GSE	currently a no-op
TURN ON GSE	begins data collection from gse sources
TURN OFF GSE	ends data collection from gse sources
SEND "string" to da_switch	sends a command string to the data acquisition switch
SEND "string" to diode_laser	sends a command string to the diode laser
SEND "string" to dye_laser	sends a command string to the dye laser
SEND "string" to temp_monitor	sends a command string to the temperature monitor
TURN ON DIODE_LASER	switches on the diode laser output
TURN OFF DIODE LASER	switches off the diode laser output
SET WAVELENGTH TO XXXX.XX	sets the diode laser to a specific wavelength
SET DIODE_OFFSET FREQUENCY TO	adjusts the diode laser output wavelength by the specified offset in GHz

#### 4. Telemetry Decomposition

The TIDI RCDS decomposes three classes of telemetry: Spacecraft, Instrument and GSE. The spacecraft telemetry is in the form of CCSDS packets supplemented with information from the ground receipt header. These are the so-called POC Telemetry Packets or PTPs. They are fixed length packets whose contents are specified by text files provided by the TIMED project at

ftp://ftp.timed.jhuapl.edu/MDC/data\_products/telemetry\_definitions/AAA

where AAA is the Application ID. Instrument packets are variable length packets that are packed in fixed length spacecraft packets for transport. As described in Section 3.2 (page 10), the TIDI instrument packets are extracted and processed by OASIS to create a stream of variable length packets. The contents of these packets are specified in reference 6. GSE packets are fixed length ASCII encoded data packets. The contents of these packets are specified in reference 8.

The procedure `init_streams.prc` defines the primary streams, listed in Table 8, page 11 and the `tidi_extr` and `tidi_09` secondary streams.

To simplify the definition of telemetry items, the TIDI RCDS includes a number of macros. The macros are listed Table 15 in the along with their arguments. The `b_num`, `b_sta`, and `i_id` macros are used to define numeric telemetry points, state telemetry points, and numeric global values respectively. The numeric global values may be used as pseudo-telemetry points with a value set by an equation as a function of other telemetry points. They may also be used for communications between CSTOL procedures with values set by the CSTOL `let` directive.

Table 15 CSTOL Macros for Telemetry Decomposition		
macro name	parameters	purpose
<code>b_num</code>	\$1 external element for item being defined \$2 item name for item being defined \$3 parent stream name \$4 bit location of item \$5 length of time in bits \$6 units for item	builds the latest data and decomposition table entries for a numerical value
<code>i_ac</code>	\$1 external element \$2 item name \$3... \$8 polynomial coefficients \$9 units	builds the analog conversion table entries for a numerical value
<code>b_sta</code>	\$1 external element for item being defined \$2 item name for item being defined \$3 parent stream name \$4 bit location of item	builds the latest data and decomposition table entries for a state variable.

<b>Table 15</b> <b>CSTOL Macros for Telemetry Decomposition</b>		
macro name	parameters	purpose
	\$5 length of time in bits	
i_sc	\$1 external element \$2 item name \$3 raw state value to convert \$4 resulting state mnemonic \$5 desirability (good   bad)	builds a state conversion table entry for a value of a state variable
I_ld	\$1 external element for item being defined \$2 item name for item being defined \$3 data class \$4 units for item	builds a latest data table entry for an global value or a pseudo-telemetry point

#### 4.1 Spacecraft Packets

The decomposition of spacecraft tm packets is specified by the CSTOL procedures listed in Table 16. The application IDs listed in the table are associated with the spacecraft IEM side 1. For spacecraft IEM side 2, add 80H to the values.

From APID 4, the RCDS extracts the instrument status word and the spacecraft main bus voltage. From APID 6, the telescope cover release pyro status and the heater status are extracted. From APID 7, spacecraft current monitors and passive temperature monitors are extracted.

<b>Table 16</b> <b>Spacecraft TM Decomposition Definitions</b>		
CSTOL Procedure	Streams Defined	Description
load_isw.prc	sc_apid4	decomposes APID 4
load_sctm.prc	sc_apid6, sc_apid7	decomposes APID6, APID7.

The procedure load\_isw.prc defines the sc\_apid4 substream. The procedure load\_sctm.prc defines the sc\_apid6 and sc\_apid7 streams. The parent stream for all three of these secondary streams is the ptp\_ret\_stream primary stream. The primary stream is defined as a bit-wise packetized stream (ptp\_ret\_stream is\_packetized=2), so that locations and lengths are specified in bits, not bytes). The parent stream is receiving Time POC Telemetry Packets (PTPs), which consist of 22 bytes of ground receipt header followed by the telemetry packet.

To add a telemetry value from the spacecraft packets to those decomposed by the TIDI RCDS one begins by selecting a name by which the telemetry point will be know to the RCDS. Names consist of an external element name and an item name. The external element name "sc\_tm" has been

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 21 of 46
--	--

defined for all spacecraft telemetry values except those extracted from the instrument status word. The external element name "isw" has been defined for those.

Next the TIMED telemetry definition files must be consulted to determine the location of the telemetry point and the conversion from a digital number to a dimensional quantity or to a state. There are seven telemetry definition files for each application ID. These files may be found at the TIMED ftp site:

ftp://ftp.timed.jhuapl.edu/MDC/data\_products/telemetry\_definitions/AAA

Each file is named according to the convention specified in the GIIS, reference 11, and summarized in Table 17.

<b>Table 17</b>	
<b>Telemetry Definition Files Naming Convention</b>	
MDC__aid_effdate_rev_type.def	
field	description
MDC	A literal string indicating that this is a TIMED Mission Data Center file
aid	A three digit hexadecimal number indicating the application id (apid) being defined
effdate	The effective data of the data in the file
rev	The file two digit file revision number.
type	A literal string indicating one of 7 possible definition types alar           alarm definition basi           basic telemetry definition information calp           calibration pair switch point definitions coef           conversion coefficient definition loca           telemetry point location definition stat           state definitions user           additional user (TIMED) data
def	A literal string indicating that the file is a telemetry definition file.

Next the location of the item in the telemetry packet is obtained from the location telemetry definition file (loca). The location is specified in the telemetry definition file as a byte offset within a telemetry packet and a bit offset within the referenced byte. The location within the PTP packet, in bits, is computed:

$$L = 8O_{\text{byte}} + O_{\text{bit}} + 177$$

Where  $O_{\text{byte}}$  is the byte offset and  $O_{\text{bit}}$  is the bit offset within the byte. The value of 177 is the length of the Ground Receipt Header that is prefixed to the telemetry packet to form the PTP packet plus one to account for the OASIS convention of numbering bits beginning with one.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 22 of 46
--	--

Once the name has been selected and the position determined, the conversion from a digital number to an analog value must be identified for numerical values. This information is found in the conversion coefficient definition file (coef) for the application id. The file indicates the range of digital numbers for which a particular set of polynomial conversion coefficients are valid and the values of the coefficients.

For discrete state telemetry, the state definitions file indicates the state name to apply to each discrete value. These names are those used in the spacecraft telemetry displays, and can be replaced by more significant state names for the TIDI RCDS.

Once all of the information is collected, instances of the macros listed in Table 15 are added to the load\_isw procedure for items to be extracted from apid 4, or to the load\_sctm procedure for items to be extracted from apids 6 or 7. For numeric items, like voltages, currents or temperatures, one instance of the b\_num macro is used to define the telemetry point. The parent stream name will be sc\_apidn, where n is the application id, 4, 6, or 7. The units field will be either "any" or "dn". The literal "any" is used for any quantity that is to be converted to a dimensional quantity. The literal "dn" is used for items, such as counters or message numbers, that may remain as a digital value. Values for the other parameters are as determined above. If the units are specified as "any", an instance of the i\_ac macro is also required to specify the conversion to a dimensional quantity from the data number. The polynomial coefficients are taken from the coef file. Note that the TIMED system permits up to eight polynomial coefficients, while the TIDI RCDS permits up to six.

For state information, such as relay position readbacks or enable bits, one instance of the b\_sta macro is used to define the telmetry point. The parent stream name will be sc\_apidn, where n is the application id, 4, 6, or 7. Values for the other parameters are as determined above. To define state mnemonics for each of the values that the telemetry point may assume, one instance of the i\_sc macro is required for each value. The fourth parameter of the macro defines the string to use to represent the raw value specified in the third parameter. The desirablity indicates the color used to display the mnemonic when state checking is enabled in OASIS, green for good and red for bad.

Installation of the updated definition requires updating the OASIS data base tables, as described in section 7, page 35.

## **4.2 Instrument Packets**

TIDI instrument packets are extracted from the spacecraft packets with application ID 500H and placed in the tidi\_extr secondary stream. This stream is a forward stream routed to the tidi\_ret return stream. The tidi\_ret primary stream is decomposed into a series of packets, one for each TIDI telemetry packet defined in the flight software (reference 6). Each of these packets is decomposed in turn to obtain the individual telemetry data points. The decomposition for each of the TIDI telemetry packets is specified in the CSTOL procedures listed in Table 18

The TIDI housekeeping packet is specified in an Excel spreadsheet and an accompanying tab delimited text file version (reference 9). The load\_ipt.prc file is created by an automated procedure, described below. This scheme was selected so that the flight software, the spacecraft simulator and the TIDI RCDS could be kept consistent as the flight software design matured. Updates to the housekeeping data are made to the spreadsheet, then the spread sheet is parsed by the automated procedure and a new version of the load\_ipt.prc procedure is created. The remaining TIDI telemetry packets are defined in CSTOL procedures that were coded to the definitions in reference 6.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 23 of 46
--	--

Each TIDI telemetry packet contains header information consisting of a sync code, a packet type, a packet length and a time code. This information is decomposed from each packet into the `tidi_header` substream, which is further decomposed to individual atoms.

The spectral science data and the CCD image data consist of the header, common science data and either a spectra or a segment of an image. The common science data is decomposed from each of these packets to the `sci_status` substream, which is further decomposed to individual atoms.

Each of the packet definition procedures includes a statement that inserts a record in the streams table defining a stream whose name is `tidi_nn` as secondary stream with the `tidi_ret` stream as the parent. The string “nn” is the TIDI TM packet type. Also included is a statement that inserts a record in the decomposition table defining an occurrence of the `tidi_header` substream whose parent is the `tidi` packet being defined and whose occurrence number is the TIDI TM packet type for the packet. The type 0 science data packet definition, `load_tidi00.prc`, includes a additional statement that inserts a record in the decomposition table defining an occurrence of the science status substream whose parent is `tidi_00` and whose occurrence number is 0. Similarly `load_tidi02.prc` inserts a decomposition table entry for the science status substream with a parent of `tidi_02` and an occurrence number of 2.

Following the stream definition records each of the packet definition procedures include one or more instances of the macros listed in Table 16 to define the telemetry data points and their conversion from digital numbers to analog values or state mnemonics.

The decomposition of the TIDI header is specified in only one procedure, `init_streams.prc`. The decomposition of the science status stream is specified in the `load_tidi00.prc` procedure.

<b>Table 18</b> <b>Instrument Packet Definitions</b>			
packet id	stream name	defining CSTOL procedure	description
0	<code>tidi_00</code>	<code>load_tidi00.prc</code>	Spectral science data
2	<code>tidi_02</code>	<code>load_tidi02.prc</code>	CCD Image data
4	<code>tidi_04</code>	<code>load_ipt.prc</code>	Instrument housekeeping and status data
5	<code>tidi_05</code>	<code>load_tidi05.prc</code>	Command confirmation
6	<code>tidi_06</code>	<code>load_tidi06.prc</code>	Memory dump data
7	<code>tidi_07</code>	<code>load_tidi07.prc</code>	CRC calculation result
8	<code>tidi_08</code>	<code>load_tidi08.prc</code>	Flight software error reports
9	<code>tidi_09</code>	<code>init_streams.prc</code>	Null telemetry packets
10	<code>tidi_10</code>	<code>load_tidi10.prc</code>	Control program globals
n/a	<code>tidi_header</code>	<code>init_streams.prc</code>	Substream which occurs in each <code>tidi</code> packet containing the common header information from each TIDI packet

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 24 of 46
--	--

<b>Table 18</b> <b>Instrument Packet Definitions</b>			
packet id	stream name	defining CSTOL procedure	description
n/a	sci_status	load_tidi00	Substream which occurs in the tidi_00 and tidi_02 substream containing the common science status data.

#### 4.2.1 The TIDI Instrument Housekeeping Packet Definition

The procedure `load_ipt.prc` is created from the housekeeping telemetry packet definition stored in file 3519 on the TIDI document server. To create the procedure one must log on to the TIDI workstation and execute two scripts, as follows:

First, ensure that your active group membership is `tidi`, by typing the command

```
newgrp tidi
```

Next, make the `tidi` configuration area the current directory and execute the script to retrieve the tab delimited version of the telemetry packet definition:

```
cd /tidi/tidi_config
get_ipt
```

The updated telemetry definition is installed (for use by the `tmReader` package) by issuing the following command:

```
ln -s 3519VV_InstParam.txt InstParam.txt
```

where "VV" is the two character version identifier.

Next, the `load_ipt.prc` procedure is created by parsing the definition file that has been retrieved. This is done with the working directory set to the directory which contains all of the CSTOL procedures:

```
cd /tidi/oasis_appl/tidi_rcds/procedures
parse_ipt /tidi/tidi_config/3519VV_InstParam.txt
```

The final step is to link the newly created procedure file to that expected by the TIDI RCDS:

```
ln -s 3519vv_instparam.prc load_ipt.prc
```

Installation of the updated definition requires updating the OASIS data base tables, as described in section 7, page 35.

#### 4.2.2 Other TIDI Telemetry Packet Definitions

TIDI Telemetry packets other than the housekeeping packet are defined by the procedures listed in Table 18. In order to modify a telemetry definition, the appropriate file must be modified. It is not anticipated that any modifications to these files will be necessary.



University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 25 of 46
--	--

To add an item to a TIDI telemetry packet, determine the location of the item in the packet by referring to the flight software specification, reference 6. The locations are specified in that document as offsets from the start of the packet data, OASIS requires locations in bits from the start of the packet, so the location to be used in the `b_num` or `b_sta` macro is determined by:

$$L = 8O_{\text{byte}} + 81$$

where  $O_{\text{byte}}$  is the offset from the start of the data portion of the packet given in the specification and 81 is the length of the TIDI TM packet header plus one to account for the OASIS convention of numbering bits beginning with one. If the data point is an analog value, the polynomial coefficients required for the conversion to a dimensional quantity are obtained. If the data point is a state, appropriate mnemonics for the values are selected.

Once all of the information is collected, instances of the macros listed in Table 15 are added to the packet definition procedure. For numeric items, like voltages, currents or temperatures, one instance of the `b_num` macro is used to define the telemetry point. The parent stream name will be `tidi_nn`, where “nn” is the TIDI telemetry packet type. The units field will be either “any” or “dn”. The literal “any” is used for any quantity that is to be converted to a dimensional quantity. The literal “dn” is used for items, such as counters or message numbers, that may remain as a digital value. Values for the other parameters are as determined above. If the units are specified as “any”, an instance of the `i_ac` macro is also required to specify the conversion to a dimensional quantity from the data number.

For state information, such as relay position readbacks or enable bits, one instance of the `b_sta` macro is used to define the telemetry point. The parent stream name will be `tidi_nn`, where “nn” is the TIDI telemetry packet type. Values for the other parameters are as determined above. To define state mnemonics for each of the values that the telemetry point may assume, one instance of the `i_sc` macro is required for each value. The fourth parameter of the macro defines the string to use to represent the raw value specified in the third parameter. The desirability indicates the color used to display the mnemonic when state checking is enabled in OASIS, green for good and red for bad.

Installation of the updated definition requires updating the OASIS data base tables, as described in section 7, page 35.

### **4.3 GSE Packets**

TIDI GSE packets are extracted from the `gse_data` stream. This stream is produced by the GSEcomm application and transmitted over the `gse_return` link. The `gse_data` stream is decomposed into a series of packets, listed in Table 19, one for each item controlled by the GSEcomm application. Each of these packets is decomposed in turn to obtain individual telemetry data points. The decomposition of each of the GSE packets is specified in the `gse_setup.prc` CSTOL procedure. It is not anticipated that any modifications to this file will be necessary.

Each of the packets consists of a fixed length header in the same format as the TIDI instrument packet header followed by one or more data items. Each data item is a 32 byte formatted ASCII string padded on the right with the ASCII null character (0H). The data items are reported in engineering units, no analog conversions are required.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 26 of 46
--	--

<b>Table 19</b> <b>GSE Packet Definitions</b>			
packet id	stream name	defining CSTOL procedure	description
F8H	gse_f8	gse_setup.prc	Data acquisition switch data
FBH	gse_fb	gse_setup.prc	Lakeshore temperature monitor
F9H	tidi_f9	gse_setup.prc	Laser diode status
FA	tidi_fa	gse_setup.prc	Dye laser status - no longer used

To add an element to one of the GSE packets, you must first determine the location of the item in the packet and the length of the item by referring to the GSEcomm Software Interface Definition, reference 8. Numbering the first item after the packet time as item 1, the location of the data item is

$$L=256 N_{\text{item}} + 81$$

where  $N_{\text{item}}$  is the number of the item and 81 is the length of the TIDI TM packet header plus one to account for the OASIS convention of numbering bits beginning with one. The length of the item is specified in bits and for data from the Data Acquisition Switch includes only the reading, ignoring the comma delimiter and the channel number that follows the reading.

Once this information is determined, a decomposition table item is added to the procedure, specifying the external element, item name, location and length. In addition, the following three fields must be set as shown (Table 20):

<b>Table 20</b> <b>Required Decomposition Table Fields for GSE Packets</b>	
Field Name	Field Value
is_formatted	true
data_type	new_eu
format	Fortran style format specifier.

Installation of the updated definition requires updating the OASIS data base tables, as described in section 7, page 35.

## 5. Displays

The TIDI RCDS system makes use of both CSTOL windows and TAE+ panels. The windows are used for logging text messages and events. The panels are used for display of both housekeeping and science data. In addition to displaying data, some TAE+ panels are components of the user interface and include controls that result in commands for the instrument or for the gse being transmitted. TAE+ panels that display data require entries in the display\_descriptions and in the

display\_definitions tables. Those TAE+ panels that are part of the user interface and do not display data require no data base table entries.

The data displays defined in the TIDI RCDS system are listed in Table 21.

<b>Table 21</b> <b>Data Display Definitions</b>				
display	TAE+ panel	type	defining CSTOL procedure	description
fs_events	n/a	CSTOL	display_fs.prc	displays flight software messages.
limit_log	n/a	CSTOL	display_limits.prc	displays limit transitions
tidi_ccd	tidi_ccd	TAE+	display_ccd.prc	displays CCD control information
tidi_cp	tidi_cp	TAE+	display_cp.prc	displays control program execution status and control program globals
tidi_mem	tidi_mem	TAE+	display_dump.prc	displays partial contents of a memory dump packet
tidi_gse	tidi_gse	TAE+	display_gse.prc	displays gse status information
spectra1	spectra1	TAE+	display_hgm.prc	displays spectral science data packet, with CCD output as a histogram.
spectra2	spectra2			
spectra3	spectra3			
spectra4	spectra4			
spectra5	spectra5			
tidi_hs	tidi_hs	TAE+	display_hs.prc	displays health and safety data. Defines TAE+ items that are in all panel headers.
tidi_pro	tidi_pro	TAE+	display_pro.prc	displays profiler status
tidi_pwr	tidi_pwr	TAE+	display_pwr.prc	displays power subsystem status
timed_stat	sc_stat	TAE+	display_scstat.prc	displays spacecraft and instrument status information extracted from the spacecraft telemetry.
schart1	schart1	TAE+	display_stp.prc	each strip chart displays up to four operator selected data items.
schart2	schart2			
schart3	schart3			
schart4	schart4			
schart5	schart5			
tidi_tel	tidi_tel	TAE+	display_tel.prc	displays telescope subsystem status

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 28 of 46
--	--

<b>Table 21</b> <b>Data Display Definitions</b>				
display	TAE+ panel	type	defining CSTOL procedure	description
tidi_temp	tidi_temp	TAE+	display_temp.prc	displays thermal control subsystem status

### 5.1 CSTOL Pages

Adding a CSTOL page for text information is relatively simple, requiring only an appropriate entry in the display\_descriptions table. The entry should specify the name of the page, the location of the page and, if desired, a message filter discriminator. You place the statements to update this table into a procedure whose name is formed by concatenating "display\_" and the display name.

### 5.2 Modifying Fixed Format TAE+ Panels

To add an item to a TAE+ panel, the display resources must be modified using the TAE+ interface builder called the Actions Workbench (AWB) and entries must be added to the display\_definitions table with the information needed to connect TAE+ display items with OASIS variables. For details, beyond the summary provided here, refer to References 1 and 5

To modify the TAE+ panel, log into the TIDI RCDS development account by using OASIS1 or OASIS2 as the user name and the provided password. Once a UNIX session has been established, type the command

```
go_rcds
```

at the UNIX prompt in a terminal window. You next start the Action Workbench by typing the command

```
awb tidi_rcds -i &
```

at the UNIX prompt. The Action Workbench will start with all panels iconified. You select a panel to modify by clicking it, then referring to the instructions in Ref. 5 you can add, remove, move or modify display elements. If you add or remove dynamic text items, which are used to display telemetry values you must also modify the contents of the display\_definitions table. This table contains the TAE+ item name, the OASIS external element and item names and information on how the information is to be formatted.

Installation of the new display requires rebuilding the OASIS application as described in section 9, page 37 and updating the OASIS data base tables, as described in section 8, page 35.

### 5.3 Adding Fixed Format TAE+ Panels

Adding a TAE+ panel requires creating the display resources and populating the display\_descriptions the display\_definitions tables. You begin by creating the display resources using the Action Workbench. Log into the TIDI RCDS development account by using OASIS1 or

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 29 of 46
--	--

OASIS2 as the user name and the provided password. Once a UNIX session has been established, type the command

```
go_rcds
```

at the UNIX prompt in a terminal window. You next start the Action Workbench by typing the command

```
awb tidi_rcds -i &
```

at the UNIX prompt. The Action Workbench will start with all panels iconified.

To create a new panel, use the awb duplicate command to duplicate the template panel named "NoName01". Select "NoName01" by clicking on it. In the Action Workbench control panel, select duplicate from the edit menu. This will create a duplicate of the template named NoName02. Select this panel. Edit its specifications by selecting modify from the edit menu. Select a short name for the panel to place in the panel name field. Display panels are generally named by appending a 3 character identifier to the string "TIDI\_". Place a longer descriptive title in the title field. This title will appear in the window title bar. Include the panel name TIDI\_XXX in parenthesis at the end of the title. Next, click the "Edit Panel Details" button and specify the help file name and the icon title replacing NoName02 with the panel name.

Now that you've created a panel, you can begin filling it with display items. Note that the template includes the health and safety header. Do not modify this region. Add label text items for labels, dynamic text items for telemetry values and dynamic display objects for telldata. Note that OASIS produces formatted text strings for all telemetry values, whether numeric or interpreted state values.

After you've created the panel and filled it with display items, you must generate the interface code and build the application. To generate the interface code, select "Generate Code..." from the Auxiliary menu. Specify that the generated c code is to be placed in a single file and that it is to be K&R code. At this point, you may exit Action Workbench and build the application as described in section 9, page 37.

After you've completed defining the display resources, including TAE+ items for each data value to be specified, you must match a TAE+ item with an OASIS global. This is done by creating entries in the display\_descriptions table. Create a STOL procedure named display\_xxx, where "xxx" is the string used in the TAE+ panel name. Include statements to initialize a display\_definitions record specifying that the PAGE\_NAME is the same as TAE\_PANEL\_NAME. For each TAE+ item specified in the panel, you must include an entry in the display\_descriptions table. See ref. 1, chapter 9 and ref. 3 for details. You do not need to supply entries for the information originally included in the template. See the file \$OASIS\_PROCS/display\_pro.prc as an example.

Installation of the new display requires rebuilding the OASIS application as described in section 9, page 37 and updating the OASIS data base tables, as described in section 8, page 35.

#### **5.4 Wildcard Display TAE+ Panels**

The Wildcard Display is a user configurable display that allows up to 5 OASIS global variables to be selected by the user for tabular display. The operation of the Wildcard display is based on the example in the OASIS Quick Reference Manual, Reference 4.

## 5.5 Strip Chart TAE+ Panels

Strip Charts are TAE+ panels that can display a time history of up to 4 channels in either engineering units or digital numbers. The TIDI RCDS currently includes 5 strip charts and it is anticipated that no more will be required. This section describes the operation of the strip charts and sketches the creation of an additional instance.

### 5.5.1 Strip Chart Operation

The TAE+ Strip chart panels consist of a header area, containing the standard header information, four channel definition areas, containing a channel button, an external element text box, a data item text box, a data type radio button and a current value display. An example strip chart is shown in Figure 3, below.

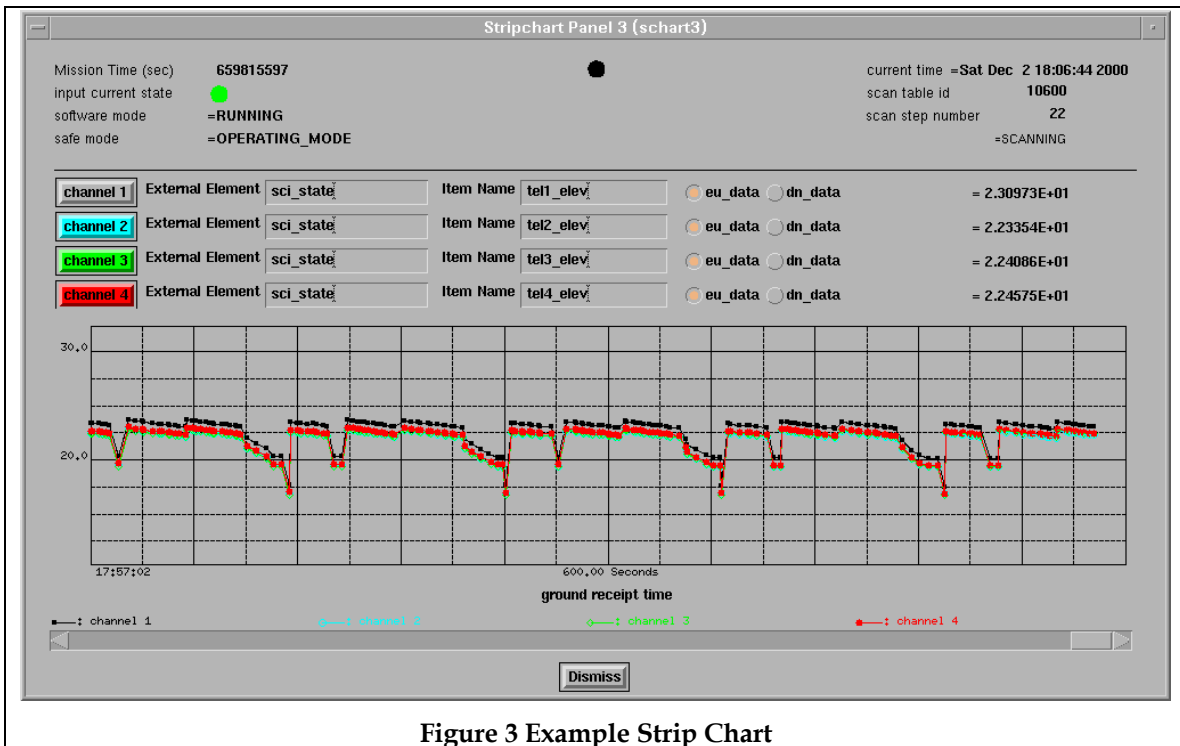


Figure 3 Example Strip Chart

The user enters the external element name and item name of the telemetry point to be displayed, selects engineering units or data number display type with radio buttons, then clicks the channel button which initializes the display. When that button is clicked, a TAE+ action is invoked which uses a macro to connect the OASIS data item with the TAE+ item used in the display. One TAE+ data item is used for each strip chart, with the index property specifying which of the four traces the OASIS data item is connected to. The independent variable for all strip charts is the OASIS system time. One TAE+ data item is used for each current value display on each strip chart.

Strip charts are implemented with one procedure, display\_stp.prc which initializes the strip charts, and several macros, strip\_value, strip\_chart1, strip\_chart2, strip\_chart3, and strip\_chart4 which are used to change the connection between the TAE+ item displayed and the OASIS data

value to be displayed. In addition each initializer buttons is connected to an action which invokes the corresponding strip chart macro and re-initializes the panel.

The display\_stp.prc procedure inserts a record into the display\_descriptions table for each strip chart. It creates a data item called strip\_chart dummy used as the initial item to be displayed in the strip charts. Since it has no value and is never updated, a strip chart display of the data item strip\_chart dummy will be blank. Entries are made in the display\_definitions for the TAE+ items scval1, scval2, scval3, scval4, scval5 for strip charts 1 through 5, respectively. Entries are also made for the TAE+ items which display the current numerical value of the item being displayed on the strip charts, *scn\_chnm* (n is the strip chart, 1 ... 5 and m is the channel 1...4)

When a user clicks the channel button on a strip chart, a TAE+ action occurs which invokes the strip chart macro and the strip value macro for the chart and channel whose button was clicked. The action further clears the panel and finally displays the panel again.

<b>Table 22</b> <b>CSTOL Macros for Strip Charts</b>		
macro name	parameters	purpose
strip_chart1	\$1 TAE+ item index, 1...4 \$2 external element for item being defined \$3 item name for item being defined \$4 type of data, EU or DN	Connects the specified data item to the TAE+ item for strip chart 1, scval1. The index number specifies which trace will be used to show the data.
strip_chart2	\$1 TAE+ item index, 1...4 \$2 external element for item being defined \$3 item name for item being defined \$4 type of data, EU or DN	Connects the specified data item to the TAE+ item for strip chart 2, scval2. The index number specifies which trace will be used to show the data.
strip_chart3	\$1 TAE+ item index, 1...4 \$2 external element for item being defined \$3 item name for item being defined \$4 type of data, EU or DN	Connects the specified data item to the TAE+ item for strip chart 3, scval3. The index number specifies which trace will be used to show the data.
strip_chart4	\$1 TAE+ item index, 1...4 \$2 external element for item being defined \$3 item name for item being defined \$4 type of data, EU or DN	Connects the specified data item to the TAE+ item for strip chart 4, scval4. The index number specifies which trace will be used to show the data.

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 32 of 46
--	--

<b>Table 22</b> <b>CSTOL Macros for Strip Charts</b>		
macro name	parameters	purpose
strip_chart5	\$1 TAE+ item index, 1..4 \$2 external element for item being defined \$3 item name for item being defined \$4 type of data, EU or DN	Connects the specified data item to the TAE+ item for strip chart 5, scval5. The index number specifies which trace will be used to show the data.
strip_value	\$1 TAE+ item name \$2 external element for item being defined \$3 item name for item being defined \$4 type of data, EU or DN	Connects the specified data item to the current value TAE+ display item, scn_chnm when n is the strip chart and m is the channel

### 5.5.2 Adding an additional strip chart

Adding a strip chart panel requires creating the display resources and populating the display\_descriptions the display\_definitions tables. You begin by creating the display resources using the Action Workbench. Log into the TIDI RCDS development account by using OASIS1 or OASIS2 as the user name and the provided password. Once a UNIX session has been established, type the command

```
go_rcds
```

at the UNIX prompt in a terminal window. You next start the Action Workbench by typing the command

```
awb tidi_rcds -i &
```

at the UNIX prompt. The Action Workbench will start with all panels iconified.

To create a new strip chart, use the awb duplicate command to duplicate the first strip chart panel named "schart1". Select "schart1" by clicking on it. In the Action Workbench control panel, select "Duplicate" from the TAE+ edit menu. This will create a duplicate of the template named NoName02. Select this panel. Edit its specifications by selecting "Modify" from the TAE+ edit menu. Name the panel *schartn*, where *n* is the next number in the sequence of strip charts already created. Change the title to reflect the strip chart number. Next, click the "Edit Panel Details" button and specify the help file name and the icon title replacing NoName02 with the panel name.

Change the name of each current value item to *scn\_chnm* when *n* is the strip chart number and *m* is the channel number by clicking the item to select it and choosing "Modify" from the TAE+ edit menu and changing the item name.

Change the name of the TAE+ item for the graph by selecting the graph by clicking it and choosing "Modify" from the TAE+ menus. Change the name to *scvaln* where *n* is the strip chart number.



University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 33 of 46
--	--

Now choose "Define Connections" on the actions workbench panel and select the channel 1 button. Select actions from the dialog that appears. A text edit window will appear with TAE+ action code in it which uses the Cstolq directive to form cstol macros that will be invoked when the channel button is selected. The code must be changed to refer to the new strip chart number . Change the string "strip\_chart1" to "strip\_chartn". Change each occurrence of the string "schart1" to "schartn". Next change the strip value macro argument "sc1\_chn1" to "scn\_chnn". Close the text editing box and the actions dialog.

Repeat the action modification for the remaining channel buttons.

Now a button to bring up the new strip chart must be added to the "Graphics" menu. To do this, select "Move" from the workbench panel. Select the icon named "graphics" and restore it. Without changing its location on the screen, lengthen the panel to provide space for another strip chart button and move the "Dismiss" button to the bottom of the panel. Duplicate the "Strip Chart 1" button and move the copy to the bottom of the panel. Select the copy and choose "Modify" from the TAE+ edit menu. Change the name to "sc1\_scn" and the title to "Strip Chart \nn. To connect this button with the new strip chart, select "Define Connections" from the TAE+ panel and click in the new button. Change the name that appears in the next panel window to the name of the new strip chart.

After you've created the strip chart panel, you must generate the interface code and build the application. To generate the interface code, select "Generate Code..." from the Auxiliary menu. Specify that the generated c code is to be placed in a single file and that it is to be K&R code. At this point, you may exit Action Workbench and build the application as described in section 9, page 37.

You must now initialize the strip chart by adding lines to the display\_stp.prc procedure. Duplicate the lines used for strip chart 1 and change the strip chart number to the number of the new strip chart. You will have 5 directives referring to "scvaln" and 4 directives referring to "scn\_chnn". You also create a macro named strip\_chartn. Use strip\_chart1 as a model.

Installation of the new display requires rebuilding the OASIS application as described in section 9, page 37 and updating the OASIS data base tables, as described in section 8, page 35.

## **5.6 Spectral Display TAE+ Panel**

Intentionally omitted

## **5.7 Command Panels**

Command panels provide the interface between the user, OASIS, and the instrument. The panels listed in Table 23 provide control of the instrument. In general, each panel has a series of TAE+ interface items which are used to collect information from the operator and an execute button, usually labeled SEND, which is connected to a TAE+ action. The action invokes a macro to create and transmit a TICL control program to perform the indicated action. The table lists the panel name, its purpose, and the command macros that are invoked. The details of the command macros are provided in Table 12, on page 15.

<b>Table 23 Command Interface Panels</b>		
Panel Name	Purpose	Macros Invoked
boot_cmd	Start and stop instrument software	bootnow noboot expire_wd
cntl_pgm	control execution of stored control programs and interact with the control program interpreter	cp_start cp_stop cp_valid report_globals send_imm set_global
heater	set duty cycle for heaters and disable automatic control	set_ccd_dc set_fw_dc set_pfa_dc set_pro_dc set_tel_dc
mech	Command mechanisms to specified position	pos_cal pos_fw pos_shut pos_tel
mem_cmds	memory updates and dumps	calc_crc dump_mem load_mem poke_mem
scan_bin	load tables and control execution of scan tables	load_bin load_scan start_scan stop_scan
send_cmd	selects a prepared command block file and transmits it.	send_cm

Adding a command to any of the panels requires creating a command macro to form ,compile and transmit the TICL commands required to perform the desired action. The display resources must next be modified using the TAE+ interface builder called the Actions Workbench (AWB).

To modify the TAE+ command panel, log into the TIDI RCDS development account by using OASIS1 or OASIS2 as the user name and the provided password. Once a UNIX session has been established, type the command

```
go_rcds
```

at the UNIX prompt in a terminal window. You next start the Action Workbench by typing the command

```
awb tidi_rcds -i &
```

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 35 of 46
--	--

at the UNIX prompt. The Action Workbench will start with all panels iconified. You select a panel to modify by clicking it, then referring to the instructions in Ref. 5 you can add, remove, move or modify interface elements. The elements should be of text presentation category and keyin presentation type. You can use the details option to specify a length of the input and an error message to display if the value is out of range. You place AWB in the "define connections" mode by selecting the appropriate radio button in the TAE+ control dialog.

Installation of the new display requires rebuilding the OASIS application as described in section 9, page 37 and updating the OASIS data base tables, as described in section 8, page 35.

## 6. Commands

### 6.1 Telemetry Server Commands

### 6.2 Real-Time TICL Commands

### 6.3 GSE Commands

## 7. Command Packaging and Transmission

## 8. Updating the OASIS Database Tables

The data base tables for the TIDI RCDS are populated by a series of CSTOL procedures. This activity can be quite time consuming, but OASIS includes facilities to create an initial data base table load. The initial load is contained in a collection of files located in the \$OASIS\_DATABASE directory. These files can be created by checkpointing the current table contents. Once created, the data base tables are loaded at OASIS startup with the configuration specified by the procedures.

The procedure for creating an updated data base configuration begins by ensuring that there will be a macros table entry for each macro in the \$OASIS\_MACROS directory and an procedures table entry for each procedure in the \$OASIS\_PROCS directory. You do this by issuing the following commands from the unix prompt:

```
makeDefineMacros
makeDefineProcs
```

These commands scan the directories containing the macros and procedures, respectively, and create procedures in the \$OASIS\_PROCS directory which fill the macros and procedures tables. Previous versions of the procedures are retained with the strings ".old" and ".older" appended to their names.

The next step is to restore the original initial table configuration. This is done so that the table build proceeds from a known valid state. To restore the initial state, you issue the following commands from the unix prompt:

```
go_rcds
cd $OASIS_DATABASE
cd distribution
cp *.dat ..
```

This sequence of commands copies the data base tables, as distributed with OASIS into the active area, completing the preparation for updating the tables

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 36 of 46
--	--

The next steps are performed within the RCDS application. Start OASIS with the following commands at the unix prompt:

```
go_rcds
oasis
```

Once OASIS is started, click in the CSTOL\_PROMPT window and issue the command

```
start init_tidi
```

This command executes the procedures created by makeDefineMacros and makeDefineProcs to fill the macros and procedures tables. Next, the number of command line interpreters (CLIs) and user privileges must be set. Enter the following commands in the CSTOL\_PROMPT window to perform this step:

```
let user_privs has_been_set=0dn
start user_privs
```

When prompted by a message in the message window, click in the CSTOL\_PROMPT window and type

```
go
```

The remaining tables are loaded and checkpointed by issuing the following commands. Click in the CSTOL\_PROMPT window and type:

```
start load_tables
```

When the procedure finishes, perform the checkpointing by typing

```
checkpoint all to OASIS_CHECKPOINT
```

The case of the last token on the line is significant.

At this point a set of initial table load files exist in the directory \$OASIS\_CHECKPOINT. They need to be copied to the location from which OASIS will load them and they need to be preserved. These steps are performed from the unix prompt. Click in a terminal window and type the following commands:

```
go_rcds
cd $OASIS_CHECKPOINT
cp *.dat $OASIS_DATABASE
mkdir ver_YYYYMMDD
mv *.dat ver_YYYYMMDD
```

where "YYYY" is the four digit year number, "MM" is the two digit month number (01, 02, ...,12), and "DD" is the two digit day of the month (01, 02, ..., 31).

The initial table load is now complete and configured .

**9. Building the OASIS Application**

**10. Supporting Utilities**

**10.1 makeDefineMacros**

**10.2 makeDefineProcs**

**10.3 make\_rcds**

**11. Global Variables**

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
\$\$GLOBAL	\$\$CEV_MODE	
\$\$GLOBAL	\$\$COMMAND_ECHO	
\$\$GLOBAL	\$\$COMMAND_MODE	
\$\$GLOBAL	\$\$LIMITS_MODE	
\$\$GLOBAL	\$\$LIMITS_WAIT	
\$\$GLOBAL	\$\$POSTCHECK_MODE	
\$\$GLOBAL	\$\$PRECHECK_MODE	
\$\$GLOBAL	\$\$STATES_MODE	
\$\$GLOBAL	\$\$TIME_STAMP	
\$\$GLOBAL	\$\$UED_MODE	
CLP_1	ERROR	
CLP_1	PROCEDURE_NAME	
CLP_1	STATUS	
CLP_2	ERROR	
CLP_2	PROCEDURE_NAME	
CLP_2	STATUS	
CLP_3	ERROR	
CLP_3	PROCEDURE_NAME	
CLP_3	STATUS	
CLP_4	ERROR	
CLP_4	PROCEDURE_NAME	
CLP_4	STATUS	
CLP_5	ERROR	
CLP_5	PROCEDURE_NAME	

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
CLP_5	STATUS	
CLP_6	ERROR	
CLP_6	PROCEDURE_NAME	
CLP_6	STATUS	
CMD	ACCEPTANCE	
CMD	HAS_BEEN_SENT	
CMD	HAS_VERIFIED	
CMD_CONF	SSC	
CMD_CONF	TIME	
CMD_CONF	VTIME	
CONTROL_PRGM	VAR_TIME	
CONTROL_PRGM	VAR_VTIME	
COUNTS	BIN_nnn	
COUNTS	BIN_COUNT	
COUNTS	GAIN_nnn	
COUNTS	MEAN	
COUNTS	PACKET_LEN	
COUNTS	SIGMA	
COUNTS	TIME	
CRC	BYTE_COUNT	
CRC	START_ADDR	
CRC	TIME	
CRC	VALUE	
CRC	VTIME	
DATA_CMDS	MODE	
DATA_CMDS	SOCKET_ADDRESS	
DIODE_LASER	CAVITY_TEMP	
DIODE_LASER	DIODE_CURRENT	
DIODE_LASER	DIODE_TEMP	
DIODE_LASER	EFF_TFRACT	

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
DIODE_LASER	EFF_TIME	
DIODE_LASER	FREQ_OFFSET	
DIODE_LASER	OP_COMPLETE	
DIODE_LASER	OUTPUT_POWER	
DIODE_LASER	PIEZO_VOLT	
DIODE_LASER	REAR_FACET_POWER	
DIODE_LASER	STATE	
DIODE_LASER	WAVELENGTH	
DIODE_OFFSET	FREQUENCY	
DYE_LASER	PACKET_SUBTYPE	
DYE_LASER	SCAN_INTERVAL	
DYE_LASER	SCAN_RATE	
DYE_LASER	SCAN_STATE	
DYE_LASER	SCAN_TFRACT	
DYE_LASER	SCAN_TIME	
DYE_LASER	WAVELENGTH	
ERROR	CODE	
ERROR	PAR1	
ERROR	PAR2	
ERROR	PAR3	
ERROR	PAR4	
ERROR	STRING	
ERROR	TIME	
ERROR	VTIME	
EVENT_MESSAGE	RECORD_FILENAME	
EVENT_MESSAGE	RECORDING	
EXT_COMM	BFL_RETRIES	
EXT_COMM	GFL_BLOCKS	
EXT_COMM	GRL_BLOCKS	
EXT_COMM	GRL_RETRIES	

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
EXT_COMM	MDC_BLOCKS	
EXT_COMM	MDC_RETRIES	
EXT_COMM	TFL_BLOCKS	
EXT_COMM	TFL_RETRIES	
EXT_COMM	TRL_BLOCKS	
EXT_COMM	TRL_RETRIES	
GSE	AMBIENT_PRES	
GSE	AMBIENT_TEMP_1	
GSE	AMBIENT_TEMP_2	
GSE	BASE1_TEMP	
GSE	BASE2_TEMP	
GSE	CF_TEMP	
GSE	CHAMBER_TEMP_6	
GSE	CSTRAP_TEMP	
GSE	DIODE_TEMP_8	
GSE	EFF_TFRACT	
GSE	EFF_TIME	
GSE	FOOT1_TEMP	
GSE	FOOT2_TEMP	
GSE	HRBACK_TEMP	
GSE	HRBOTTOM_TEMP	
GSE	HRLEFT_TEMP	
GSE	HRRIGHT_TEMP	
GSE	HRTOP_TEMP	
GSE	HSG_TEMP	
GSE	OPM_GAIN	
GSE	OPTICAL_POWER	
GSE	SHUTTER_ENABLE	
GSE_CMDS	MODE	
GSE_CMDS	SOCKET_ADDRESS	



<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
GSE_DATA	DEBUG_LEVEL	
GSE_DATA	ID_OFFSET	
GSE_DATA	ID_SIZE	
GSE_DATA	IS_PACKETIZED	
GSE_DATA	LENGTH_CORRECTI O	
GSE_DATA	LENGTH_OFFSET	
GSE_DATA	LENGTH_SIZE	
GSE_DATA	LENGTH_UNIT	
GSE_DATA	MODE	
GSE_DATA	SOCKET_ADDRESS	
HISTOGRAM	DUMMY	
ISW	ALIVE	
ISW	ATRIGGER	
ISW	BAD_CNT	
ISW	BOOT	
ISW	CAL_LAMPS	
ISW	CAL1	
ISW	CAL2	
ISW	CAL3	
ISW	CAL4	
ISW	ERR_CNT	
ISW	ERROR_SENT	
ISW	FW1_PERROR	
ISW	FW1_POS	
ISW	FW2_PERROR	
ISW	FW2_POS	
ISW	ILIMIT	
ISW	MECH_BUSY	
ISW	MISSION_TIME	
ISW	REC_CNT	

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
ISW	SCAN_ID	
ISW	SW_MODE	
ISW	TEL1_PERROR	
ISW	TEL1_SHUTTER	
ISW	TEL2_PERROR	
ISW	TEL2_SHUTTER	
ISW	TEL3_PERROR	
ISW	TEL3_SHUTTER	
ISW	TEL4_PERROR	
ISW	TEL4_SHUTTER	
ISW	WORD1	
ISW	WORD2	
ISW	WORD3	
ISW	WORD4	
MEM_DUMP	ADDR_1	
MEM_DUMP	ADDR_2	
MEM_DUMP	ADDR_3	
MEM_DUMP	ADDR_4	
MEM_DUMP	BLANK_BYTE	
MEM_DUMP	BYTE_COUNT	
MEM_DUMP	BYTE <sub>nnn</sub>	
MEM_DUMP	LAST_ADDR	
MEM_DUMP	PACKET_SIZE	
MEM_DUMP	START_ADDR	
MOP_TDSP	DATE_TIME	
MOP_TDSP	TAE_TIME	
PTP_RET_STREAM	DEBUG_LEVEL	
PTP_RET_STREAM	ID_OFFSET	
PTP_RET_STREAM	ID_SIZE	
PTP_RET_STREAM	IS_PACKETIZED	

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
PTP_RET_STREAM	LENGTH_CORRECTI O	
PTP_RET_STREAM	LENGTH_OFFSET	
PTP_RET_STREAM	LENGTH_SIZE	
PTP_RET_STREAM	LENGTH_UNIT	
PTP_RET_STREAM	MODE	
PTP_RET_STREAM	SOCKET_ADDRESS	
SC_TM	CS_PYRO_PRI_SEL	
SC_TM	CS_PYRO_SEC_SEL	
SC_TM	EBOX_TEMP	
SC_TM	INST_CURR	
SC_TM	INST_POWER	
SC_TM	MAIN_BUS_VOLT	
SC_TM	OP_HTR_CURR	
SC_TM	OP_HTR_POWER	
SC_TM	OP_HTR_STATUS	
SC_TM	PKT_STORE_COUNT	
SC_TM	PKT_STORE_LIMIT	
SC_TM	PRO_IF_TEMP	
SC_TM	PRO_TEMP	
SC_TM	SV_HTR_CURR	
SC_TM	SV_HTR_POWER	
SC_TM	SV_HTR_STATUS	
SC_TM	TEL1_IF_TEMP	
SC_TM	TEL1_PED_TEMP	
SC_TM	TEL2_IF_TEMP	
SC_TM	TEL2_PED_TEMP	
SC_TM	TEL3_IF_TEMP	
SC_TM	TEL3_PED_TEMP	
SC_TM	TEL4_IF_TEMP	
SC_TM	TEL4_PED_TEMP	

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 44 of 46
--	--

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
SC_TM	WS_PYRO_PRI_SEL	
SC_TM	WS_PYRO_SEC_SEL	
SCI_STATE	BIN_TABLE_ID	
SCI_STATE	DATA_INVALID	
SCI_STATE	ELV_ERROR	
SCI_STATE	EXPOSE_COUNT	
SCI_STATE	FW_ERROR	
SCI_STATE	FW_TIME_ERROR	
SCI_STATE	FW1_POS	
SCI_STATE	FW1_POS_ERROR	
SCI_STATE	FW2_POS	
SCI_STATE	FW2_POS_ERROR	
SCI_STATE	IN_SAA	
SCI_STATE	INT_TIME	
SCI_STATE	LAMP_STATUS	
SCI_STATE	PIXEL_NUMBER	
SCI_STATE	SCAN_TABLE_ID	
SCI_STATE	SHUT_TIME_ERROR	
SCI_STATE	SHUT1_POS	
SCI_STATE	SHUT2_POS	
SCI_STATE	SHUT3_POS	
SCI_STATE	SHUT4_POS	
SCI_STATE	SUN_AVOID	
SCI_STATE	TEL_TIME_ERROR	
SCI_STATE	TEL1_ELEV	
SCI_STATE	TEL2_ELEV	
SCI_STATE	TEL3_ELEV	
SCI_STATE	TEL4_ELEV	
SPACECRAFT	MASTER_WARN	
STRIP_CHART	DUMMY	

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
SUB_CLP	NUMBER	
TEL_1	OC_LATCH	
TEL_2	OC_LATCH	
TEL_3	OC_LATCH	
TEL_4	OC_LATCH	
TICL	CMD_LINE	
TIDI_FWD	DEBUG_LEVEL	
TIDI_FWD	MODE	
TIDI_FWD	SOCKET_ADDRESS	
TIDI_PACKET	LENGTH	
TIDI_PACKET	SYNC	
TIDI_PACKET	TFRACT	
TIDI_PACKET	TIME	
TIDI_PACKET	TYPE	
TIDI_RET	DEBUG_LEVEL	
TIDI_RET	ID_OFFSET	
TIDI_RET	ID_SIZE	
TIDI_RET	IS_PACKETIZED	
TIDI_RET	LENGTH_CORRECTI O	
TIDI_RET	LENGTH_OFFSET	
TIDI_RET	LENGTH_SIZE	
TIDI_RET	LENGTH_UNIT	
TIDI_RET	MODE	
TIDI_RET	SOCKET_ADDRESS	
TIMED	IEM_SIDE	
TMLG	C_MSG	
TMLOGGER	CONFIG	
USER_CLP	ERROR	
USER_CLP	PROCEDURE_NAME	
USER_CLP	STATUS	

University of Michigan Space Physics Research Laboratory <b>TIDI RCDS</b> <b>Design and Maintenance</b>	Drawing No. 055-3957 Filename 3957X2-RCDS Design&Maint Page 46 of 46
--	--

<b>Table 24</b> <b>OASIS Global Variables</b>		
External Element	Item Name	Description
USER_PRIVS	HAS_BEEN_SET	
WC_DISPLAY	DUMMY	
WC_DISPLAY	LABEL_1	
WC_DISPLAY	LABEL_2	
WC_DISPLAY	LABEL_3	
WC_DISPLAY	LABEL_4	
WC_DISPLAY	LABEL_5	