

**SANDIA REPORT**

SAND97-1271

Unlimited Release

Printed December 1997

**RECEIVED**

DEC 29 1997

**OSTI****TAOS/LS1 Development Final Report**

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Gary S. Phipps, Stephen M. Gentry, John M. Falls, Paul J. Claassen, George J. Alder

Prepared by

Sandia National Laboratories

Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,  
a Lockheed Martin Company, for the United States Department of  
Energy under Contract DE-AC04-94AL85000.

**MASTER**

Approved for public release; further dissemination unlimited.

**Sandia National Laboratories**

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

**NOTICE:** This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from  
Office of Scientific and Technical Information  
P.O. Box 62  
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from  
National Technical Information Service  
U.S. Department of Commerce  
5285 Port Royal Rd  
Springfield, VA 22161

NTIS price codes  
Printed copy: A05  
Microfiche copy: A01

## **DISCLAIMER**

**Portions of this document may be illegible  
electronic image products. Images are  
produced from the best available original  
document.**

SAND97-1271  
Unlimited Release  
Printed December 1997

## **TAOS/LS1 Development Final Report**

Gary S. Phipps, Stephen M. Gentry  
Optics and Emerging Technologies Department  
John M. Falls, Paul J. Claassen, and George J. Alder  
Sensors and Electronics Department  
Sandia National Laboratories  
P. O. Box 5800  
Albuquerque, NM 87185-0980

### **Abstract**

The Laser Sensor #1 (LS1) is a system designed and built by Sandia to detect and report laser illumination of an orbiting satellite. It was launched March 1994 as part of the U.S. Air Force Phillips Laboratory, Technology for Autonomous Operational Survivability (TAOS) satellite program. The engineering details of the system are described in this report. Operation characteristics and results have been reserved for inclusion in a classified Air Force report prepared by the TAOS Program Office of Phillips Laboratory.

**Intentionally Left Blank**

## Table of Contents

<b>TAOS/LS1 DEVELOPMENT FINAL REPORT.....</b>	<b>9</b>
<b>1.0 LS1 SYSTEM.....</b>	<b>9</b>
1.1 LASER SENSOR #1 DESCRIPTION .....	9
1.2 LS1 TEAM.....	10
1.3 DESIGN OVERVIEW.....	11
1.4 SYSTEM SPECIFICATIONS .....	12
1.5 STATE-OF-HEALTH .....	14
1.5.1 Analog and Digital Monitors .....	14
1.5.2 Discrete States .....	18
1.6 POWER .....	19
1.7 MECHANICAL DESIGN .....	23
1.8 OPTICAL DESIGN .....	27
1.8.1 Theoretical Basis .....	27
1.8.2 Practical Implementation of the Slit Grating Optical Design .....	31
1.9 ELECTRONIC DESIGN.....	37
1.9.1 Analog Electronics.....	37
Sensor Array Analog (SAA) Module.....	39
1.9.2 Digital Electronics.....	40
T800 Transputer.....	40
CLC.....	40
Clock Divider.....	41
The Silicon Array Processor (SAP) .....	42
Preprocessor Signal Interface Module (PSI).....	44
<b>2.0 SIGNAL PROCESSING.....</b>	<b>47</b>
2.1 LS1 EVENT SIGNAL PROCESSING SOFTWARE (ESP).....	47
Auto-Threshold Determination .....	48
2.2 EVENT DETECT/DECODE ALGORITHMS .....	51
<b>3.0 TESTING.....</b>	<b>53</b>
3.1 ENVIRONMENTAL .....	53
Ambient Thermal Tests:.....	53
Vibration:.....	53
Pyroshock: .....	53
Thermal Vacuum: .....	53
Results: .....	54
3.2 EMC TESTING.....	54
3.3 TAOS AUTOMATED TEST UNIT GROUND STATION (TATU).....	54
<b>4.0 COSTS .....</b>	<b>55</b>
<b>5.0 SCHEDULE.....</b>	<b>56</b>
<b>6.0 LS1 SYSTEM OPERATIONS .....</b>	<b>57</b>
6.1 LASER SENSOR OBJECTIVES .....	57
6.2 LS1 OPERATIONAL MODES .....	57
Event Detection Mode .....	58
Triggered Background Mode .....	58
Commanded Background Mode.....	59
Periodic background Mode.....	59
Image and Flood Illumination Calibration .....	60
6.3 LS1 PERFORMANCE ASSESSMENT.....	61
6.4 LS1 LESSONS LEARNED .....	62

MIL Fabrication.....	62
Quality Monitoring.....	62
Borrowing Modules from Other Systems.....	62
Redundancy and Cross-Strapping.....	62
Ground and Power Planes.....	63
High-Frequency Electronics.....	63
Sensor Noise Limits.....	63
Communication Protocols.....	63
Silicon Arrays.....	64
Structured Software Analysis.....	64
Low Earth Orbit Spacecraft.....	64
Lenses.....	64
Thermal Modeling.....	64
Reprogramming Connector.....	64
<b>APPENDICES.....</b>	<b>65</b>
<b>A1 BOX STRUCTURE .....</b>	<b>65</b>
<b>A2 OPTICAL DETAILS .....</b>	<b>66</b>
A2.1 OPTICAL SUBASSEMBLY.....	66
A2.2 TORIC LENS.....	67
A2.3 FIELD STOP.....	68
<b>A3 LS1 TEST SYSTEM AND DATA PROCESSING SOFTWARE .....</b>	<b>69</b>
A3.1 INTRODUCTION.....	69
A3.2 TAOS/LS1 AUTOMATED TEST UNIT (TATU).....	69
A3.2.1 Hardware Description.....	69
A3.2.2 Software.....	69
Screen.....	69
Definition of TATU Commands.....	70
Description of command arguments:.....	74
Function Keys.....	74
Command Files.....	74
Outputs.....	74
A3.3 PLOT UTILITY.....	75
Inputs.....	75
Outputs.....	75
Program Execution.....	75
Blowing Up a Message - Using F1.....	76
Using F2.....	76
State-of-Health and System Configuration Messages.....	76
Raw Data Messages.....	76
Finding a Message Type.....	77
Printing the Browse Window.....	78
Find Sequence Anomalies.....	78
Toggling between the Command and Browse Windows.....	78
Errors.....	78
Exiting PLOT.....	78
Description of PLOT Software Routines:.....	79
A3.4 FLIGHT DATA PROCESSING SOFTWARE.....	79
A3.4.1 PARSE_15.....	79
A3.4.2 LOCATE.....	80
A3.5 DECLASSIFICATION SOFTWARE.....	80
A3.5.1 SANITIZE.....	81
A3.5.2 CHECK.....	82
A3.5.3 JOIN.....	82
A3.6 OTHER UTILITY PROGRAMS.....	82

A3.6.1	TRCOPY .....	82
A3.6.2	SPLIT .....	82
A3.6.3	SEARCH .....	83
A3.6.4	STRIPOUT .....	83
A3.7	MAIN TATU SOFTWARE MODULES .....	84
	MAIN.C .....	84
	KBD_MGR.C .....	84
	CMDINTRP.C .....	85
	CMDFILES.C .....	85
	CREATMSG.C .....	85
	OUT_MGR.C .....	85
	IN_MGR.C .....	85
	IN_QUEUE.C .....	85
	MSGFILTR.C .....	85
	MSGFILES.C .....	85
	SCREEN.C .....	85
	EXCEPT.C .....	85
	SERIAL.C .....	86
	TIMEMARK.C .....	86
	PROTOBRD.C .....	86
	NUMBERS.C .....	86
	IEEE_488.C .....	86
	IEEEIO.C .....	86
A4.0	LS1 SYSTEM CHECK-OUT .....	87
A4.1	TEST OBJECTIVES, METHODS AND CONSTRAINTS .....	87
A4.2	EVENT MODE TESTING .....	88
A4.3	PERIODIC BACKGROUND MODE TESTING .....	92
A4.4	TRIGGERED BACKGROUND MODE TESTING .....	94
	DISTRIBUTION: .....	98



## List of Figures

FIGURE 1: LS1 BLOCK DIAGRAM .....	11
FIGURE 2: ANALOG OUTPUT INTERFACE .....	13
FIGURE 3: BI-LEVEL LS1 TO S/C INTERFACE CIRCUIT .....	19
FIGURE 4: LS1 POWER SYSTEM BLOCK DIAGRAM .....	20
FIGURE 5: BI-LEVEL S/C TO LS1 INTERFACE CIRCUIT .....	23
FIGURE 6: LS1 HOUSING .....	24
FIGURE 7: LS1 MODULE LAYOUT .....	24
FIGURE 8: LS1 MECHANICAL REFERENCE SYSTEM .....	26
FIGURE 9: SLIT-GRATING/LINEAR ARRAY SUBASSEMBLY WITH SLIT IMAGE LOCATIONS .....	27
FIGURE 10: TWO ROTATED ARRAYS ALLOW AOA VECTOR DETERMINATION .....	29
FIGURE 11: SUBASSEMBLY LAYOUT (ND FILTER AND FIELD STOP NOT SHOWN) .....	31
FIGURE 12: TORIC LENS .....	33
FIGURE 13: SLIT RONCHI SURFACE.....	33
FIGURE 14: CUSTOM EG&G RETICON SI ARRAY .....	34
FIGURE 15: PIXEL CLOSEUP.....	34
FIGURE 16: TYPICAL FOV ROLLOFF.....	35
FIGURE 17: CAL LAMP ASSEMBLY .....	36
FIGURE 18: UNFOLDED CAL LAYOUT .....	37
FIGURE 19: SUBASSEMBLY MODULE.....	38
FIGURE 20: ARRAY READOUT ARCHITECTURE.....	38
FIGURE 21: SIGNAL CONDITIONING SCHEMATICS .....	40
FIGURE 22: CLC BLOCK DIAGRAM .....	41
FIGURE 23: THE SAP-10 FILTER ALGORITHM .....	42
FIGURE 24: SAP BLOCK DIAGRAM.....	43
FIGURE 25: BLOCK DIAGRAM OF ESP PROCESS.....	47
FIGURE 26: AUTO-THRESHOLD SEARCH CURVE.....	49
FIGURE 27: PIECE-WISE LINEAR CURVE TO FACILITATE AUTO-THRESHOLD CALCULATION.....	49
FIGURE 28: TYPICAL TRIGGERED EVENT DATA.....	58
FIGURE 29: DATA COLLECTED IN THE PERIODIC BACKGROUND MODE .....	59
FIGURE 30: FLOOD ILLUMINATION CALIBRATION DATA .....	60
FIGURE 31: IMAGE CALIBRATION DATA .....	61

## List of Tables

TABLE 1: LS1 DESIGN TEAM .....	10
TABLE 2: LS1 PHYSICAL SPECIFICATIONS .....	13
TABLE 3: TEMPERATURE SPECIFICATIONS.....	14
TABLE 4: ANALOG MONITORS .....	15
TABLE 5: DIGITAL MONITORS .....	18
TABLE 6: DISCRETE MONITOR FUNCTIONS.....	18
TABLE 7: LS1 POWER MODES .....	20
TABLE 8: POWER SUPPLY FUNCTIONS ON INTERFACE CONNECTOR .....	21
TABLE 9: POWER SUPPLY FUNCTIONS ON INTERFACE CONNECTOR .....	22
TABLE 10: BI-LEVEL INPUTS TO THE LS1 FROM THE SPACECRAFT.....	22
TABLE 11: ELECTRONIC MODULE COMPLEMENT: 18 DESIGNS, 34 MODULES .....	25
TABLE 12: LS1 MASS PROPERTIES.....	26
TABLE 13: COMMANDABLE PARAMETERS USED IN THE AUTO-THRESHOLD CALCULATION.....	50

# TAOS/LS1 Development Final Report

## 1.0 LS1 SYSTEM

### 1.1 Laser Sensor #1 Description

The Laser Sensor #1 (LS1) project originated as both a technology demonstration for Phillips Lab and a proof of concept for Special Sensor F (SSF) developed by Sandia for the Defense Meteorological Satellite Program (DMSP). The DMSP/SSF project was underway at Sandia and the DMSP Program Office was interested in flying a technology demonstration prior to the first flight of an SSF sensor. This demonstration flight required a quick turn-around sensor development as well as a quick turn-around satellite development. At the same time Phillips Laboratory was seeking experimental sensor technologies, including laser sensors, for the Technology for Autonomous Operational Survivability (TAOS) satellite program. The result was an RFP (request for proposal) to Sandia from the Air Force for a laser sensor system which became the LS1 project. The undertaking was convenient for DMSP/SSF in that the TAOS satellite would launch sufficiently in advance of the SSF-equipped DMSP satellites that the SSF technology would be proven before it was committed to launch.

It was decided that because of the time constraint for development, LS1 would consist of a subset of the SSF sensor. Much of the SSF design approach could be proven by flying the pulse, visible channel whose design was most mature at that time. Because of its benefit to SSF, the DMSP Program Office at Space and Missile Systems Center (SMC) agreed to fund development of LS1 while LS1 on-orbit operations funding was supplied by the TAOS Program Office at Phillips Lab.

The LS1 experiment sought to determine if the slit-grating concept could meet the required sensitivity goals in the clutter-noise limited environment of earth-looking, low-earth-orbit (LEO) satellites. Considering visible wavelengths for the sunlight earth, clutter noise is the dominant source of spatial- and time-varying signal against which a laser signal must be detected. Other laser sensor systems separate the background from the laser signal using temporal coherence (monochromaticity) as the only discriminator. Such a technique has limitations for operation in low-false-alarm-rate, high-sensitivity modes where rejection of clutter, glint, and high-energy particles is necessary. Clutter noise for LS1 results not only from spatial variation in the earth reflectance but also from spacecraft motion which modulates the reflected solar radiation within the field-of-view (FOV) of the sensor.

In addition to coherence the LS1 system was designed to use the measured angle-of-arrival and multiple-sensor coincidence to improve its performance. To implement these

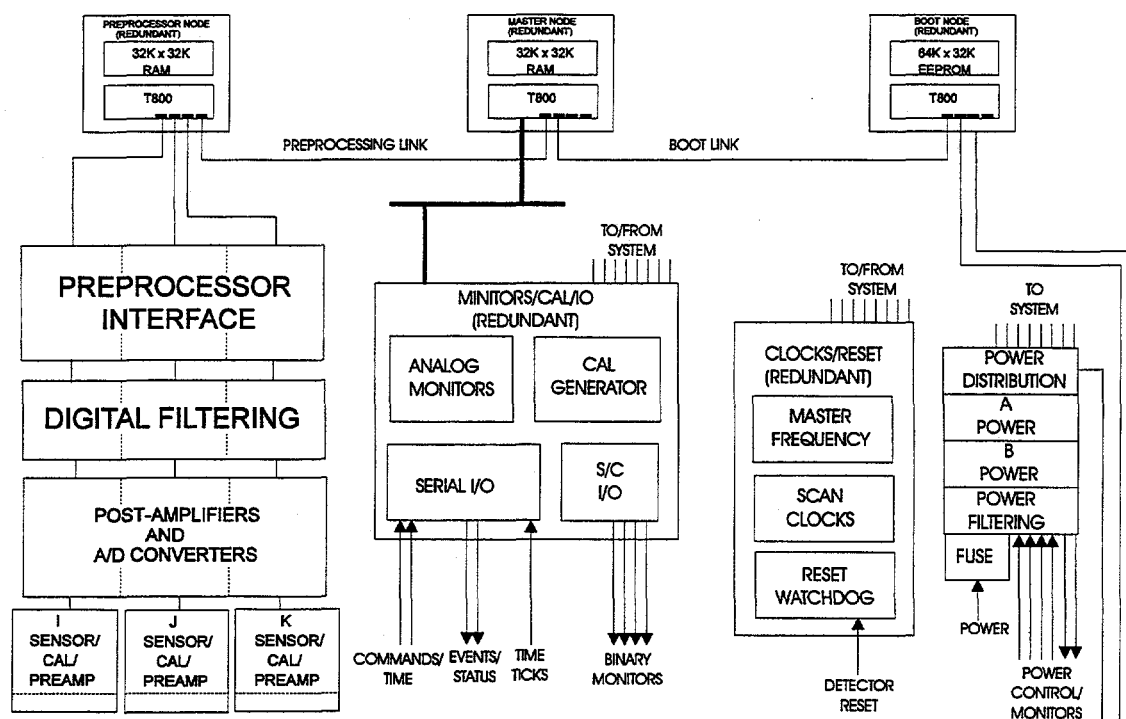
additional discriminators, the slit-grating concept was developed; a system which produces a distinctive three-line signal on three separate linear arrays when a monochromatic point-source is within the sensor FOV. The scale of the patterns indicates the wavelength of the light; the position of the patterns relative to the center of the arrays indicates the angle-of-arrival (AOA) vector. A combination of the coincidence of the signals on the three arrays and the knowledge of the distinct spatial nature of the expected laser signal improves discrimination against clutter and therefore improves performance in low signal-to-clutter-noise operation.

## 1.2 LS1 Team

Table 1 identifies the members of the design team and delineates their areas of contribution to the project.

Name	Area of Contribution
Falls, John	Analog Electronics
Warner, Pete Prior, Ray	Transputer Hardware Design
Bangate, Delfin	Digital Signal Processing Electronics
Gentry, Steve	Optical Design, Detection System Specs
Solomon, Otis	Signal Discrimination Algorithm Software
Rogers, Mike	Signal Discrimination Algorithm Software
Mitchell, Roger	Flight Embedded Software Design
Phipps, Gary	Project Leader of Laser Detection Projects
Claassen, Paul	GSE Software, Digital Hardware, Flight Data Processing Software
Kern, Jeff	Project Leader at Project Inception
Welton, Ted	Mechanical Design
Lee, Duane	Mechanical Design
Grahek, Duane	Power Supply Design

**Table 1: LS1 Design Team**



**Figure 1: LS1 Block Diagram**

### 1.3 Design Overview

A complete listing of electrical and hardware schematics as well as a listing for all of the software source code used in LS1 is beyond the scope of this report. A full description of the minute design details are contained in several file drawers of Sandia schematics and drawings. The drafting system used to generate most of the mechanical drawings is no longer in use at Sandia and the mechanical drawings, for the most part, are easily accessible only as paper copies. The detailed software source code is available in electronic form but the printed version is many hundreds of pages. Only the details needed to understand the operation of the system will be covered here. As of November 1997 all of the additional information concerning the system resides at Sandia: contact the Optics and Emerging Technologies Department, Mail Stop 0980, Albuquerque, NM 87185-0980. Release of information not contained in this report will be contingent upon available manpower, approval of the sponsoring organization, and/or applicable security guidelines at the time of request.

A block diagram of the LS1 system is schematically illustrated in Figure 1. The system consists of a suite of three identical subassemblies each comprised of a linear array, a toric lens, a slit diffraction grating and the necessary support and signal processing electronics. The detector arrays for the three subassemblies are oriented in a plane with the linear array axes 120 degrees apart, aligned normal to the spacecraft pointing vector.

Input optical signals produce a series of point signals along the length of each array. The amplitude and position of these points determines the angle-of-arrival vector, amplitude, and wavelength of the source. An onboard calibration system generates optical signals to simulate both event and background data using laser- and light-emitting-diodes.

All array signals are digitized, digitally filtered, and preprocessed in each subassembly. Further digital signal processing is accomplished by multiple T800 Inmos transputers which pass the data by serial I/O to the Bus Interface Module (BIM). The LS1 is capable of being controlled via commands from the ground. Multi-parameter commands, which originate from the mission ground station, are sent from the spacecraft command/data handler processor to the BIM via a 1553 data communications bus. Following receipt and identification of the command destination, the BIM sends the commands to the LS1 via a custom interface. Laser event data and state-of-health messages are generated and transmitted by LS1 to the BIM. These messages are formatted into 1553 message packets for transmission to the spacecraft mass memory unit.

The Logical Systems "C" language programming environment was chosen for the flight software implementation because of the extensions available, embedded and real-time applications support, and for the efficiency and speed of code generated. Along with a preprocessor, compiler, assembler, linker and ROM code generator, the Logical Systems software has a windowed node debugger, a flood-fill loader, a load balancing library and virtual channel communications facilities.

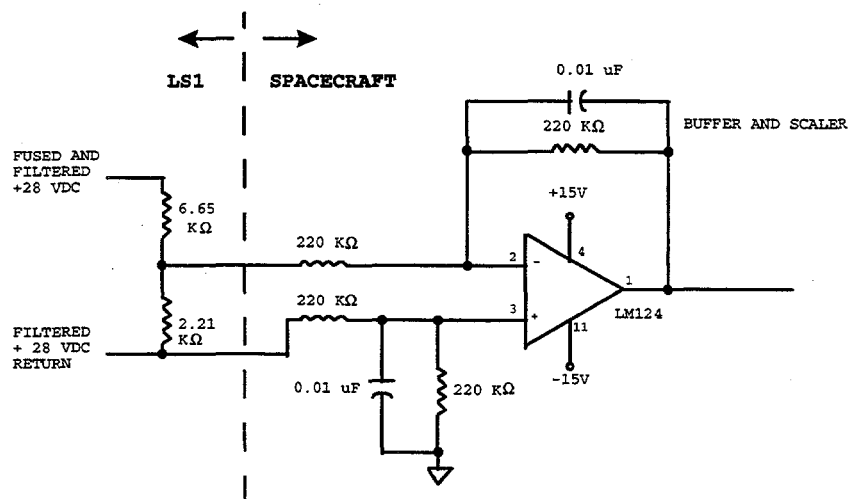
## 1.4 System Specifications

Although the detailed performance specifications of LS1 are classified, the Table 2 physical specifications are not. The LS1 payload provides fully redundant electronic subsystems (A and B). In addition to internal LS1 fusing, fuses for the LS1 payload are provided on the spacecraft side of the interface. For purposes of EMI shielding, the LS1 box is electrically connected to the spacecraft chassis and the secondary winding of the power transformer is grounded to the LS1 box. Primary power is isolated from secondary power by greater than 1 M $\Omega$  with digital outputs referenced to the secondary. Digital signals are isolated from ground at the receiving end by greater than 1 M $\Omega$  through the use of opto-couplers. Analog outputs are isolated from secondary power by greater than 1 M $\Omega$ , and are isolated from ground at the receiving end by greater than 200 K $\Omega$  through the use of a differential amplifier.

The LS1 has one temperature transducer, an AD590, mounted external to the payload which is conditioned and monitored by the spacecraft. The LS1 has two monitors for 28 volt DC input power; one for subsystem A and another for subsystem B. These two

Weight	20.6 lb.
Dimensions	14.1" l x 9.3" w x 8.6" h
Power	12.7W to 13.8W
Temperature	-20 ° C - +50 ° C, operational
	-40 ° C - +60 ° C, survival
Availability	99.9%
Radiation	1 Krad total dose
Reliability	
-Redundancy	redundant digital, power, and interface circuitry, partial redundancy of analog sensors (one of three may fail), minimization of non -redundant power control circuitry
-Stress Testing	
Vibration	Qualification: 14.1 GRMS Acceptance: 10.0 GRMS
Thermal Vacuum	8 Cycles Minimum
Pyro Shock	1000Gs 1-10 kHz resonant plate

**Table 2: LS1 Physical Specifications**



**Figure 2: Analog Output Interface**

power monitors provide signals within the range of 0 to 10 volts DC. The interface circuit for each of the analog monitor outputs is shown in Figure 2. The temperature specifications for the operation of the LS1 are summarized below in Table 3.

Temperature Sensor	Non-operating	Operating
Box Temperature		-20 ° C to +50 ° C
Baseplate Temperature	-20 ° C to +30 ° C	0 ° C to +30 ° C
Baseplate Survival	-20 ° C to +50 ° C	0 ° C to +30 ° C

**Table 3: Temperature Specifications**

## 1.5 State-of-Health

### 1.5.1 ANALOG AND DIGITAL MONITORS

The following LS1 state-of-health data is displayed on the contractor-supplied engineering telemetry window (ETW) test equipment:

- Spacecraft voltage for redundancy A or B
- A temperature monitor located on the LS1 base plate
- BIM interface monitor indicating synchronization and correct operation for both redundancies
- A heartbeat monitor indicating active timing signals
- An event flag monitor signifying the detection of an event

Other monitors are sampled at regular intervals, digitized, and stored in memory for later transmission to the ground station. A one page state-of-health report is generated which provides the analog and digital voltages at the outputs of the power supplies, the analog and digital voltages at the i, j, k arrays, pertinent temperatures throughout the LS1 system and other parameters relevant to proper system operation. The following tables list the parameters monitored and the conversion constants for the state-of-health telemetry data. A listing of the analog monitors and conversion factors is shown in Table 4 below. The digital monitors are listed in Table 5.

Analog Monitor	Conversion	Nominal	Low Limit	High Limit
Logic +10 Volts	0.08026 * Count	+10 Volts	+9 Volts	+11 Volts
Logic +5 Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor i +5a Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor i +5d Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor i +12 Volts	0.09405 * Count	+12 Volts	+11 Volts	+13 Volts
Sensor j +5a Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor j +5d Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor j +12 Volts	0.09405 * Count	+12 Volts	+11 Volts	+13 Volts
Logic -10 Volts	-0.08026 * Count	-10 Volts	-9 Volts	-11 Volts
Sensor i -12 Volts	-0.09405 * Count	-12 Volts	-11 Volts	-13 Volts
Sensor i -8 Volts	-0.06419 * Count	-8 Volts	-7.2 Volts	-8.8 Volts
Sensor j -12 Volts	-0.09405 * Count	-12 Volts	-11 Volts	-13 Volts
Sensor j -8 Volts	-0.06419 * Count	-8 Volts	-7.2 Volts	-8.8 Volts
Sensor k -12 Volts	-0.09405 * Count	-12 Volts	-11 Volts	-13 Volts
Sensor k -8 Volts	-0.06419 * Count	-8 Volts	-7.2 Volts	-8.8 Volts
Logic - 12 Volts	-0.09405 * Count	-12 Volts	-11 Volts	-13 Volts
Sensor k +5a Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor k +5d Volts	0.03949 * Count	+5 Volts	+4.5 Volts	+5.5 Volts
Sensor k +12 Volts	0.09405 * Count	+12 Volts	+11 Volts	+13 Volts
Logic +12 Volts	0.09405 * Count	+12 Volts	+11 Volts	+13 Volts
Spare Voltage Monitor	0.00000 * Count	0 Volts	0 Volts	0 Volts
Array Temperature	25 +[(2.3495 * Count) - 298.2]	25° C	-30° C	+60° C
Box Temperature	25 +[(2.3495 * Count) - 298.2]	25° C	0° C	+50° C
RAM Temperature	25 +[(2.3495 * Count) - 298.2]	35° C	-30° C	+60° C
T800 Temperature	25 +[(2.3495 * Count) - 298.2]	50° C	-30° C	+60° C
A Power Temperature	25 +[(2.3495 * Count) - 298.2]	35° C	-30° C	+60° C
B Power Temperature	25 +[(2.3495 * Count) - 298.2]	35° C	-30° C	+60° C

**Table 4: Analog Monitors**



Digital Monitor Name	Nominal Value
ack nack timeout	= 0.150000 seconds
TBE(missing time mark)	= 60 seconds
TBE(coarse time same)	= 60 seconds
TBE(low priority queue full)	= 60 seconds
TBE(high priority queue full)	= 60 seconds
TBE(low priority queue pop error)	= 60 seconds
TBE(high priority queue pop error)	= 60 seconds
TBE(aborted outgoing message)	= 60 seconds
TBE(process stalled)	= 60 seconds
TBE(event processing error)	= 60 seconds
TBE(scan data not received)	= 60 seconds
TBE(top of scan not received)	= 60 seconds
TBE(all sensor buffers in use)	= 60 seconds
TBE(input byte buffer full)	= 60 seconds
TBE(all scan buffers in use)	= 60 seconds
TBE(missed sensor scan)	= 60 seconds
TBE(TBD 1)	= 60 seconds
TBE(TBD 2)	= 60 seconds
TBE(TBD 3)	= 60 seconds
TBE(TBD 4)	= 60 seconds
TBE(TBD 5)	= 60 seconds
TBE(TBD 6)	= 60 seconds
TBE(TBD 7)	= 60 seconds
TBE(TBD 8)	= 60 seconds
TBE(TBD 9)	= 60 seconds
TBE(TBD 10)	= 60 seconds
uart command instruction	= 17h
uart mode instruction	= 7fh
TBM(spare msg 1)	= 60 seconds
TBM(event)	= 1 seconds
TBM(raw event data)	= 60 seconds
TBM(triggered background data)	= 60 seconds
TBM(periodic background data)	= 60 seconds
TBM(raw image cal data)	= 60 seconds
TBM(raw cal data)	= 60 seconds
TBM(system configuration)	= 3600 seconds
TBM(full state of health)	= 60 seconds
TBM(interval state of health)	= 60 seconds
TBM(spare msg 2)	= 60 seconds
maximum status update period	= 15.000000 seconds

Digital Monitor Name	Nominal Value
BIM interface status update period	= 10.000000 seconds
event status update period	= 10.000000 seconds
LSI heartbeat status update period	= 20.000000 seconds
spare integer 5	= 1125
spare integer 6	= 20392
spare integer 7	= 29200
angle of sensor i	= 5.750432 radians
angle of sensor j	= 4.708230 radians
first to zero threshold ratio	= 0.400000
threshold calculation tolerance	= 0.100000
threshold convergence parameter	= 0.500000
threshold calculation interval	= 1000 scans
angle of sensor k	= 3.663231 radians
desired scan threshold rate	= 56
number of pixels to sum	= 2
number of lines to process	= 5
wavelength difference tolerance	= 0.030000 microns
direction cosine diff tolerance	= 1.500000
minimum ratio of 1st to 0th order	= 0.200000
maximum order separation	= 16000.000000 microns
1st order difference tolerance	= 200.000000 microns
minimum wavelength	= 0.300000 microns
maximum wavelength	= 1.200000 microns
spare char 1	= 0
spare char 2	= 1
current redundancy	= 10 (A)
spare char 4	= 0
spare integer 1	= 0
spare integer 2	= 10
spare integer 3	= 0
spare integer 4	= 0
distance from lens to sensor i	= 26.736820 mm
distance from lens to sensor j	= 26.776409 mm
distance from lens to sensor k	= 26.826429 mm
offset of sensor i	= 0.338720 mm
offset of sensor j	= 0.052320 mm
offset of sensor k	= 0.120440 mm
maximum ratio 0th to 1st order	= 1.500000
spare float 1	= 0.000000
minimum # 4 line events	= 1.000000
minimum illumination	= 0.000000 mW/cm <sup>2</sup>

Digital Monitor Name	Nominal Value
three array coincidence	= 1.000000 (0/1 = off/on)
maximum offset difference	= 100.000000 microns
maximum direction cosine	= 0.850000
minimum order separation	= 1500.000000 microns
spare double 1	= 0.000000
spare double 2	= 0.000000
current operation mode	= 00h = event mode
sensor enable mode	= 00h = I enabled, J enabled, K enabled
last cal command	= 0
filter mode command	= 0
auto threshold mode	= 00h = auto threshold
manual zero threshold command	= 4095
mission enable mode	= 00h = enabled

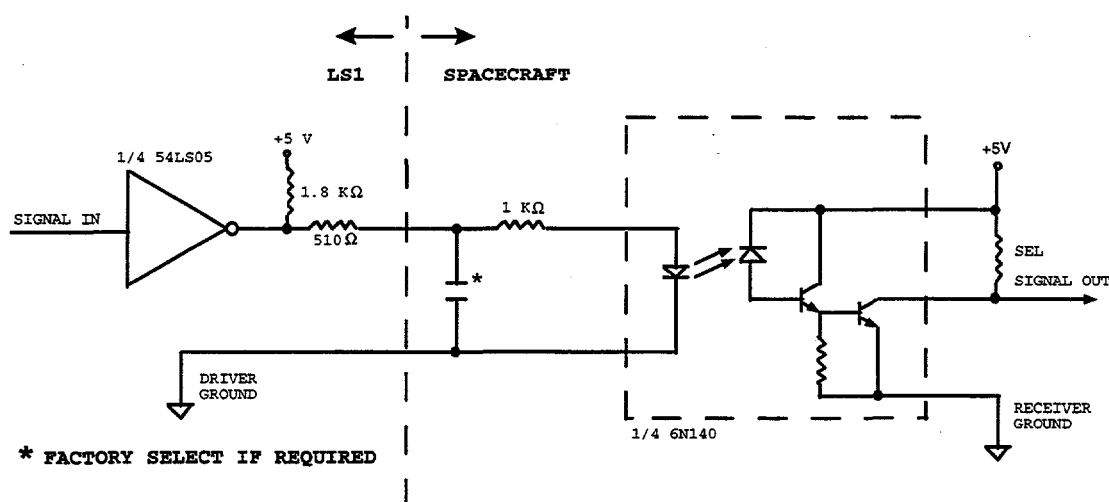
**Table 5: Digital Monitors**

### 1.5.2 DISCRETE STATES

Bi-level outputs from LS1 to the spacecraft include the six discrete monitor functions shown in Table 6. The interface circuit for these monitor outputs is shown in Figure 3. The heartbeat monitor is a binary output square wave, with a programmable period. The event monitor is active (logic high) during a laser illumination event. The duration of the active level is programmable. The BIM interface status signal is active (logic low) if the LS1 detects that a coarse time message has not been received from the BIM within a defined time interval.

Signal Name	Function
Heartbeat Mon. A	Health monitor A
Heartbeat Mon. B	Health monitor B
Event Mon. A	Laser Event flag A
Event Mon. B	Laser Event flag B
BIM I/F Mon. A	I/F health A
BIM I/F Mon. B	I/F health B

**Table 6: Discrete Monitor Functions**



**Figure 3: Bi-Level LS1 to S/C Interface Circuit**

## 1.6 Power

The power subsystem performs the following functions:

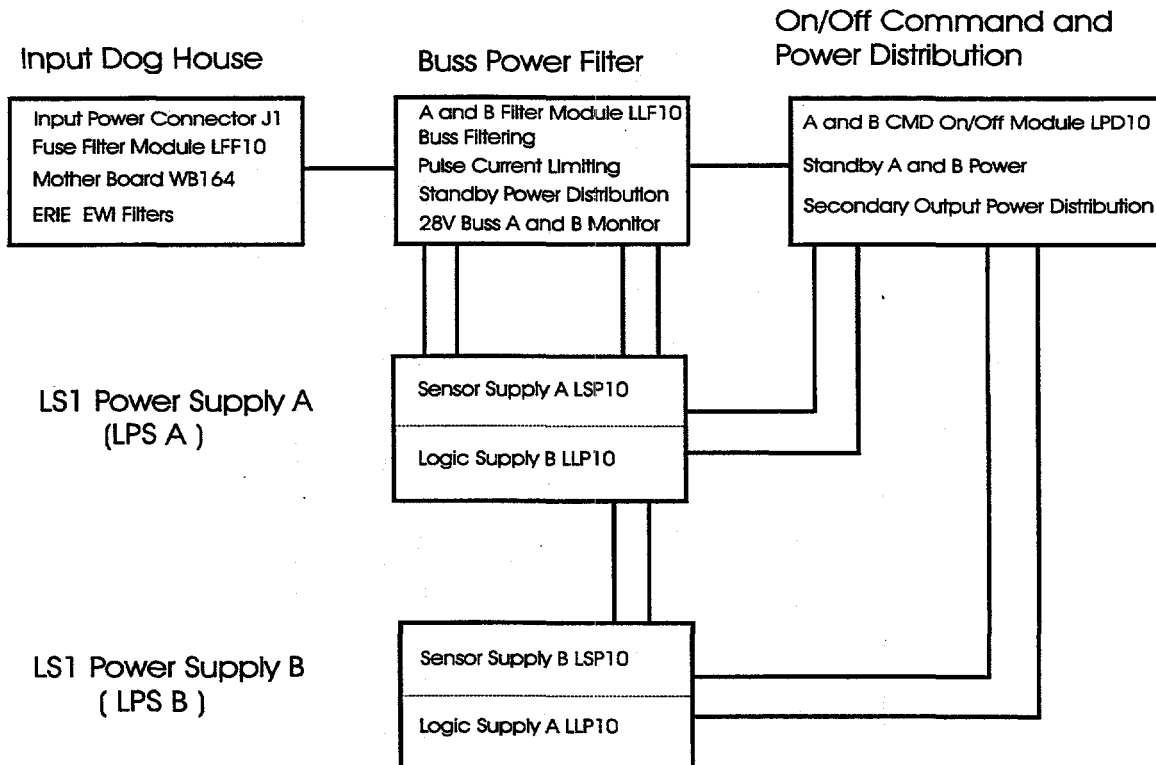
1. Interface to the spacecraft power busses
2. Interface to the spacecraft discrete command lines
3. Provides redundant system power at +5V, +12V and -12V
4. Provides a fuse network for power distribution to non-redundant, cross-strapped subsystems.
5. Provides spacecraft bus voltage monitors to interface to spacecraft analog monitors
6. Provides conditioned-power voltage and current monitors for interface to the LS1 state-of-health sub-system

LS1 has two independent power supplies A and B. Only one supply can be used at a time. The spacecraft always maintains the bus voltages on both of these redundancies during launch and at all times afterward. There is no capability for the spacecraft to turn off the LS1 power busses during normal orbital operation but each power bus is independently fused.

The power consumption of the LS1 is 12.7 to 13.8 watts over temperature and bus voltage. The current and power drawn for specific system modes and for specific bus voltages for the LS1 are summarized in Table 7 below:

	+23 Volts	+28 Volts	+33 Volts
Standby	21 Ma./483 W	25 Ma./700 W	27 Ma./891 W
Power A	.592 A/13.62 W	.475 A/13.30 W	.417 A/13.67 W
Power B	.580 A/13.34 W	.490 A/13.72 W	.423 A/13.96 W

**Table 7: LS1 Power Modes**



**Figure 4: LS1 Power System Block Diagram**

A block diagram of the LS1 power subsystem is shown in Figure 4. The LS1 system requires redundant power supply subsystems, designated "A" and "B". Only one redundancy is active at any time, except for Standby Power, which has both redundancies operating. In addition to the supplies, a distribution network was required to route power to non-redundant and cross-strapped subsystems. Table 8 and Table 9 list the power supply functions on the interface connectors.

Pin	Definition	Pin	Definition
1	Spare	34	Spare Digital Monitor A
2	Spare	35	Spare, Digital Monitor Return A
3	Spare	36	Spare
4	LS1 On A	37	LS1 Reset A
5	LS1 On Return A	38	LS1 Reset Return A
6	Shield Ground	39	Shield Ground
7	LS1 Off A	40	LS1 Reset B
8	LS1 Off Return A	41	LS1 Reset Return B
9	Shield Ground	42	Shield Ground
10	LS1 On B	43	Heartbeat Monitor B
11	LS1 On Return B	44	Heartbeat Monitor Return B
12	Shield Ground	45	Spare
13	LS1 Off B	46	Event Monitor B
14	LS1 Off Return B	47	Event Monitor Return B
15	Shield Ground	48	Spare
16	Spare	49	Spare Command B
17	28V Monitor A	50	Spare Command Return B
18	28V Monitor Return A	51	BIM I/F Status Monitor B
19	Spare	52	BIM I/F Status Monitor Return B
20	28V Monitor B	53	Spare
21	28V Monitor Return B	54	Spare Digital Monitor B
22	Spare	55	Spare Digital Monitor Return B
23	Spare Command A	56	Spare
24	Spare Command Return A	57	Spare
25	Heartbeat Monitor A	58	Spare Analog Monitor A
26	Heartbeat Monitor Return A	59	Spare Analog Monitor Return A
27	Spare	60	Spare
28	Event Monitor A	61	Spare Analog Monitor B
29	Event Monitor Return A	62	Spare Analog Monitor Return B
30	Spare	63	Spare
31	BIM I/F Status Monitor A	64	Spare
32	BIM I/F Status Monitor Return A	65	Spare
33	Spare	66	Spare

**Table 8: Power Supply Functions on Interface Connector**

Pin	Definition	Pin	Definition
1	Spare	8	Unregulated Power B
2	Spare	9	Unregulated Power Return B
3	Unregulated Power A	10	Spare
4	Unregulated Power Return A	11	Spare
5	Spare	12	Spare
6	Spare	13	Spare
7	Spare		

**Table 9: Power Supply Functions on Interface Connector**

Six discrete commands from the spacecraft control the LS1 unit. In order for the On discrete command to function, the spacecraft bus voltage must be within limits and selected to the system. The LS1 power-on sequence requires the *LS1\_28V\_Mains\_On* command followed by the *LS1\_A\_On* discrete command. The DC voltage level must also be between +23 and +33 volts. When the +28 volts is applied, a small amount of standby current flows to the power supply control circuitry. With spacecraft voltage selected and the *LS1\_A\_On* command sent, the system is fully operational within 15 seconds. Both redundancies cannot be commanded at the same time. A waiting period of two minutes is required after a power off before the system can again be commanded *On*.

Six Bi-level inputs to the LS1 from the spacecraft control power functions are shown in Table 10. The interface circuit for these control inputs is shown in Figure 5.

Signal Name	Function
LS1 on A	On Command A
LS1 off A	Off Command A
LS1 on B	On Command B
LS1 off B	Off Command B
LS1 reset A	P/L Reset A
LS1 reset B	P/L Reset B

**Table 10: Bi-Level Inputs to the LS1 from the Spacecraft**

Each of these inputs has the following specifications:

Low Logic Level  $\geq 1.0$  mA drive current from source (Logic 0 at Signal Out)

High Logic Level = 0 mA (Logic 1 at Signal Out)

Maximum Drive Current  $< 5$  mA

Pulse Duration 100 ms  $\pm$  10%.

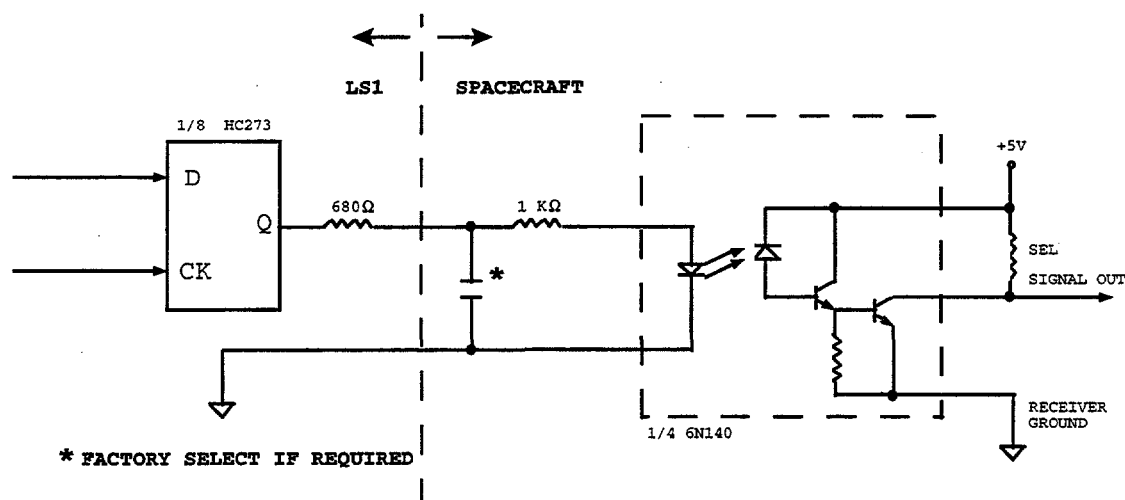


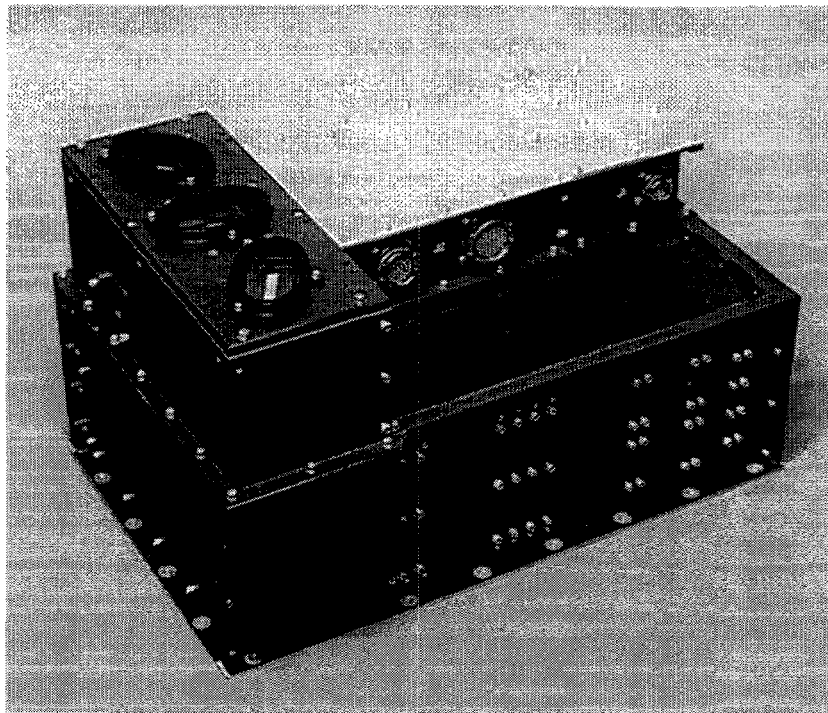
Figure 5: Bi-Level S/C to LS1 Interface Circuit

The power bus interface provides bus fuses close to the physical interface connector to protect against faults in the LS1 system. All bus lines are fused at the Sensor Interface Unit (SIU). The +28V busses are fused with 3 ampere fuses and the +5V busses are fused with 1 ampere fuses. It is intended that these fuses will be relied on by the spacecraft to protect the SIU and the power bus wiring harness, but that the LS1 will not rely on them for protection of the LS1 system. The interface also provides EMI filtering to protect against conducted susceptibility and conducted emissions.

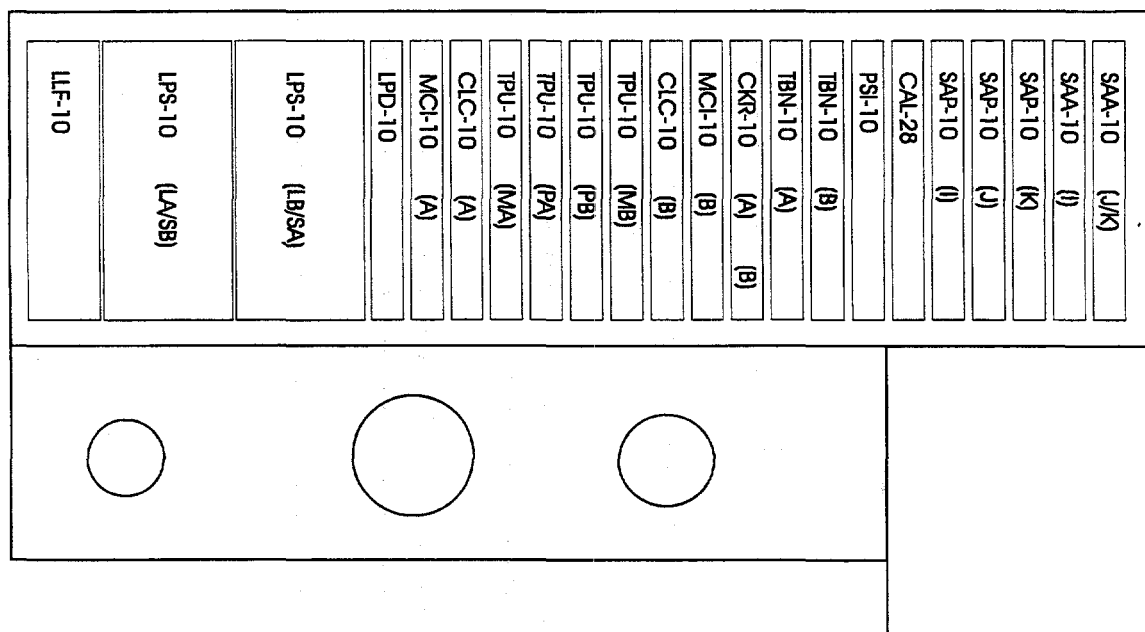
## 1.7 Mechanical Design

The LS1 system consists of a single instrumentation stack with redundant power supplies; optical sensors, analog electronics, and all digital processing are contained in one enclosure. The LS1 housing is shown in Figure 6. Although normally the housing would be machined from a single piece of aluminum, this housing was designed to use a number of subassemblies which were bolted together. It was decided that multiple subassemblies were more appropriate for the LS1 program in light of the compressed time schedule and desire to reduce costs.





**Figure 6: LS1 Housing**



**Figure 7: LS1 Module Layout**

The mechanical construction of the electronic modules is based on the EE size module designed by Sandia National Laboratories (3.5" x 7" x Variable thickness). The layout of the electronic modules, the power supplies and the power line filter module are shown in Figure 7. The names and functions of the modules are listed in Table 11. Based on the function of the module, it may have a plastic or metal frame and may employ covers for signal shielding, internal heat sinking structures to carry internal heat to the metal module frames, and/or radiation shielding in the form of tantalum tunnels to help protect the transputers from radiation upset.

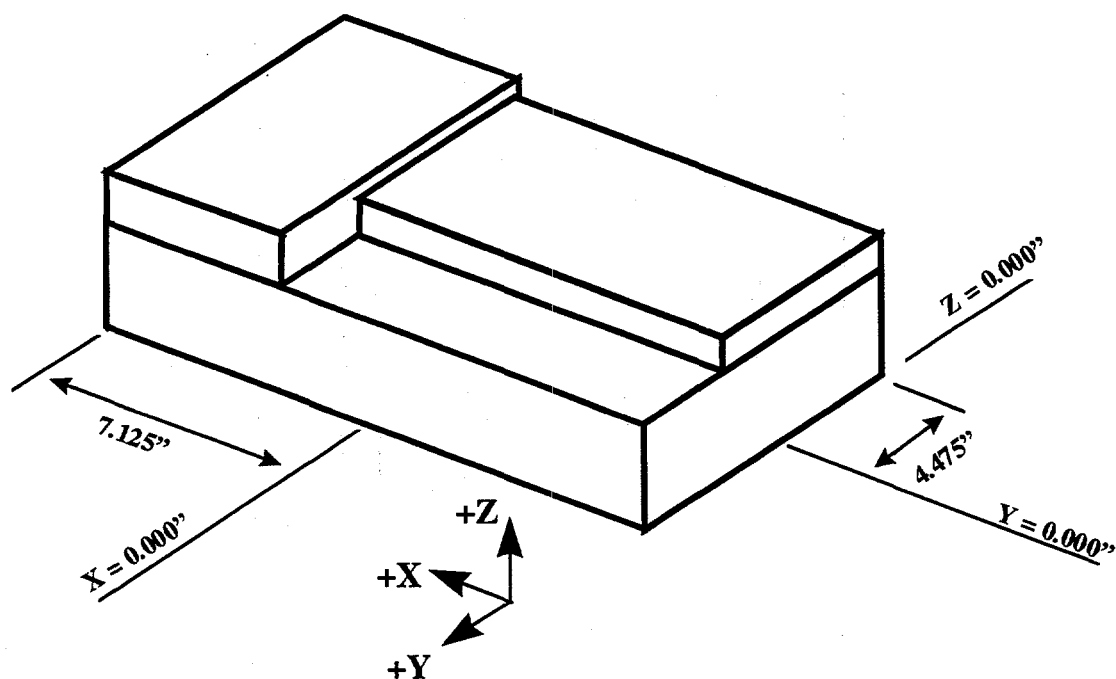
Module Name	Number	Function
TPU-10	4	Central Processing Unit
TBN-10	2	Boot ROM Processor
SIB-25	3	Silicon Array Interface Board
SIB-26	3	Silicon Array Interface Board
SAA-10	2	Silicon Amplifiers & ADC
SAP-10	3	Silicon Array Processor
CLC-10	2	SAP Clock Module
TOTAL:	19	
PSI-10	1	Preprocessor/SAP Interface
CAL-28	1	Optical Calibration
MCI-10	2	System Monitors & I/O
CKR-10	1	I/O Clocking & LSI Reset
LLP-10	2	Logic Power Conditioning
LSP-10	2	Sensor Power Conditioning
LPD-10	1	Power Distribution
LFF-10	2	Bus Power Fusing/EMI Filtering
LLF-10	1	Bus Power Limiter/Filtering
MB-164	1	Fuse Motherboard
MB-165	1	System Motherboard
TOTAL:	15	

**Table 11: Electronic Module Complement: 18 Designs, 34 Modules**

Mass properties of the LS1 system are listed in Table 12. The physical measurement reference system used for the center of gravity and moments of inertial is illustrated in Figure 8.

<b>Envelope Dimensions:</b>	14.25"L	9.34"W	8.448"H
<b>Weight:</b>	20.6 lbs.		
<b>Center of Gravity:</b> (see Figure 8 for coordinate reference)	$\bar{X} = +0.25$ inches	$\bar{Y} = +0.02$ inches	$\bar{Z} = +2.90$ inches
<b>Moments of Inertia:</b>	$I_{xx} = 222 \text{ lb-in}^2$	$I_{yy} = 435 \text{ lb-in}^2$	$I_{zz} = 469 \text{ lb-in}^2$

**Table 12: LS1 Mass Properties**

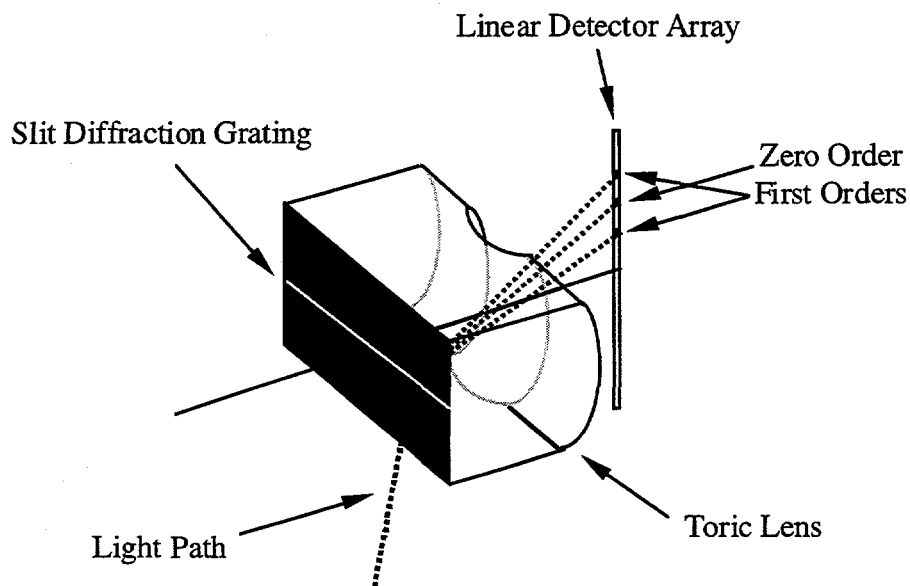


**Figure 8: LS1 Mechanical Reference System**

## 1.8 Optical Design

### 1.8.1 THEORETICAL BASIS

The main optical elements of an LS1 subassembly are depicted in Figure 9. The basis for the LS1 optical design is the concept that if a linear array is placed behind a slit aperture, illumination of that slit causes a spot to be formed on the array at a location



**Figure 9: Slit-Grating/Linear Array Subassembly with Slit Image Locations**

indicative of the component of the illumination angle-of-arrival vector (AOA),  $\vec{A}$ , which lies along the axis of the linear array. By adding a diffraction grating to the slit, with grating lines running parallel to the slit axis, some of the light that would have propagated directly to the linear array is diffracted into two 'first-order' spots. [ 'Order' in this sense refers to the diffracted order number,  $m$ . Although higher orders do exist, they are small in amplitude relative to the first order and will not be further discussed. ] The location of these first-orders relative to the undiffracted or 'zero order' spot is governed in part by the wavelength of the illuminating light. The location of any of the diffracted orders on the array can be determined from the following conical diffraction relationships.

$$x_m = x_0 + z \frac{\mu k - \frac{m\lambda}{n'a}}{\sqrt{1 - (\mu l)^2 - \left[\mu k - \frac{m\lambda}{n'a}\right]^2}} \quad (1.0)$$

$$y_m = z \frac{\mu l}{\sqrt{1 - (\mu l)^2 - \left[\mu k - \frac{m\lambda}{n'a}\right]^2}} \quad (1.1)$$

$$\mu = \frac{n}{n'} \quad (1.2)$$

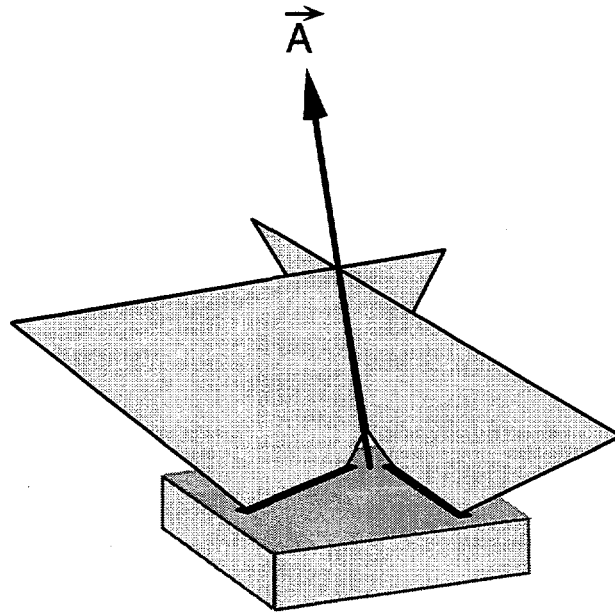
- ( $x_m, y_m$ ) = center location of slit image relative to ray intersection with the grating surface [m]
- $x_{m=0}$  = array offset along x axis from intersection with optical axis [m]
- $z$  = separation between slit and linear array along optical axis [m]
- $m$  = diffracted order number
- $a$  = grating constant [m]
- $\lambda$  = illumination wavelength [m]
- $n$  = index of refraction on the source side of the diffraction grating
- $n'$  = index of refraction on the detector side of the diffraction grating
- $\hat{A} = \hat{k}i + \hat{l}j + \hat{m}k$  = unit AOA vector with direction cosines  $k, l, m$
- $\hat{i}$  = unit vector along the x axis (perpendicular to slit and grating lines)
- $\hat{j}$  = unit vector along the y axis (parallel to slit and grating lines)
- $\hat{k}$  = unit vector along the z axis (optical axis of system)

Of particular note in equation 1.1 is that the higher diffracted orders will begin to experience a lateral shift along the y-axis relative to the zero order as the angle of incidence in the y-plane increases. For example, if  $k=0$  (the incoming ray is totally in the y-z plane, the y location of the slit image as a function of order number is given by

$$y_m = z \frac{\mu l}{\sqrt{1 - (\mu l)^2 - \left[ \frac{m \lambda}{n' a} \right]^2}} \quad (1.3)$$

This relationship will be used later concerning the use of the toric lens in the slit-grating subassemblies.

When two subassemblies are rotated relative to each other, as shown Figure 10, the location of the two undiffracted zero order spots ( $m=0$ ) can be used to solve for the AOA unit vector,  $\hat{A}$ .



**Figure 10: Two Rotated Arrays Allow AOA Vector Determination**

First the x direction cosines in the coordinate system for each subassembly,  $k$  and  $k'$ , are found.

$$k = \left[ \frac{x - x_0}{\mu z} \right] \sqrt{1 - \frac{(x - x_0)^2 \sin^2(\theta') + \left[ z \frac{x' - x'_0}{\mu z'} - (x - x_0) \cos(\theta) \right]^2}{\left[ (x - x_0)^2 + z^2 \right] \sin^2(\theta') + \left[ z \frac{x' - x'_0}{\mu z'} - (x - x_0) \cos(\theta) \right]^2}} \quad (2.0)$$

$$k' = \left[ \frac{x' - x'_0}{\mu z'} \right] \sqrt{1 - \frac{(x' - x'_0)^2 \sin^2(-\theta') + \left[ z' \frac{x - x_0}{\mu z} - (x' - x'_0) \cos(\theta') \right]^2}{\left[ (x' - x'_0)^2 + z'^2 \right] \sin^2(-\theta') + \left[ z' \frac{x - x_0}{\mu z} - (x' - x'_0) \cos(\theta') \right]^2}} \quad (2.1)$$

- $x_0$  = array offset for first subassembly [m]  
 $x$  = zero order location on first subassembly [m]  
 $z$  = separation between slit and linear array for first subassembly [m]  
 $x'_0$  = array offset for second rotated subassembly [m]  
 $x$  = zero order location on second rotated subassembly [m]  
 $z'$  = separation between slit and linear array for second rotated subassembly [m]  
 $\theta'$  = rotation angle about the optical axis from first to second subassembly [deg]

With knowledge of the angular rotation between the two subassemblies,  $\theta'$ , the y direction cosine in the first coordinate system,  $l$ , is computed.

$$l = \frac{k - k'}{\sin(\theta')} \quad (2.2)$$

Finally, the z direction cosine in the first coordinate system can be determined.

$$m = \sqrt{1 - (k^2 + l^2)} \quad (2.3)$$

The three direction cosines completely specify the unit AOA vector.

$$\hat{a} = k \hat{i} + l \hat{j} + m \hat{k} \quad (2.4)$$

Knowledge of a first order location then yields the source wavelength.

$$\lambda = \left( n' \frac{a}{m} \right) \left[ \mu k - (x_m - x_0) \sqrt{\frac{1 - (\mu l)^2}{z^2 + (x_m - x_0)^2}} \right] \quad (3.0)$$

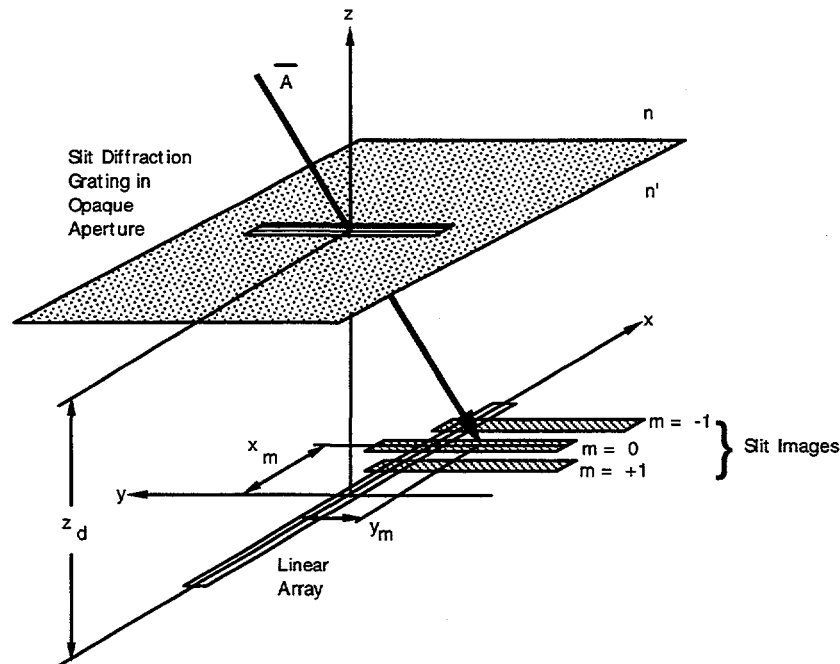
$x_{m=1,-1}$  = order location for either  $m = +1$  or  $-1$  order

$m$  = corresponding order number

Knowledge of two zero-order locations on two subassemblies rotated relative to each other, and knowledge of one first-order location on one of the sensors, completely defines the solution for both wavelength and the AOA vector.

### 1.8.2 PRACTICAL IMPLEMENTATION OF THE SLIT GRATING OPTICAL DESIGN

The LS1 optical design flows from the desire to achieve the maximum wavelength resolution while maintaining the desired FOV and wavelength specifications. The resulting optical configuration used for the flight unit consists of a Ronchi diffraction grating on a ZnS toroidal lens. A conceptual layout of the optical subassembly is shown in Figure 11. Detailed schematics of the subassembly layout are found in an appendix. The following paragraphs discuss the diffraction grating, toric lens, neutral density filter, detector array, field stop, and optical calibration source designs.



**Figure 11: Subassembly Layout (ND Filter and Field Stop Not Shown)**



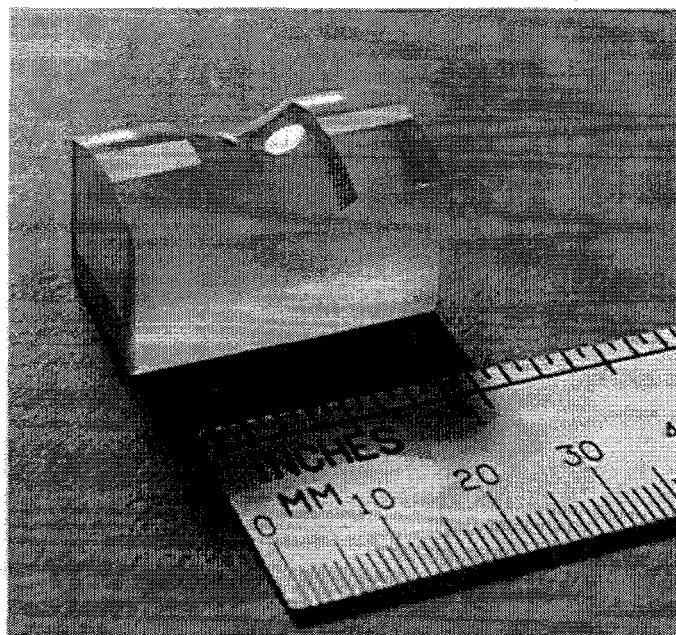
The diffraction grating design was driven by two factors: 1) a desire to achieve the maximum wavelength resolution possible by maximizing diffraction grating spatial frequency, and 2) the need for diffractive properties to remain as constant as possible over the specified field-of-view and wavelength ranges. Phase grating structures were investigated but found to have large variations over field angle and wavelength. Ronchi ruling diffraction gratings were chosen due to fabrication simplicity, relative uniformity over wavelength and field angle, and the ability to directly specify the first order amplitude from the line-to-space ratio of the grating. A 50% grating duty cycle was chosen to maximize energy transferred to the first orders. Grating frequency was selected to be 500 line pair/mm based on discussions with grating manufacturers who indicated that this was the maximum frequency achievable. The width of the grating was chosen to be 0.3 mm.

The gratings were fabricated by Hyperfine Inc. of Boulder, Colorado, using a direct-ruling approach. Significant difficulty was encountered trying to achieve satisfactory line quality using this method. Careful monitoring of the ruling tip pressure was required to avoid cutting through the metallic film and into the substrate. The final technique used was to rule only part-way through the material and to ion-etch the remaining material. Additional problems were encountered by the manufacturer in obtaining ruling tools of the width precision required.

After analyzing the conical diffraction process for the grating specified above, it was determined that the first-orders would not propagate at the edge of the FOV in the y-direction. This condition can be determined by allowing

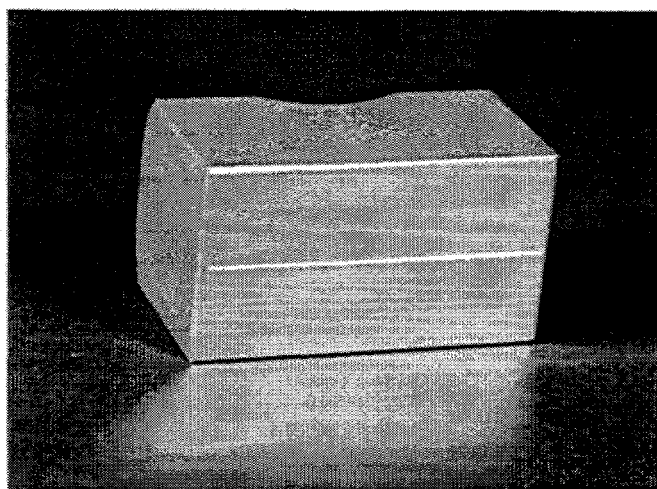
$$(\mu l)^2 + \left[ \frac{m\lambda}{n'a} \right]^2 > 1 \quad (4.0)$$

in equation 1.3. If one maintains the wavelength and FOV specifications, the only alternatives that allow the first orders to propagate are decreasing the grating frequency (increase a) impacting wavelength resolution, or increasing the refractive index into which the orders propagate (increase n'). Rather than impact wavelength resolution, the second approach was pursued by the development of the Cleartran-ZnS toric lens element, shown in Figure 12 and Figure 13. Schematics of the toric lens design are found in an appendix. The lens was designed to provide a high-index medium which allowed the orders to propagate, while allowing the rays to exit normal to the toric surface of the lens. Not only did the toric lens solve the propagation problem, but it increased the wavelength resolution by a factor proportional to the ZnS index of refraction, 2.2-2.4. Additional losses were introduced into the system due to absorption and Fresnel surface losses. These losses were not of consequence since the final system required a neutral density (ND) optical filter as detailed below.



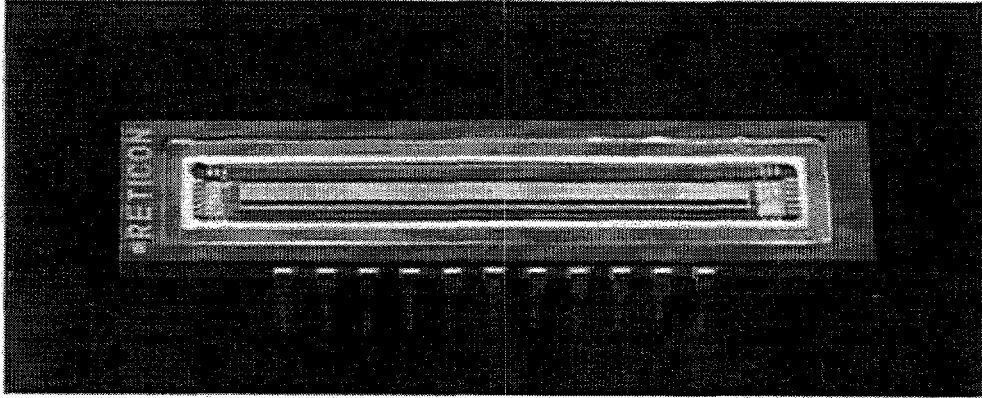
**Figure 12: Toric Lens**

The lenses were fabricated by Optical Filter Corporation (OFC) Diamond Turning Division, Keene, NH. The original design studies were performed by J. L. Wood Optical Systems, Santa Ana, California.

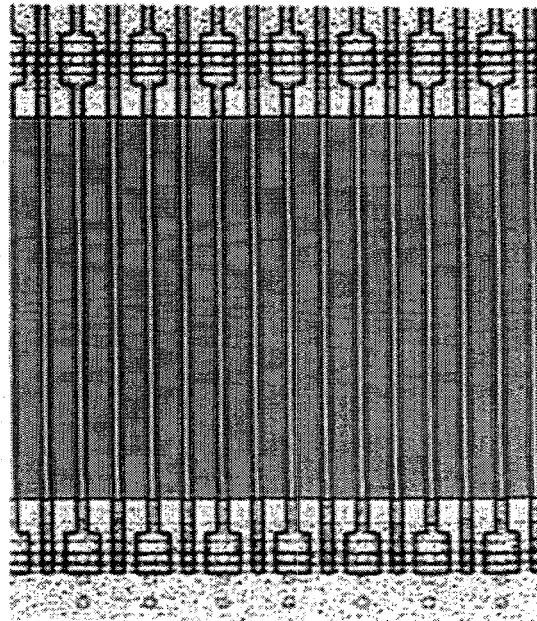


**Figure 13: Slit Ronchi Surface**

The detector array used was a custom 1024-element, silicon photodiode array manufactured by EG&G Reticon Inc., shown in Figure 14. The  $30\text{ }\mu\text{m}$  pixel pitch yields a total active array length of 30.72 mm. Pixel fill factor is 75% with a pixel width of  $360\text{ }\mu\text{m}$ . The array is covered by a 0.020" fused silica window. A close-up of the pixels is shown in Figure 15.



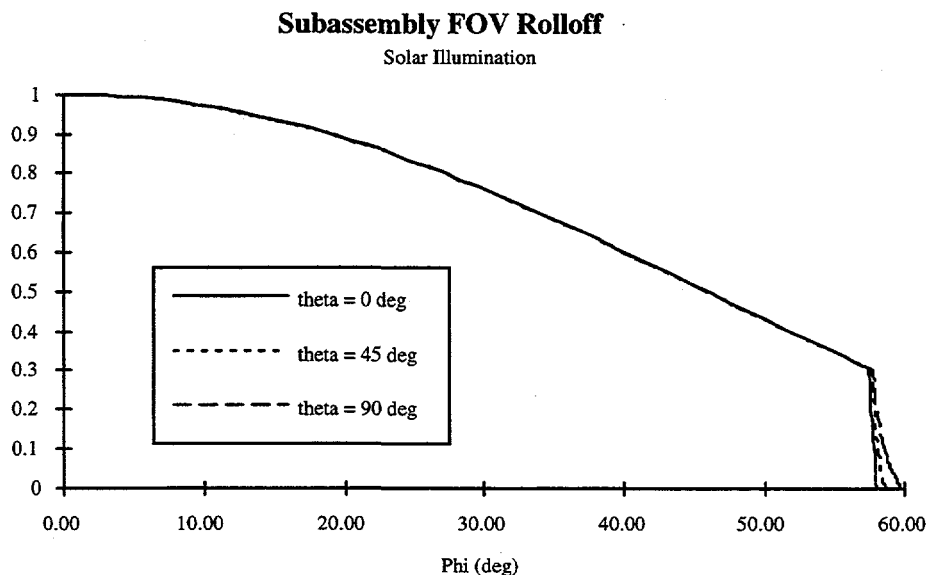
**Figure 14: Custom EG&G Reticon Si Array**



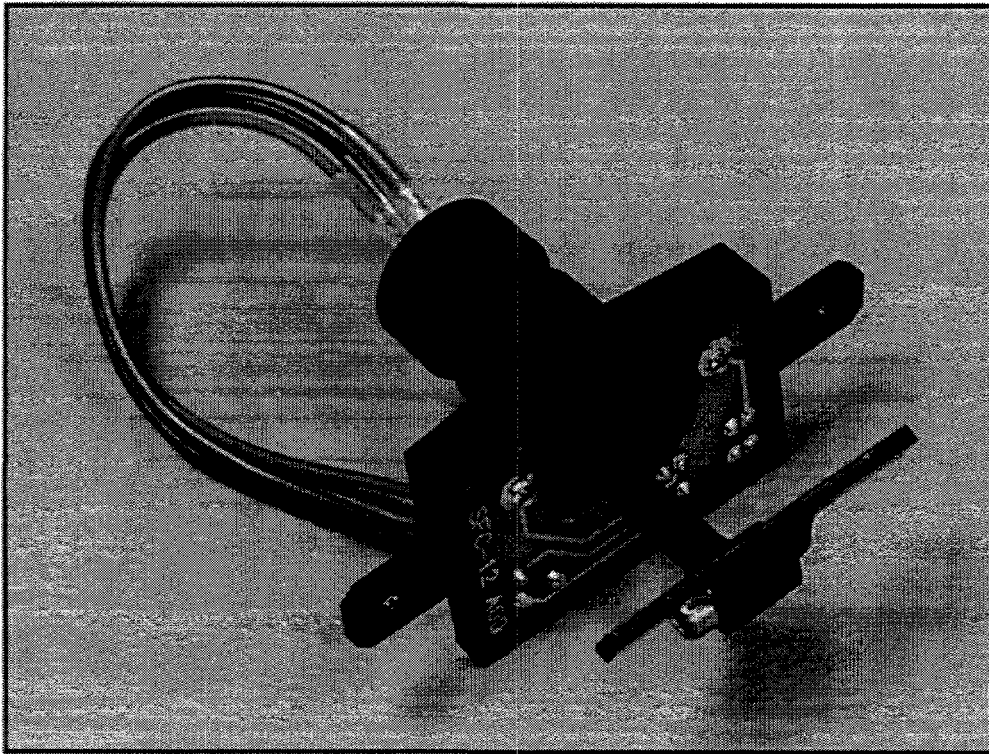
**Figure 15: Pixel Closeup**

Analysis of the slit-grating/toric lens combination with the linear array showed that the dominant noise source in the system would be clutter noise due to the interaction of the sensor FOV and the orbiting earth albedo. In an effort to maximize dynamic range and minimize noise-equivalent irradiance, an ND filter was inserted into the optical system. The system uses a 10-bit A/D converter with 4 bits of gain shifting to provide a 14-bit output signal. An optical filter transmission of 3% was chosen to keep all background signals in the least significant 10-bits. The pixels on either end of the array are optically masked by the ND filter retaining clips. Optical information is only processed for pixels ~100 through ~920.

An external field stop was implemented to prevent blinding of the sensor from direct solar illumination beyond the earth's limb outside of the intended FOV. Due to the asymmetric nature of the toric lens design, an asymmetric field stop was required. The field stops are the raised, oval shapes around the periphery of each array and can be seen in Figure 6 at the left-hand side of the box. Specific design parameters for the field stop are given in schematics found in an appendix. A typical FOV roll-off curve showing the field stop cutoff at  $59^\circ$  is shown as Figure 16. The variation in the roll-off as a function of theta (azimuth) is complex; the effective widths of the field stop, grating, and sensor change with the position of the rays around the edge of the FOV.

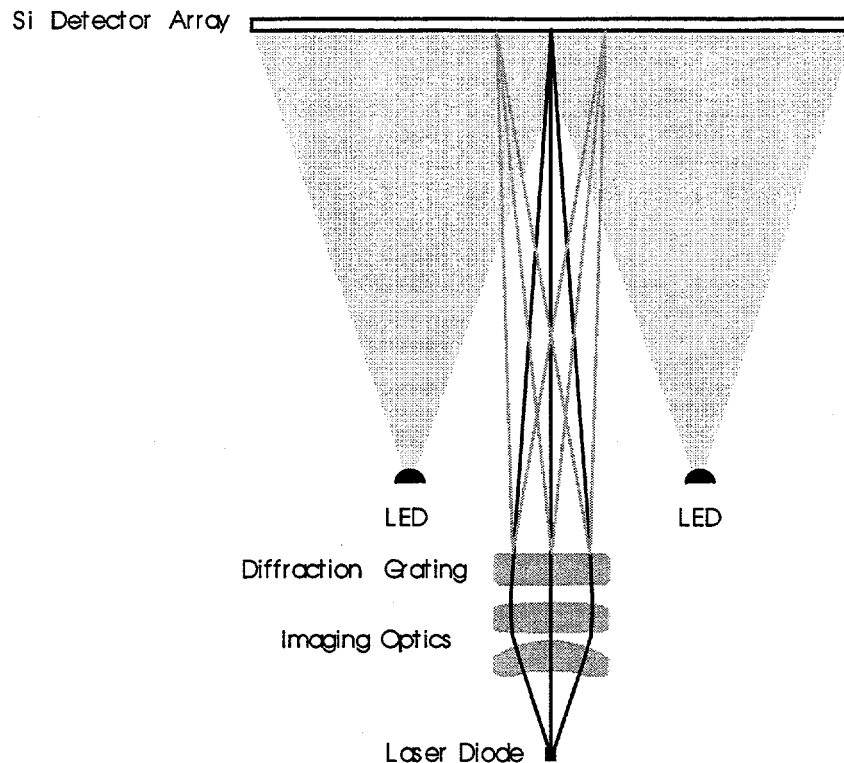


**Figure 16: Typical FOV Rolloff**



**Figure 17: Cal Lamp Assembly**

Each of the three optical subassemblies is comprised of a toric lens, a slit diffraction grating, an ND filter, and a silicon linear array. The toric lens is designed to allow light from very wide angles to be focused onto the linear array. Internal optical sources were added to allow the sensor to be stimulated by two different types of illumination, coined 'flood' and 'image' cal sources. These sources are depicted in Figure 17 and Figure 18. The flood cal sources consist of two LEDs projected onto the linear array to illuminate all of the active pixels. Each image cal source is a laser diode projected through a diffraction grating and focusing optics to provide a diffraction pattern on the array. This is to simulate a laser source illuminating the system from nadir. These sources provide and end-to-end check of pixel response and event processing algorithms.

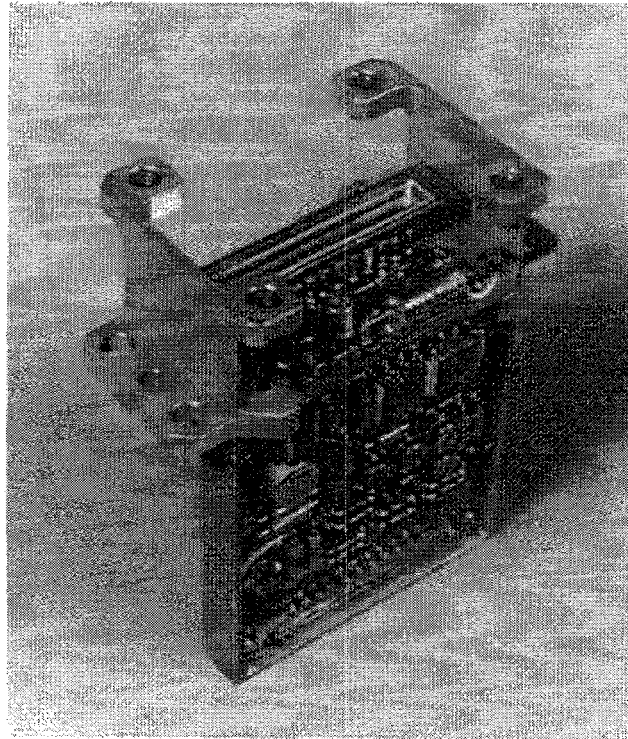


**Figure 18: Unfolded Cal Layout**

## 1.9 Electronic Design

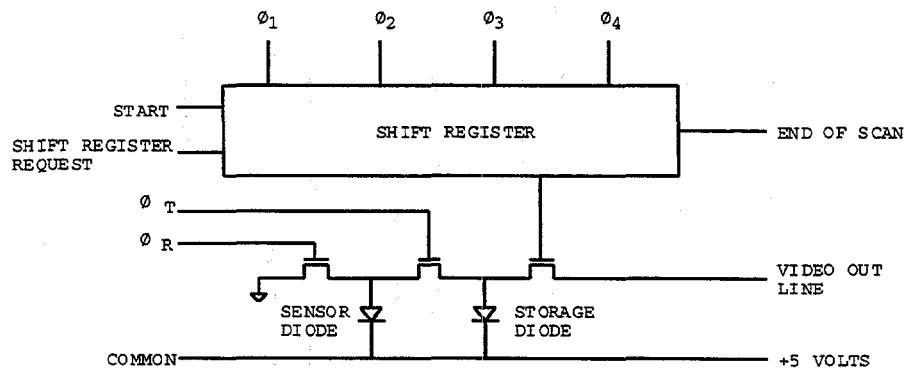
### 1.9.1 ANALOG ELECTRONICS

Each focal plane array is housed in a subassembly frame and consists of a custom, linear, EG&G silicon photodiode array of 1024 elements, a preamplifier, timing circuits, and an amplifier circuit. A photon-generated current charges the photodiode capacitance of each array element. These charges are sequentially switched to the video line where they are conditioned by the preamplifier. The three optics of each subassembly are oriented in a plane with the long axes of their linear arrays 120 degrees apart aligned normal to the spacecraft pointing vector. Figure 19 is a photograph of the subassembly module.



**Figure 19: Subassembly Module**

Level shifted timing clock signals are supplied to the array from the array interface board to sequentially transfer the photodiode charge to the video line. A schematic of the array readout architecture appears in Figure 20. Additional analog signal amplification and conditioning are performed by successive amplifiers. Amplification, analog-to-digital conversion levels, and ultimately usable dynamic range for the system, were determined by the saturation level of the array and the noise floor of the system and associated electronics.



**Figure 20: Array Readout Architecture**

The silicon linear array consists of 1024 elements, 360 microns in width with an inter-element spacing of 30 microns. The photodiodes are alternately switched to a common video line through a Reticon M-series multiplexer built into the array package. This multiplexer is controlled by a four-phase clock and a start pulse. The array is a staring, integrating device with an integration time of 10.4448 milliseconds per frame. The pixels are sequentially connected to the video line at a 9.6 microsecond rate.

A low-noise, high-bandwidth preamplifier is located adjacent to the array video line output. The preamplifier buffers the high impedance video line and conditions the signal before routing through a coaxial cable to an analog processing module. The preamplifier board, SIB-25, also supplies the +5 and -7 volt level shifted signals needed by the silicon array.

### ***Sensor Array Analog (SAA) Module***

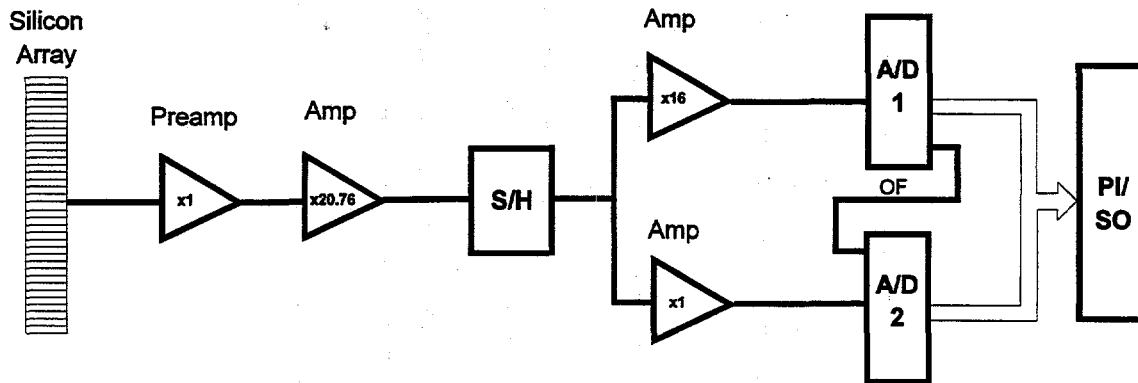
The analog processing is done in the Silicon Array Analog (SAA) module located in the main processing box. The module contains additional amplifiers, a sample-and-hold, an analog-to-digital converter, and parallel-to-serial data conversion circuitry.

Signal processing consists of an inverting amplifier with a frequency response of 504 kHz. This output goes to a capacitor that is switched to ground via a double-correlated sampler. This signal is then sent to a sample-and-hold amplifier before the high-pass pre-quantization filter and X16-gain amplifier. A MOSFET switch across the high-pass filter allows a larger capacitor to be selected during the collection of background data to lower the frequency of signals passed to the analog-to-digital converter.

Two analog-to-digital converters are used for a dynamic range of 16,384. The analog-to-digital converters are only 10 bits, but 14 bits of data are generated. When the overflow bit of the X16 analog-to-digital converter goes high, the X1 analog-to-digital converter data is directed to the output data lines and the 4 LSB's are zeroes. When the X16 analog-to-digital converter is used, the 4 MSB's from the X1 amplifier are pulled low making the upper 4 bits zeroes. These 14 lines are converted to serial data and sent as data and complement data to the processing modules. A schematic diagram of the signal conditioning circuitry is shown in Figure 21.

Further digital signal processing is accomplished by multiple T800 INMOS transputers, operating at a clock frequency of 17 MHz. Processor algorithms determine angle-of-arrival, wavelength, and amplitude. The data is passed from the transputer by serial I/O to the Bus Interface Module (BIM).





**Figure 21: Signal conditioning schematics**

### 1.9.2 DIGITAL ELECTRONICS

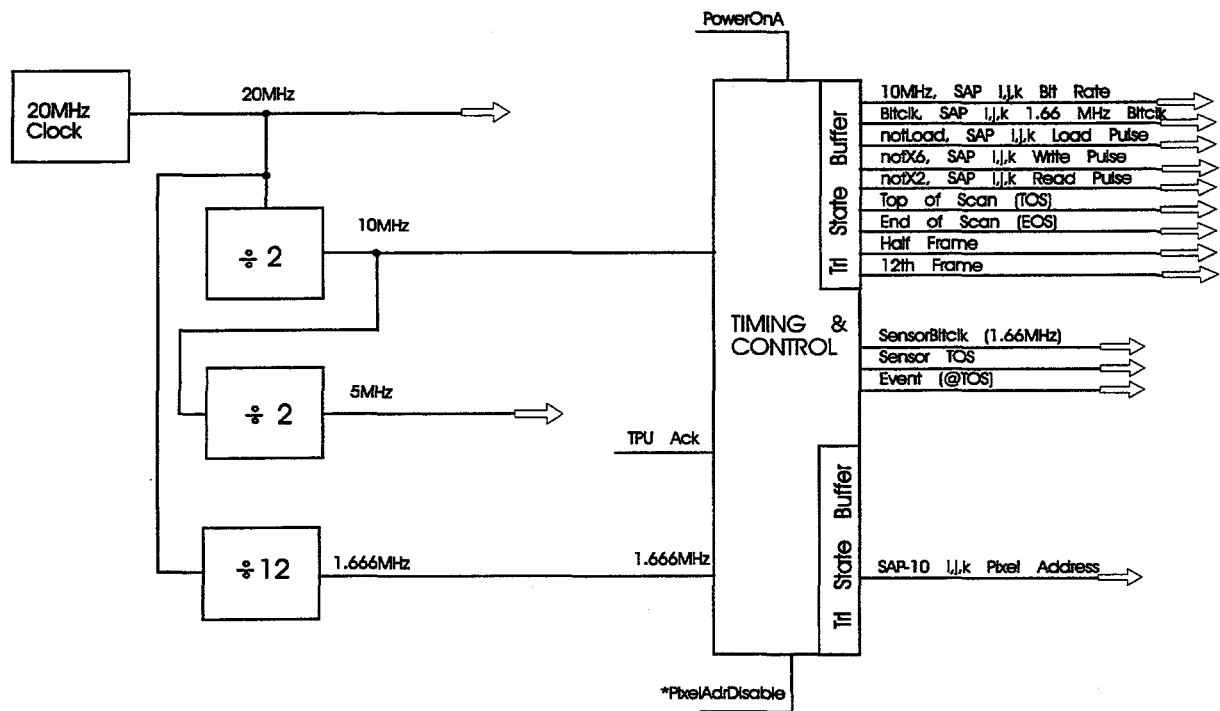
#### ***T800 Transputer***

The digital electronics design is based on the T800 INMOS transputer. The T800 is a TTL-compatible, 32-bit architecture CPU operating at 17 MIPS peak, 8 MIPS sustained (approximately 4 DAIS MIPS). Built of CMOS-on-epilayer, it is immune from latchup and has a radiation tolerance greater than 30k rads (prompt) and greater than 300k rads (Si) total dose. The T800 includes an internal 64-bit floating point co-processor which conforms to IEEE Std 754. It uses four serial communication links and consumes approximately 600 milliwatts from a single + 5 volt supply.

Three T800 transputers, labeled the Transputer Boot Node (TBN), the Transputer Master Node (TMN), and the Transputer Preprocessor Node (TPN) make up the intelligence of the LSI digital system. The primary function of the TBN is to hold the firmware for the other two transputers in non-volatile memory and to load the firmware to each computer following a system reset command. The TBN runs at a clock frequency of 17.5 MHz and addresses two banks of Electrically Erasable and Programmable Read Only Memory (EEPROM) which hold the firmware code. The total capacity of the eight EEPROM memory modules is 256 K Bytes.

#### ***CLC***

Clock and timing circuits from the CLC module are required to synchronize the various modules and sub-circuits. A CLC block diagram is shown in Figure 22. A circuit operating at a fundamental frequency of 20 MHz was designed using a Bliley BK34-1B



**Figure 22: CLC Block Diagram**

crystal. In the design, the crystal is tuned slightly in the inductive operating region to provide better frequency stability over the temperature range of operation and to reduce power consumption. All timing signals connected to the SAP modules are tri-state buffered and synchronously derived from the unbuffered 1.666 MHz clock. This unbuffered 1.666 MHz clock is also fed to the SAA module through a true complement driver. The CLC module has a power-up reset circuit that is not used in the LS1 system; the power-up reset is provided by the PSI module instead.

### ***Clock Divider***

For highest reliability a non-redundant module must have interfaces from redundant modules. It is imperative that each signal which feeds into a non-redundant module be driven by a tri-state buffer. This buffer must be powered at all times with the same power as the receiving non-redundant module. In LS1 the signal selection is controlled by signal *Power-on-A* for the A side and by *Power-on-B* for the B side.

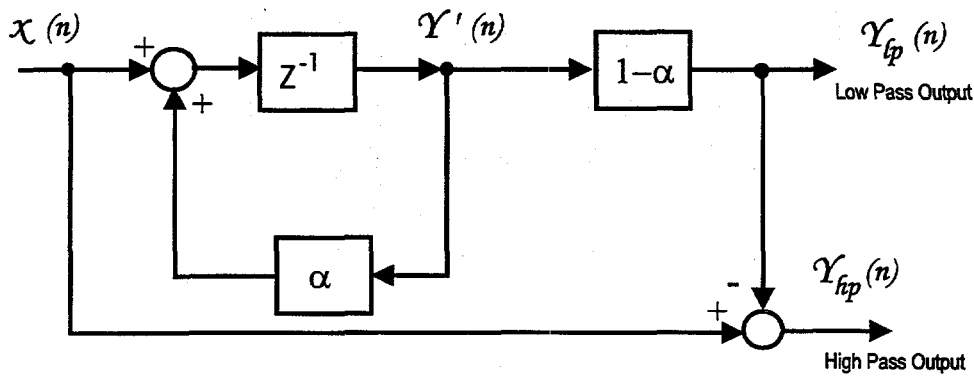
The 20 MHz clock is divided in a binary counter to yield 10 MHz and 5 MHz and divided by three to yield 6.666 MHz. This later signal is further divided by four to yield a 1.666 MHz clock. The 5 MHz clock is used in the TPU modules. Both the 1.666 and the 10 MHz clocks are essential to the operation of the SAP (i, j, k) modules. The 10 MHz clock is used to clock the P channel (i, j, k) data into the respective (i, j, k) TPU input, the

1.666 MHz clock is used to clock the pixel data from the SAA (i, j, k) into the respective (i, j, k) SAP input. Since these clocks are routed from redundant to non-redundant modules, the interfaces require tri-state buffers.

### ***The Silicon Array Processor (SAP)***

#### **SAP-10 Filter Algorithm**

The SAP-10 implements an IIR 1st order filter with both high-pass and low-pass taps. The LS1 sensor utilizes only the high-pass tap; the SSF sensor is to use both the high-pass and low-pass taps. The digital filtering is performed per pixel in time; the SAP-10



**Figure 23: The SAP-10 Filter Algorithm**

implements 1023 parallel IIR filters in hardware by time-division-multiplexing the available hardware. The sample period,  $T_s$ , is 10.4448 ms. The coefficient,  $\alpha$ , was chosen so that it could be implemented without using a true multiplication. The value of  $\alpha = (1-1/16) = .9375$  was chosen since it can be accomplished by a four-position bit-shift followed by a single subtraction.

The low-pass difference equations and Z-domain transfer function are given by:

$$y_{lp}(n) = (1-\alpha)x(n-1) + \alpha y_{lp}(n-1)$$

$$H_{lp}(z) = \frac{y_{lp}(z)}{x(z)} = \frac{(1-\alpha)z^{-1}}{1-\alpha z^{-1}}$$

The high-pass difference equation and transfer function are:

$$y_{hp}(n) = x(n) - x(n-1) + \alpha y_{hp}(n-1)$$

$$H_{hp}(z) = \frac{y_{hp}(z)}{x(z)} = \frac{(1 - z^{-1})}{1 - \alpha z^{-1}}$$

The SAP module collects digitized pixel data from the analog SAA and performs a combination Low Pass (temporal LPF) and High Pass (temporal HPF) filter. The block diagram for the SAP is shown in Figure 24. All of the essential timing signals required by this module are generated from the CLC module described above.

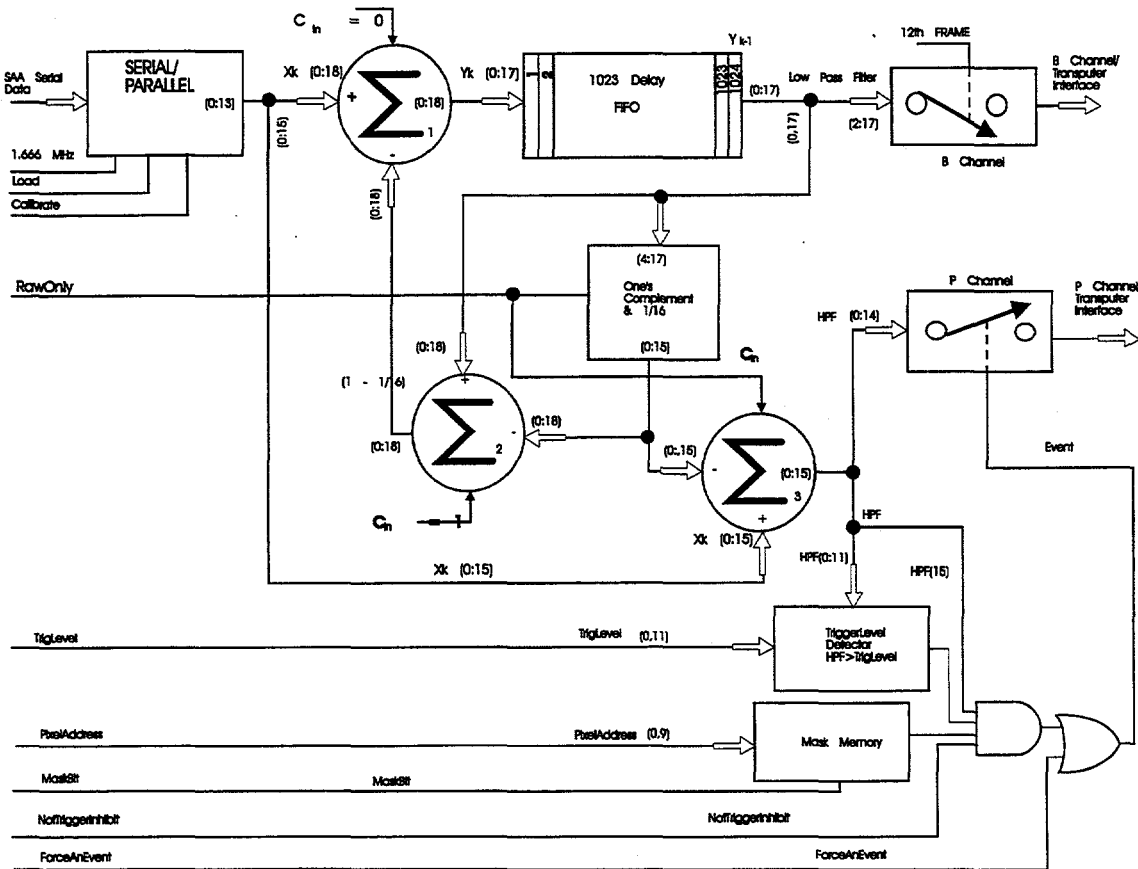


Figure 24: SAP Block Diagram

The serial pixel data is sampled at the middle of the bit-time period by the 1.666 MHz clock. During the load pulse, which occurs every 16 clock cycles, a pixel data word is formed in the 16-bit parallel register. Only the most significant 14 bits of the pixel data word are of interest; the two least significant bits are ignored. The truncated pixel word is fed to the Sigma 1 and Sigma 3 circuits.

The low-pass filter circuitry consists of the Sigma 1, the 1023 LPF delay FIFO, the Ones Complement circuit, and the Sigma 2. The circuitry is configured so that the FIFO provides only a 1023 bit delay. The first load pulse into the Serial/Parallel registers is disabled, allowing the 1024<sup>th</sup> location of the FIFO to be used as a holding register. The Sigma 2 circuit and the Ones Complement circuit combine in determining the "1-1/16" coefficient of the algorithm. The "1-1/16" factor is obtained by shifting the LPF delay FIFO output data four bits to the right at the input of the Ones Complement circuit. This signal is summed again with the LPF delay FIFO output data plus a carry-in of 1 to calculate the "1-1/16" coefficient. While the coefficient is being processed the current pixel data is added simultaneously to form new processed data. The new processed data is written into the LPF delay FIFO at the arrival of the not-Write (\*X6) pulse. The sequence of load, read, process, and write constitutes the essence of the digital filter, the entire sequence being performed in one pixel period (9.6  $\mu$ sec). As the "1-1/16" factor is being processed above, it is also being subtracted from the current pixel data to determine the HPF data. This HPF data is written into the event FIFO at the arrival of the not-Write (\*X6) pulse together with the pixel mask bit.

### ***Preprocessor Signal Interface Module (PSI)***

The PSI module forms the digital control interface between the preprocessing transputer in the TPU module and the digital signal processing in the SAP modules. It allows the transputer to write configuration data to the PSI module which then decodes and drives it to all three SAP modules. The PSI module resides on the data bus of the transputer and appears as part of the memory (i.e. it is memory mapped). There are two identical A and B redundancy address decoders with data latches which are powered by the appropriate +5VA or +5VB digital power supply. The six bytes of configuration data are decoded to form a calibrate strobe, data inhibits, trigger inhibits, raw data strobe, SAP reset, RAM enable, force event strobe, pixel mask bit and address, write masks and trigger levels.

The address decoder and a set of data latches for only one redundancy are powered at any time. Since there are A and B redundancy transputers, but only one set of SAP modules, the signals from both the powered and unpowered address decoders/data latches are multiplexed through *nonredundant* CD4019, 2:1 multiplexers to the SAP modules. The signal lines from the unpowered set of data latches are pulled low with resistors. The multiplexer is switched by sensing the +5VA and +5VB power supplies. The multiplexers are powered by a +5V bus generated by diode OR-ing the +5VA and +5VB digital buses and will be ON regardless of which redundancy is powered up. In addition to running on OR-ed power, each CD4019 has its own 1/8 amp fuse. Should any (nonredundant) multiplexer fail and blow its fuse, only one byte of configuration data

will be lost. Due to power-up defaults, LS1 should retain at least some semblance of functionality if one or more bytes of data are lost.

The occurrence of an event will only be detected if the HPF output is positive and exceeds the 12-bit trigger level threshold. It may be desirable to prohibit a particular pixel from generating valid data if that pixel is noisy or not responding properly. Disabling a pixel is accomplished by writing a zero bit to the memory mask which is controlled by the PSI during the write mode. During a normal scanning mode, the PSI releases the memory mask into the read mode and the pixel address in the CLC regains control. An event can be forced by pulsing the *Force-an-Event* pin through the PSI module. The generation of an event in this manner increases testing flexibility by allowing the raw data from the sensor to be displayed and analyzed even when no 'real' events are occurring.

Top of Scan (TOS) is the major-frame-cycle time of the system. All generated pulses and timing signals are in sync with TOS, including the pixel address lines which are initialized to zero. There are ten address lines connected into the pixel mask memories of the three SAPs. These address lines are tri-state buffered. If a pixel mask is to be written into the pixel mask memory of the SAP, an active low *Pixel Address Disable* signal from the PSI module must be activated. When the PSI completes writing, it must release the pixel address tri-state buffers to CLC control. The ten address lines change state every 9.6  $\mu$ sec.

The P Channel is an interface between the HPF event FIFO data and the Transputer. For every scan, new HPF data is written in the HPF event FIFO (1023 words). If no event is detected, the event FIFO is flushed at the TOS. On the other hand if an event is detected at the EOS, the P Channel is allowed to transfer the event FIFO data into the Transputer. The B Channel is an interface between the LPF delay FIFO data and the Transputer. Data transfer is allowed every 12th frame or 12th scan. Although the B Channel is not used in the LS1 system, it is important that this channel is functioning properly.

Since the SAA and the SAP are on the same motherboard in the LS1 system, the differential receiver U39 in the SAP module is not required. However, a jumper wire must be connected between the pad to U39-7 and the junction pad of R6-P3. The resistor in slot 13 must also be replaced with a fuse (part number PICO-267-0.25A). (Refer to SSRR # 7435-0S)

The Sensor Top of Scan (Sensor TOS) pulse is a signal provided for the SAA module. This pulse occurs four pixel periods (9.6  $\mu$ sec per pixel) before the TOS. The Sensor TOS pulse width is one cycle of the 1.666 MHz clock. The low-to-high transition of the 1.666 MHz clock must be in the middle of the pulse for the SAA module to operate properly. This pulse is not tri-state buffered but is driven by a true complement buffer.

**The following timing signals are connected to the SAP modules:**

The already-described Top of Scan (TOS) signal.

The not-Load (\*X1) pulse loads the sampled pixel data into the serial/parallel circuit of the SAP module.

The not-Read (\*X2) pulse reads the previous pixel data from the 1023 delay FIFO of the SAP module.

The not-Write (\*X6) pulse writes the processed pixel data which results from the processing of the previous pixel data and the current pixel data into the 1023 delay FIFO of the SAP. At the same time the processes pixel data is also being written into the event FIFO of the SAP.

The End of Scan (EOS) indicates the end of pixel scan. It is used to disable \*X1, \*X 2 , and \*X6 pulses at the first pixel scan and at the end of the 1024 pixel scan. Only 1023 pixels are used; the first pixel is ignored.

The Half Frame pulse is an essential time indicator for the software data collection by the Transputer. If a trigger or event has occurred, the P channel data transmission starts at the EOS and ends at the Half Frame.

Every 12 scans the 12th Frame signal allows the transmission of the B Channel data into the Transputer. Although the LSI system does not need the B Channel, the 12th Frame signal is essential to the operation of the SAP modules.

The Event timing signal is generated at the TOS and stays active-high until an *acknowledge* response from the TPU is asserted. This signal functions as an event interrupt; it provides a hardware to software operation in the Transputer.

## 2.0 SIGNAL PROCESSING

### 2.1 LS1 Event Signal Processing Software (ESP)

The primary purpose of the event signal processing software is to convert data from the three silicon photodiode sensors (i, j, k) into three event-line software arrays. The event-line arrays are inputs to the Event Detect/Decode Algorithms described in the next section. The event-line arrays contain the location (expressed as pixel numbers) and the amplitude (in arbitrary units) of the five largest pixels or group of adjacent pixels that exceed the first-order threshold of the silicon sensors. To facilitate this line detection/forming process, the software performs certain auxiliary functions, such as threshold calculations. The silicon array trigger thresholds can be commanded from the ground or can be automatically determined on-board. Threshold levels determined on-board depend on array signal input levels and various other commandable criteria. The ESP process is diagrammed schematically below in Figure 25.

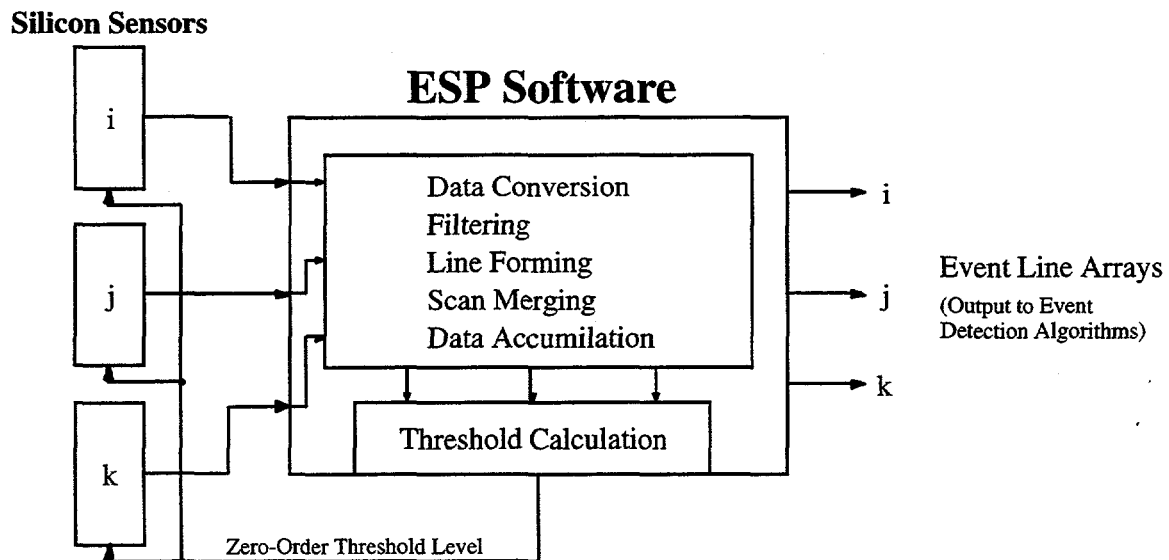


Figure 25: Block diagram of ESP Process

The ESP consists of numerous software functions (i.e., subroutines or C-language code) that perform specific processing on the input data. The packed-character-format raw data is first unpacked and converted to floating-point format. The floating-point data is filtered through a moving-sum filter to concentrate all energy into a signal line on a single pixel. (The width of the filter window is variable and commandable from the ground.) The output of the filter is input to a line-forming routine which checks all pixels for an amplitude greater than the 'first-order threshold' (FOT) and forms a 'line' if it detects a pixel or group of adjacent pixels above FOT. (In instances where there are a number of



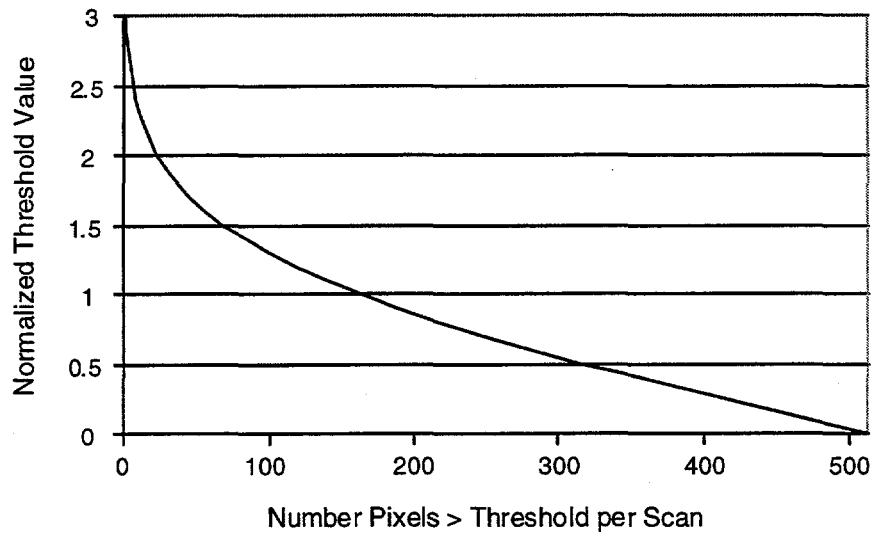
adjacent pixels above FOT, a separate 'line' is formed each time a negative inflection point is encountered.) As 'lines' are formed, they are inserted in amplitude-descending order into a list that continues to expand until the last line is formed or until the commanded maximum size of the list is reached. At this point, new lines are inserted only if their amplitude is larger than the smallest line in the list; otherwise, the potential line is discarded. Simultaneously with forming lines, the algorithm also accumulates statistics on pixels greater than FOT for subsequent use in the auto-threshold calculation routine.

After the event/signal 'lines' are formed, the ESP checks to see if lines from additional sensor scans need to be merged. Since event energy could be split between two successive scans, the software does not output any line arrays until at least two successive scans have been checked for lines. When a sensor triggers after a quiet period, the data is processed, but is not passed to the event detection software until a second scan can be examined. If the sensor does not trigger on the next scan, the data from the previous scan is output if at least one other sensor also triggered on that scan. If the sensor triggers on two successive scans, the data is processed and lines from the two scans are merged by adding the amplitudes of overlapping lines from the two scans (lines 'overlap' if their array locations match to within  $\pm 1$  pixel). The summed lines and other lines form a new list which is resorted. The 'n' largest lines are passed to the event detection algorithm, n being a commandable number. (In case there are fewer than n lines, *all* lines are passed to the algorithm.) If only one sensor triggers on any scan, the lines are discarded. The merging routine will continue merging scans as long as the sensor is triggering.

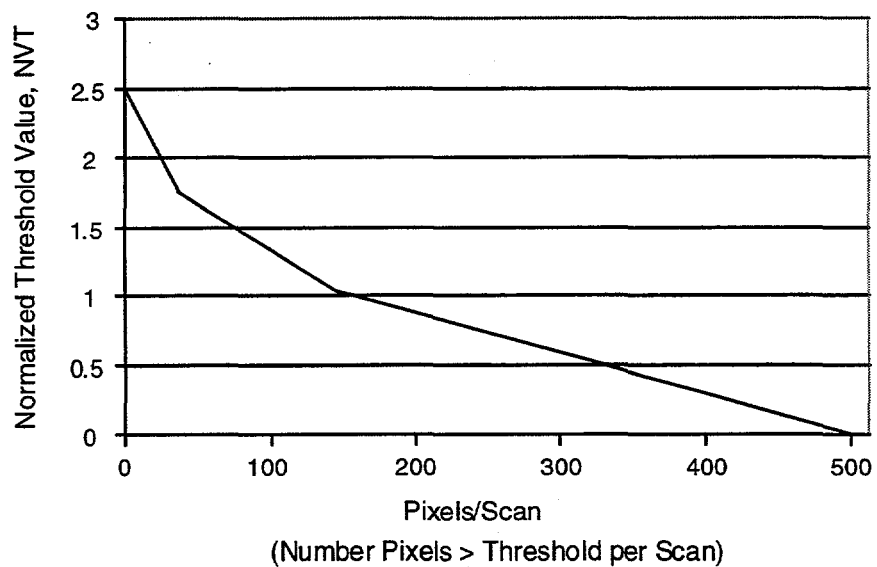
#### AUTO-THRESHOLD DETERMINATION

The auto-threshold algorithm portion of the ESP merits special explanation because of its complexity and the fact that it requires several commandable parameters whose derivations are neither obvious nor trivial. The algorithm attempts to establish a quiescent operating point that will result in a predetermined number of triggers per unit time in the absence of actual events. The algorithm assumes that the amplitude of the system noise has a Gaussian distribution. The pixel statistics used are those generated in the 'form lines' routine mentioned above.

An auto-threshold level is calculated by evaluating the Gaussian integral (error function) at a given signal level to determine the probability of a given pixel exceeding FOT due to system noise. That probability is multiplied by the number of pixels (1023) to calculate the expected rate at which pixels will randomly exceed FOT for each scan. The normalized threshold value (NVT) is defined as threshold/RMS noise. When NVT is plotted versus the 'pixel rate' the 'search curve' of Figure 26 is produced. The normalization should remove absolute noise levels from the calculation since the desired threshold (first- or zero-order) is expressed as a multiple of RMS noise and for a normally distributed, zero-mean signal the standard deviation = RMS noise. For convenience and to make the auto-threshold calculations tractable, the actual curve is approximated by three straight-line segments, as shown in the Figure 27.



**Figure 26: Auto-threshold Search Curve**



**Figure 27: Piece-wise Linear Curve to Facilitate Auto-threshold Calculation**

Table 13 summarizes the commandable parameters relevant to the ESP software:

Parameter	Units	Description
first to zero threshold ratio, tr	none	ratio of first- to zero-order threshold ( tr > 0 )
threshold calculation tolerance, tt	none	allowable error between actual and desired threshold. ( 0 < tt < 1 ). If  th <sub>act</sub> - th <sub>dsr</sub>  /th <sub>act</sub> < tol, no calculation is performed
threshold convergence parameter, tc	none	determines rate of threshold convergence ( 0 < tc < 1 )
threshold calculation interval, ti	scans	number of scans between threshold calculation. ( ti > 100 )
desired scan threshold rate, dr	pixel/scan	target number of pixels>first-order threshold/scan ( 0 < dr < 500 ) this parameter determines auto-threshold level)
number of pixels to sum, np	pixels	window width of moving sum filter ( np > 1 )
number of lines to process, nl	lines	number of event lines to pass to event detection algorithms ( 3 < nl < 15 )
autothreshold mode	none	flag to enable/inhibit autothreshold. Calculation ( '0' = enabled )
zero order threshold commanded. ZOT	counts	manual threshold setting ( 0 < ZOT < 4095 )

**Table 13: Commandable Parameters Used in the Auto-Threshold Calculation**

Since the ESP routine must examine each pixel value in a search for amplitudes larger than FOT, it can easily accumulate statistics on pixels greater than FOT as well as the number of scans that have triggered. When the specified update interval has elapsed, the algorithm divides the total number of pixels greater than FOT by the number of scans that have been triggered to obtain a  $Pixels/Scan_{old}$  allowing  $NTV_{old}$ , an estimated value of the present NTV, to be determined from Figure 27.  $Pixels/Scan_{new}$  is calculated as follows using the convergence parameter, tc, and desired  $Pixels/Scan$  value, dr as follows:

$$Pixels/Scan_{new} = Pixels/Scan_{old} + tc * ( dr - Pixels/Scan_{old} )$$

$Pixels/Scan_{new}$  is then used with Figure 27 to determine  $NTV_{new}$  and finally:

$$FOT_{new} = FOT_{old} * ( NTV_{new} / NTV_{old} )$$

The speed with which the auto-threshold converges to the desired threshold is directly dependent upon the convergence parameter,  $t_c$ , since  $0 < t_c < 1$ . Separate FOTs are recalculated every 'ti' array scan intervals. The largest of the three calculated auto-threshold values is used by the processing software as the FOT for all arrays unless a manual threshold level, ZOT, has been specified instead.

## 2.2 Event Detect/Decode Algorithms

Once line sets from at least two sensors have been formed, a Detect/Decode algorithm examines combinations of lines to determine if they could possibly be diffracted laser-line signatures. The analysis flows as follows.

1. Select a pair of sensors: ij, ik, or jk
2. Select two lines from two different arrays.
3. Estimate direction cosines assuming a nominal index of refraction and assuming that the lines are zero order (equations 2.0 and 2.1).
4. Choose a candidate first order line that has an amplitude greater than the minimum first-to-zero order amplitude ratio and less than the maximum first-to-zero order amplitude ratio. The candidate line's position relative to the zero order candidate on the array must also be greater than the minimum order separation and less than the maximum order separation.
5. Estimate the wavelength of the source that could have created the three lines using equation 3.0 and check to see if it is greater than the minimum wavelength and less than the maximum wavelength.
6. Update the index of refraction and direction cosine estimations used in step 3.
7. Using the candidate zero order locations, the candidate first order location, and the estimated wavelength, compute an estimated first order location on the array from which only one zero order line was selected. If this position matches a line on the array to within the location tolerance, a candidate event is recorded. This event is called a four-line event.
8. Repeat steps 1-7 for all combination of lines on the first sensor pair. While analyzing all the combinations of lines on two arrays, two four-line events can be made up from the same lines. If the wavelength and direction cosines match to within tolerance, the events are combined.

9. Repeat steps 1-8 for the two remaining sensor pairs. This yields three lists of four-line events.
10. If any of the four-line events have matching wavelength and direction cosines, they are combined into a three-array event. A list of three-array events is compiled.
11. From the four lists of events: ij, ik, jk, events which have too large of a first-order prediction error, or too small an amplitude (such as would be formed by noise lines), are eliminated.
12. Of the remaining events, the three-array event with the lowest first-order prediction error is reported first. If no three-array events are detected, the four line event that has the lowest first-order prediction error is reported.

## 3.0 TESTING

### 3.1 Environmental

The following environmental testing was performed on the LS1 prior to delivery to the Air Force.

#### *Ambient Thermal Tests:*

- 15 hot cycles and 8 cold cycles
- Full functional tests on first cycle and last 3 cycles
- Temperature ramped at 1° C/min
- Functional tests at ambient
- One hour soak unpowered at -40° C, Cold start tests
- 12 hour soak powered at -20° C, Functional tests
- 12 hour soak powered at +55° C, Functional tests
- System cycled through redundancies, voltages, and modes between functional tests

#### *Vibration:*

- Random vibration tests: Qualification: 14.1 G RMS; Acceptance: 10.0 G RMS
- 1 minute each axis
- No failures detected on post analysis
- Optical alignment performed before and after vibration with no deviations detected

#### *Pyroshock:*

- Toric lens failed at original full shock specification
- Analysis of the shock environment concluded that the sensor subassembly specification should be 50% of that for the baseplate
- Sensor specification set at 1000 Gs 1-10 kHz resonant plate method
- Lens passed at reduced shock levels

#### *Thermal Vacuum:*

- 11 temperature cycles
- Low temperatures varied from -20° C to -40° C
- High temperatures varied from +30° C to +46° C
- Functional tests performed at ambient
- 12-hour soaks powered at low temperature
- 12-hour soaks powered at high temperature
- One 12-hour unpowered cold soak was performed at -40° C to verify cold start
- Full functional tests done on cycles 1, 4, 6, 8, and 11
- Temperature ramped at 0.33° C /min

- System cycled through redundancies, voltages, and operational modes between all functional tests

**Results:**

System operates reliably and meets requirements over the range -25° to +35° C  
System powers 'ON' reliably at -40° C after 12 hour unpowered soak

### 3.2 EMC Testing

Electromagnetic testing for all of the TAOS satellite was conducted at the Goddard Space Flight Center. All payloads were powered during the testing. Included were emission tests to measure the EM output from the LS1. The major frequencies of interest were from 100 MHz to 10 GHz, with additional testing from 14 kHz to 100 MHz. The EM susceptibility of LS1 was tested by operating in the triggered background mode at a manual threshold level of 40 hexadecimal. During the emissions tests state-of-health and configuration data were collected every 15 seconds. In two days of emissions testing no anomalous data was observed.

### 3.3 TAOS Automated Test Unit Ground Station (TATU)

The LS1 TAOS Test Unit (TATU) is used to collect and display LS1 data received from the 1553B bus via the BIM. A program running on a VAX called *LS1 Sender 2 Wire* extracts the LS1 data from the appropriate 1553B messages and transfers those messages via RS232 to the TATU. Details of the ground station operation are provided in an appendix.

## 4.0 COSTS

There was a tremendous overlap in the design, prototyping, fabrication, and test phases of the project. Because of the overlap, it is difficult to accurately account for the man-hours required for each phase of the project. LS1 program time and money were reduced by borrowing a portion of the system concept and technology from another program, and developing and qualifying our own fabrication sources. Conversely, since different approaches were tried simultaneously to produce system solutions in minimum time the total cost were raised somewhat.

Our best estimate of a cost breakdown is:

	<u>Actual</u>	<u>If Done Stand-alone</u>
Labor	1,413K	2,489K
<u>Direct Support:</u>		
Drafting	81K	105K
Computer Charges	1K	1K
Technical Information	1K	1K
Shop Charges	148K	148K
Test (Did Not Include EMC)	3K	4K
<u>Direct Charges:</u>		
Travel	12K	21K
JIT	17K	30K
Taxes, G&A	157K	277K
Non-Module, Non-Mechanical Purchases	237K	237K
Module Fabrication (Excluding Mechanical)		
Layout	30K	49K
Fabrication	24K	28K
Assembly	11K	13K
Mechanical Drafting	131K	190K
Purchased Mechanical Parts (Excluding Module Fab, Connectors, Wiring, etc.)	62K	84K
In House Shop Charges (Estimated, Fab of Test Parts, Misc.)	9K	30K
Development Costs (Estimated, Test Fixtures, Prototype Parts)	1K	25K
<u>Total</u>	2,338K	3,732K

Note: LS1 utilized several mechanical, optical, and module designs which were developed jointly with another project. The "Stand Alone" cost is an estimate of what would have been spent without the benefit of the joint development.



## 5.0 SCHEDULE

### Time Required for Various Tasks

Concept Definition (9/23/89 - 10/23/89):	1.0 months
Note: Concept was largely generated as part of another system.	
System Requirements Analysis (10/23/89 - 1/19/90):	2.9 months
Electronics:	
Preliminary Design (10/23/89 - 3/30/90):	5.2 months
Detailed Design (1/29/89 - 5/1/91):	17.1 months
Breadboard Fabrication and Test (8/1/90 - 4/9/91):	8.3 months
Flight Layout (8/8/90 - 5/24/91):	8.8 months
Flight Fabrication (1/28/91 - 6/6/91):	4.3 months
Flight Assembly (3/15/91 - 6/12/91):	2.9 months
Conformal Coat (8/14/91 - 8/18/91):	0.2 months
Mechanical:	
Design (10/23/89 - 3/21/91):	16.9 months
Fabrication (8/28/90 - 5/10/91):	8.4 months
Optics:	
Design (9/15/89 - 9/15/90):	12.0 months
Fabrication (9/15/90 - 5/21/91):	8.2 months
Software:	
Requirements Analysis (1/19/90 - 6/11/90):	4.7 months
Design and Coding (6/11/90 - 9/2/91):	14.7 months
Test Unit:	
Preliminary Design (6/15/90 - 7/3/90):	0.6 months
Hardware Build (7/3/90 - 11/1/90):	3.9 months
Software (7/3/90 - 6/1/91):	10.9 months
System Functional Testing (6/17/91 - 7/16/91):	1.0 months
Optical Calibration and Test (8/23/91 - 9/9/91):	0.5 months
System Environmental Testing:	
Thermal and Related Fixes (7/16/91 - 8/11/91):	0.9 months
Vibration (9/4/91):	0.0 months
Mass Properties (9/4/91):	0.0 months
Thermal-Vac and Related Fixes (9/10/91 - 10/6/91):	0.9 months

## **6.0 LS1 SYSTEM OPERATIONS**

### **6.1 Laser Sensor Objectives**

The LS1 laser sensor, shown above in Figure 6, was designed by Sandia National Laboratories to fulfill a primary mission of detection and characterization of pulsed, visible, and near-infrared laser illumination of the Air Force Phillips Laboratory's TAOS spacecraft. The orbital objectives of the LS1 were to prove the detection, characterization, and location of laser illumination events, with a secondary mission to characterize background clutter and to collect other signals for data analysis. The sensor and spacecraft were launched in March 1994 and had an expected mission lifetime of one year.

The LS1 design was based on the slit-grating optical concept used in the SSF laser sensor which is to be flown aboard the DMSP spacecraft slated for launch after the year 2000. In advance of the SSF system deployment, the LS1 experiment sought to verify that the slit-grating concept could meet its sensitivity goals in the clutter-noise limited environment of earth-looking, low-earth-orbit (LEO) satellites.

The LS1 system is designed to characterize incoming pulsed optical signals and determine their time-of-arrival, angle-of-arrival, wavelength, and amplitude. Experiments conducted using ground-based lasers have confirmed the performance specifications of the LS1 sensor. The LS1 system helped answer feasibility questions for the follow-on SSF system, to refine detection and discrimination algorithms, and to predict future sensor performance by collecting raw background data. The LS1 system also provided data on false event sources and glint data and allowed statistics to be developed on sensor false alarm rates.

### **6.2 LS1 Operational Modes**

Wavelength, amplitude, and time-of-arrival performance as well as significant information on background clutter has been characterized. Although the AOA performance appears good, inaccuracies in the reported satellite attitude data have made quantifying the exact AOA performance somewhat difficult. Numerous efforts at compensating for the satellite attitude data deficiencies have been tried with varying amounts of success. The details of these efforts are contained in a classified final report on the TAOS mission published by the U.S. Air Force.

The data collected for operational assessments of LS1 were acquired using one of the following operational modes. This data was processed using test equipment located at the Contractor Support Facility in Mountain View, California and later transferred to Sandia over a secure phone line for analysis.

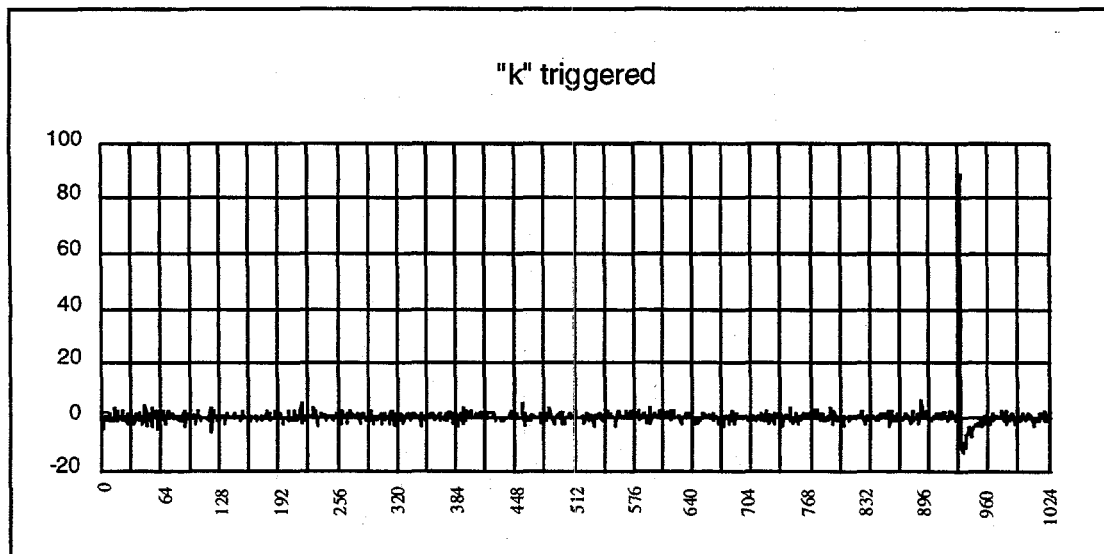
### ***Event Detection Mode***

The basic operational mode of LS1 is to detect and characterize laser illumination. Optimal system configuration allows operation with a low probability of false events. Three-array coincidence mode requires a trigger from all three arrays before an event is reported. Two-array coincidence mode only requires triggers from two selected arrays. The sensors were operated in both the three-array and two-array coincidence modes.

### ***Triggered Background Mode***

Data collection in the triggered background mode begins when a trigger signal exceeds the commanded threshold levels. The event processing algorithms are not enabled during the triggered background mode. This mode collects information on background signals which may not be laser events, i.e. energetic particles. Triggered background data collections were conducted on several occasions.

The triggered background mode allows raw data to be collected from any array exceeding threshold. This allows the amplitude and frequency of random triggers to be characterized even though event processing is disabled during triggered background collection. A trigger level of 40 was selected as the commanded threshold and a frame of data from any triggered array was collected at a maximum rate of one frame per minute.



**Figure 28: Typical Triggered Event Data**

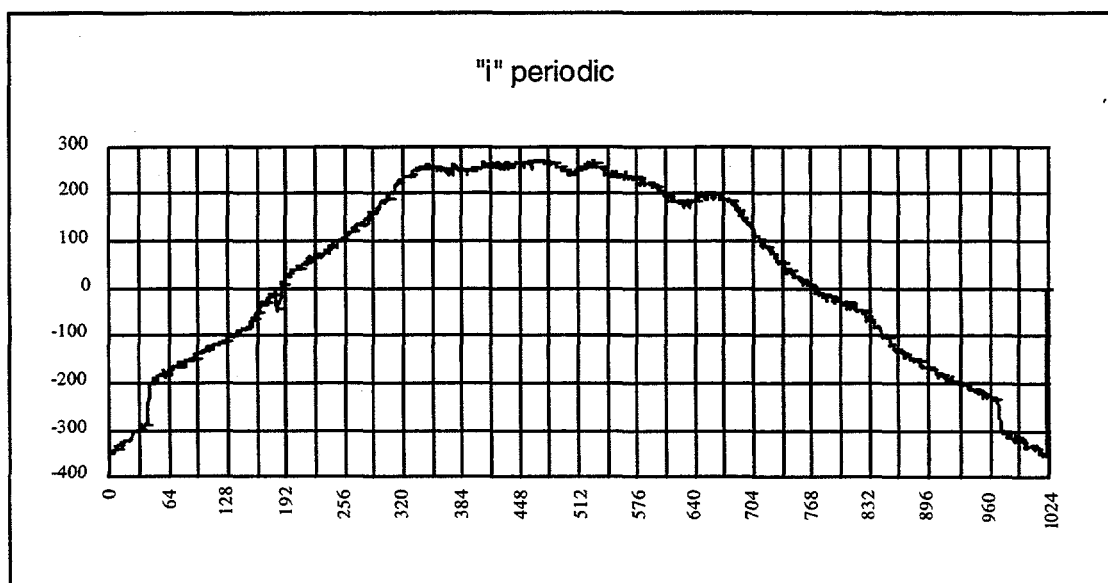
The information collected generates a data base of triggered signals most of which are the result of energetic particle hits. An energetic particle deposits energy to the sensor, and produces signals from, only a few pixels. A typical triggered event is shown in Figure 28. Certain earth positions, particularly over the South Atlantic, generate a very high number of triggers. With raw data collected at a maximum rate of one frame per minute, triggers occurring more often than once per minute are not processed. However, a trigger counter advances once every 10 milliseconds if a scan is above the threshold. Over areas such as the South Atlantic as many a 600 triggers have been counted in a one minute period. In a single 24-hour period the trigger counter has recorded in excess of 10,000 triggers at a threshold level of 22.

### ***Commanded Background Mode***

This data mode is similar to the periodic background mode described below but only one or two frames of data are collected. This mode was not used on-orbit; instead data collections were made using the periodic background mode.

### ***Periodic background Mode***

The periodic background mode allows two consecutive frames of data to be collected from all arrays at a selectable interval to characterize the slowly varying background signals. The rate, or *period*, of data collection can be selected from 1 second to as many as  $2^{32}$  seconds (136 years) in 1 second increments. On several occasions periodic background data were collected every minute over a 24 hour interval. No event or triggered data processing occurs in this mode and the spatial and temporal filters are disabled. The resulting low frequency background signals will appear superimposed on the array fixed pattern noise.

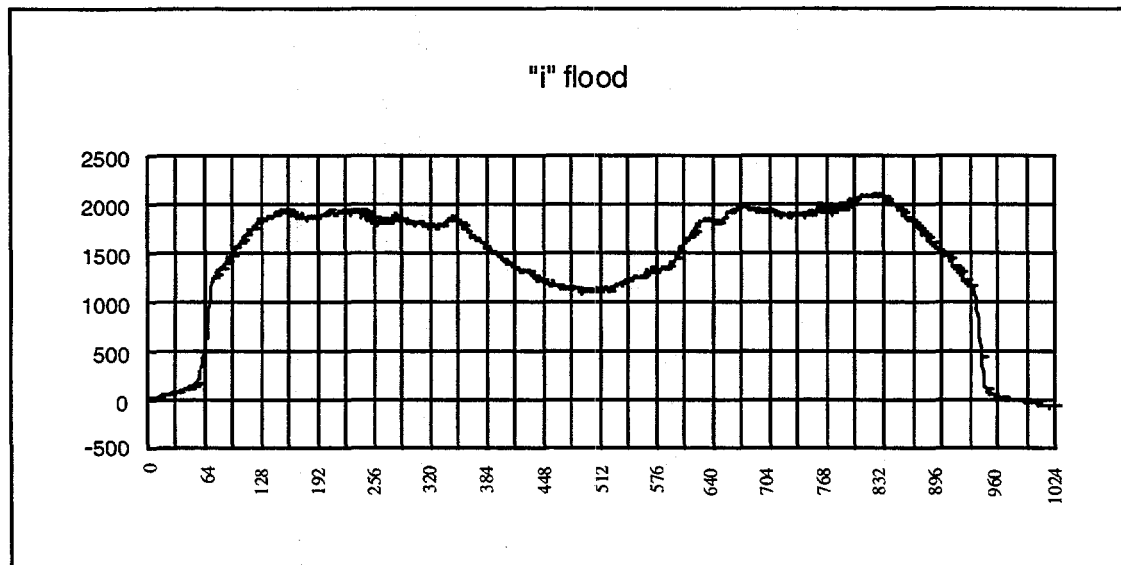


**Figure 29: Data Collected in the Periodic Background Mode**

The signals collected in the periodic background mode generated a data base representing all illumination conditions over the orbit of the satellite. They tracked the earth's albedo as seen by the sensor. A typical periodic background collection for an illuminated scene is shown in Figure 29.

### ***Image and Flood Illumination Calibration***

Two types of internal, optical calibration signals are generated for the LS1 system. These are to stimulate the photodiode array and verify operation of the processing electronics. During either calibration a flag bit is set to indicate that the data is the result of testing as opposed to actual data. The mirror used to redirect the calibration signals onto the linear array was shown earlier as Figure 17.

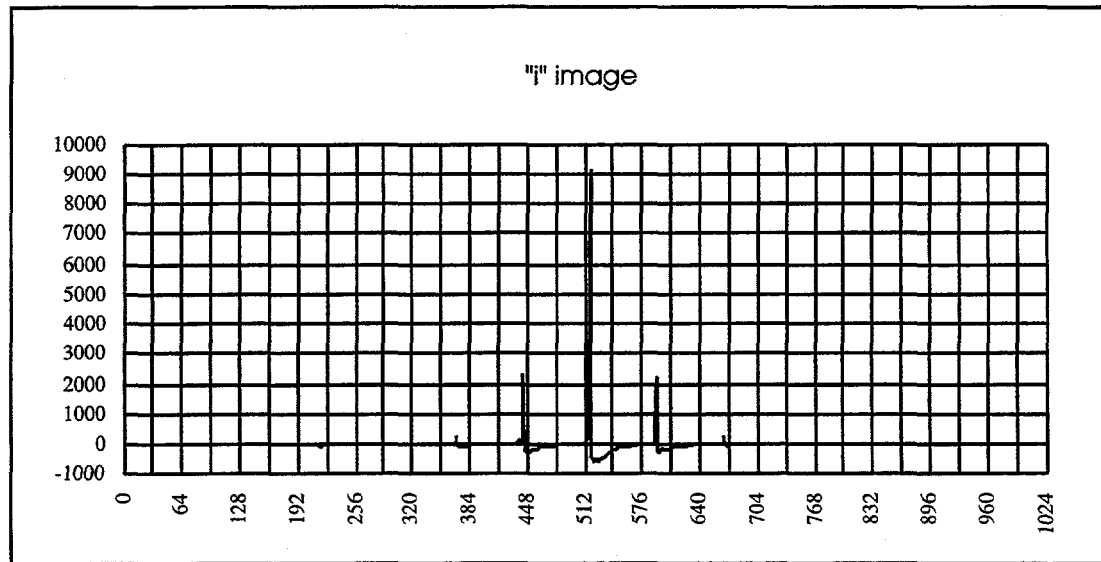


**Figure 30: Flood Illumination Calibration Data**

The *flood illumination* is used to verify the operation of individual photodiodes of the linear array and is generated by two TIL-24, gallium arsenide diodes each illuminating approximately one half of the array. The start signal for the flood illumination is synchronized with the top-of-frame signal for the array and two frames of data are collected. Since the array is an integrating detector, the signal acquired is an increasing signal (in time) on the first frame and a decreasing signal on the second frame. If these two frames are combined, the total optical signal falling on the array can be reconstructed. The spatial divergence of the diodes generates the light output shown in Figure 30 without the aid of intermediate optics.

The *image calibration* signals are generated using an Hitachi, HL7836MG98, gallium aluminum arsenide laser diode. The laser diode output is passed through imaging optics and a finely-spaced diffraction grating to generate a diffraction pattern on the linear array

similar to that generated by an external laser signal. Very fine control of lens focus generates an imaged spot size of only a few photodiodes. The laser diode start signal is synchronized with the top-of-frame signal. The measured optical signal is allowed to trigger the system and be processed by the detection algorithms. A typical output signature observed from an image calibration is shown as Figure 31.



**Figure 31: Image Calibration Data**

Although the image calibration system does have temperature compensation circuitry, there is no absolute amplitude calibration for either the flood or image signals. All of the recorded on-orbit image calibration measurements were within nominal specifications.

### 6.3 LS1 Performance Assessment

Multiple tests of LS1 were planned and executed over the life of the system. Some of these tests were measurements of background conditions while others used various ground-based laser sites as test sources. Most of the orbiting test with laser illumination were performed during terminator conditions -- times when the satellite could be visually located using a telescope. The visual location made the laser-pointing problem somewhat easier. All of the laser tests required coordination between many different organizations and would not have happened without the excellent planning and cooperation of all involved. A single mission involved coordination of the lasing facility, ground data station, voice links between the two, as well as personnel from Phillips Laboratories and Sandia. These events verified the feasibility of testing a satellite laser sensor to verify the accuracy of measured wavelength, angle of arrival, and amplitude data from ground sources.

<b>LS1 was successfully tested by US ground laser sources with:</b>
over 100 events collected
multiple wavelengths
amplitude, wavelength and angle of arrival within specifications
thousands of background frames collected

## 6.4 LS1 Lessons Learned

### *MIL Fabrication*

External MIL standard fabrication was used for LS1 with many of the modules being laid out according to MIL standard practices. MIL standard PC board layout and fabrication has worked well; the LS1 system has endured environmental testing and several hundred hours of pre-launch operation without any failures that can be attributed to this process. On orbit operation has shown no failures due to fabrication.

### *Quality Monitoring*

Fabrication requiring this level of quality has to be monitored carefully by someone knowledgeable in the fabrication process. Several vendors were evaluated for many years in order to establish three fabrication houses that could provide the required quality in the required delivery time.

### *Borrowing Modules from Other Systems*

It was beneficial not having to develop all of the modules as part of this project. The down side of this is that it is very difficult to design a module to be generic. A few minor changes were needed in the borrowed modules to make them more useful. The biggest problem occurred in the area of power distribution. Most modules tend to be designed assuming a particular power distribution. With all the cross-strapping that was done to supply 3 sensors from 2 power supply redundancies, it was very difficult to tailor the power distribution scheme to the borrowed modules.

### *Redundancy and Cross-Strapping*

Cross-strapping 3 sensors to 2 logic and 2 power supplies caused numerous misunderstandings in the interfaces between modules and added to the complexity of the system. For example, the SAP modules expected a single 5 volt sensor supply. The SAPs could not be powered by the sensor 5 volts supply because that added too much noise to the sensors. In order to move them to the 5 volt logic supply, the power supply

had to be modified to OR the A and B logic power. This change introduced an inadvertent fault mode which was discovered after delivery and had to be fixed at the integrating contractor.

Another problem with cross-strapping is ground loops. There were literally dozens of signals which crossed the sensor/logic interface. Special attention was given to each of them in order to minimize current sourced by one supply and sunk by another. Cross-strapping also creates single-point failure possibilities that must be eliminated. That single point in this system is where the *two* logic/power systems connect with each of the *three* sensor systems. The necessity of cross-strapping should be considered carefully rather than assumed.

### ***Ground and Power Planes***

The main criticism against ground and power planes is that they require extra PC board layers. There is also an increase in capacitance which must be tolerated, but often that is helpful. It was found that by using ground and power planes, coupling between PC traces was greatly reduced and "textbook" signals became the norm rather than the exception.

### ***High-Frequency Electronics***

The Transputers and their support chips used a lot of power. We had to heat-sink several of these parts, but their high internal temperatures still required us to specify a lower maximum temperature limit from the spacecraft contractor.

The Transputer clock ran at 17.5 MHz. That created 29 ns timing pulses in many areas of the box. This is on the verge of having to treat each signal trace as a transmission line. Gate delays had to be carefully accounted for because many of the delays were as long as the timing pulses or longer

### ***Sensor Noise Limits***

It is possible to approach the noise limits published for the sensor and preamplifier components if (a) the power supply designer works closely with the sensor designer and tailors the power supply accordingly, (b) a grounding plan is developed early in the project which is flexible and can be finalized with jumpers, and (c) the modules and motherboard are laid out very meticulously to provide adequate isolation for sensitive signal and power lines. Opto-coupled devices solve numerous interface isolation and noise problems but use a bit more power than other interface devices.

### ***Communication Protocols***

We implemented sync words, checksums, handshaking, message sequence numbers, and re-transmissions. All of these proved useful for tracing transmission problems and improving noise immunity. Had we been forced to conform to a fixed link format, it



probably would not have been possible to use the handshaking and re-transmissions as effectively. In this case, it would have been better to substitute an error correction scheme.

### ***Silicon Arrays***

The silicon array fixed pattern noise was significant and varied with the layout of the clock signals. It is difficult to remove any background level from a sensor array, because you cannot put an analog high pass filter on each pixel, you must resort to other techniques such as digital filtering.

### ***Structured Software Analysis***

Structured analysis was used to communicate the software requirements to the Command and Control software designer. This proved useful in two ways; the software met the requirements that were generated as a result of the analysis, and it was possible to track the software development easily and prioritize tasks.

### ***Low Earth Orbit Spacecraft***

The radiation levels were low enough that several new non-rad-hard technologies could be used. Single Event Upset and latch-up had to be considered since the proton and cosmic ray environment is not much different than other orbits.

The velocity of LEO spacecraft is much higher than the velocity at higher orbits. The most dramatic effect is the angular velocity at which points on the ground move past the spacecraft (or at which the spacecraft moves past points on the ground). The particular problem for this sensor was a higher noise level due to fast moving ground clutter.

### ***Lenses***

Cleartran™ zinc sulfide lenses turned out to be difficult to polish and were not as transmissive as originally thought. The transmission problem was overcome by rejecting poorly polished lenses.

### ***Thermal Modeling***

Thermal modeling at the payload, and even the component, level was essential. This was especially true since we used such power hungry devices as Transputers and fast RAM.

### ***Reprogramming Connector***

This has been used 6 times since delivery. In most cases the system was reprogrammed to remove minor bugs discovered during integration. The customer was very pleased that, not only did the system work well, but the minor annoyances were easily fixed.

## A1 Box Structure



## A2.1 Optical Subassembly



**NOTES:**

1. UNLESS OTHERWISE SPECIFIED, ALL DIMENSIONS ARE IN MILLIMETERS.
2. MATERIAL: CVD CLEARTRAN ZAS.
3. DIAMETER ALL EDGES 0.5/0.5.
4. SURFACE ROUGHNESS: 10-5.
5. WITHIN CLEAR APERTURES.
6. ALL NON-CLEAR-APERTURE SURFACES GRIND TO SURFACE ROUGHNESS 10-5.
7. EDGE CHIPS TO BE < 0.5.

**TABLE 1: DIMENSIONS**

VIEW	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100

**TABLE 2: SURFACE ROUGHNESS**

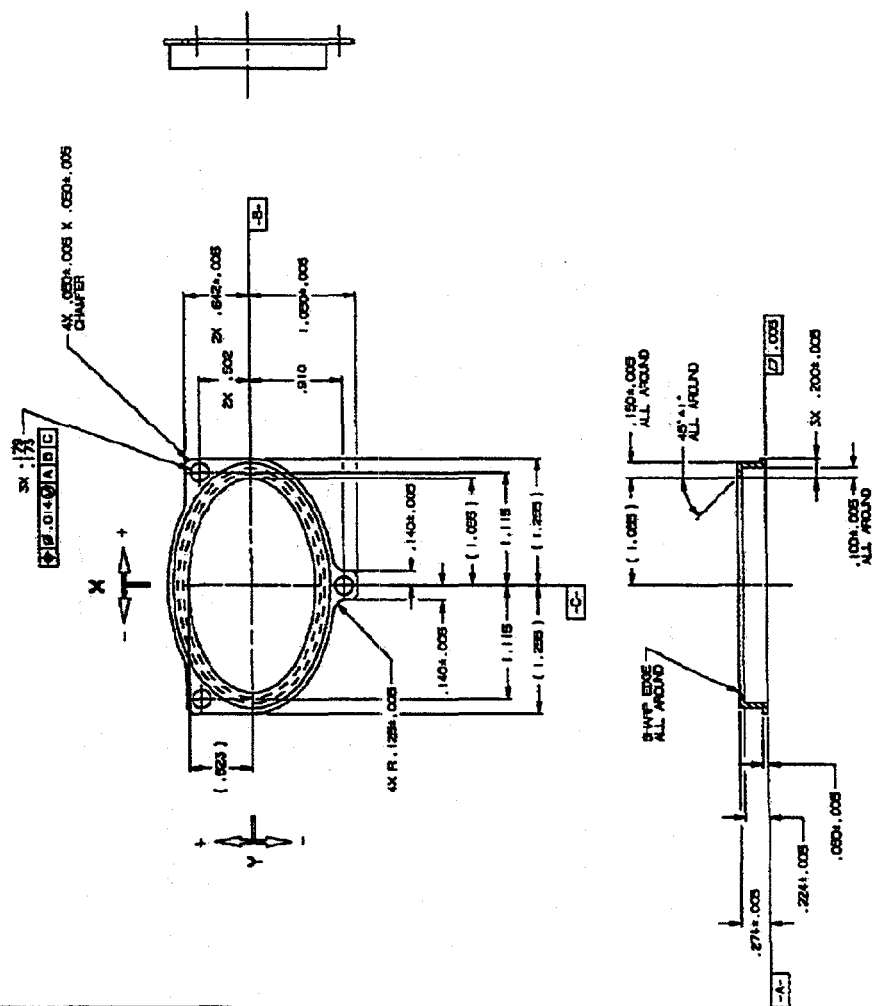
VIEW	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100

COORDINATES FOR BL 102	X	Y	Z
1	0	653	
2	230	567	
3	527	581	
4	432	578	
5	485	577	
6	1520	540	
7	571	521	
8	617	502	
9	680	483	
10	701	462	
11	740	441	
12	776	418	
13	812	384	
14	845	360	
15	878	341	
16	908	312	
17	940	278	
18	970	241	
19	1000	187	
20	1027	159	
21	1055	0	

[illegible]

CROSS INDEX PART CLASS	PETITIONS				
	1st FILED	PREPARED BY	DESCRIPTION	DATE	DATE FILED
R3.922Z-000	A	D. KESLER D. LEE	2082 9223		0.

1. GENERAL REQUIREMENTS FOR 9000000.
2. MATERIAL, PLATE, ALUMINUM, 6061-T651 PER  
ASTM A-26011.
3. ANODIZE PER MIL-A-8628, TYPE 1, CLASS 2, DYE  
BLACK.
4. STRESS RELIEVE FINISHED PART AS FOLLOWS:  
A. COOL PART TO -100°F AND SOAK FOR 1 HOUR.  
B. REMOVE PARTS FROM COOL SOAK AND CLENCH IN  
BOILING WATER FOR SOAK 1 HOUR.  
C. REPEAT PROCESSES A AND B A MINIMUM OF THREE  
TIMES.

[illegible]

## **A3            LS1 Test System and Data Processing Software**

### **A3.1           Introduction**

To support the testing and operation of the LS1 sensor, a test system and associated data processing software was designed and created. In addition to the test system, various programs were written to process the flight data and to assist in the declassification of flight data.

### **A3.2           TAOS/LS1 Automated Test Unit (TATU)**

TATU is a PC-based test system that allows real time commanding of LS1 and limited real-time display of information such as LS1 messages and monitors.

#### **A3.2.1           HARDWARE DESCRIPTION**

The test system, known as TATU, is based upon an IBM PC/AT. A custom designed and wire-wrapped AT-bus interface board is installed in the PC. This board allows the PC to interface and communicate with LS1 and control the operation of LS1. Interface board functions include conversion to and from the opto-coupled serial communications links, generation of opto-coupled ON, OFF and RESET pulses for both redundancies, and generation of a 1 Hz interrupt to the PC for timemark generation and clocking purposes. The input and output of the opto-couplers are routed to the PC's serial/parallel board. A 10 foot custom cable carries the signals between TATU to LS1.

In addition to the PC, a Hewlett Packard 6633A programmable power supply is used to provide the  $28 \pm 4$  bus voltage to LS1. An IOTech™ IEEE-488 interface board is installed in the PC to control the power supply. A Hewlett Packard LaserJet II is used with the PC for printing. The PC system is housed in a rolling PC cabinet for portability and ease of use.

#### **A3.2.2           SOFTWARE**

Individual software modules used for TATU are described below.

##### ***Screen***

The PC display screen is split into windows:

- A command line for the operator to type commands
- A log window where informative and error messages are printed
- A serial window where LS1 messages scroll by, a monitor window showing the 28V monitor, the heart beat, the event monitor, the time, redundancy information (A/B, on, off and reset), command file status (file, delay and pause)
- Various pop-up windows.

### ***Definition of TATU Commands***

clear	Clears the log and serial windows and refreshes the monitor window.
cont	Continue after pause; clears the pause and resumes execution of the command file.
delay xxxx	Delays execution of the next command for xxxx seconds (a floating point number); normally used in command files but can also be used as a "timer" from the keyboard.
enable/disable beep	Turn the audible beep associated with an error on or off.
enable/disable byte file	Opens "tatufile.by" and writes all bytes received over the serial port to this file in addition to the normal processing of LS1 messages. Used for diagnostic purposes only.
enable/disable checksum	Enables/disables the checksum checking at the end of an LS1 messages; if disabled, the message output to the screen may be garbage. Normally used for diagnostic purposes.
enable/disable clock	Turns the 1 Hz protoboard clock on or off and enables/disables the interrupt handler for that clock; in effect, this enables/disables the clock shown in the monitor window. The clock must be enabled before the timemark and coarse time may be enabled.
enable/disable coarse time	Enables/disables the creation and sending of the coarse time message packets to LS1 (triggered by the 1 Hz clock). Normally used only as a diagnostic. [See "enable/disable timemark" and "enable/disable timemark pulse".]
enable/disable log	Opens the "tatufile.log" log file and sends all error messages and echoes of all operator commands (including command files) to that file.
enable/disable power	Enables or disables the output of the HP6633A power supply that supplies power to LS1.
enable/disable serial	Enables the sending of commands to LS1 and the receiving of messages from LS1 by setting the proper serial port registers and enabling the serial interrupt.
enable/disable timemark	Sets a bit on the protoboard to send the 1 Hz clock to LS1 as the timemark and also sets a flag to enable the creation and sending of coarse time messages to LS1.

enable/disable timemark pulse	Sets a bit on the protoboard to send the 1 Hz clock to LS1 as the timemark but does not enable the coarse time messages. Normally used only as a diagnostic. [See "enable/disable timemark" and "enable/disable coarse time".]
echo ack on/off	A confirmation message is written to the log window whenever an ack is sent or received.
echo all segments on/off	When off, only the first segment of state-of-health, system configuration, and raw data messages are echoed to the serial window. Used to control the volume of messages written to that window.
echo cmd on/off	Echoes all operator or command file commands to the log window with a time tag.
echo input on/off	Writes all bytes received from the serial port to the serial window; used for diagnostic purposes only.
echo nack on/off	A confirmation message is written to the log window whenever a nack is sent or received.
init ieee	Initializes the IEEE-488 bus between TATU and the HP6633A.
note <i>text</i>	Writes <i>text</i> to the log window verbatim; used in command files to write notes to the screen.
pause	Halts execution of the command file until the operator presses F1 to release the pause. A command file can be forcibly paused by the operator pressing F1 at any point.
power on/off	Pulses the discrete ON or OFF command line to the appropriate redundancy of LS1. Command may be shortened to simply "on" or "off".
proc <i>filename</i>	Opens the specified command file, interprets the contents and executes all recognizable commands
quit	Exit TATU.
reset ls1	Pulses the discrete RESET command line to the appropriate redundancy of LS1.
reset hp6633a	Resets the state of the HP6633A. Normally not needed.
reset files	Resets the byte, log, message and monitor output files of TATU to the beginning of the files.



save screen filter = <i>filename</i>	Saves the output filter of the serial window to the specified file.
select a/b	Sets a bit on the protoboard that switches communications between the A and B redundancies of LS1. Affects the ON, OFF, RESET, serial, heartbeat, event monitor and 28 volt monitor.
send ack/nack	Sends an ack or nack byte through the serial port; used only for diagnostic purposes.
send cmd aa bb cc...	Sends the specified numbers aa, bb, cc, etc. to LS1 in the form of a command. See the following description of how to specify the numbers.
send event/bad cmd xx/ garbage/event/exception	Sends mock commands through the serial port; used solely for diagnostic purposes.
set current = xx	Sets the current limit on the HP6633A to xx amps (xx is a floating point number).
set monitor = continuous/delta	If the monitor is set in delta mode, the 28 volt monitor and event monitor will only be output to the tatufile.mon file when one or the other changes. In the continuous mode, they are output every second, which can create a huge file very quickly. Default is delta mode.
set parity reset = on/off	Tells the input manager software what to do when a byte from the serial register is encountered with a parity error. Normally ON.
set resend limit = xx	Sets the limit on re-transmission of commands to LS1 (in the event of nacks or time-outs) to xx.
set screen filter	This command pops up a window in which the "filter" to the serial window may be set; the filter allows or disallows the echoing of specified types of received LS1 messages to the serial window. This command was never used.
set screen filter = <i>filename</i>	Saves the current serial window filter to the specified filename.
set timemode = normal/hex	Changes the format in which the system time is displayed in the monitor and log windows between HH:MM:SS and DDDD:III TAOS/STEP 0 system time formats. HH:MM:SS format is default.
set timeout = xx	Sets the time allowed for the receiving of an ack or nack from LS1 to xx (floating point) seconds.

set voltage = xx	Sets the voltage output of the HP6633A power supply to xx (floating point) volts.
show counter/counters	Prints the transmit and receive counters (in software) to the log window. [Diagnostic]
show hp6633a	Reads the state from the HP6633A power supply and prints it to the log window.
show input [flags]	Prints the flags associated with the input manager to the log window. [Diagnostic]
show main [flags]	Prints the main control flags in the TATU software to the log window. [Diagnostic]
show reg/registers	Reads the serial port registers and prints them to the log window. [Diagnostic]
view msg	Pops up a window and allows the user to scroll through the message file collected so far. This operates very slowly and required extensive disk access; it was never used.
# text	This defines a comment statement in a command file. It may be placed on a line by itself or after a command on the same line. Blank lines may also be placed in the command file.

### ***Description of command arguments:***

send cmd xx (i)yy (f)zz.zzz (d)tt.ttt

Any number xx is interpreted as a hexadecimal number; it must be between 00 and FF (case not important). Any number yy following "(i)" is interpreted as a four byte integer. Any number zz.zzz following "(f)" is interpreted as a four byte floating point, and any number tt.ttt following "(d)" is interpreted as an eight byte double precision floating point. Any sequence of numbers may be placed in the command (subject to the LS1 command definitions). The bytes from the interpreted numbers will be placed into a LS1 command message in the order of entry and scheduled for output to LS1. Zeroes will be inserted to pad the length of the command message up to the normal 11 data bytes.

### ***Function Keys***

F1: Pause/Unpause the execution of a command file (toggle).

F2: Stop or abort execution of the command file.

F3: Prematurely abort delay command (as in a command file) and resume execution.

F4: Print listing of primary system flags to log window (diagnostic).

F10: Kill timemark (1 Hz clock and coarse time messages) for emergency and diagnostic purposes usually.

UpArrow: TATU remembers the last ten commands, and the up arrow recalls the previous command.

DownArrow: Recall the next command from memory.

### ***Command Files***

TATU can execute stored command files. All commands entered from the keyboard plus a few others may be used in a command file along with comments. At start up, TATU looks for a boot file called "tatuboot.cmd" and executes whatever commands it finds there. Also, a file called "sideinfo.cfg" contains information about what redundancy is/was current and in what state is was left (on, off, etc.).

### ***Outputs***

TATU creates three output files: tatufile.msg which contains all LS1 messages collected, tatufile.log which contains a copy of all logged information messages and errors, and tatufile.mon which contains a list of the 28V monitor, the event monitor, and the heartbeat monitor. In addition, an optional fourth file named tatufile.byf may be created which contains all bytes received by the serial port without parsing or interpretation applied.

### A3.3 PLOT Utility

The PLOT.EXE program is used to display and browse through a flat (ASCII) file of LS1 messages, display detailed blow-ups of messages, plot raw data, and create selected output files of data. It is written in Borland Turbo C and compiled in Borland's C++ V5.0 environment. The source and executable codes for PLOT are not classified; however, the input files, output files and information dumped to the printer may contain classified information.

#### ***Inputs***

The input to PLOT is a flat file containing LS1 messages. The query for the file name offers "tatufile.msg" as a default, but any name may be entered. The file may contain incomplete messages and PLOT will attempt to make some minimal sense out of them.

#### ***Outputs***

There are no automatic output files created by PLOT. Raw event or calibration data may be dumped to files in binary or ASCII format and with or without time tags. Displays of messages or plots of raw data may be dumped to the printer with the keyboard *Print Screen* key. (Note: To dump plots to the printer, a suitable printer driver such as EGALASER.COM must be loaded first.) A listing of part or all of the messages displayed in the browse window may also be dumped to the printer.

#### ***Program Execution***

Run PLOT to initiate execution of the program. An optional file name may be entered as an argument after PLOT (e.g. "plot myfile.msg<cr>"). If the file cannot be found in the current directory, PLOT will append ".msg" to the file name and attempt to open that file.

At execution, a small blue window appears in the middle of the screen; this is the command window. If no file name was supplied on the command line you will be prompted for the name of one; pressing the carriage return uses the default name "tatufile.msg". (PLOT will attempt to append ".msg" to the name if need be.)

The specified file is opened and a two-step parsing process begins. First, the program searches the file to find how many messages and fragments of messages are in the file, after which a buffer is allocated in memory to hold vital information about those messages. Secondly, the vital information on all the messages is read and placed in the buffer. If the buffer allocation fails, it means that there are too many messages in this file for the PLOT program to handle. In this case, the SPLIT utility may be used to split the message file into pieces that PLOT can handle.

After the two-step parsing process ends, the blue command window disappears and the browse window appears. Each yellow line represents one LS1 message as well as the high-lighted, red

line which is the selected message. You may move the high-lighted line and/or move around in the file with the following keys:

Desired Action	Key(s) Used
Line Up/Line Down	Up Arrow/Down Arrow
Page Up/Page Down	PgUp/PgDn
Top/Bottom of Current Page	Home/End
Top of File	Control Home or Control PgUp
End of File	Control End or Control PgDn

In the browse window, the type of message, time tag, sequence number, place in the file as a percentage, and the segment number are displayed.

#### ***Blowing Up a Message - Using F1***

To interpret and display the contents of a *non-segmented* message (e.g., a command echo, exception message, or event message) while in the browse window, select that message and press F1. If F1 is used on a segmented message (see below), it will cause the message to be displayed in a raw, hexadecimal format.

#### ***Using F2***

To display the collective data from a series of segmented messages (i.e., raw data messages, system configurations, and state-of-health) select the first segment and press F2.

#### ***State-of-Health and System Configuration Messages***

After selecting the first segment and pressing F2, the data from all three or four segments will be collected. An error message will be generated if a segment is missing. The state-of-health or system configuration items will be displayed a page at a time; press any key to step to the next page. The pages will wrap around. Press Esc to exit to the browse window.

#### ***Raw Data Messages***

Select the first available message of a series of segmented messages even if there happen to be missing segments before the first segment. All the available messages will be read with zeroes filling in for data from missing segments. (If some segments are missing, a notice will appear on the screen.) Once the segments are read, a black window will appear with a menu:

Options:

(g)raph data

display (s)tatistics  
generate (b)inary file of raw data  
generate binary file with (t)ime headers  
generate (a)scii file of raw data  
(d)isplay raw data  
(p)rint raw data to lpt1:  
(Esc)  
(Esc)

The graphing option simply graphs the available data for the arrays and scans that triggered; the plots may be printed by pressing the keyboard's *Print Screen* key as long as the proper printer driver has been loaded as mentioned earlier. The statistics option calculates and displays means, standard deviations, minimums, maximums, and spreads in addition to showing the arrays and scans that triggered, the number of pixels found in each array and scan, and the number of message segments found. The binary and ASCII file options take the available raw data and writes it to a file name specified by the operator. The display and print raw data do just as they indicate. The *Esc* key returns you to the browse window.

### ***Finding a Message Type***

PLOT can search the file for the next instance of a specified type of message. While in the blue *command window*, type "find <command type>". A list of possible variations follows. The words in italics are optional and are not necessary for command recognition.

find event  
find raw event *data*  
find raw image *data*  
find raw cal *data*  
find triggered *background data*  
find periodic *background data*  
find image *cal*  
find system *configuration*  
find forced system *configuration*  
find forced soh  
find boot *soh*  
find full *soh*

find interval *soh*

find *next*

When a find command is executed, the blue command window will be replaced by the browse window, and the browse window will shift to the next message of the specified type starting from the currently selected message. If the browse window and the current message do not move, then no more messages of that type are to be found. Once a find command has been executed, the next message of the specified type may be found by pressing F6 while in the browse window or by typing "find next" or just "find" in the command window.

### ***Printing the Browse Window***

It is possible to print a listing of all or part of the browse window. First, a block of messages must be marked in the browse window. The start of a block is marked by selecting that message and pressing F3. The end of a block is marked with F4. (A marked block will change color.) To print the selected block, press F5. To deselect the block (and return the color to normal), simply select the *block start* or the *block end* so that the end of the block is before the start of the block.

### ***Find Sequence Anomalies***

At times it is desirable to be able to find places in the file where messages have been lost or dropped out. While you can simply look for missing numbers in the sequence number column of the browse window, a much simpler way is to press F7. The next place with missing sequence numbers (starting from the current message) will be located, the browse window will shift to that location in the file, and the message just prior to the missing message(s) will be selected (made red). If no more missing sequence numbers are to be found, the window and the selected message will not shift.

### ***Toggling between the Command and Browse Windows***

To toggle between the command and browse windows in either direction, press the *Esc* key.

### ***Errors***

While in the command window, an unrecognizable command will elicit one of several, expressive error messages along with a BEEP. Other conditions can also elicit an error message. Typically, error message will be displayed on-screen, and pressing any key will dispel the message and return you to the browse window with no harm done.

### ***Exiting PLOT***

To exit PLOT, Esc to the command window and type "quit<CR>".

### ***Description of PLOT Software Routines:***

#### **V\_MAIN.C**

This main module controls the overall function of PLOT and interprets and executes the user's commands.

#### **V\_SOH.C**

The state of health (SOH) module collects the four messages in a state-of-health and displays them. The selected line in the browse window must be placed on the first segment of the four segments.

#### **V\_CONFIG.C**

The configuration module displays the information in a system configuration. The first segment must likewise be selected.

#### **V\_RAWDAT.C**

This module collects and displays the 26 or 51 segments of raw event data, calibration data, periodic background data, or triggered background data.

#### **V\_EXPLOD.C**

Single segment messages (e.g. event, exceptions, command echoes, a single segment of a raw data) are "blown up" or displayed on the screen. The data in a segmented message is shown in raw, hexadecimal form.

#### **V\_WINDOW.C**

All windowing functions are contained in this module.

## **A3.4 Flight Data Processing Software**

### **A3.4.1 PARSE\_15**

The PARSE\_15 program is a utility that takes an input file of MIL-STD-1553B packets containing LS1 messages (typically the LS1.EXP file from the TAOS Data Center), sorts and collects them into BIM messages (each containing one or more LS1 messages), strips the LS1 information out and cleans up the resulting LS1 message file. Its command syntax is:

`parse_15 inputfilename [/partial] [/notimetag]`

The *inputfilename* is simply the name of the target 1553 file. The "/partial" option causes the information in incomplete BIM message (i.e. one or more 1553B packets is missing) to be output



along with contents of complete BIM messages. The LS1 messages in an incomplete BIM message will probably be mis-located compared to the rest of the messages, and one or more of the messages will be missing. A second option is "/notimetag". Normally, a six byte binary time tag is expected at the front of each 1553 packet; however, if the time tags are missing for some reason, this option allows those packets to still be correctly processed. The '/' options are not order dependent.

Executing PARSE\_15 without any arguments causes a synopsis of its command syntax to be printed to the screen.

#### A1.4.2 LOCATE

The LOCATE program takes (for the same time span) an LS1 message file (such as that from PARSE\_15), an ephemeris file (such ADT.EPH or ADT.SV1 from the TAOS Data Center/Phillips Lab), and the attitude file ADT.ATT (from the TAOS Data Center again), and combines the information to compute earth locations. The LS1 message file is scanned for events. When an event is found the ephemeris and attitude files are scanned for information occurring at the time of the event.

Several variations of this program have been created to take advantage of different variations of information available from the TAOS Data Center and the CSF (Contractor Support Facility). Since several variations exist, only the following brief description will be provided.

The event vector reported by LS1 is rotated from LS1's coordinate system to the satellite's body frame by a fixed transformation. From the body frame, the vector is rotated (using either roll/pitch/yaw or quaternions) to the satellite's orbital frame which is defined by the earth normal vector (z), the velocity vector (x) and the opposite-rotation axis (y). (Since the loss of the IMU in June, 1994 the roll, pitch, and yaw have been stabilized at 0, 0, and -90 degrees respectively with an observed  $\pm 5^\circ$  of error.) The vector is then rotated from the orbital frame into the Earth Centered Inertial (ECI, or celestial) coordinate system. Once in ECI coordinates the vector is projected to earth and the latitude and longitude on the earth's surface are found in the ECI system. Based upon the event time tag provide by LS1, the Greenwich Apparent Sidereal Time (GAST) is calculated, and the ECI longitude is adjusted by the GAST to become the Earth Centered Earth Fixed (ECEF) or actual ground location longitude. The units for the computed errors are 'degrees pointing error at the sensor' and 'kilometers on the ground'.

#### A3.5 Declassification Software

Several programs were written in Borland C to assist in the sanitization and declassification of LS1 message files. Typically, triggered background data and periodic background data are the subjects of the declassification process. The process itself ensures that no event data or raw event data is written to an output file.

### A3.5.1 SANITIZE

The SANITIZE program reads from an LS1 message input file, throws away all event information, and writes the remaining messages to output files. Each output file is usually no more than 3850 messages (the maximum that PLOT can handle) and no larger than 1 megabyte (the maximum that the TRCOPY file removal utility can handle). The command syntax is:

```
sanitize inputfilename [outputfilename] [/all] [/tri] [/per] [/fsxxxx] [/myyyy]
```

The *inputfilename* is the name of the LS1 message file. ".msg" will be appended if needed. An optional output file name may be supplied; if not supplied, the root name of the input file will be used. Depending on the /all, /tri or /per option selected, the suffix of the output files will be ".axx", ".txx" or ".pxx" respectively, where xx ranges from 01 to 99.

Options are:

- /all        Output all messages *except* event messages and event data.  
            This option is the default and need not be specified. ".axx" files are produced.
- /tri        Outputs *only* triggered background messages. ".txx" files are produced.
- /per        Outputs *only* periodic background messages. ".pxx" files are produced.
- /fsxxxx    "xxxx" is the maximum output file size allowed in units of  
            kilobytes. Default value is 1024 (i.e. 1 megabyte), and the minimum  
            value is 64.
- /myyyy    "yyyy" is the maximum output file size allowed in terms of the number  
            of messages. Default is 3850, and the minimum is 51.

/tri, /all and /per are mutually exclusive -- only one of these may be specified. The other options are not order dependent. Note that the sequence of messages in the output files may be slightly reordered compared to the order in the input file. Executing sanitize without any arguments causes a synopsis of its command syntax to be printed to the screen.

### A3.5.2 CHECK

Syntax: *check inputfilename*

This program is meant to check the output files of SANITIZE. It performs two passes through the file using two different parsing techniques to identify the messages in the file. Each pass must detect the same number of LS1 messages, and no event messages or raw event data messages, or other message fragments may be detected in order for the file to be declared unclassified. If declared unclassified, the file may be removed from the LS1 ground station with the TRCOPY utility.

### A3.5.3 JOIN

The SANITIZE utility chops a large LS1 message file into smaller, 1MByte pieces for removal from the classified LS1 ground station. JOIN was created to concatenate those pieces into a single contiguous file. The syntax is:

*join outputfilename inputfilename1 inputfilename2 [inputfilename3...]*

At least two input file names must be specified. If no arguments are specified, a synopsis of the JOIN command syntax is printed to the screen.

## A3.6 Other Utility Programs

### A3.6.1 TRCOPY

Trusted Copy (TRCOPY) is a commercial software package that allows an unclassified 5¼" disk to be used to transfer data into a classified PC system without causing the disk itself to become classified. This is done by making sure that all sectors of the disk contain data leaving no available sectors that might inadvertently receive classified data while in the PC disk drive.

### A3.6.2 SPLIT

A message file too long for PLOT to handle may be divided into manageable pieces by using SPLIT. SPLIT will parse out blocks of messages and writes the blocks to separate files, each of which may then be browsed with PLOT. The command syntax is:

*split inputfilename [outputfilename] [/mxxxx]*

If no output file name is specified the root of the input file name will be used with ".001", ".002" and so forth appended. The /mxxxx option specifies the maximum number of LS1 messages allowed per output file. xxxx defaults to 3850 (the maximum that PLOT can handle) and the

minimum value is 100. Executing split without arguments causes a synopsis of its command syntax to be printed to the screen.

### A3.6.3 SEARCH

Syntax: search *inputfilename1* [*inputfilename2...*] [/outputfilename]

The SEARCH program peruses the specified LS1 message input files for certain state-of-health and system configuration data, tabulates statistics, and prints them to the screen. They are also written to an optional output file specified by /outputfilename. Executing search without arguments will cause a description of its syntax to be printed to the screen.

### A3.6.4 STRIPOUT

Various LS1 message files contain large amount of periodic and triggered background data that has been declassified. However, a method was needed to strip the background data from the message file and convert it into a form more easily readable by other programs such as Excel™ and MathCad™. STRIPOUT does just that using a Borland C routine. The syntax of the command is:

stripout *inputfilename* [*outputfilename*] [/per] [/tri] [/bin] [/dec] [/hex] [/rms] [/nnn] [/inc] [/nt]

*inputfilename* is expected to be an LS1 message file. The raw data output is normally placed into six files: three for the periodic data from i, j and k arrays, and three for the triggered data from the i, j and k arrays. *outputfilename* will have ".pi", ".pj", ".pk", ".ti", ".tj" and ".tx" appended to it to form the periodic and triggered background files. If no output file name is specified, the root name of the input file name will be used.

Available options are:

- /per      Create only periodic background data files.
- /tri      Create only triggered background data files. Both are created by default.
- /bin      Output the data in PC compatible binary format. This is the default and need not be specified.
- /dec      Output the data in decimal text format.
- /hex      Output the data in hexadecimal format.
- /rms      Create auxiliary output text files named rms\_peak.px and rms\_peak.tx containing the rms and peak values of all scans as well as the rms of the rms values and the average and peak of the peak values.

- /nnn**      nnn is a numerical threshold value that specifies that only background scans with one or more pixels equal to or above the nnn value may be output. Default value is 000 (i.e. all scans are output). nnn may range from 000 to 4096.
- /inc**      Normally, only the scans from backgrounds with no missing data (i.e. no missing LS1 message segments) are output. If the /inc option is included, then any complete scans available from backgrounds with missing segments will also be output.
- /nt**        Normally, each scan of data is preceded by an eight byte time/array/scan number tag in a YMDHMSAS format. The /nt (no time) option precludes that tag from being placed with the raw data in the output files.

Executing stripout without any arguments will cause a synopsis of the above command descriptions to be printed to the screen.

### A3.7      Main TATU Software Modules

The TATU software was written in Borland's TurboC V2.0™ and compiled using the command line make, compiler and linker. A brief description of the code modules follows.

#### **MAIN.C**

The primary purpose of the main module is to control the various functions of TATU. A revolving loop tests combinations of various flags to see what tasks need to be done. These flags are set by hardware interrupt handlers such as the serial port or by other software routines that have performed some function and created an output that needs to be handled by another module. Various tasks include: handling keystrokes from the keyboard, interpreting operator commands, handling and disposing of bytes placed in a buffer by the serial interrupt handler, scheduling outgoing serial commands to LS1 with the serial interrupt, interpreting and displaying incoming messages from LS1, servicing the 1 Hz interrupt by creating and scheduling a time message to LS1, updating the monitor window periodically, and calling the exception handler.

#### **KBD\_MGR.C**

The keyboard manager buffers keystrokes and passes command lines to the command interpreter (CMDINTRP.C) by setting a flag.

### ***CMDINTRP.C***

This module parses and interprets command lines and calls the necessary procedures to perform the indicated function.

### ***CMDFILES.C***

The module handles the interpretation and execution of command files by passing command lines to command interpreter.

### ***CREATMSG.C***

Command messages to LS1 are created by this module and flagged to be scheduled by the output manager (OUT\_MGR.C).

### ***OUT\_MGR.C***

The output manager receives pointers to messages and schedules those messages with the serial interrupt handler for output to LS1.

### ***IN\_MGR.C***

The input manager takes bytes from the serial byte buffer and parses messages from the byte stream. The messages are passed to the input queue.

### ***IN\_QUEUE.C***

The input queue buffers LS1 messages from the input manager for later processing and disposal.

### ***MSGFILTR.C***

The message filter pulls LS1 messages from the input queue and selectively displays them on the screen based upon an operator selectable filter.

### ***MSGFILES.C***

The message files procedure allows a file of LS1 messages to be opened and browsed while TATU is also communicating with LS1. This function is very slow and was never used.

### ***SCREEN.C***

All the screen windowing procedures reside in this module. There are the command, serial, log, monitor, and pulldown windows.

### ***EXCEPT.C***

This module contains all the exception messages. When the exception flag is set this procedure prints the appropriate error message to the log window.

### ***SERIAL.C***

Procedures with which to set serial port registers are in this module along with an inline assembly code serial port interrupt handler. Because of the inline assembly code, the compiler in Borland's Integrated Development Environment (IDE) could not be used. Instead, the command line compiler was called by the make file to compile the modules via an intermediate assembly code step.

### ***TIMEMARK.C***

The 1 Hz clock interrupt is handled by this procedure. A coarse time message is assembled and flagged to be scheduled by the output manager. An inline assembly code interrupt handler resides here.

### ***PROTOBRD.C***

All I/O functions between the PC and the interface board are handled by procedures in this module.

### ***NUMBERS.C***

All numerical input from the keyboard is interpreted by procedures in this module. These procedures are more powerful than the standard "atoi", "atof" and "atod" C language functions.

### ***IEEE\_488.C***

This module contain procedures to send commands to the HP6633A power supply using the IOTech™ supplied software in IEEEIO.C

### ***IEEEIO.C***

This module contains IOTech™ supplied software with which to drive the IEEE-488 board.

## **A4.0 LS1 SYSTEM CHECK-OUT**

### **A4.1 Test Objectives, Methods and Constraints**

The LS-1 sensor payload provides detection and reporting of optical illumination events. The system is redundant, providing A and B modules of which only one may be on at any given time. The LS-1 system will be powered on using the Initialization Procedure and will be optimized during these tests. The following System check-out Procedure will demonstrate functions of the LS-1 sensor which include commanding, internal calibration, configuration reporting, filter modes, triggered background mode, periodic mode, and event reporting for the A module. The B module will also be powered on using the Initialization Procedure, then disabled. All testing and optimization collected on the A module are directly transferable to the B module.

A small subset of the LS-1 commands will be demonstrated and are outlined as follows:

- Configuration Command
- Calibration Command
- Enable/Disable Sensors Command (not used during check-out)
- Enable Mission Command (not used during check-out)
- Disable Mission Command (not used during check-out)
- Triggered Background Collection Command
- Periodic Background Collection Command
- Manual Threshold Command
- Pixel Mask Command (not used during check-out)
- Filter Mode Command
- Force Trigger Command
- Force Two Trigger Command
- Event Mode Command

All LS-1 report message types will be demonstrated and include:

- State-of-Health Message
- Configuration Message
- Event Message
- Raw Event Data Message
- Triggered Background Collection Message
- Periodic Background Collection Message
- Exception Message



### Constraints

The LS-1 Check-out procedure has the following constraints:

- 0° to +30° C unit temperature
- +23V to +32V bus voltage
- BIM must be running and showing no serious Health and Status errors
- LS-1 optical aperture must not be solar pointing

The LS-1 On-Orbit Check-out procedure is divided into functionally separate sequences. These sequences are as described below.

## A4.2 Event Mode Testing

**I.** The event mode requires the event detection offsets to be sent prior to an image calibration command. The raw event data rate should be set to a 1 minute interval prior to a calibration sequence. This test collects data using event processing. The event mode testing is conducted with the pre-quantization filter 'on'. Section "I" is written to be a stand alone procedure. If an Image Calibration is sent, the array offsets must be sent to configure the system for external event detection. The following sections lower the threshold value to determine lower limit values.

Set threshold level = 100h command (256)	A1 00 01 00 00 00 00 00 00 00 00
Set event mode	E0 00 00 00 00 00 00 00 00 00 00

The following commands set the system to detect an image calibration.

Set raw event data message rate to 1 minute	40 04 02 3C 00 00 00 00 00 00 00
Set "i" array offset value to -0.12 mm	40 28 00 8F C2 F5 BD 00 00 00 00
Set "j" array offset value to -0.54 mm	40 29 00 71 3D 0A BF 00 00 00 00
Set "k" array offset value to -0.21 mm	40 2A 00 3D 0A 57 BE 00 00 00 00
Set max ratio of 1st to 0th order to 1	40 2B 00 00 00 80 3F 00 00 00 00
Send an image calibration command	80 01 00 00 00 00 00 00 00 00 00

The following commands set the system to detect an external event.

Set "i" array offset value to 0.338720 mm	40 28 00 B5 6C AD 3E 00 00 00 00
Set "j" array offset value to 0.052320 mm	40 29 00 7F 4D 56 3D 00 00 00 00
Set "k" array offset value to 0.120440 mm	40 2A 00 3F A9 F6 3D 00 00 00 00
Set max ratio of 1st to 0th order to 1.5	40 2B 00 00 00 C0 3F 00 00 00 00

Wait at least 60 seconds.

The time between raw event data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set raw event data message rate to 10 minutes      40 04 02 58 02 00 00 00 00 00

During this configuration changes to the threshold level may need to be made. Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

**II.** Change the threshold, by a factor of two, and set the system up to detect an image cal. This is the same test as the above, with a lower threshold value.

Set raw event data message rate to 1 minute	40 04 02 3C 00 00 00 00 00 00
Set threshold level = 80h command (128)	A1 80 00 00 00 00 00 00 00 00
Set "i" array offset value to -0.12 mm	40 28 00 8F C2 F5 BD 00 00 00 00
Set "j" array offset value to -0.54 mm	40 29 00 71 3D 0A BF 00 00 00 00
Set "k" array offset value to -0.21 mm	40 2A 00 3D 0A 57 BE 00 00 00 00
Set max ratio of 1st to 0th order to 1	40 2B 00 00 00 80 3F 00 00 00 00
Send an image calibration command	80 01 00 00 00 00 00 00 00 00

The following commands set the system to detect an external event.

Set "i" array offset value to 0.338720 mm	40 28 00 B5 6C AD 3E 00 00 00 00
Set "j" array offset value to 0.052320 mm	40 29 00 7F 4D 56 3D 00 00 00 00
Set "k" array offset value to 0.120440 mm	40 2A 00 3F A9 F6 3D 00 00 00 00
Set max ratio of 1st to 0th order to 1.5	40 2B 00 00 00 C0 3F 00 00 00 00

Wait at least 60 seconds.

The time between raw event data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set raw event data message rate to 10 minutes      40 04 02 58 02 00 00 00 00 00

During this configuration, changes to the threshold level may need to be made. Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

**III.** Change the threshold, by a factor of two, and set the system to detect an image cal. This is the same test as the above, with a lower threshold value.

Set raw event data message rate to 1 minute	40 04 02 3C 00 00 00 00 00 00 00
Set threshold level = 40h command (64)	A1 40 00 00 00 00 00 00 00 00 00
Set "i" array offset value to -0.12 mm	40 28 00 8F C2 F5 BD 00 00 00 00
Set "j" array offset value to -0.54 mm	40 29 00 71 3D 0A BF 00 00 00 00
Set "k" array offset value to -0.21 mm	40 2A 00 3D 0A 57 BE 00 00 00 00
Set max ratio of 1st to 0th order to 1	40 2B 00 00 00 80 3F 00 00 00 00
Send an image calibration command	80 01 00 00 00 00 00 00 00 00 00

The following commands set the system to detect an external event.

Set "i" array offset value to 0.338720 mm	40 28 00 B5 6C AD 3E 00 00 00 00
Set "j" array offset value to 0.052320 mm	40 29 00 7F 4D 56 3D 00 00 00 00
Set "k" array offset value to 0.120440 mm	40 2A 00 3F A9 F6 3D 00 00 00 00
Set max ratio of 1st to 0th order to 1.5	40 2B 00 00 00 C0 3F 00 00 00 00

Wait at least 60 seconds.

The time between raw event data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set raw event data message rate to 10 minutes	40 04 02 58 02 00 00 00 00 00 00
---	----------------------------------

During this configuration, changes to the threshold level may need to be made. Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

**IV.** These commands test the auto-threshold mode in the event mode. This is a stand-alone procedure.

Set auto-threshold mode	A0 00 00 00 00 00 00 00 00 00 00
Set raw event data message rate 1 minute	40 04 02 3C 00 00 00 00 00 00 00
Set "i" array offset value to -0.12 mm	40 28 00 8F C2 F5 BD 00 00 00 00
Set "j" array offset value to -0.54 mm	40 29 00 71 3D 0A BF 00 00 00 00
Set "k" array offset value to -0.21 mm	40 2A 00 3D 0A 57 BE 00 00 00 00
Set max ratio of 1st to 0th order to 1	40 2B 00 00 00 80 3F 00 00 00 00
Send an image calibration command	80 01 00 00 00 00 00 00 00 00 00
Set "i" array offset value to 0.338720 mm	40 28 00 B5 6C AD 3E 00 00 00 00
Set "j" array offset value to 0.052320 mm	40 29 00 7F 4D 56 3D 00 00 00 00
Set "k" array offset value to 0.120440 mm	40 2A 00 3F A9 F6 3D 00 00 00 00
Set max ratio of 1st to 0th order to 1.5	40 2B 00 00 00 C0 3F 00 00 00 00

Wait at least 60 seconds.

The time between raw event data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set raw event data message rate 10 minutes                      40 04 02 58 02 00 00 00 00 00 00

Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

From the data collected, a decision can be made as to the optimum threshold with which to leave the system or whether the manual or auto-threshold mode should be used. The final commands will place the sensor in the event mode with the pre-quantization filter 'on'. A dry-run rehearsal for a LAZAP will be conducted during the optimization period to become familiar with the data path and communication with the other sites. All portions of the LAZAP will be tested, including tracking with the most current ephemeris and attempts to stimulate the sensor. (This early test exercise should prove useful even though the official tests from LAZAP will be conducted during terminator conditions.)

Send threshold maximum level command	A1 FF 0F 00 00 00 00 00 00 00 00
Set pre-quantitization filter on	81 00 00 00 00 00 00 00 00 00 00
Set event mode	E0 00 00 00 00 00 00 00 00 00 00

The optimum threshold level for event detection mode will be sent at this time.

Set threshold level = optimum (xxx)                      A1 xx 00 00 00 00 00 00 00 00 00

If solar pointing is expected, the disable sensor command should be sent to assure the LS1 sensor does not generate excess amounts of data and fill the MMU.

Send disable sensors command	C7 00 00 00 00 00 00 00 00 00 00
Set raw event data message rate 1 minute	40 04 02 3C 00 00 00 00 00 00 00
Set raw event data message rate 2 minutes	40 04 02 78 00 00 00 00 00 00 00
Set raw event data message rate 3 minutes	40 04 02 B4 00 00 00 00 00 00 00 00
Set raw event data message rate 4 minutes	40 04 02 F0 00 00 00 00 00 00 00 00
Set raw event data message rate 5 minutes	40 04 02 2C 01 00 00 00 00 00 00
Set raw event data message rate 10 minutes	40 04 02 58 02 00 00 00 00 00 00

### A4.3 Periodic Background Mode Testing

**I.** Periodic Background Mode allows background to be collected on a periodic basis. In this mode the system collects data at the commanded periodic rate and sends two frames of data to memory. No event or triggered data processing occurs in this mode. Each section is written to be a stand-alone procedure.

Set periodic background rate to 1 minute	40 04 04 3C 00 00 00 00 00 00 00
Set threshold level = 100h command (256)	A1 00 01 00 00 00 00 00 00 00 00
Set periodic background collection mode	E2 00 00 00 00 00 00 00 00 00 00

Wait at least 60 seconds.

The time between periodic background data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set periodic background rate to 10 minutes	40 04 04 58 02 00 00 00 00 00 00
--	----------------------------------

Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

**II.** The following test checks the periodic background mode. It has the high-pass filter removed and allows background to be collected on a periodic basis.

Send threshold maximum level command	A1 FF 0F 00 00 00 00 00 00 00 00
Set periodic background rate to 1 minute	40 04 04 3C 00 00 00 00 00 00 00
Set prequantitization filter off	81 01 00 00 00 00 00 00 00 00 00
Set periodic background collection mode	E2 00 00 00 00 00 00 00 00 00 00

Wait at least 60 seconds.

The time between periodic background data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set periodic background rate to 10 minutes	40 04 04 58 02 00 00 00 00 00 00
--	----------------------------------

Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off line using the TATU test unit located at GTE.

Set pre-quantization filter on	81 00 00 00 00 00 00 00 00 00 00
--------------------------------	----------------------------------

From the data collected, a decision can be made for the starting threshold value in the triggered background mode. The final commands will place the sensor in the event mode with the pre-quantization filter on.

Set periodic background rate to 1 minute	40 04 04 3C 00 00 00 00 00 00 00
Set periodic background rate to 2 minutes	40 04 04 78 00 00 00 00 00 00 00
Set periodic background rate to 3 minutes	40 04 04 B4 00 00 00 00 00 00 00
Set periodic background rate to 4 minutes	40 04 04 F0 00 00 00 00 00 00 00
Set periodic background rate to 5 minutes	40 04 04 2C 01 00 00 00 00 00 00
Set periodic background rate to 10 minutes	40 04 04 58 02 00 00 00 00 00 00

## A4.4 Triggered Background Mode Testing

**I.** The Triggered Background Mode does not require the event detection array offsets to be sent prior to an image calibration command. The triggered background rate should be set to a 1 minute interval prior to a calibration sequence. This test collects data on the background and does not allow event processing. The first part of this test should be conducted with the pre-quantization filter 'on'. Section "I" is written to be a stand-alone procedure. Subsequent sections lower the threshold value to determine lower limit values.

Send threshold maximum level command	A1 FF 0F 00 00 00 00 00 00 00
Set triggered background rate to 1 minute	40 04 03 3C 00 00 00 00 00 00
Set pre-quantization filter 'on'	81 00 00 00 00 00 00 00 00 00
Set threshold level = 100h command (256)	A1 00 01 00 00 00 00 00 00 00
Set triggered background collection mode	E1 00 00 00 00 00 00 00 00 00

Wait at least 60 seconds

Send an image calibration command	80 01 00 00 00 00 00 00 00 00
-----------------------------------	-------------------------------

Wait at least 60 seconds

The time between triggered background data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Send a force trigger command	83 00 00 00 00 00 00 00 00 00
Set triggered background rate to 10 minutes	40 04 03 58 02 00 00 00 00 00

During this configuration, changes to the threshold level may need to be made. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

**II.** If background data triggers are low, change the configuration by sending the following commands.

Set triggered background rate to 1 minute	40 04 03 3C 00 00 00 00 00 00
---	-------------------------------

Wait at least 60 seconds.

Set threshold level = 80h command (128)	A1 80 00 00 00 00 00 00 00 00
Send an image calibration command	80 01 00 00 00 00 00 00 00 00





Wait at least 60 seconds.

The time between triggered background data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set triggered background rate to 10 minutes      40 04 03 58 02 00 00 00 00 00 00

During this configuration, changes to the threshold level may need to be made. Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

**V.** If background data triggers are low, change the configuration by sending the following commands.

Set triggered background rate to 1 minute      40 04 03 3C 00 00 00 00 00 00 00

Wait at least 60 seconds.

Set threshold level = 80h command (128)      A1 80 00 00 00 00 00 00 00 00 00  
Send a force two triggers command      87 00 00 00 00 00 00 00 00 00 00

Wait at least 60 seconds.

The time between triggered background data collects can be set in one second intervals and may be changed depending on the amount of data desired to be collected. Alternate commands for the rate are shown below. Convert the decimal time in seconds to hexadecimal.

Set triggered background rate to 10 minutes      40 04 03 58 02 00 00 00 00 00 00

During this configuration, changes to the threshold level may need to be made. Monitor the system and analyze the data. Some data may be monitored on the ETW screens, but data analysis must be done off-line using the TATU test unit located at GTE.

Send threshold maximum level command      A1 FF 0F 00 00 00 00 00 00 00 00  
Set pre-quantization filter 'on'      81 00 00 00 00 00 00 00 00 00 00  
Set threshold level = 100h command (256)      A1 00 01 00 00 00 00 00 00 00 00  
Set event mode      E0 00 00 00 00 00 00 00 00 00 00

**VI.** If solar pointing is expected, the disable sensor command should be sent to assure the LS1 sensor does not generate excess amounts of data which could fill the MMU.

Send disable sensors command

C7 00 00 00 00 00 00 00 00 00 00

These commands change the interval between triggered background data collection.

Set triggered background rate to 1 minute	40 04 03 3C 00 00 00 00 00 00 00
Set triggered background rate to 2 minutes	40 04 03 78 00 00 00 00 00 00 00
Set triggered background rate to 3 minutes	40 04 03 B4 00 00 00 00 00 00 00
Set triggered background rate to 4 minutes	40 04 03 F0 00 00 00 00 00 00 00
Set triggered background rate to 5 minutes	40 04 03 2C 01 00 00 00 00 00 00
Set triggered background rate to 10 minutes	40 04 03 58 02 00 00 00 00 00 00

## DISTRIBUTION:

- 15 Cpt. Keith Stanley  
U. S. Air Force Research Laboratory  
AFRL/VSD-I Bldg. 592  
3550 Aberdeen SE  
Kirtland AFB, NM 87117
- 20 Mr. Dale Mumford  
GTE Government Systems, MS 6236  
100 Ferguson Drive  
Mountain View, CA 94043
- 1 HQ Space & Missile Systems Center  
Attn: Col. Benjamin James/CI  
2420 Vela Way  
Suite 1467-A8  
El Segundo, CA 90245-4659
- 
- |   |         |                                 |
|---|---------|---------------------------------|
| 1 | MS 0965 | Paul W. Elder, 5711             |
| 1 | 0965    | Kurt R. Lanes, 5711             |
| 1 | 0966    | Delfin Q. Bangate, 5717         |
| 1 | 0972    | C. Andrew Boye, 5705            |
| 1 | 0972    | Jeffrey P. Kern, 5705           |
| 1 | 0972    | George J. Alder, 5335           |
| 1 | 0972    | Anthony J. Medina, 5722         |
| 1 | 0972    | Paul J. Claassen, 5722          |
| 1 | 0972    | John M. Falls, 5722             |
| 1 | 0972    | R. Rex Kay, 5722                |
| 1 | 0972    | Ted W. Welton, 5722             |
| 1 | 0980    | Philip J. Green, 5725           |
| 1 | 0980    | Stephen M. Gentry, 5725         |
| 5 | 0980    | Gary S. Phipps, 5725            |
| 1 | 9018    | Central Technical Files, 8940-2 |
| 1 | 0899    | Technical Library, 4916         |
| 2 | 0619    | For DOE/OSTI                    |