MODIS
SCIENCE DATA SUPPORT TEAM
PRESENTATION

May 29, 1992

AGENDA

Page

ACTION ITEMS:

04/24/92 [Lloyd Carpenter] Prepare the Team Leader's Software and Data Management Plan for review. (An updated draft version was provided on May 19, 1992.) STATUS: Open. Due Date: May 10, 1992.

04/24/92 [Lloyd Carpenter] Prepare the Team Leader's Science Computing Facility Plan for review. (An updated draft version was provided on May 19, 1992.) STATUS: Open. Due Date: May 10, 1992.

04/24/92 [Tom Goff] Develop a detailed schedule through to the delivery of Version 1 to the DAAC for Level-1A and -1B software design and development, identification of risk areas in Level-1A and -1B design, and prototyping of risks. (A Task List is included in the handout.) STATUS: Open. Due Date: 05/22/92

04/24/92 [J. J. Pan] Develop a detailed schedule for the Level-2 Processing Shell design and development, identification of risk areas in the Level-2 Processing Shell design and development, and prototyping of risks, through to the delivery of Version 1 to the DAAC. (Included in the handout.) STATUS: Open. Due Date: 05/22/92

04/24/92 [J. J. Pan] Develop a detailed schedule for a typical algorithm integration into the Level-2 processing shell. STATUS: Open. Due Date: 06/05/92

04/24/92 [Lloyd Carpenter & Team] Develop a staffing plan for the accomplishment of the tasks shown on the schedule. STATUS: Open. Due Date: 06/12/92

MODIS Airborne Simulator status (Liam Gumley)

Progress up to 28 May 1992

(1) Visit to Ames Research Center

I was at Ames from 05/21/92 to 05/23/92 to assist in the characterization and testing of the MAS following modifications at Daedalus. Also present for this purpose were Tom Arnold (GSFC) and Chris Moeller (U. Wisconsin-Madison).

Initial plans were to test the MAS during a checkout flight on 5/22. However at the flight readiness review on the morning of 5/22, the flight was scrubbed due to a persistent malfunction in the Valero radiometer. The test flight was delayed until 5/27. I have arranged for Ames to FedEx me a copy of the data tape, which I will examine in the Azores.

During the day on 5/22, it was decided by Chris Moeller, Tom Arnold and myself to do some ground tests on the MAS. The tests involved were
(1) viewing a target of ice-water (to check IR channels)
(2) viewing a black target with all external light sources covered (to check visible/near-IR channels).

The MAS instrument configuration was as follows:

Channel # number	Band # number	Wavelength (microns)	Bandwidth @50% response (microns)	Bits
01	-	Bit bucket	-	8
02	02	0.665	0.055	8
03	07	0.875	0.041	8
04	09	0.945	0.043	8
05	10	1.623	0.057	8
06	20	2.142	0.047	8
07	31	3.725	0.151	8
08	49	13.952	0.517	8
09	42	8.563	0.396	10
10	45	11.002	0.448	10
11	48	13.186	0.352	10
12	46	12.032	0.447	10

Data was first recorded while the MAS viewed a container of ice/water. Approximately 14000 scanlines were acquired in this mode. The ice water was then removed, and a black cloth taped over the entrance port. Approximately 5000 scan lines were acquired in this mode.

The data tape created was read on the Field Data System (FDS) the same day at Ames. However it was found that only one file could be read from the tape, containing 1323 scanlines. After some discussion, the Ames group decided that the Exabyte recorder was not functioning as expected during power-on/power-off. The recorder normally rewinds the tape as soon as it is turned on, and is then supposed to fast forward to the end of the last data write. However the recorder appeared to be either rewinding itself at the wrong time, or not fast-forwarding to the correct position after rewind. Pat Grant decided to recommend that the MAS recorder not be cycled on/off during flight as was previously planned.

The 1323 scanlines of data that were read from tape were examined on the FDS. Imagery showed that the new 13 micron channels (ch 8,11) were considerably more noisy than any of the other channels. This was confirmed by examination of the black body counts for these channels. Calibration coefficients were computed for each of the IR channels, and appeared to be reasonable (with the exception of the noisy data in channels 8,11).

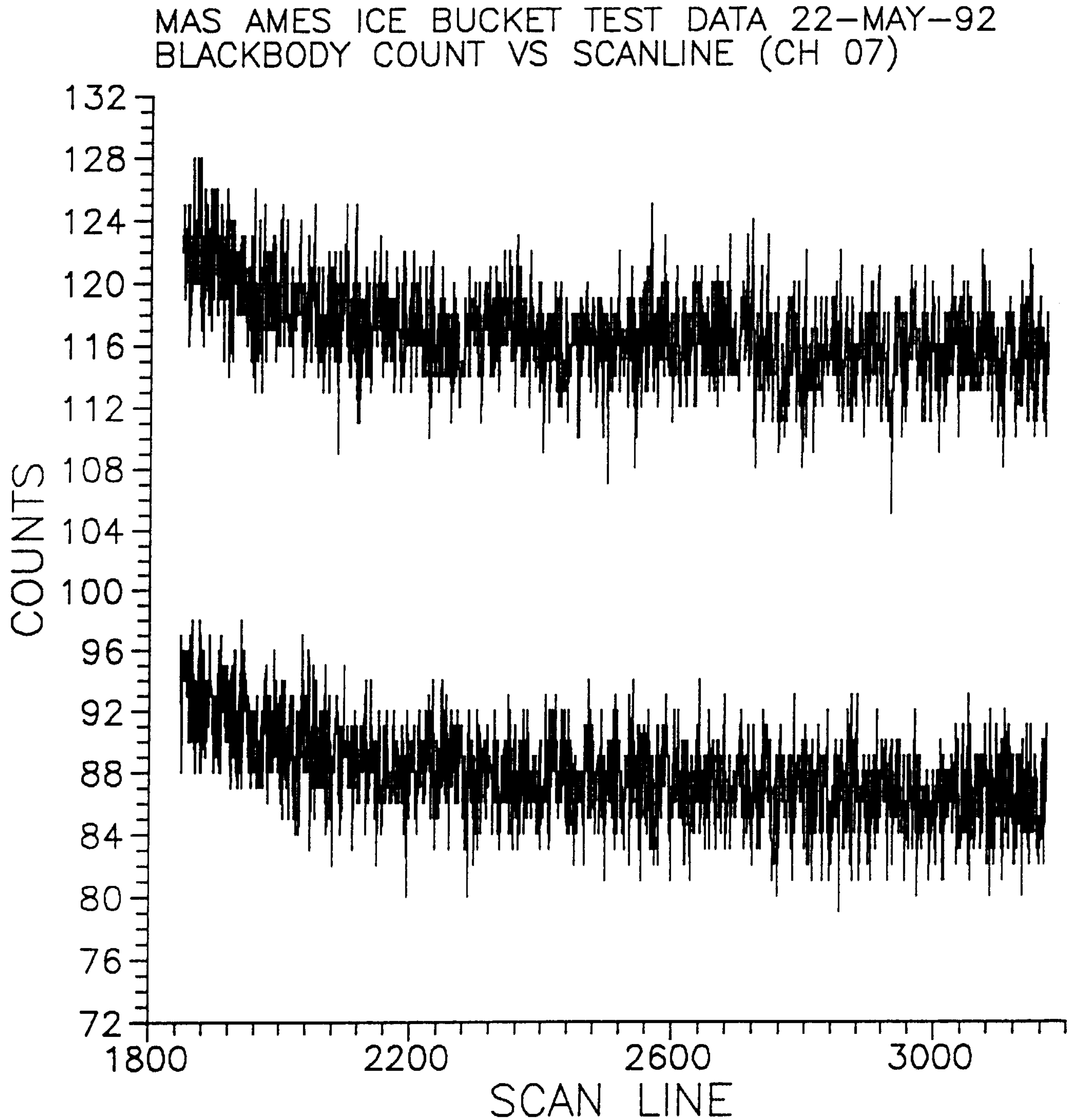
(2) MAS deployment to Terceira, Azores for ASTEX

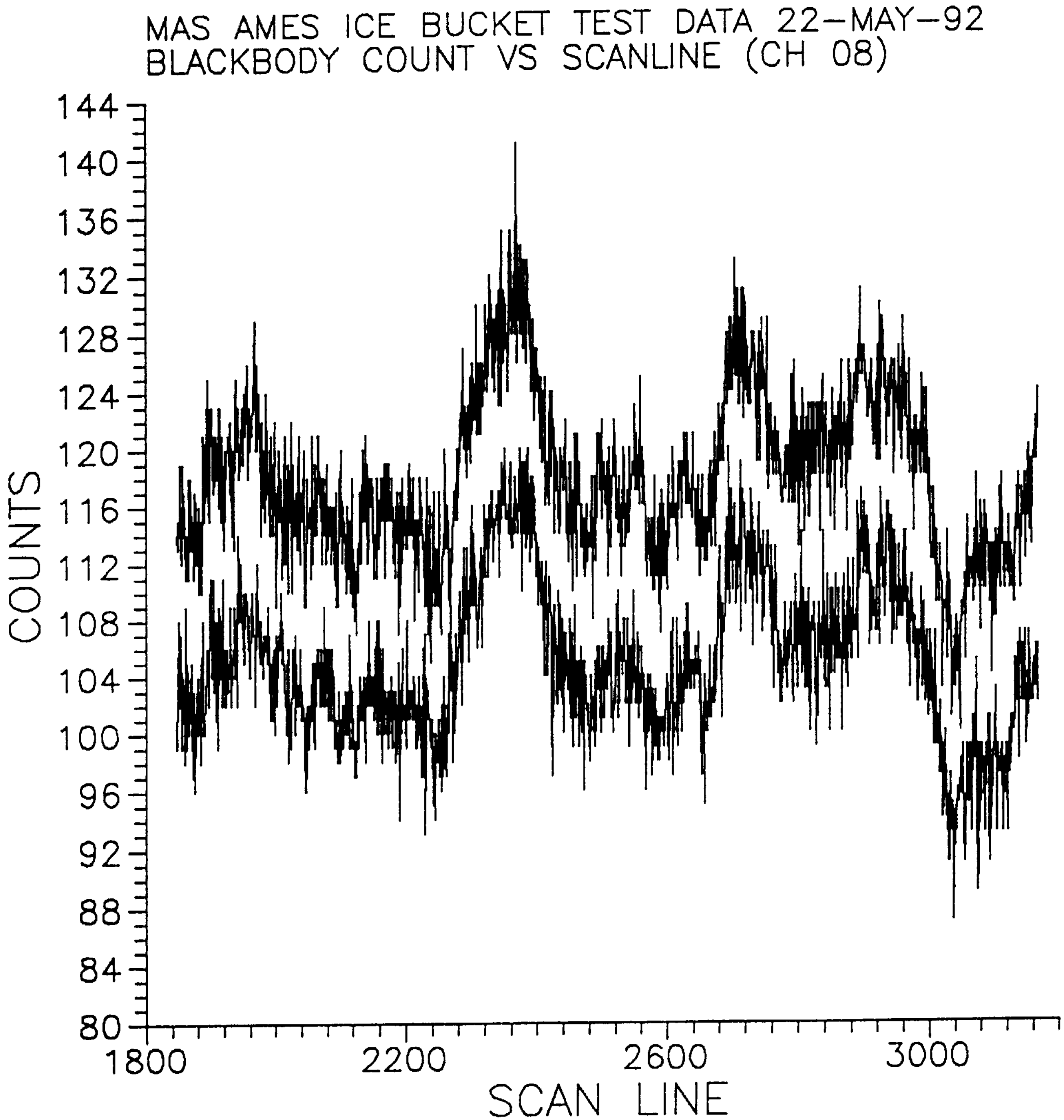
The ER-2 leaves for the Azores on 5/29 (via Wallops Island). It is planned to have the MAS acquire data on both legs of the ferry flight to Terceira.

The FDS was shipped to Terceira by RDC on 05/27, and should arrive on 06/01. The first task will be to look at the MAS test flight data tape and ferry flight tape.

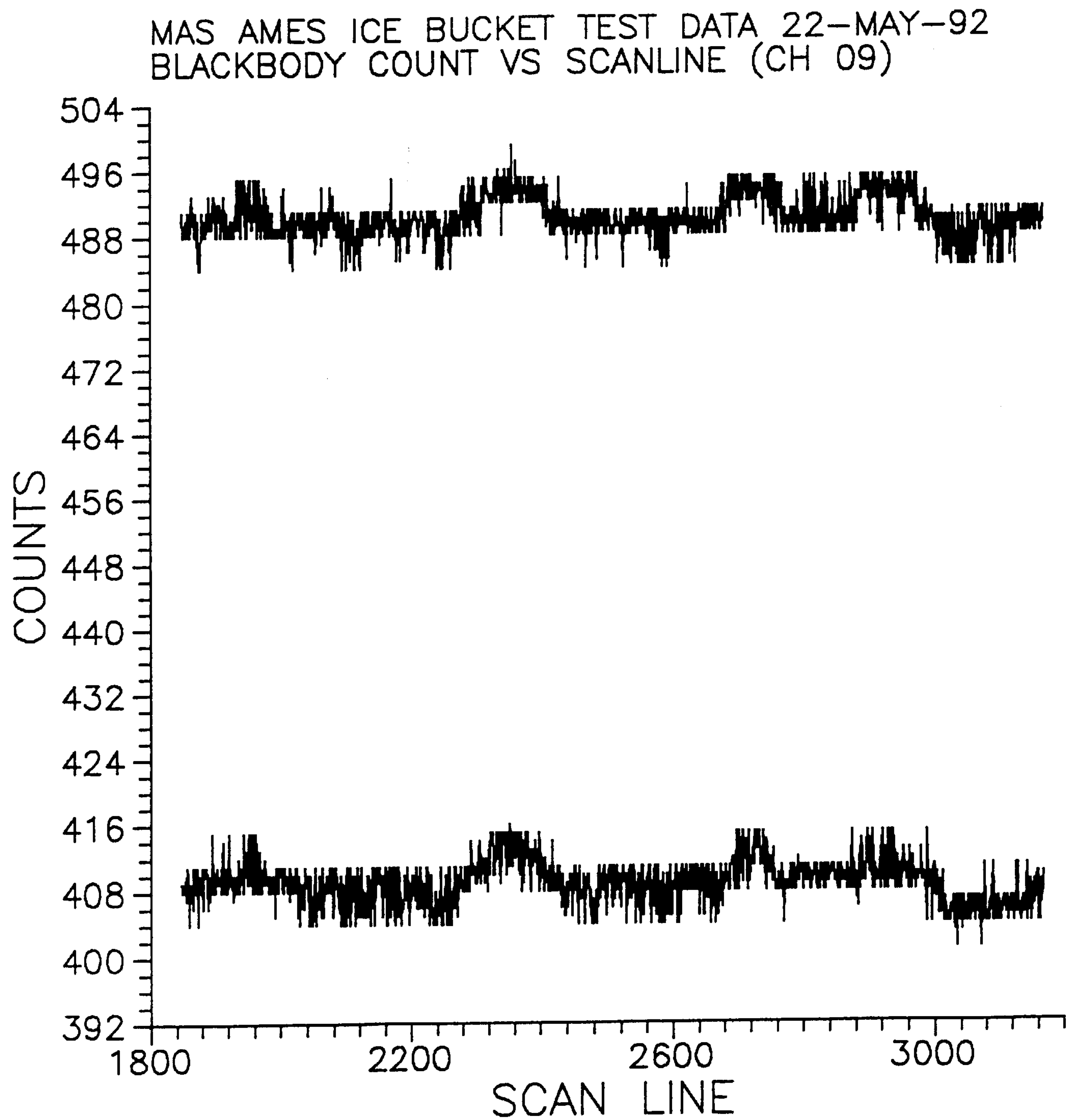
I will be leaving for the Azores on 05/29, and returning on 06/12.

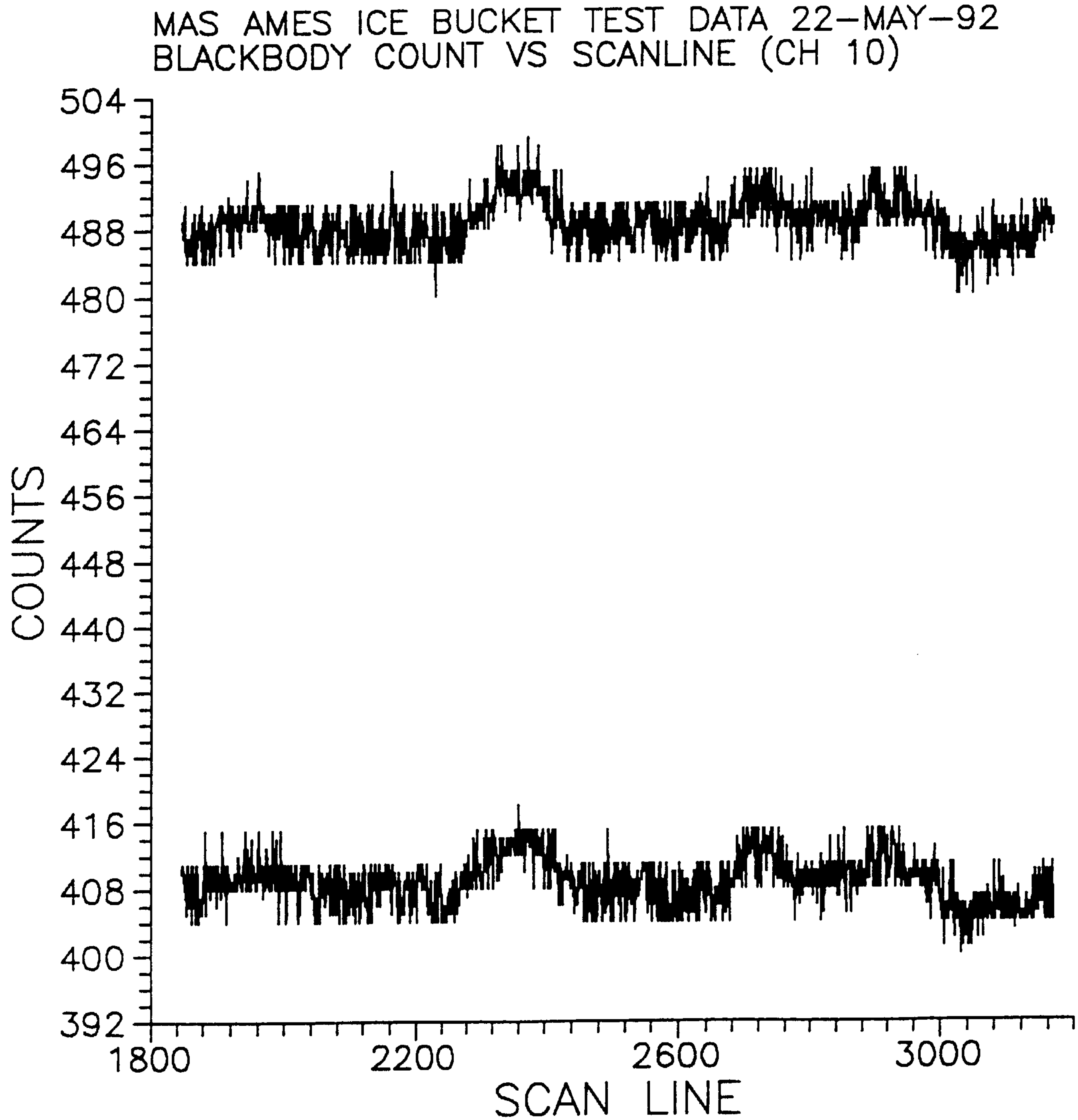
```
Enter channel, record : 12,50
Scanline HHMMSSSS   BB1T   BB2T Gain  Slope          Intercept         Lat      Lon
    1896 13020110   2.29  15.14 1.00  1.614655E-01  3.522282E+01-999.99 -999.99
E  er pixel1, pixel2 : 310,390
  86.73    85.76    86.41    84.63    83.99    86.57    86.57    85.76    85.92    85.92
  84.47    85.76    85.76    84.95    86.57    86.73    86.73    87.86    87.05    87.70
  87.38    87.70    85.92    86.89    87.05    87.05    87.21    85.92    87.05    87.05
  86.08    87.05    87.86    86.08    85.92    85.60    87.38    85.44    86.73    88.67
  87.70    85.92    86.08    85.28    86.89    86.57    86.73    87.54    87.54    86.25
  86.41    86.57    87.54    86.41    86.73    87.38    87.70    86.25    86.25    86.08
  87.86    87.86    86.25    87.86    86.89    86.41    86.89    87.21    87.21    87.86
  87.86    85.92    87.70    87.05    87.86    87.70    87.38    86.89    86.41    87.38
  87.70
Mean =    86.71 Variance =      .78 RMS =      .88 mW/m2/sr/cm-1
 273.07   272.38   272.84   271.56   271.10   272.95   272.95   272.38   272.49   272.49
 271.45   272.38   272.38   271.80   272.95   273.07   273.07   273.87   273.29   273.75
 273.53   273.75   272.49   273.18   273.29   273.29   273.41   272.49   273.29   273.29
 272.60   273.29   273.87   272.60   272.49   272.26   273.53   272.15   273.07   274.44
 273.75   272.49   272.60   272.03   273.18   272.95   273.07   273.64   273.64   272.73
 272.84   272.95   273.64   272.84   273.07   273.53   273.75   272.73   272.73   272.60
 273.87   273.87   272.73   273.87   273.18   272.84   273.18   273.41   273.41   273.87
 273.87   272.49   273.75   273.29   273.87   273.75   273.53   273.18   272.84   273.53
 273.75
Mean =  273.05 Variance =      .37 RMS =      .61 Kelvin
Enter pixel1, pixel2 :
```


MAS AMES ICE BUCKET TEST DATA 22-MAY-92
BLACKBODY COUNT VS SCANLINE (CH 07)
COUNTS
132
128
124
120
116
112
108
104
100
96
92
88
84
80
76
72
1800
2200
2600
3000
SCAN LINE

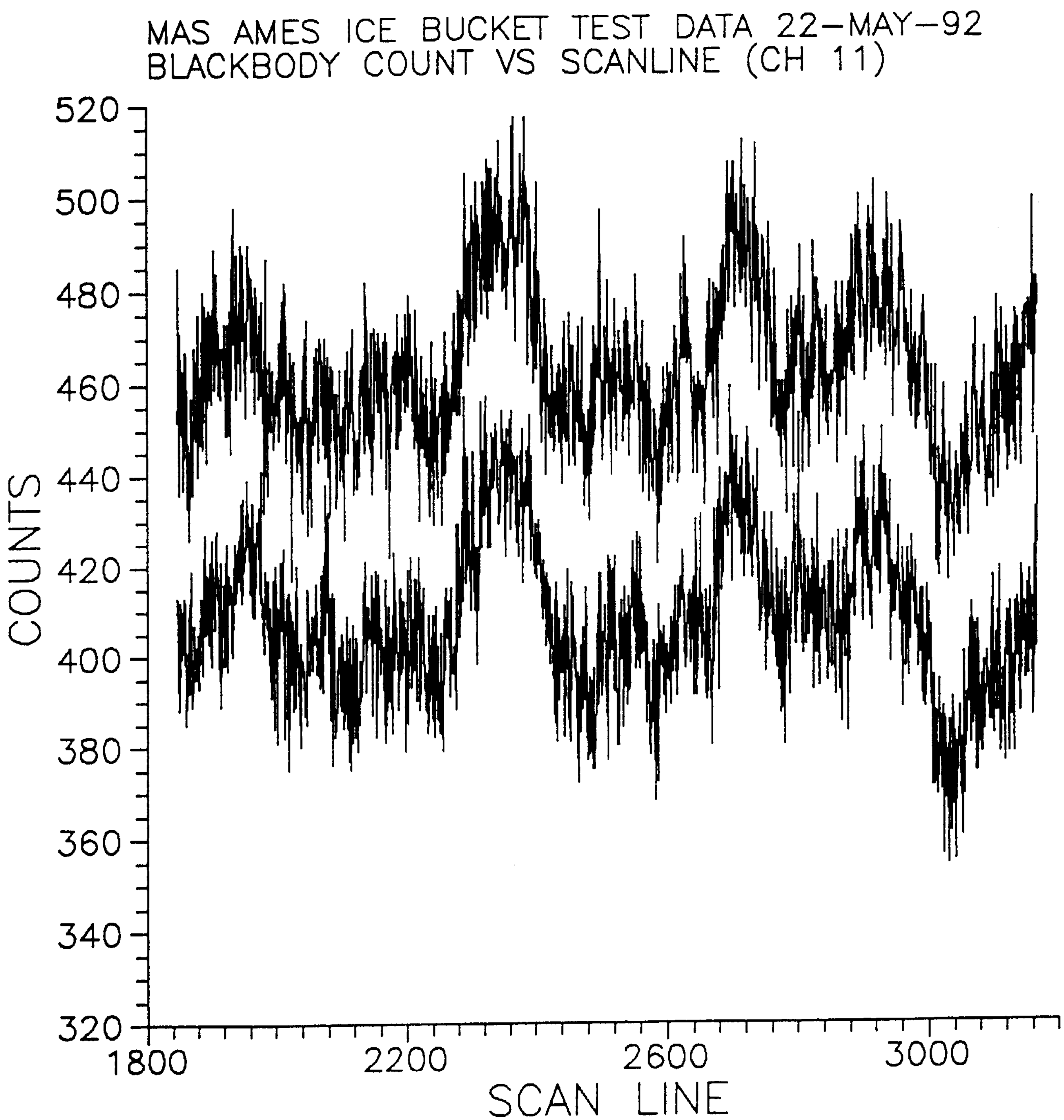


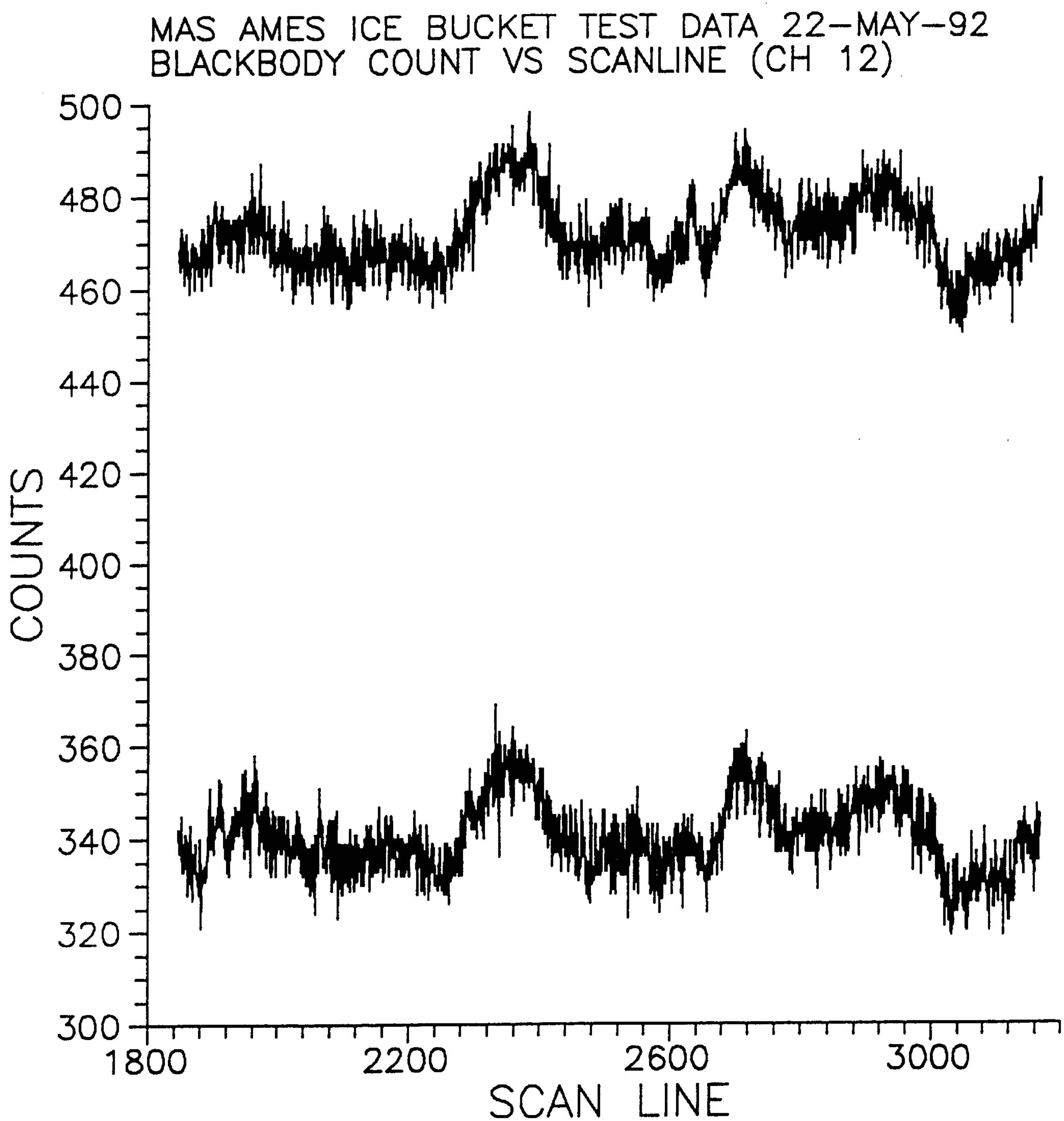
MAS AMES ICE BUCKET TEST DATA 22-MAY-92
BLACKBODY COUNT VS SCANLINE (CH 08)
COUNTS
144
140
136
132
128
124
120
116
112
108
104
100
96
92
88
84
80
1800
2200
2600
3000
SCAN LINE


MAS AMES ICE BUCKET TEST DATA 22-MAY-92
BLACKBODY COUNT VS SCANLINE (CH 09)
COUNTS
504
496
488
480
472
464
456
448
440
432
424
416
408
400
392
1800
2200
2600
3000
SCAN LINE



MAS AMES ICE BUCKET TEST DATA 22-MAY-92
BLACKBODY COUNT VS SCANLINE (CH 10)
COUNTS
504
496
488
480
472
464
456
448
440
432
424
416
408
400
392
1800
2200
2600
3000
SCAN LINE


MAS AMES ICE BUCKET TEST DATA 22-MAY-92
BLACKBODY COUNT VS SCANLINE (CH 11)
COUNTS
520
500
480
460
440
420
400
380
360
340
320
1800
2200
2600
3000
SCAN LINE


MAS AMES ICE BUCKET TEST DATA 22-MAY-92
BLACKBODY COUNT VS SCANLINE (CH 12)
COUNTS
500
480
460
440
420
400
380
360
340
320
300
1800
2200
2600
3000
SCAN LINE

MODIS Level-2 Processing Shell Design and Development

J. J. Pan
Research and Data Systems Corp.
(301) 982-3700

Purpose:

Integrate data processing algorithms, coordinate subsystems, and control data flow to minimize processing redundancy.

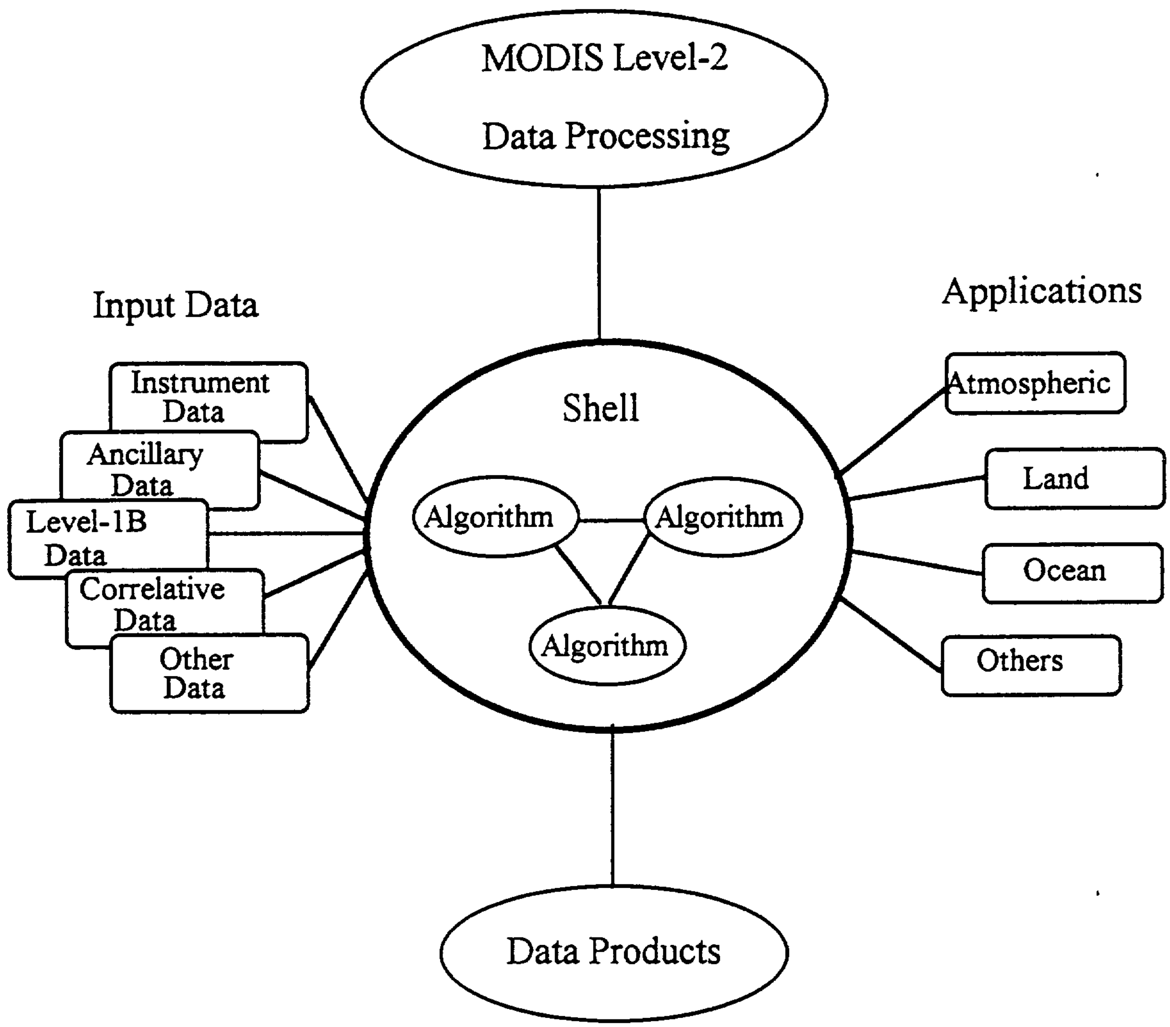
Goals:

1. Manipulate input/output data efficiently.
2. Execute algorithms systematically.
3. Generate data products correctly.
4. Make the system user-friendly.

Schedule:

1. Design --	
System Requirements Review	10/92 - 03/93 (6 months)
Preliminary Design Review	04/93 - 09/93 (6 months)
Critical Design Review	10/93 - 03/94 (6 months)
2. Development --	
Beta Version	04/94 - 12/94 (9 months)
Version 1.	01/95 - 12/95 (12 months)
Version 2.	01/96 - 12/96 (12 months)

MODIS LEVEL-2 SHELL



System Requirements Review

The purpose of the System Requirements Review (SRR) is to specify all requirements in the system integration for achieving the goals of Shell Design and Development in Level-2 Processing.

Phase 1. System Concept Development

1. Review the required input data and data products of each algorithm.
2. Prepare the schema for data flow control, algorithm implementation, and data products generation.
3. Generate a hierarchy structure of the system and indicate the relationships between subsystems.
4. Communicate with SDST Members.
5. Prepare a draft of the system concept development.
6. Hold an informal review.

Phase 2. System Requirements Definition

1. Specify functional, performance, and operational requirements.
2. Prepare a draft of the system specifications.
3. Hold an informal review.

Phase 3. System Concept Overview

1. Refine the hierarchy structure and data flow diagram based on the review comments of Phase 1.
2. Discuss the operational reliability of the system.
3. Complete a draft of the system concept development.
4. Discuss the draft with SDST Members.

Phase 4. System Requirements Analysis

1. Evaluate each requirement of Phase 3, particularly with emphasis on resolving ambiguities, discrepancies, and to be determined items.
2. Analyze the limitation of the system.
3. Finalize the system specifications.

4. Generate a draft of the system requirements analysis.
5. Hold an informal System Requirements Review.
6. Combine the drafts of Phases 3 and 4 and generate documentation on System Requirements.

Phase 5. System Requirements Review

-- Before Review

1. Prepare a SRR form which includes
   (1). Participants,
   (2). Agenda,
   (3). Materials distribution, and
   (4). Key issues to be discussed.

-- At Review

1. Discuss the key issues.
2. Check the risk areas in the system design.
3. Review any problems and items to be determined.

-- After Review

1. Generate a SRR report.

(3). Report the test results to the Team Member.
(4). Repeat steps (1) to (3) if necessary.
(5). Prepare a report to summarize the code testing.

2. Software required in Shell Design and Development
   (1). Determine the programming language and check tools required.
   (2). Develop a prototype of the shell to perform the simulation of the system integration.

Phase 5. Preliminary Design Review

-- Before Review

1. Prepare a PDR form which includes
   (1). Participants,
   (2). Agenda,
   (3). Materials distribution, and
   (4). Key issues to be discussed.

-- At Review

1. Discuss the key issues.
2. Check the risk areas in the system design.
3. Review any problems and items to be determined.

-- After Review

1. Generate a PDR report.

Critical Design Review

Basically, the Critical Design Review (CDR) is to justify the major design changes since PDR, particularly with emphasis on minimization of redundancy in data input processing.

Phase 1. Preliminary Design Revision

1. Revise the system design based on the PDR.
2. Identify the risk areas in the system design.

Phase 2. Before Review

1. Prepare a CDR form which includes
   (1). Participants,
   (2). Agenda,
   (3). Materials distribution, and
   (4). Key issues to be discussed.

Phase 3. At Review

1. Discuss the key issues.
2. Confirm the final implementation plan.
3. Discuss the risk areas in the system design.
4. Review any problems and items to be determined.

Phase 4. After Review

1. Finalize the implementation plan.
2. Generate a CDR report.
3. Initialize the Beta version development.

Beta Version

The Beta Version will be tested frequently during the system integration period because the TM code could be delivered and updated at different times. The schedule arranged for the Beta version is based on the final implementation plan in CDR. The tasks include system integration, code review, system delivery, interfaces exercises, and execution test in the operational environment.

Phase 1. System Integration

1. Test the performance of each subsystem.
2. Check the interconnections between subsystems.
3. Examine the data flow between subsystems.
4. Prepare a draft of the system integration.
5. Hold an informal review.

Phase 2. System Integration Test

1. Before test :
   (1). Develop a test plan.
   (2). Define the test strategy and approach.
   (3). Group the test data and applications.
   (4). Prepare a draft of the User's Manual.
2. At test :
   (1). Record test results.
   (2). List constraints, concerns, problems, and their effects on the design.
   (3). Evaluate data products.
3. After test :
   (1). Evaluate the system performance.
   (2). Hold an informal review.
   (3). Retest if necessary.
   (4). Generate a report of the integration test.

Phase 3. System Integration Revision

1. Revise the integration based on the results of Phase 2.
2. Update the report of the system integration.

Phase 4. Acceptance Test

1. Repeat the necessary procedure of Phase 2.
2. Generate the acceptance test report.
3. Submit the code for code review.
4. Update the User's Manual.

Phase 5. System Delivery and Training

1. Arrange the requirements for system delivery.
2. Deliver the system and User's Manual (Beta Version) to PGS.
3. Execute the system test in the operational environment.
4. Train operations staff.
5. Revise the User's Manual based on users' response.

Version 1

Basically, the major task in Version 1 is to correct all problems which happened in the Beta Version. In addition, the system maintenance will start, and all relevant documents will be finished.

Task 1. Make any revisions which are required as a result of Team Member algorithm modifications.
Task 2. Correct all problems in the Beta Version.
Task 3. Develop test plans, test data, and valid test results for checking all viable paths in the software.
Task 4. Perform test plans and deliver the software.
Task 5. Establish the maintenance procedure.
Task 6. Finish all required documents.
Task 7. Complete operation interfaces.
Task 8. Submit final version for Configuration Management.
Task 9. Provide a demo to users.

Version 2

In Version 2, the system is ready for launch. The tasks include

Task 1. Make any revisions which are required as a result of Team Member algorithm modifications.
Task 2. Correct all problems in the Version 1.
Task 3. Develop test plans, test data, and valid test results for checking all viable paths in the software.
Task 4. Perform test plans and deliver the software.
Task 5. Complete operation interfaces.
Task 6. Update final version if necessary
Task 7. Install the integrated system,
Task 8. Document the operational procedure,
Task 9. Train operations staff.

Draft

Level 2 Processing Shell Design & Development

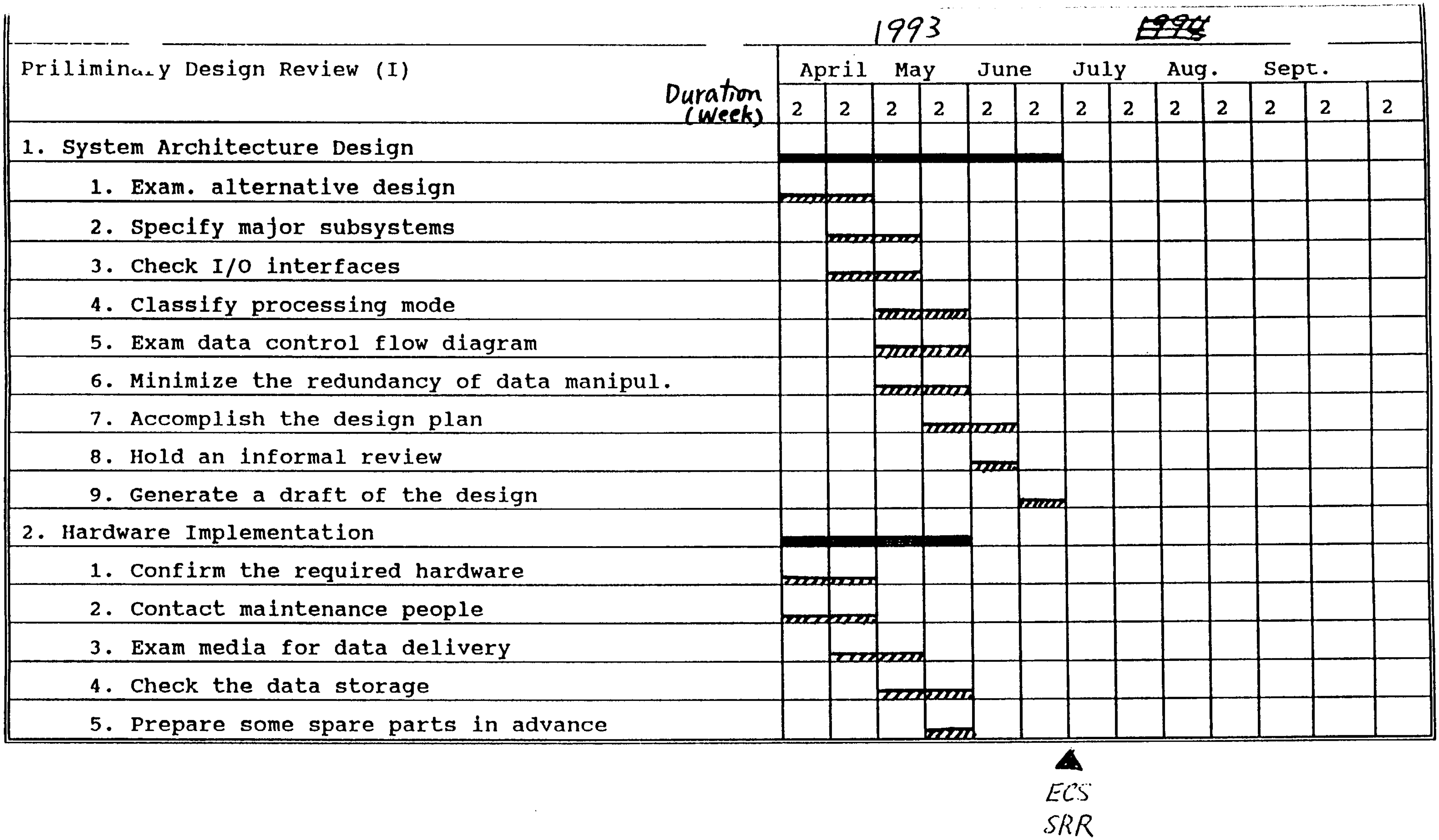
System Requirements Review (I) — Duration (week)	1992 Oct		Nov		Dec		Jan 1993		Feb		March		
	2	2	2	2	2	2	2	2	2	2	2	2	2
1. System Concept Development	■	■	■	■	■	■	■						
1. Review data and algorithms	▨	▨	▨	▨									
2. Prepare the schema			▨	▨	▨								
3. Generate the system structure				▨	▨	▨							
4. Communicate with SDST members				▨	▨	▨							
5. Prepare a draft					▨	▨	▨						
6. Hold an informal review							▨						
2. System Requirements Definition		■	■	■	■	■							
1. Specify all requirements		▨	▨	▨									
2. Prepare a draft			▨	▨	▨								
3. Hold an informal review						▨							
3. System Concept Overview								■	■	■			
1. Refine system structure								▨					
2. Evaluate the reality of the system								▨					
3. Complete the draft									▨	▨			
4. Discuss the draft										▨			

■■■■■ Period of each Phase

▨▨▨▨▨ Period of each Task

▲ Milestone

	1992						1993						
System Requirements Review (II)	Oct.		Nov.		Dec.		Jan.		Feb.		March		
Duration (Week)	2	2	2	2	2	2	2	2	2	2	2	2	2
4. System Requirements Analysis													
1. Evaluate requirements													
2. Analyze the limitation													
3. Finalize the system specifications													
4. Generate a draft													
5. Hold an informal review													
6. Generate a document on System Require.													
5. System Requirements Review													
1. Before review													
2. At review													
3. After review													


1993
1994
Priliminary Design Review (I)
Duration (week)
April May June July Aug. Sept.
2 2 2 2 2 2 2 2 2 2 2 2 2
1. System Architecture Design
1. Exam. alternative design
2. Specify major subsystems
3. Check I/O interfaces
4. Classify processing mode
5. Exam data control flow diagram
6. Minimize the redundancy of data manipul.
7. Accomplish the design plan
8. Hold an informal review
9. Generate a draft of the design
2. Hardware Implementation
1. Confirm the required hardware
2. Contact maintenance people
3. Exam media for data delivery
4. Check the data storage
5. Prepare some spare parts in advance
ECS
SRR

1993

Preliminary Design Review (II)	Duration (Week)	April		May		June		July		Aug.		Sept.		
		2	2	2	2	2	2	2	2	2	2	2	2	2
3. Data Management														
1. Check test data														
2. Exam. data format and validation														
3. Check the volume of data														
4. Software Implementation														
1. Test TM code														
2. Develop preliminary shell program														
5. Preliminary Design Review														
1. Before review														
2. At review														
3. After review														

▲ ECS SRR

1993								1994					
Critical Design Review	Oct.		Nov.		Dec.		Jan.		Feb.		March		
(Duration) week	2	2	2	2	2	2	2	2	2	2	2	2	2
1. Preliminary Design Revision													
1. Revise the system design													
2. Identify the risk areas													
1. Before review													
1. Prepare a CDR form													
2. At Review													
1. Discuss the key issues													
2. Finalize implementation plan													
3. Discuss the risk areas													
4. Review any problems													
3. After review													
1. Finalize the implementation plan													
2. Generate a CDR report													
3. Initialize the Beta Version													

Initial delivering of TM β code

Beta Version (I)

Month	1	2	3	4	5	6	7	8	9	10	11	12
	1994											
1. System Integration	X	X	X	X	X							
1. Test of subsystems	X	X	X	X								
2. Test interconnections between subsystems	X	X	X	X								
3. Examine data flow control	X	X	X	X								
4. Prepare a draft			X	X	X							
5. Hold an informal review					X							
2. System Integration Test						X	X	X				
1. Before test						X						
2. At test							X					
3. After test								X				
3. System Integration Revision									X			
1. Revise the integration									X			
2. Update the draft									X			
4. Acceptance test										X		
1. Repeat phase 2.										X		
2. Generate the acceptance test report										X		
3. Submit the code for code review											X	
4. Update the User's Manual											X	

▲ Initial delivery of β version of TM Code (month 1)

▲ ECS PDR (month 7)

▲ SCF β (end of month 12)

	1994												
Beta Version (II) Month	1	2	3	4	5	6	7	8	9	1	11	12	
5. System Delivery and Training													
1. Arrange requirements for delivery													
2. Deliver the system and User's Manual													
3. Execute the system test on site													
4. Train operations staff													
5. Revise the User's Manual													

Initial delivery β version of TM Code

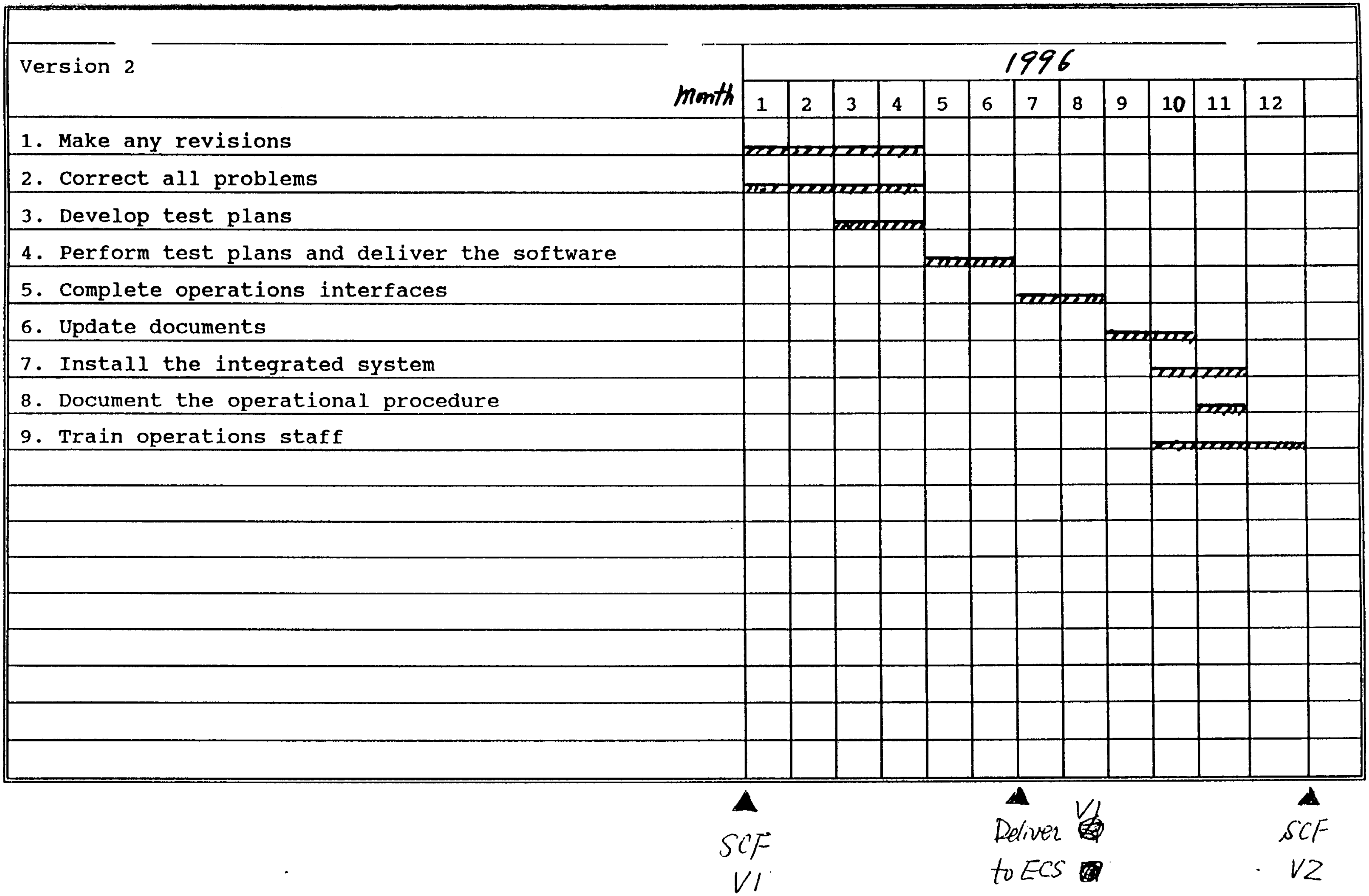
SCF β

Version _	1995												
Month	1	2	3	4	5	6	7	8	9	10	11	12	
1. Make any revisions	▨	▨	▨	▨	▨	▨							
2. Correct all problems	▨	▨	▨	▨	▨	▨							
3. Develop test plans	▨	▨	▨	▨	▨	▨							
4. Perform test plans and deliver the software				▨	▨	▨	▨						
5. Establish the maintenance procedure							▨	▨					
6. Finish all documents									▨	▨			
7. Complete operation interfaces										▨	▨		
8. Submit final version												▨	
9. Provide a demo													▨

▲ SCF β (start of month 1)

▲ Initial delivery of TM β code to DACC β (start of month 7)

▲ SCF V1 (end of month 12)


Version 2
1996
Month
1
2
3
4
5
6
7
8
9
10
11
12
1. Make any revisions
2. Correct all problems
3. Develop test plans
4. Perform test plans and deliver the software
5. Complete operations interfaces
6. Update documents
7. Install the integrated system
8. Document the operational procedure
9. Train operations staff
SCF V1
Deliver V1 to ECS
SCF V2

Work Task List
for the MODIS Level-1A Design

Thomas E. Goff
28 May, 1992

NOTES: dependencies are given in curly braces {}, one man duration times are given in brackets [], and comments are given in *italics*.

1. Software Requirements Specification [3 weeks]
   Provide a full description to SDST specs - combining appropriate elements of the IEEE and NASA requirements specification standard guidelines.

2. Data Rate & Volume Spreadsheet {Excel spread sheet} [1 week]
   Clean up and greatly expand the information in the existing or a newly created spread sheet to include a full description of all quantities. The emphasis will be on clarity. This will be used as an input to the EIS writeup.

3. CASE design {cadre's teamwork, training}
   Perform the next revision of the design in a full CASE environment. The major CASE subitems are given below.

   3.1 Structure Charts [1 week per bubble, 22 bubbles]
   For each processing element, by bubble levels, as derived from the existing preliminary design. The next version of the design will be expanded to the code level. A List of the bubbles follows:

   3.1.1 Control Message Decomposition
   3.1.2 Abort / Cleanup Determination
   3.1.3 Processing Mode Setup
   3.1.4 Status Derivation
   3.1.5 Data Availability Check
   3.1.6 Memory Requirement Determination
   3.1.7 Memory Request
   3.1.8 Data Store Setup
   3.1.9 Packet Parameter Determination
   3.1.10 Packet Structure Verification
   3.1.11 Time Stamping / Formatting
   3.1.12 Packet to Scan Cube Location
   3.1.13 Granule virtual Translation
   3.1.14 Completeness Indicator Updating
   3.1.15 Scan Cube Completeness Determination
   3.1.16 Ancillary Data Appending
   3.1.17 Cube Header Creation
   3.1.18 Metadata Updating
   3.1.19 Granule completeness Determination
   3.1.20 Metadata Derivation
   3.1.21 Granule Header Appending
   3.1.22 Granule and Header Transmission

   3.2 Data Dictionary {Teamwork} [concurrent with the structure charts]
   This will be performed as a necessary component of the structure chart design.

3.2.1 Data Structure Layout {Teamwork, FrameMaker}

These are the data arrays that will be ingested or generated by the Level-1A process. They consist of instrument data in addition to the contents of the various internal and external messages.

3.2.1.1 Level-0 Packet contents {data from SBRC} [2 weeks]
by packet ID (calibration vs science day/night, etc)
3.2.1.2 DADS Interrogation Messages {PGS tool kit} [4 weeks]
3.2.1.3 Level-1a Data Product contents {Level-0 Packet contents} [1 week]
3.2.1.4 Metadata contents [4 weeks]
3.2.1.5 Cube and Granule Header contents [1 week]
3.2.1.6 Initiation Message {PGS tool kit} [1 week]
3.2.1.7 Termination Message {PGS tool kit} [1 week]
3.2.1.8 Dynamic Status Inquiry/Response {PGS tool kit} [1 week]
3.2.1.9 Processing Log entries {on going} [1 week]

3.2.2 Data Flow [2 weeks]

These items of flow control are defined in the data dictionary, formatted into messages and passed to and received from external processes.

3.2.2.1 Instrument Events {MCST definitions}
3.2.2.2 Processing Events
3.2.2.3 System Memory Allocations {PGS tool kit}

3.2.3 Processing Items [2 weeks]

These items must be derived as part of the data dictionary effort. They are not covered by the normal data processing bubbles.

3.2.3.1 Initialization and Setup of Metadata and Granule Outlines
3.2.3.2 Granule Virtual Paging

4. Assumptions / Tracking List {on going} [concurrent]

A continuing and expanding effort that is kept concurrent with the design effort.

5. Coding {TLCF computer / SoftBench}

The generation of computer source code in an environment that is coupled to the structured design charts. Integrated debugging of code and configuration management of code and complementary files will be a part of this effort.

5.1 Source File Creation/Debugging {TLCF compilers} [30 weeks]
5.2 MakeFiles {Make manuals} [2 weeks]
5.3 Code QA Checking (tools) {aux computers} [3 weeks]
5.4 code walk throughs {on going} [12 weeks]
5.5 Source Code Feature Extraction {TLCF PERL/ scripting languages} [2-5 weeks]
5.6 Key Index Database {TLCF computer} [5 weeks]

6. External Interface Document {PGS tool kit} [2 weeks]

This document will describe and detail the interactions among the MODIS Level-1A processor and the external (to the MODIS Level-1A processor) environment. The external entities are listed below:

6.1 Wall Time
6.2 DTM

6.3 DADS
6.4 PMS
6.5 SCA

7. Prototyping

This effort provides early derivation and testing of selected portions of the design. Items generated during this prototyping effort will be incorporated into the final design.

7.1 IFOV Ground Plots {Scan Geometry}
7.1.1 Plotting Package {library} [1 week]
7.1.2 Projection Routines [4 weeks]
7.1.3 MODIS Scan Geometry {SBRC / MCST} [4 weeks]

8. Scan Geometry derivation [4 weeks]

Obtaining / generation of the equations to convert from instrument geometry parameters to Earth intersection via IFOV vectors.

8.1 Anchor Point Determination {SBRC geometry}
8.1.1 Earth Model {PGS Tool Kit, Earth model decision}
8.2 Moon Looking {flight dynamics code and/or algorithms}

9. PGS Tool Kit Interfacing {ECS contractor Tool Kit Spec Delivery}

Included in this topic are items that will be necessary to complete the MODIS Level-1A processor design and/or verification. This includes, but is not limited to, the following items:

9.1 S/C Ephemeris Simulator {flight dynamics / homegrown}

To provide S/C platform position and attitude values at either the S/C telemetry packet intervals, or at the MODIS requested UTC time.

9.2 Digital Terrain Model, database

Terrain Elevation and possibly Slope at user requested positions. This is to be provided in several, user selectable, coordinate systems.

10. Testing/Debugging {computer} [30 weeks]

Normal debugging will be handled within the CASE / SoftBench environment. Auxiliary items not available from the CASE environment are listed below:

10.1 Packet Telemetry Simulator {SBRC telemetry lists} [8 weeks]

GTSIM is a real-time simulator and needs to be checked for possible use for MODIS. Otherwise, a telemetry simulator for MODIS that can generate staged telemetry via the (simulated) DADS needs to be developed.

10.2 L-1A Data Product Validator/Invertor {Level-1A design} [8 weeks]

This utility item receives the Level-1A Scan Cube and verifies the contents by any of various analytical or visual techniques. This may contain COTS image processing facilities.

The Level-1A to Level-0 data invertor will be used recursively with the MODIS Level-1A Data Product Generator to validate the Level-1A Data Products. This invertor should be written by an independent validation authority.

11. Documentation {FrameMaker or equivalent on TLCF}

11.1 EIS for MODIS {MODIS spread sheet} [3 weeks]
11.2 Software Design Description {CASE design} [3 weeks]
11.3 Software V&V Plan {Code Review unless performed independently} [8 weeks]
  11.3.1 Traceability Matrix {CASE design}
11.4 User {delivered with code releases} [2 weeks]

12. Configuration Management {CM tool / networked} [2 weeks]

13. Training

This is a list of formal training that needs to be provided to allow optimal efficiency of the development staff.

13.1 formal TLCF computer classes [4 weeks]
13.2 Cadre's Teamwork Usage Classes [2 weeks]
13.3 SoftBench Usage [2 weeks]
13.4 Configuration Management Tool [2 weeks]
13.5 Programming QA Tools [2 weeks]

RISKS:

* Direct communication with SBRC.
  Channels need to be created between the SDST and SBRC for direct access to instrument information.

* TLCF availability of the complete computer system.
  - System Administration.
  - Vendor support for hardware maintenance, software updates, and consulting.

* Timeliness of Teamwork training.
  This is the first training that needs to be completed.

* Verification of packet simulator.
  A bit by bit comparison with explained discrepancies of the simulated packets against SBRC supplied packets.

* Verification of the Level-1A validator.
  Perhaps SBRC generated test data can be used?

* The Independence of the Validation Team.

* PGS tool kit specifications.
  Early information about the PGS Tool Kit will allow this MODIS design effort to avoid delays.

* Timeliness and thoroughness of the programming style reviews.
  This effort requires a comprehensive review of code performed within a minimum turn around.

file: c:\modis\task_l1a.wp