NASA/CR-97- 205877

NAG5-2941

SUMMER INSTITUTE IN ENGINEERING AND COMPUTER APPLICATIONS

LEARNING THROUGH EXPERIENCE

NOVEMBER 1995

A SUMMER INSTITUTE

IN

ENGINEERING AND COMPUTER APPLICATIONS

1995

FINAL REPORT

Prepared by:

Joan S. Langdon, Ph.D
Bowie State University

TABLE OF CONTENTS

SUMMER INSTITUTE IN ENGINEERING AND COMPUTER APPLICATIONS

ADMINISTRATIVE PROCEDURES

Student Participant Recruitment and Selection

The Summer Institute in Engineering and Computer Applications (SIECA) Program, sponsored jointly by Bowie State University (BSU) and NASA's Goddard Space Flight Center (GSFC), is a ten week internship program which supports graduate and undergraduate students. This year, there were ten (10) graduate and eighteen (18) undergraduate participants. The students were recruited from colleges and universities in the United States and the Commonwealth of Puerto Rico. The list of recruitment schools appears in the Appendix (page III).

The students selected to participate in the program must meet the following list of criteria:

1. American citizenship.

2. Grade point average of 3.00 or better.

3. A major in any of the areas: electrical/mechanical/aerospace engineering; physics; computer science with a mathematics/science orientation; and mathematics.

4. Undergraduate students must have completed their sophomore year and be enrolled in a 4-year accredited institution.

   Graduate students must have a bachelor's degree in any one of the areas mentioned in number three (3) above.

5. Two letters of reference.

Student participants fall into two categories: new and returnee. Typically, about forty (40) percent of the participants are returnees to the SIECA program. Typically, students are allowed to participate in the institute for a maximum of two years. This year six percent (6%) of the undergraduate participants and 20 percent (20%) of the graduate participants were returnees.

Letters of solicitation and applications were sent to various individuals (chairpersons of science, engineering and mathematics departments, career planning and placement directors, professors, etc.) at the institutions listed in the Appendix (page III). Follow-up telephone calls were made to these institutions. The SIECA Program Director attended conferences for the purpose of recruitment. Applications were sent to approximately ninety individuals who made direct inquiries to the SIECA Director and the outreach program staff at Goddard Space Flight Center (GSFC). These individuals heard about the SIECA Program from previous participants, faculty members at their institutions, etc. Some recruitment was also done by the outreach program director and other outreach program staff members at GSFC.

The student applications are initially checked to determine whether each of them meets the list of selection criteria given previously. All students who do not meet the requirements receive letters informing them of the reason that their particular application could not be considered for the program. All other applications are accepted. The actual selection of the students is done by the outreach program director and the SIECA director. Students are selected based on their qualifications and area of interest.

This year one hundred fifty-five (155) applications (one hundred thirty-four [134] undergraduate and twenty-one [21] graduate) were received from aspiring participants. Thirty (30) of these applications (twenty-nine [29] undergraduate and one [1] graduate) did not meet one or more of the program requirements. One hundred twenty-five (125) applicants were eligible for participation in the program. The list of schools represented by the applications can be found in the Appendix (Page I).

Eighteen (18) undergraduate students and ten (10) graduate students were accepted for the 1995 institute. Twenty-six students started their internship on May 30, 1995. These students completed the 10 week session which ended August 4, 1995. Two additional participants served a seven week internship starting June 26, 1995 and ending August 11, 1995.

SEMINARS/SPECIAL COURSES/TOURS/COLLEGE FAIR

Several seminars were offered on site to give the student participants information about the technical and nontechnical sides of GSFC. The seminars covered a wide range of subjects, from NASA's Past and Future to job opportunities. Mr. Joseph Rotherberg, the Director of GSFC, held a round table discussion with the students which was very informative.

A College Fair was held for the third year during this internship. Students in the various programs sponsored at GSFC hosted this affair for local area junior and senior high school students.

A list of the seminars given during the ten-week period appears in the Appendix (Page IX).

FORTRAN, Pascal, and C are the languages used most often in daily activities at the center. The SIECA participants attend training courses in these languages as required by the participating mentors on an as needed basis. This policy allows the participants to begin an immediate relationship with their mentors in the training environment. They take advantage of the full ten-week period, learning and working on individual Goddard related projects.

This year fifty-seven percent (57%) of the SIECA students participated in an overnight tour of the NASA/Wallops facility. They viewed the aircraft hangar which houses equipment used for NASA related aviation research. They also took an overall tour of the facility which included the Range Control Center. The students participated in a visit to the University of Maryland-Eastern Shore for a campus tour, in particular, the Engineering Department. Wallops deals with sub-orbital flight.

FACILITIES/TRANSPORTATION

Office space and equipment needed by the participants were furnished by GSFC. For the past four (4) summers, out-of-town students have been housed at Seven Springs Village, a local apartment complex. Transportation to GSFC from this central location and vice versa was provided by the center on a daily basis. All projects involving participants took place on the site.

STAFF AND ADMINISTRATION

The SIECA program has one director and part-time secretarial assistance. The director holds a terminal degree in the area of Mathematics and a masters degree in Computer Science. The 1990 institute gave the director her first experience with the program.

COLLABORATION

Bowie State University (BSU) is responsible for student recruitment, housing arrangements, and the administrative functions required to arrange for each of the student participants to receive college credit for the ten week internship. BSU is also responsible for the disbursement of the participants' stipends and travel funds. GSFC provides technical advisors (mentors), local transportation, and office space for the students.

PARTICIPANT/PROJECT MONITORING AND EVALUATION

Each participant was matched with a mentor employed in a technical area closely allied to the participant's major area of stated interest. The student then interned with his/her mentor for the ten week period on a GSFC research project. Approximately six weeks into the institute, each participant submitted an abstract to the program director which described the type of work with which the student was involved. The students and mentors were interviewed periodically during the internship by the SIECA director to ensure that the student was having a good experience at GSFC.

During the final week of the program each participant gave a brief talk on his/her project. The audience for undergraduate students included the chief or assistant chief of the student's assigned division or directorate, the student's mentor, the program director, and any other interested GSFC personnel. The audience for graduate students included the same individuals as the undergraduate audience as well as the director of the participant's assigned code. Each student was also required to submit a paper to the SIECA director and to complete an evaluation form on his/her summer experience. A copy of each student participant's abstract and paper appears in the Appendix (pages XIV and XLI respectively).

The evaluation process was a cooperative effort. The mentors submitted performance evaluations on their student interns. Each mentor was also interviewed on the subject of his/her intern's progress and performance. The SIECA director developed a summary evaluation based on the evaluations, interviews, and various other factors. The participants may be invited to return to the summer institute a second year if their summary evaluation is above average. This evaluation also stands as a basis for the student's grade for the ten week internship. Undergraduate students receive four (4) hours of college credit for this internship, and graduate students receive one (1) hour. A copy of each of the evaluation forms appears in the Appendix (Pages X and XIII).

Each student participant received a Certificate of Achievement for completing the internship and each mentor received a Certificate of Appreciation.

FISCAL AND DEVELOPMENT ACTIVITIES

College credits were received through BSU by each participant. The college tuition was paid by the SIECA grant. Summer 1995 tuition for an undergraduate student was $413.50 (4 credits), while the amount for a graduate student was $168.50 (1 credit). Participants in the program received a training stipend as follows:

First-time undergraduate participants	$380/week
Returnee undergraduate participants	$390/week
First-time graduate participants	$450/week
Returnee graduate participants	$470/week

Travel support for student participants (undergraduate and graduate) for summer 1995 was $6201.52.

JOB READINESS/JOB INTERNSHIP DEVELOPMENT AND PLACEMENT

Mr. Joseph Rothenberg, the Center Director, discussed job opportunities during his round table session. Most of the SIECA students expressed the desire to either return to the institute during the summer of 1996 or return to GSFC as a co-op or permanent employee. Just as in previous summers, several mentors made special requests in favor of working with the same student for a subsequent year, if at all possible.

Since 1990, SIECA has had eleven (11) participants (undergraduate and graduate) who co-oped with NASA and five (5) undergraduate participants who have received jobs with NASA or NASA contractors. Aprille Ericsson-Jackson, a former graduate student in the program who is now an employee at GSFC, recently became the first Afro-American female Ph.D. Aerospace Engineer. These students are helping to realize a major goal of the program, mainly, to help NASA/GSFC gain contact with individuals who have backgrounds in areas which form the heart of GSFC operations. Since 100% of the participants involved in GSFC's co-op program and 50% of the participants receiving employment are minority individuals, the program is meeting another major goal. SIECA's original purpose was to bring minority individuals with the appropriate science and engineering backgrounds into contact with the NASA work environment.

STUDENT FOLLOW-UP/TRACKING

A follow-up survey was done in 1993 and the results were published in the Final Report for the summer of 1993. The next longitudinal study will be conducted in 1996.

A P P E N D I X

SIECA STUDENT DATA

1995 SEICA APPLICATIONS BY INSTITUTION

INSTITUTIONS	NUMBER OF APPLICANTS	NUMBER OF PARTICIPANTS	NUMBER OF RETURNEES
Arizona State University	3	0	0
Armstrong State College	1	0	0
Bennett College	2	1	0
Bowie State University	2	2	0
Cal. State Northridge	1	1	0
Cal. State Polytechnic Univ.	1	0	0
Carnegie Mellon University	1	0	0
Central State University	2	2	0
City College of New York	1	0	0
Clemson University	1	1	1
Columbia University	1	0	0
Cooper Union	1	1	0
Cornell University	1	0	0
Dillard University	2	0	0
Elizabeth City State Univ.	4	0	0
Fayetteville State Univ.	1	0	0
Florida A&M University	2	0	0
Florida State University	1	1	1
Fort Valley State College	1	1	0
George Washington Univ.	1	0	0
Georgia Inst. of Tech.	1	0	0
Hampton University	5	2	0
Hofstra University	1	0	0
Howard University	3	0	0
Illinois Wesleyan Univ.	1	0	0
Inter American Univ. of P.R.	4	1	0
Jackson State University	5	1	0
Lemoyne College	1	0	0
Lincoln University	3	1	0
Loyola College in Maryland	1	0	0
Morgan State University	6	1	0
Mount Holyoke College	1	0	0
Norfolk State University	12	1	0
NC State A&T University	4	2	0
Prince George's Comm. Col.	1	0	0
South Carolina State Univ.	2	0	0
Southern Illinois Univ.	1	0	0

INSTITUTIONS	NUMBER OF APPLICANTS	NUMBER OF PARTICIPANTS	NUMBER OF RETURNEES
Spelman College	7	0	0
Swarthmore College	1	0	0
Tennessee State University	1	0	0
Towson State University	1	0	0
Tuskegee University	1	1	0
University of Detroit	1	0	0
University of D.C.	4	0	0
University of Maryland (Baltimore County)	2	1	0
University of Maryland (College Park)	4	1	0
University of Michigan	1	0	0
University of North Carolina	1	0	0
University of Puerto Rico	38	3	0
University of TX, Arlington	1	0	0
University of TX at El Paso	5	3	1
University of TX, Pan Amer.	2	0	0
University of Virginia	1	0	0
Virginia Polytechnic Inst. and State University	1	0	0
Western Kentucky Univ.	1	0	0
TOTALS:	155	28	3

There were one hundred thirty-five (135) undergraduate applications received. Of these, seventy-four (74) were male applicants and sixty-one (61) were female applicants. Twenty-nine (29) of these applicants did not meet the basic qualifications. From the remaining pool of one hundred six (106) applicants, the 1995 Summer Institute obtained eleven (11) male participants and seven (7) female participants.

There were twenty-one (21) graduate applications received. Of these, fourteen (14) were male applicants and seven (7) were female applicants. One (1) of these applicants did not meet the basic qualifications. From the remaining pool of twenty (20) applicants, the 1995 Summer Institute obtained seven (7) male participants and three (3) female participants.

Eighty-six (86%) of the SIECA participants were minority students.

1995 SIECA RECRUITMENT SCHOOLS

Benedict College
Bennett College
Bowie State University
Central State University
Cheyney University
City College of New York
Coppin State College
Delaware State College
Dillard University
Elizabeth City State University
Fayetteville State University
Fisk University
Florida A&M University
Hampton University
Howard University
Jackson State University
Morehouse College
Morgan State University
Norfolk State University
North Carolina A&T State University
Savannah State College
Shaw University
South Carolina State College
Spelman College
Tennessee State University
University of Arizona
University of the District of Columbia
University of Maryland-Eastern Shore
University of Puerto Rico
University of Texas
University of Texas-Pan American
University of Texas-El Paso
Virginia State University
Virginia Union University

1995 SIECA PARTICIPANTS AND MENTORS

UNDERGRADUATE STUDENTS	MENTOR	PHONE	CODE	BLDG/RM
Edwin Beckford	David Batchelor	62988	632	26/Base
Cedric Blair	Brad Ferrier	64034	912	22/332
Lolita Clayborn	Dr. Forrest Hall	62974	923	22/Trail
Trena Covington	Dieter Bilitza	441-4193	630	26/Base
Ricardo Diaz	Phil Nessler	64693	205.2	18/173
Francisco Fernandez	Tim Leath	65302	735.3	11/E105
Stacy Flowe	Jean Swank	69167	666	2/263
Roberto Gomez	Robert MacDowall	62608	695	1/257
Marvin Jackson	Angelo Wade	68058	754.2	7/181
Michelle Jones	Boyd Pearson	64737	303	6/W16
Dana Murph	Dr. Forrest Hall	62974	923	22/Trail
Nnaemeka Nwosu	A. Ericsson-Jackson	69154	712.3	11/S104
Miguel Ordaz	Jim Bishop	63212	521	23/E441
Patsy Polston	Peter Shirron	67327	713	7/123
Marcellus Proctor	Robert Stone	65873	743	5/W83
Demetrius Shaffer	K. Petraska-Veum	63348	933	28/W248
Danielle Whipp	Mona Kessell	66595	632	26/Base
Arturo Yanez	Chuck Manns	61370	442	29/100

GRADUATE STUDENTS	MENTOR	PHONE	CODE	BLDG/RM
Gilbert Castillo	Jan McGarry	65040	920.1	22/C227D
Shahar Harel	Brad Torain	66990	541.3	12/116A
Samone Jones	Patrick Coronado	69323	935	28/W186C
Prem Lall	Dr. Sui	62122	913	22/C151
Dorothy Langendorf	David Landis	6/3349	923	22/G91
Tony Miller	Shahin Samadi	68510	920.2	22/G70N
Alberto Rodriguez	Javier Lecha	61002	723.2	5/L355B
Kenneth Russell	Alan Cudmore	64339	735.3	11/E109
Rontrill Swain	Jon Valett	66564	552	23/E229
Ebony Waller	Ron Oliversen	66290	684.1	21/G36

SIECA CERTIFICATES OF ACHIEVEMENT - 1995

UNDERGRADUATES

Edwin Beckford
Cedric Blair
Lolita Clayborn
Trena Covington
Ricardo Diaz
Francisco Fernandez
Stacy Flowe
Roberto Gomez
Marvin Jackson
Michelle Jones
Dana Murph
Nnaemeka Nwosu
Miguel Ordaz
Patsy Polston
Marcellus Proctor
Demetrius Shaffer
Danielle Whipp
Arturo Yanez

GRADUATES

Gilbert Castillo
Shahar Harel
Samone Jones
Prem Lall
Dorothy Langendorf
Fred Anthony Miller
Alberto Rodriguez
Kenneth Russell
Rontrill Swain
Ebony Waller

VII

SIECA CERTIFICATES OF APPRECIATION - 1995

RECIPIENT\CODE

David Batchelor/632
Valerie Thomas/632
Brad Ferrier/912
Dr. Forrest Hall/923
Dieter Bilitza/630
Phillip Nessler/205.2
Steven Schoolcraft/205.2
Charlie Papadimitris/205.2
Tim Leath/735.3
Jean Swank/666
Dr. Roger Hess/690
Dr. Robert MacDowall/695
Brent Ignacio/695
Cathie Meetre/695
Angelo Wade/754.2
Boyd Pearson/303
George Coger/303
Ted Hammer/303
Charles Coleman/303
Aprille Ericsson-Jackson/712.3
Darrell Zimbelman/712.3
Dave McGlew/712.3
Jim Bishop/521
Dr. Mike DiPirro/713
Peter Shirron/713
Tom Hait/713
Jim Tuttle/713
Robert Stone/743
Tom Gostomski/743
Cindi Lewis/743
Karen Petraska-Veum/933
Emma Kolstad/933
Mona Kessell/632
Bob Candey/632
Chuck Manns/442
Cynthia Ivy/442
Jan McGarry/920.1
Brad Torain/541.3

VIII

SIECA CERTIFICATES OF APPRECIATION - 1995 (continued)

Vishal Desai/541.3
John Steedman/541.3
Patrick Coronado/935
Nick Short, Jr./935
Dr. Chung-Hsiung Sui/913
David Landis/923
Jeff Newcomer/923
Amy Ruck/923
Fred Irani/923
Shahin Samadi/920.2
Curt Tilmes/920.2
Dan Kozak/920.2
Michael Wilson/920.2
Javier Lecha/723.2
Willie Blanco/723.2
Rajeev Sharma/723.2
Alfonso Hermida/723.2
Alan Cudmore/735.3
Tim Leath/735.3
Jon Valett/552
David Matusow/552
John Bristow/552
Ron Oliversen/684.1
Peter Chen/684.1

1995 SIECA UNDERGRADUATE PRESENTATIONS

PARTICIPANT	TOPIC
Edwin Beckford	Skylab X-ray Images Made Realily Accessible
Cedric Blair	Range Related Effects of Radar Estimated Rainfall
Lolita Clayborn	Supplying our Vegetated Areas
Trena Covington	Enhancing the User Friendliness of NSSDC's Model Archive
Ricardo Diaz	Indoor Air Quality Evaluation of Building 18/Fire Pumps Performance Evaluation/Replacement of the Boilers at Building 24
Francisco Fernandez	Use of 3D Computer Modelling in Spacecraft Development
Stacy Flowe	GSE.Basae for the XTE Satellite
Roberto Gomez	Pattern Recognition
Marvin Jackson	Electromagnetic Interference (EMI) and Hewlett Packard's Visual Engineering Environment (HP VEE)
Michelle Jones	Analysis of GSPAR Compliance with ISO 9000
Dana Murph	Biospheric Processes in the General Circulation Model
Nnaemeka Nwosu	The Vibrational Effect of Solar Array Motion on Tropical Rainfall Measuring Mission (TRMM)
Miguel Ordaz	The Design and Implementation of the CCSDS Simulator Chip

1995 SIECA UNDERGRADUATE PRESENTATIONS (continued)

Patsy Polston	High Resolution Penetratiaon Depth Thermometer Testing
Marcellus Proctor	My Experiences with the Spartan 206 Spacecraft Project
Demetriius Shaffer	My Summer Contribution to Goddard Space Flight Center
Danielle Whipp	Space Plasma Detector Program
Arturo Yanez	Hubble Space Telescope (HST) Vehicle Electrical System Test (VEST)

XI

1995 SIECA GRADUATE PRESENTATIONS

PARTICIPANT	TOPIC
Gilbert Castillo	Programming for the GLAS and the 1.2m Telescope
Shahar Harel	Network Management for the EOS Backbone Network (Ebnet)
Samone Jones	
Prem Lall	The Effect of the Diunral Cycle on Precipitation Rates in the Toga Coare IOP
Dorothy Langendorf	BOREAS Project, C Programming Style and Maintenance
Fred Anthony Miller	SATE: System Administration Task Environment
Alberto Rodriguez	Simulation of a Tracking Device Controller for PAMS Experiment
Kenneth Russell	The Remote Telemetry Monitoring System
Rontrill Swain	The Use of Swingby and Advanced Viedo Technology to Make a Quicktime Movie
Ebony Waller	Design of a Lunar-Based Telescope

XII

1995 SUMMER SEMINAR SCHEDULE

DATE	TOPIC	PRESENTER
6/2/95	Bridging Visual Communication Workshop	Gallaudet Univ.
6/13/95	From Pre-College to Ph.D. in 8 Years	Dr. Horace Moo-Young
7/5/95	Update on Mission to Planet Earth	
7/11/95	College Fair	Various University Student Reps.
7/18/95	Technology Transfer	Nona Minnifield

Questions for Mr. Rothenberg from summer interns - July 6, 1995

1. What role does NASA plan to take in helping to improve the environment for future generations?

2. How will the recent cutbacks affect future cooperative education positions at Goddard? How does the co-op program work?

3. What are the plans for the Wallops Flight Facility?

4. What kind of graduate fellowship/scholarship programs are available at GSFC? How many of these grants are awarded to minorities?

5. What are some of the major electrical engineering subjects that correspond with what goes on at NASA/Goddard?

6. Who founded the EEO program at GSFC and what was the ultimate goal? Has the program lived up to expectations and what benefit is EEO to NASA/Goddard?

7. Assuming it is impossible to recreate Earth's gravitational pull in space, what is the likelihood of humans being born in space and living on Earth without physiological complications?

8. Being aware of the restructuring of NASA personnel but still being interested in a career involving aerospace, what would be a suggested path of study that would give me the best opportunity for NASA employment? Would I have a better chance if I seeked employment as a contractor?

9. I would like to know the opportunities to work in the area of telecommunications at NASA. Which is the best NASA center to work in that area?

10. Are all of the NASA facilities funded solely by the federal government or do outside corporations help to fund their labor force?

11. What are the reports you get from mentors about summer interns and/or co-op students?

12. What do you see NASA doing in the area of space exploration in the year 2000?

13. There are large budget cuts and new legislation being passed daily by the U.S. government. How will this affect the objectives of NASA/Goddard and the employment of minorities in science, engineering, and mathematics?

14. How has the efficiency of the various space flight missions here at Goddard increase over the past 10 years?

Questions for Mr. Rothenberg from summer interns - July 7, 1995

1. With all of the recent attention surrounding affirmative action, what efforts is Goddard making to increase the number of minority professional employees?

2. How will the budget problems that the agency is facing impact the Goddard EEO summer programs?

3. What projects are of the greatest importance at this point in time to the Goddard community and ideally, what would be the desired outcome of each?

4. Are there any computer science divisions at Goddard that solely perform computer science research? If not, why aren't there any?

5. What is full-cost accounting? What impact does full-cost accounting have on NASA, Goddard, and their employees?

6. When NASA put the first man on the moon, there seemed to be resounding public support behind a common goal - namely, beating the Russians. Do you think that NASA's budget would be under fire now if there was a unified public push similar to that in the 60's?

7. It is safe to say that most American people have very little idea of what NASA does outside of launching manned space vehicles every so often. Would it benefit NASA to "advertise" some of the projects that are taking place that affect and improve life? Has there ever been an attempt to do so?

8. With all the budget cuts and layoffs at NASA and within corporate America, what is the ratio of cuts/layoffs to hires and what is a student's chance of full-time employment in years to come?

9. If you could do your career again, would you go into private industry?

10. Is NASA still planning a manned launch to Mars, and if so, are there plans to design a launch vehicle (such as the Saturn 5) to transport the astronauts there?

PLEASE RETURN THIS FORM TO THE SIECA COORDINATOR (Code 120) BY: ____________

Summer Institute in Engineering & Computer Applications (SIECA)
SIECA Student Self Evaluation
Bowie State University, Bowie, Maryland
NASA's Goddard Space Flight Center

NAME: ____________________ DATE: ____________ MAJOR: ____________

POSITION TITLE: ____________ SEMESTER: ____________ YEAR: ________

CODE/BRANCH: ____________________ DIVISION: ____________

Briefly list the major duties you performed during your SIECA experience:

____

____

____

YOUR SIECA WORK EXPERIENCE

RATING SCALE: 1-needs improvement; 2-average; 3-good; 4-excellent; 5-N/A

Comments are very helpful to us. Please try to give some specific remarks that will support your rating.

	Rating (1-5)	Comments
SIECA COORDINATOR: answered my questions, informed me well, helped me to deal with my concerns and helped to guide me during my internship.		
ORIENTATIONS: at work, I received a complete orientation.		
JOB DUTIES: were clearly defined.		
MY MENTOR: was available to discuss questions or problems		
MY MENTOR: welcomed my ideas and comments, gave feedback and information.		

Check the appropriate statements which accurately describe your performance.

INTERPERSONAL SKILLS

___Worked very poorly with others
___Had some difficulty working with others
___Got along satisfactorily with others
___Worked well with others
___Worked exceptionally well with others

QUANTITY OF WORK

___Low out-put slow
___Below average
___Normal amount
___More than average
___Unusually high out-put

DEPENDABILITY

___Unreliable
___Sometimes neglectful or careless
___Usually dependable
___Above average in dependability
___Completely dependable

ABILITY TO LEARN

___Very slow to learn
___Rather slow in learning
___Average in understanding work
___Learned work readily
___Learned work exceptionally well

MATURITY-POISE

___Quite poised and confident
___Had good self-assurance
___Average maturity and poise
___Seldom asserted myself
___Timid
___Brash

JUDGEMENT

___Consistently used poor judgement
___Often used poor judgement
___Usually made the right decision
___Above average in making decisions
___Exceptionally mature in judgement

COMMUNICATION SKILLS
Oral/Written

____/____Very poor
____/____Below average
____/____Average
____/____Very good
____/____Excellent

QUALITY OF WORK

___Very poor
___Below average
___Average
___Very good
___Excellent

INITIATIVE

___Must be induced to work
___Waits for instruction
___Did all assigned work
___Went ahead independently at times
___Proceeded well on my own

ATTITUDE-APPLICATION TO WORK

___Definitely not interested
___Somewhat indifferent
___Average in diligence and interest
___Very interested and industrious
___Outstanding in enthusiasm

ATTENTION IN REGARD TO TIME

___Always tardy
___Often tardy
___Sometimes tardy
___Usually punctual
___Always punctual

MISCELLANEOUS

___Showed patience and humor
___Adjusted well to new assignments
___Accepted feedback well
___Some difficulty accepting changes
___Was well prepared for internship

PLEASE TAKE A MOMENT TO ANSWER THE FOLLOWING:

1) This SIECA work experience made my courses at (university or college) __________________ more meaningful. PLEASE RATE: (YES) 4 3 2 1 (NO)

2) This SIECA work experience helped me decide to continue in my career choice/major. PLEASE RATE: (YES) 4 3 2 1 (NO)

3) This SIECA work experience convinced me to change my career choice/major.
PLEASE RATE: (YES) 4 3 2 1 (NO)

4) I worked harder and learned more because I received academic credit.
PLEASE RATE: (YES) 4 3 2 1 (NO)

5) What suggestions do you have for improving the SIECA process?

6) What would you say to other students about your SIECA experience? (We may use this for advertising.)

7) May we use other quotes from this report for advertising purposes?
YES______ NO______

PLEASE RETURN THIS FORM TO THE SIECA COORDINATOR BY:_________________

THANK YOU!!!!!

EVALUATION OF PERFORMANCE FOR STUDENT INTERNS

STUDENT NAME________________________ PROGRAM____

MENTOR NAME_________________________ TITLE_____________________________ CODE________

Check the appropriate statements which accurately describe the individual's performance.

RELATIONS WITH OTHERS

___Works very poorly with others
___Has some difficulty working with others
___Gets along satisfactorily with others
___Works well with others
___Exceptionally well accepted by others

INITIATIVE

___Must be pushed frequently
___Hesitates
___Does all assigned work
___Goes ahead independently at times
___Proceeds well on his/her own

DEPENDABILITY

___Unreliable
___Sometimes neglectful or careless
___Usually dependable
___Above average in dependability
___Completely dependable

QUANTITY OF WORK

___Low out-put slow
___Below average out-put
___Normal amount
___More than average
___Unusually high out-put

JUDGEMENT

___Consistently uses poor judgement
___Often uses poor judgement
___Usually makes the right decision
___Above average in making decisions
___Exceptionally mature in judgement

ABILITY TO LEARN

___Very slow to learn
___Rather slow in learning
___Average in understanding work
___Learns work readily
___Learns work exceptionally well

ATTITUDE - APPLICATION TO WORK

___Definitely not interested
___Somewhat indifferent
___Average in diligence and interest
___Very interested and industrious
___Outstanding in enthusiasm

MATURITY - POISE

___Quite poised and confident
___Has good self-assurance
___Average maturity and poise
___Seldom asserts himself/herself
___Timid
___Brash

QUALITY OF WORK

___Very poor
___Below average
___Average
___Very good
___Excellent

COMMUNICATION SKILLS

___Very poor
___Below average
___Average
___Very good
___Excellent

ATTENTION IN REGARD TO TIME

___Always tardy
___Often tardy
___Sometimes tardy
___Usually punctual
___Always punctual

Recommend Appointment next year Yes( ) No( )

Would you be willing to serve as a future mentor? Yes( ) No( )

COMMENTS____

____

Signature (Supervisor)_______________________________ Date__________________

PLEASE RETURN THIS FORM TO J.S. LANGDON, CODE 120

SUMMER INSTITUTE IN ENGINEERING AND COMPUTER APPLICATIONS
Sponsored by Bowie State University
and Goddard Space Flight Center

REQUEST FOR TRAVEL REIMBURSEMENT

NAME____ DATE:_________

Social Security No.________________________________

Please reimburse the following amount which covers travel expenses from _____ ____ to Goddard Space Flight Center and return.

Amount: _____________

Method of Travel: ___Plane

___Train

___Car (Miles traveled round trip ___________)
x mileage rate of _______________

___Other (Explain)_____________________________
____

Additional Expenses:

___Local/Ground Transportation

Please check if applicable:

___Receipts attached

____ ____________
Payee's Signature Date

____ ____________
(Approval) Signature Date

XX

1995 SIECA STUDENT ABSTRACTS

Skylab X-ray Images Made Readily Accessible

Skylab, the first long range orbital observatory, took over 12,000 exposures of the sun between May 1973 and February 1974. This was accomplished by using nine telescopes, each uniquely designed to capture images of the sun within pre-designated wavelengths. Two of these telescopes, the S-054 (wavelengths 2 to 60 A) and S-056 (wavelengths 6 to 33 A) provided the X-ray images. My project revolved around the images taken with the S-054. The objective of my project was to make these images readily accessible to the public though gif. files, internet, and CD ROM.

Prior to my arrival, David Batchelor (my mentor) had made it possible to view these images on PC screens. This required a complex sequence of case sensitive UNIX and IDL commands to be manually implemented.

I wrote a UNIX program, using Shell Script, that executed the sequence of UNIX commands upon typing only its file name at the UNIX prompt. Next, I wrote a IDL program, using the IDL buffer, that executed the sequence of IDL commands upon typing only its file name at the IDL prompt. Then I nested the UNIX program as a function of the IDL program, resulting in the display of an image upon typing only one word. This effectively eliminated syntax errors and saved valuable time in my future research.

At this stage, the images still could only be viewed and temporary gamma corrections be made. Therefore, I had to modify the IDL program to compile the current format into a gif file. This was a major undertaking and accomplishment.

At last, from EBCIDIC to gif, the Skylab images taken by the S-054 telescope are readily accessible and will soon be on CD ROM.

Edwin Beckford

July 14, 1995

Range Related Affects of Radar Estimated Rainfall

Student Researcher:	Mentor:
Cedric Blair	Brad Ferrier
Junior, Meteorology Major	Mesoscale Atmosphere Branch
Jackson State University	NASA--Goddard Space Flight Ctr.
Jackson, MS	Greenbelt, MD

The Tropical Rainfall Measuring Mission is a satellite not being studied jointly by the U.S. and Japan. The study carries out a systematic approach to the tropical rainfall required for major strides in weather and climate research. The ultimate objective of the project is to estimate rainfall rates using the satellite.

Currently rain rates can be estimated by radar, but adjustments need to me made. If accurate measurements can be made by radar, then these systems can be used for the satellite. During 1992 and 1993 two cruises were made on the Pacific Ocean in the tropic region. Both ships had radar on them, referred to as the Toga and Mit radar. To determine the range dependent affects, the data sets chosen for analysis were those where the area coverage of the TOGA and MIT radars overlapped os that the areas of rainfall will be obserbed at different distances. The hope is tht alogarithms can be developed to more accurately estimate rainfall rates particularly when these areas are measured by the radar. As a result of looking at the maps, there appears to be a great deal of range dependent affects, with lover rainfall rates estimated for the same area but different distance locations from the radar.

ENHANCING THE USERFRIENDLINESS OF NSSDC'S MODELS ARCHIVE

Trena Covington

ABSTRACT

The National Space Science Data Center (NSSDC) maintains an archive of solar-terrestrial models. There are more than 70 model software packages which are distributed on diskette, CD-ROM, and through anonymous ftp. Several of the international standard models, such as the International Reference Ionosphere (IRI), the Mass-Spectrometer Incoherent Scatter (MSIS), and the International Geomagnetic Reference Field (IGRF), can also be run online in NSSDC's Online Data and Information Services (NODIS) account.

The goal of my project is to enhance the userfriendliness of the archive. The first step is to use file compression software to compress all files for a specific model into one self-extracting file. This will simplify the electronic transfer of these software packages.

Feedback from the user community has shown that one of the most desired capabilities is the graphical display of model parameters. The main part of my project is to focus on the development of callable Interactive Data Language (IDL) programs that would allow a user to display the model parameters that he or she had created. The IDL programs will read parameter files as produced by current model software and then plot parameters selected by the user.

Abstract :

Three main projects were proposed for this summer internship. The first one dealt with the balancing of the air conditioning system (ACS) of Building 18. Some occupants in that building were complaining about uncomfortable temperature inconsistencies, ranging from very cold to hot temperatures. Also an important task was determining the amount of outside fresh air (OFA) that enters the system. The OFA was compared with codes and professional suggestions to find if it was within a desirable range.

The second project dealt with fire pump analysis. There are four fire pumps at the Wallops Flight Facility that have not been tested to the National Fire Protection Association (NFPA) criteria since they were installed more than ten years ago. The goal in this task was to develop performance curves of the actual state of these pumps, compare the actual data to the original performance curves, and to determine if the actual state is acceptable or if further investigation is needed.

The last of these projects dealt with the replacement of the boilers at Building 24 (Power Plant). Due to the long operating time of the boilers (approximately 30 years) they have lost efficiency. Replacing them with new ones was the best option. The goal in this task was to check the boilers replacement specification book and compare the replacement procedures to the National Fire Codesfor incompatibilities.

Ricardo E. Díaz
Code 205.2

Use of 3D Computer Modeling in the Software Development Process

Francisco Fernandez

In the Flight Software Section here at Goddard, developers work to create reliable software tasks to make unmanned spacecraft capable of running autonomously. The development process involves long hours of research and develpment work on both the hardware and software components of the spacecraft. Each part is carefully built and/or programmed to meet its requirements. However, since each part is developed separately, testing becomes a difficult task. In particular, it is challenging to find ways to test software that must control a physical aspect of the spacecraft, such as attitude control or antenna management. Additionally, once the spacecraft is launched and in orbit, monitoring it may be difficult to visualize for even experienced ground controllers, who receive only numerical data from the spacecraft about its position, orientation, etc. One way to deal with these difficulties is to use 3D computer modelling to simulate a spacecraft.

My project for the summer has been to work on a 3D graphic simulation of the X-Ray Timing Explorer (XTE). In particular, my task has been to take an existing 3D computer model of XTE and add to it the capability to connect to a spacecraft ground control workstation via internet socket connections. With this capability, the model is able to obtain either simulated or actual information about XTE's position, orientation, antenna gimble angles, and so forth, and use this information to accurately reflect what is occuring with the spacecraft.

My final paper will be labeled:

GSE.BASE FOR THE XTE SATELLITE.

(GROUND SUPPORT EQUIPMENT DATABASE FOR THE XRAY TIMING EXPLORER SATELLITE.)

Stacy A. Flowe
7/13/95
S.I.E.C.A. Student
Bldg.2, Code 666
Ex. 6-3588

Stacy A. Flowe
7/11/95
S.I.E.C.A Student
Bldg. 2, Code 666

ABSTRACT

My intership with NASA for the summer of 1995 was spent working in bldg.2, under my mentor, Dr. Jean Swank. We worked out of code 666, Laboratory for High Energy Astrophysics. Dr. Swank is the project scientist for the Xray Timing Explorer Satellite. I was to assist her in creating a program that will run all test data received in the past three years, on the ground. The program will produce an output which will be a summary of the data. My final paper will discuss the issues I encountered on this project and read in the following order.

My summer at NASA began with several weeks of studying. I was to make use of the literature that was given to me, the on site library and the learning center. The first few days were spent becoming familiar with the sun computer and the various programs available on it.

After the basics were covered, I began comprehending the functions of the XTE satellite.

Finally, I began to work with the project's program. The program was based on astrophysics formulae and had to be done using Paw, a tool used by astrophysicists to write scientific formulas for programs. The structures that needed to be added, had to be done in the Fortan programming language. Not being familiar with this language, I spent several more days learning it.

While learning the language, Fortran, I added a chart format to a model program, that I constructed, similar to the layout that my mentor requested the original program conform to. After perfecting the model, I added the code to the original program. With my mentors approval of the chart code added, I began to work on adding histogram formulae that my mentor constructed.

The summary of data, when completed, will become an entry into a SQL data base, so scientists can easily find particular data they are interested in studying.

NASA
Goddard Space Flight Center
Greenbelt, Maryland

ABSTRACT

Roberto Gómez Báez
[redacted]
SIECA (UG)
Mentor: Dr. MacDowall
Code 695
July 14, 1995
Final Report Title: Pattern Recognition

Abstract

The project that I have been assigned to deals with the daily data received from the spacecraft Ulysses. Ulysses is a joint mission between the European Space Agency (ESA) and NASA. To understand the Sun's environment and its influence on the Earth, the Sun not only has to be studied around the ecliptic plane, which is the plane where the Earth and other planets of our solar system orbit around the Sun, but also around the solar poles. This is the goal of the Ulysses mission, the first spacecraft to navigate over the solar poles of the Sun.

Different types of observations are made aboard Ulysses, including those of the Unified Radio and Plasma Wave Investigation (URAP). This is an experiment to detect both distant radio emissions (via remote sensing), as well as locally generated plasma waves. URAP is the experiment taking place at NASA. Dr. Robert MacDowall is the principal investigator.

Dr. MacDowall developed a computer program to determined the plasma frequency from the data we receive from Ulysses. Scientists will use the plasma frequency number to determine the density which is a fundamental parameter. This program, coded in IDL, fails when certain conditions are present in the data. Some of these conditions are Jovian Emissions from Jupiter, Type III Solar Bursts, and changes in mode in the spacecraft itself. My job is to overcome this conditions in order to obtain accurate results from the daily data. To achieve this goal I had to study the program in detail to be able to determine weaknesses and implement new solutions to the problem. Several changes and additions have been implemented on a daily basis in the program greatly increasing the accuracy of the results; however, we are still working on other aspects of the program, such as the data we received from Ulysses, to make it more efficient, accurate, and precise.

ABSTRACT

Electromagnetic Compatibility is the ability of systems, subsystems, circuits, and components to function as designed, without malfunction or unacceptable degradation of performance due to electromagnetic interference (EMI), within their intended operational environment. Electromagnetic Interference (EMI) consists of any unwanted, spurious, conducted, and/or radiated signals of electrical origin that can cause unacceptable degradation of systems or equipment performance.

Two types of tests that are performed to test for incompatibility of systems are: Emissions and Susceptibility. Conducted and Radiated Emissions Tests are used to 1.) measure the levels of narrowband and broadband conducted emissions which may exist on the D.C. power lines and return leads of the test sample, and 2.) demonstrate that the levels of electric and magnetic field emissions do not exceed the specified limits from the test sample and interconnecting cable. Conducted and Radiated Susceptibility Tests are used to 1.) demonstrate the ability of test samples to survive a(n) conducted A.C. sinusoidal ripple appearing on the input power lines, and 2.) ensure that the test sample does not exhibit any degradation of performance, malfunction, or undesirable effects when immersed in an electric field.

With the use of an 8904A Multifunctional Synthesizer, a General Purpose Interface Bus (GPIB), and Hewlett Packard's Visual Engineering Environment (HP VEE) software package, automation of radiated susceptibility tests will facilitate the laborious, manual testing of test engineers. HP VEE, an iconic programming language, is optimized for instrument control to simplify test procedure setup. The objective is to implement HP VEE in the testing environment to provide the capabilities of easy collection, analysis and presentation of data in other programs for further processing.

MARVIN JACKSON

ANALYSIS OF GSPAR COMPLIANCE WITH ISO 9000

Presented By: Michelle A. Jones

Code 300 - Office of Flight Assurance

Mentor: Boyd Pearson, Assistant Chief
Data Systems Assurance

ABSTRACT

The International Organization for Standardization , commonly known as ISO, is a worldwide federation of national standards bodies established to provide guidance for quality management and models for quality assurance. By December 1994, eighty countries formally adopted ISO 9000 with over 70,000 registrations worldwide. ISO 9000 has become such a vital tool in quality management, especially to the customer because ISO 9000 registration means an International Standard for Quality has been defined and measured, a method that helps customers choose between competitors' service offerings has been established, and a 'Freedom from fear' attitude that the customer's service provider will be 'inconsistent' in performance has been created.

The Guidelines for Ground Systems Performance Assurance Requirements, or GSPAR, is designed to address hardware and software assurance requirements for ground systems. This manual represents a baseline picture that can be tailored to form the requirements for an assurance program for any Goddard Space Flight Center (GSFC) ground system project. However, the GSPAR can only be used at GSFC.

The purpose of this project is to determine whether or not the GSPAR complies with the standards presented in the ISO 9000 manual? In examining the GSPAR with the ISO 9001 and ISO 9000-3 (for software), the focus will be to determine what requirements of the GSPAR, if any, need to be modified in order to comply with the standards of the ISO 9000 family.

NAME: NNAEMEKA H. NWOSU
MENTOR: APRILLE ERICSSON-JACKSON
CODE: 712.3
PROJECT: <u>THE EFFECT OF ARRAY MOTION ON TRMM</u>

ABSTRACT

Most satellites require solar arrays to power them during their lifetime in orbit. Solar arrays are positioned in such a way that the satellite gets the most energy from the sun. In the case of the Tropical Rain forest Measuring mission (TRMM) satellite, its solar panels have to be re-positioned as the satellite emerges from the dark phase to sunlight phase.

During the dark phase, the solar panels are in a "feathered" position, in order to minimize the effect of drag on the satelite. As TRMM emerges from this phase, its solar panels begin tracking and positioning itself to the correct angular position, so as to receive maximum solar power from the sun. The re-positioning of the solar panels to the desired position requires the use of Harmonic drives (H.d.) and Step motors. As in any mechanical system, vibrations arise from the movement of the Step motor and H.d. .

In this paper, the vibration of these flexible solar array appendages (magnified by the Step motor and H.d.) on the entire satellite are examined. Using a FORTRAN based software called TREETOPS, the TRMM satellite can be modeled and simulated as in real life. The main FORTRAN program in TREETOPS is first modified to accept various cases of

angular displacements and angular rate of displacements for the solar panel. An input file containing the exact specifications of the different components of the satellite, is made. Incorporation of all of the above generates data by running TREETOPS. These results are plotted using MATLAB. The effect of the vibration can then be studied be analyzing the plots.

THE DESIGN AND IMPLEMENTAION OF THE CCSDS SIMULATOR CHIP

(ABSTRACT)

Miguel Angel Ordaz

The University of Texas at El Paso

Summer Institute in Engineering

and Computer Applications

Summer 1995

Code 521.2

This summer I have been working with a project that has introduced me to the design concept used in the Systems Applications Section of the Microelectronic Systems Branch of the Data Systems Technology Division, also known as Code 521.2. My summer project entails the design and implementation of the CCSDS Simulator Chip. I am designing the chip generically so that it may be used in a variety of systems and perform multiple functions. The CCSDS Simulator Chip will be capable of generating instrument packets given the packet header. These packets will then be formatted into frames as recommended by the Consultative Committee for Space Data Systems(CCSDS). This project intends to reduce both the complexity and size of current telemetry simulator technology.

I have been using the state-of-the-art Cadence Software on the Sun workstations in the VLSI Design Laboratory to design and simulate the logic of the project. The first two weeks were mostly a tutorial in which I used my time to get acquainted with the Cadence Software tools, meeting with my mentor to discuss the project and reading excerpts suggested by him on digital theory and Actel technology that I would use in the design and implementation of the CCSDS Simulator Chip. Since then I have worked extensively in the VLSI Design Laboratory designing and simulating circuits that will eventually become the CCSDS Simulator Chip.

My project has also involved learning about the design process used in Systems Applications Section which includes more than design and simulation. I have attended design reviews that provide the designing engineer with constructive criticism on a project from other engineers in the section. During this summer I have experienced, through the different faces of my project, technical education in a professional environment.

Patsy Polston
Tuskegee University
SIECA-UG
Goddard Space Flight Center - Code 713

ABSTRACT

We worked on the testing of a high resolution Penetration Depth Thermometer (PDT). The PDT is based on the temperature dependence of the magnetic penetration in a superconductor. The active element in the sensor of the PDT is a thin film of a superconductor. Our goal for the development of the PDT was to achieve a transition temperature of 2.2° K. We hoped to accomplish this by depositing a thin film of aluminum onto a substrate. As the film that was deposited and tested decreased in thickness, the transition temperature increased. Hence, we searched for the film thickness whose transition temperature was what was desired. The test procedures and results will be presented.

<u>Abstract Report</u>

<u>By: Marcellus Proctor (SIECA)</u>

The title of my final technical report is , "**<u>My experiences with the Spartan 206 Spacecraft Project</u>**".

For the past six weeks, I have been working with the completion of the Spartan 206 Spacecraft which is scheduled to launch in November 1995. I have been mainly helping out with testing the spacecrafts onboard systems and drawing schematics of different test circuit boxes.

I started working with the Spartan 206 Spacecraft last summer. One of the jobs I had to do was to build and put together harness for the interior of the spacecraft. A harness is a bunch of wires joined by a special conector which is the connected to the onboard systems boxes. The harness is basically the communication system of the spacecraft. All the differnet experiments and housekeeping equipment inside the spacecraft talk to each other through the harness.

For the first two weeks of my internship I helped out with the testing of the spacecrafts onboard housekeeping equipment which were done in Building 7. I had the opportunity to go inside the meduim size cleanroom in Building 7 to help with testing. Then after that I was resposible for checking scematics designs of differnet test boxes used on the Spartan 206 Spacecraft.

Demetrius Shaffer
SIECA-UG

My Summer Contribution to Goddard Space Flight Center

Computer networking plays an important role in today's rapidly growing technological advances. It is critical in the transferring, accessing, and receiving of information via different computers across the world. The Goddard Space Flight Center(GSFC) Center Network Environment(CNE) is the center wide computer network. Each computer on site (and some off site) is registered into the network and given a specific Internet Protocol address. Because of the massive number of connected computers, plenty of information is stored in numerous different databases to keep track of each computer, which is the only source for troubleshooting any problems that go on in the network. My job this summer was to update the data stored in the CNE database(CNEDB), the official GSFC database of all computers in the network. This clean-up was essential to maintain the network more efficiently. My work consisted of noting duplicate names, correcting misspelled names and e-mail addresses, updating old e-mail addresses, and correcting any obvious errors in the data.

Another project I had for the summer dealt with the CNE document, or pages, listed on the Internet. The CNE already had different sites that could be viewed on the Internet. My job was to create another site on the Internet which would give a listing of the top features of the CNE and link any user to those documents. This project consisted of learning the language used to write web documents, hypertext markup language, better known as HTML. In my paper, I will discuss more in detail of the function of the CNE, each of my projects at GSFC, and how each of my projects related to the role of the CNE.

PLASMA SIMULATION PROGRAM

Danielle M. Whipp
Summer Project
Code 632
Bldg. 26, Room G1
July 5, 1995

ABSTRACT

This summer I was assigned to work for Mona Kessel in the Space Physics Data Department (CODE 632). My project for the summer involves taking a FORTRAN simulation program, which at this point in time contains two plasma detectors, and add in the new Hawkeye LEPEDEA detector. A period of time had passed since I last used FORTRAN; therefore before doing anything I had to get re familiarized with the syntax. After refreshing my skills I began to tackle the code. My first step was to see what the old code did. The old code allowed the user to choose from two types of detectors, AMPTE and CLUSTER. Both detector's main job is to calculate density, velocity, and temperature of plasma. Next, Mona stated that she would like to update, rearrange, and shorten the old code. After that, I took data sets and tested out the old detectors to make sure they were still running correctly. Now that I have completed that part, I am starting to work on implementing the LEPEDEA detector. The steps for doing that is as follows: do background research on the old detectors and LEPEDEA, go through the code which includes the calculations for LEPEDEA, then meet with my mentor so she can guide me in programming the code for LEPEDEA.

NATIONAL AREONAUTIC AND SPACE ADMINISTRATION
SUMMER INSTITUTE IN ENGINEERING
AND COMPUTER APPLICATIONS
GODDARD SPACE FLIGHT CENTER

HUBBLE SPACE TELESCOPE (HST)

VEHICLE ELECTRICAL SYSTEM TEST (VEST)

CODE #442

ARTURO YANEZ NAVARRETE
SIECA-UG STUDENT
[redacted]

JULY 14, 1995

ABSTRACT

The Hubble Space Telescope (HST) Vehicle Electrical System Test (VEST) Facility provides test support and maintenance of the HST during its long term mission. The VEST Facility is in building #29 at Goddard Space Flight Center (GSFC) under the direction of Code 442. VEST operators and Integration & Test Engineers are preparing the facilities for the Second Service Mission programmed to begin on September 1995 up to March 1997. Their work consists in flight software and hardware verification testing.

As part of the Summer Institute in Engineering and Computer Applications (SIECA) I was working with the crew on (1) understanding how the VEST Ground System functioned in reference to the HST on-orbit, (2) NSI's responsabilities maintaining electrical struction in clean room, (3) Science Instrument Team to identify floor space for the new equipment and (4) assisting VEST operators Technicians in the Clean Room to set-up test equipment as the installation of the Flight Spare DF-224 and the Rate Gyro Assembly (RGA).

Abstract

Programming for the GLAS and the 1.2m Telescope
by
Gilbert Castillo

My projects this summer involved work on Satellite Laser Ranging (SLR) and the GLAS (Geoscience Laser Altimeter System). SLR provides data to determine earth's gravity field, crustal motion, polar motion and more. GLAS is a future project whereby a laser will be placed on board an orbiting satellite in order to constantly monitor the earth's changing surface.

At NASA's 1.2m telescope site at the Goddard Space Flight Center, laser ranging data is gathered and stored for experiments in atmospheric modelling, satellite spin determination, relativity and others. The data includes information such as the day and time of the laser firing, the vital round-trip time of flight of the laser, the telescope's pointing angles and other system information. In 1983, an international format standard for full rate SLR (Satellite Laser Ranging) data was made. This standard was named MERIT (Monitoring of Earth Rotation and Intercomparison of Techniques) and has been the operational format for the global community since then. The 1.2m telescope, however, produced its data in another internal format, making the data unaccessible to the global community. My project was to convert the 1.2m telescope's data from the internal format to the 130 byte MERIT II format. One of the first users of the 1.2m telescope data in this format will be Dr. Alley of the University of Maryland who will look at relativistic effects in the Russian GLONASS satellite's orbit.

I have also continued on the project I worked on last summer - GLAS (Geoscience Laser Altimeter System). My project involves the terrain generation code for a 3-D version of a simulator for GLAS. This code uses AOL (Airborne Oceanographic Lidar) ice data and attempts to regenerate the ice surface as closely as possible. This regenerated surface will be read by the simulator's space-to-time conversion routine to generate the system's response to a 3-D terrain. Tests will then be run to determine what hardware configurations will provide the most accurate data on the modeled surfaces. I will also do some preliminary testing of the 3-D space-to-time conversion routine on the simulator.

Network Management for the EOS Backbone Network (Ebnet)

Shahar Harel

SIECA-Graduate Student

Mail Code 541.3

Mentor: Veshal Desai

7/14/95

The Mission to Planet Earth (MTPE) is a far ranging project established by NASA in order to study the planet as an integrated system of atmosphere, oceans and continents interacting through energy exchange. The Earth Observing System (EOS) includes a constellation of satellites that will collect the pertinent data which scientific investigators will use in their research. Providing and maintaining a reliable network on the ground is an essential component of this mission.

The current design uses a distributed, open systems architecture. The data is sent from a group of satellites, known as the Tracking and Data Relay Satellite System (TDRSS), to White Sands Complex in New Mexico and then transmitted to the EOS Data and Operation Systems (EDOS) at Fairmont, West Virginia for further processing. The ECOM network transmits the data from the West Virginia site to nine different Distributed Active Archive Centers (DAACs) for storage, including one at Goddard Space Flight Center. It is also responsible for transmitting data from White Sands Complex directly to these sites in real time. The EOS Science Network (ESN) provides communications between the different DAAC's.

This design is being upgraded and simplified by the EOS Backbone Network (EBnet) which will consolidate both ECOM and ESN into one network. While this change is taking place the network must be continually managed to accommodate operations, simulations, and testing. The Simple Network Management Protocol (SNMP) is ideally suited for this purpose. Hewlett Packard's Network Node Manager implements this protocol and was used in conjuction with Remedy's Action Request System (ARS) Trouble Ticketing software to monitor and repair the network.

A Study Of The Effect Of The Diurnal Cycle On Weather Activity In The Pacific Ocean

Prem Lall / SIECA Program

ABSTRACT

During my last internship at Goddard, I created a series of computer programs that analyzed radar data to compute parameters for various meteorological experiments. Since that time, my programs have been used to evaluate data collected in the Toga Coare IOP (Intensive Operation Period). The resulting information was then used to calculate precipitation rates in certain regions of the Pacific Ocean.

This year, my project involves using these figures to gather information about the causes of rain activity in tropical climates. Specifically, it is my objective to determine the relationship between rainfall rates and the Diurnal Cycle . This cycle is an ongoing process in nature that is driven by subtle changes in atmospheric temperature and pressure. However, the effect that the Diurnal Cycle has on rainfall can be difficult to isolate, since many other factors can influence the weather.

In order to determine the nature of this connection, I first needed to see if there was a relationship between rain rates and temperature variations in the surrounding environment. By examining changes in sea-surface temperature readings, I hoped to establish a correlation between variations in the Diurnal Cycle and periods of heavy rainfall. Since our original data was gathered during a series of periodic scans, I was able to generate a sequence of probability distributions. These were then combined to create a graphical representation of the rainfall activity during a four month period. Similar graphs were also created for the sea-surface temperature, to see if any sort of correspondence could be detected. I also tried to see if I could observe any kind reoccurring rainfall patterns over time. In this way, I hoped to uncover some empirical evidence that would help us establish a link between the Diurnal Cycle and precipitation rates.

ABSTRACT

BOREAS Project:

C Programming Style and Maintenance

Dorothy Langendorf

SIECA Program

Goddard Space Flight Center

Hughes STX

11 July 1995

Hughes STX, a contractor with Goddard Space Flight Center (Code 923), has been tasked with the design and execution of the BOREAS Field Experiment. BOREAS is a study of the Boreal forest at two primary sites in central Canada.

Extensive data is collected at the test sites It is then categorized and loaded into a database. The database used is Oracle version 7. This data is made available to scientists by means of the World Wide Web and through CD-ROM production.

Several computer programs have been written to check and implement various aspects of the data handling. Programming style of some of the programs makes them difficult to read and understand. Another concern is running the programs on the newly-installed Oracle version 7 (replacing version 6).

One such program concerns quality assurance of the data; another, downloading data for CD-ROM production. My assignment for the summer is to insure that both programs run on the Oracle Version 7 database. Once that is accomplished, I should enforce a proper programming style in order to make the programs more easily understood and maintainable. Both programs are written in C and use ProC (embedded SQL commands).

SAT: System Administration Tool

Tony Miller

1.0 Abstract

Maintaining large numbers of UNIX workstations requires much attention to detail. With complex installations from multiple manufacturers, the stock administration tools which ship with the operating systems are insufficient for handling even simple tasks. The common duties of adding, deleting, and manipulating user information become a matter of hand editing configuration files on several different machines.

My project involves the creation of an administration tool which will provide a single repository for information concerning all users and machines operated by the Laboratory for Terrestrial Physics, Code 920.2. The user interface, the part that is seen and used for interaction, utilizes the World Wide Web. Through the use of HTML files and PERL scripts, any networked machine which can execute the latest version of the Netscape WWW browser can modify system files. This means that even a PC or Mac on the network can aid in administrative duties. Preserving the security of such a system is vital to preventing unauthorized individuals from gaining access. SAT utilizes security features of the Web server run by the LTPCF to ensure that only people with proper privileges can manipulate the system's database. Current functionality includes the ability to add, delete, and modify users. To allow for rapid modification of many users, Netscape tables are employed to present an entire database on one screen, with an option to modify any field value of any number of users at one time. Log files are kept to record an changes made to the database, thus providing a record of all interactions with the tool and the ability to restore previous information.

The tool currently works as a self contained system, but will provide a platform for expansion. Future plans include keeping track of all the equipment operated by the facility as well as related information such as the purchasing records and service contract data..

Alberto Rodríguez
University of Wisconsin-Madison
SIECA 1995

June 14, 1995

Abstract

The objective of this project is to develop a mathematical model to simulate the dynamic behavior of the gimbal in the Attitude Measurement System (AMS) of the Passive Aerodynamically Damped Satellite Experiment (PAMS). The purpose of the PAMS shuttle test flight is to demonstrate the feasibility of using aerodynamic stabilization and magnetic damping to passively stabilize small satellites. The data obtained from the PAMS experiment will be used to verify the analytical predictions made by Langley Research Center (LaRC). The PAMS experiment will be flown aboard the shuttle in the spring of 1996.

In order to accomplish this, a satellite test unit (STU) is going to be ejected from the shuttle bay and its trajectory tracked as it decays into the earth atmosphere. The STU has an array of corner cubes mounted on its face and its attitude is going to be measured by shooting a laser beam at it. The STU reflections are going to be read from the corner cubes using a charged coupled diode (CCD) array camera. To center the reflection of the STU into the field of view (FOV) of the CCD array camera, the reflected laser beam is first bounced off a gimbaled mirror and steered to the center of the camera FOV by adjusting the gimbal angles. The gimbal angles are adjusted by means of a feedback system that measures the position of the beam relative to the CCD FOV using a Quadrature Avalanche Photo Diode (QAPD). Using a computer control system the gimbal motors are positioned to null the system's position error.

The mathematical model of the AMS gimbal that I am developing includes the optics and QAPD, gimbal mechanics, control electronics and accounts for errors induced by the shuttle attitude control system. The designed mathematical model is being simulated using an object based programming tool called Simulink, designed by Matlab's Mathworks. The model parameters are being curve fitted and different scenarios are being simulated to find the system's best tracking performance.

Summer Internship Project Abstract

Kenneth Russell
July 14, 1995
SIECA - Graduate Student

North Carolina Agricultural And Technical State University

CODE 735.3
Mentor: Alan Cudmore

The Flight Software Section, 735.3, was in need of an application that would allow them to have remote access to Ground Support Equipment (GSE) telemetry via personal computer (pc) when designated workstations are not available or are in use. While the thought process behind this project was still going on, it was also realized that this application, if designed, could save travel time from home back to work when problems arose during simulations and other tests.

One of the main ideas of the project was portability. The ability to use any pc at home or work was appealing because it would not require a person to use a specific machine in a certain area. Any pc with network access would be available.

Once the hardware issue was sorted out, the matter of what type software to use was presented. Microsoft's Visual Basic was chosen because it was relatively inexpensive, accessible and it develops effective user interfaces. These interfaces or screens would have to be designed to enable a person with minimal computer experience to easily maneuver though the system.

In June of 1995, this idea became reality and the Spacecraft Interface Monitoring System (SIMS/95) project began. Once complete it would allow access to any telemetry providing a valid network connection could be established to the ASIST GSE workstation(s).

SIECA INTERN : RONTRILL SWAIN

DATE : JULY 14, 1995

NASA - GODDARD SPACE FLIGHT CENTER

* FINAL REPORT TITLE
* ABSTRACT

SOFTWARE ENGINEERING BRANCH

MENTOR : JON VALETT

CODE 552.3

The Final Report Title:

THE

USAGE OF SWINGBY

AND

ADVANCED VIDEO TECHNOLOGY

IN

ROUTE TO MAKING

A

QUICKTIME MOVIE

ABSTRACT

For the past few years, Swingby has been referred to as a tool for analyzing and creating missions within the Flight Dynamics Division. It's ability to describe the orbital movements of a spacecraft with the moon, sun, earth and other gravitational bodies has provided scientists with several ideas and areas to explore within the Software Engineering Arena. Futhermore, by using Swingby, these orbital movements can be modeled to illustrate how a spacecraft is transferred from a low earth parking orbit into geostationary orbit and how a particular technique can be used to send a spacecraft to the Earth - Sun libration point.

However, these functions only provide a small portion of the ways in which Swingby can be used. Relative to this study, Swingby was used to create 3 missions that provided reasonable images to be captured by advanced video application. These missions displayed different orbital movements of the spacecraft with respect to specific goals and variables of particular events.

Thus, this study is designed to illustrate the versatility of Swingby -- **A Mission Analysis and Design tool** -- in order to encourage more usage of the tool in the near future.

Lunar-Based Telescope Design

Ebony Alexis Waller
Code 684.1

I will be responsible for designing and constructing a model for the lunar-based telescope. The telescope will be made out of grapite epoxy and will have approximately a 14 inch primary mirror. The telescope will be about 11 inches long and 15 inches wide. I am using AutoSketch software to make my designs and hope to convert my drawings to AutoCAD.

LIX

1995 SIECA STUDENT PAPERS

Range Related Effects of Radar Estimated Rainfall

Student Researcher:
Cedric Blair
Junior, Meteorology Major
Jackson State University
Jackson, MS

Mentor:
Brad Ferrier Code 912
Mesoscale Atmospheric Branch
NASA-Goddard Space Flt. Ctr.
Greenbelt, MD

ABSTRACT

The Tropical Rainfall Measuring Mission is being studied Jointly by the US and Japan. The study carries out a systematic approach to the tropical rainfall required for major strides in weather and climate research. The ultimate objective of the project is to estimate rainfall rates using a satellite.

Currently rain rates can be estimated by radar, however adjustments need to be made. If accurate measurements can be made by radar, then these systems can be incorporated for the satellite. During 1992 and 1993 two cruises were made on the west Pacific Ocean near the equator. Both ships were equipped with radar, referred to as the TOGA and MIT radars. To determine the range dependent affects, the data sets chosen for analysis were those in which the area of coverage of the TOGAand MIT radar's overlapped. The hope is that algorithms will be developed to more accurately estimate any range related affects in the rainmaps developed from the radar scans. As a result of looking as the maps, there appears to be a great deal of range dependent affects, with lower rainfall rates estimated with increasing distance from the radar.

BACKGROUND

The tropical rainfall measuring mission (TRMM) incorporates two aspects of measuring rainfall, satellite and radar estimations. It was found that satellite estimates were a factor of two higher than that of the radar derived rainmaps. The hope is to bridge the gap through a correction scheme. While this is significant, agreement must be obtained between the two radar's first, particularly with data that is subjected to range dependent affects. The rain maps used are also being studied by oceanographic and meteorological teams. Rain maps used in this study were generated in code 910.1 of GSFC by the TRMM office.

GOALS

Our purpose is to identify range dependencies in rainfall maps obtained from the two ship radar's during TOGA-COARE (Tropical Ocean Global Atmosphere-Coupled Ocean & Atmospheric Response Experiment). By analyzing a common area between two radars a correction scheme will be developed and tested to bridge the discrepancies between the two radar's and hopefully between the radar and satellite data.

ANALYSIS

Figure 1 shows the location of the two ships. During this project there were three cruises taken during the period of November 1992 to February 1993. Cruise one is still being analyze. Cruise 2, the cruise we concentrated on, was done from December 20, 1992 to January 9, 1993. Cruise 3 was done from January 29, 1993 to February 18, 1993. The cruises were made over the equator in the far western pacific. The red dot signifies TOGA radar locations and the blue the MIT location. Figure 2 illustrates how the common area and the range of the two radar's. Both radar's have an effective range of 145 km the area in which the overlap of the radar's occurs is the common area (in green). Within the common area various cells will be studied and observed.

Figure 3 illustrates average rain rates for baseline radar distances between 137-147 kilometers. The averages are taken every 5 kilometers over a five by five grid. All graphs tend do show strong range dependent affects with TOGA rain rates and MIT rain rates dropping significantly with distance. These dependencies are illustrated in the MAX*/TOGA ratios. The ratios are near 1:1 for areas close to the TOGA radar, but increase with distance as rate estimates for the TOGA decrease and rate measurements for the MIT increase. The apparent range related affects are surprising, particularly when correction for attenuation and other range dependent factors were already incorporated. Based on the TOGA and MIT rain rate maps, a correction scheme was developed to reduce the presence of range dependencies and that is evident in figure four. The result after applying range dependent corrections for Cruise 2 data was encouraging. The TOGA and MIT rain maps both showed more symmetry, thus the correction reduced the range dependent affects. The TOGA/MIT ratio map indicates more scattering of the ratios. The MAX*/TOGA ratio indicated a larger area of 1:1 ratios, thus the correction led to a greater agreement of rainfall rates between the two radars. Although the MAX*/TOGA map was an improvement, range affects were still evident on the outer edges close to the MIT radar. However in general after the correction was applied range dependency fell dramatically.

In analyzing cruise 3 range dependent affects, the range affects were less noticeable, primarily due to a lack of rainfall in the field. With cruise three there were also higher levels of noise. Analysis on this cruise are still being conducted at the time of this paper.

RESULTS AND CONCLUSION

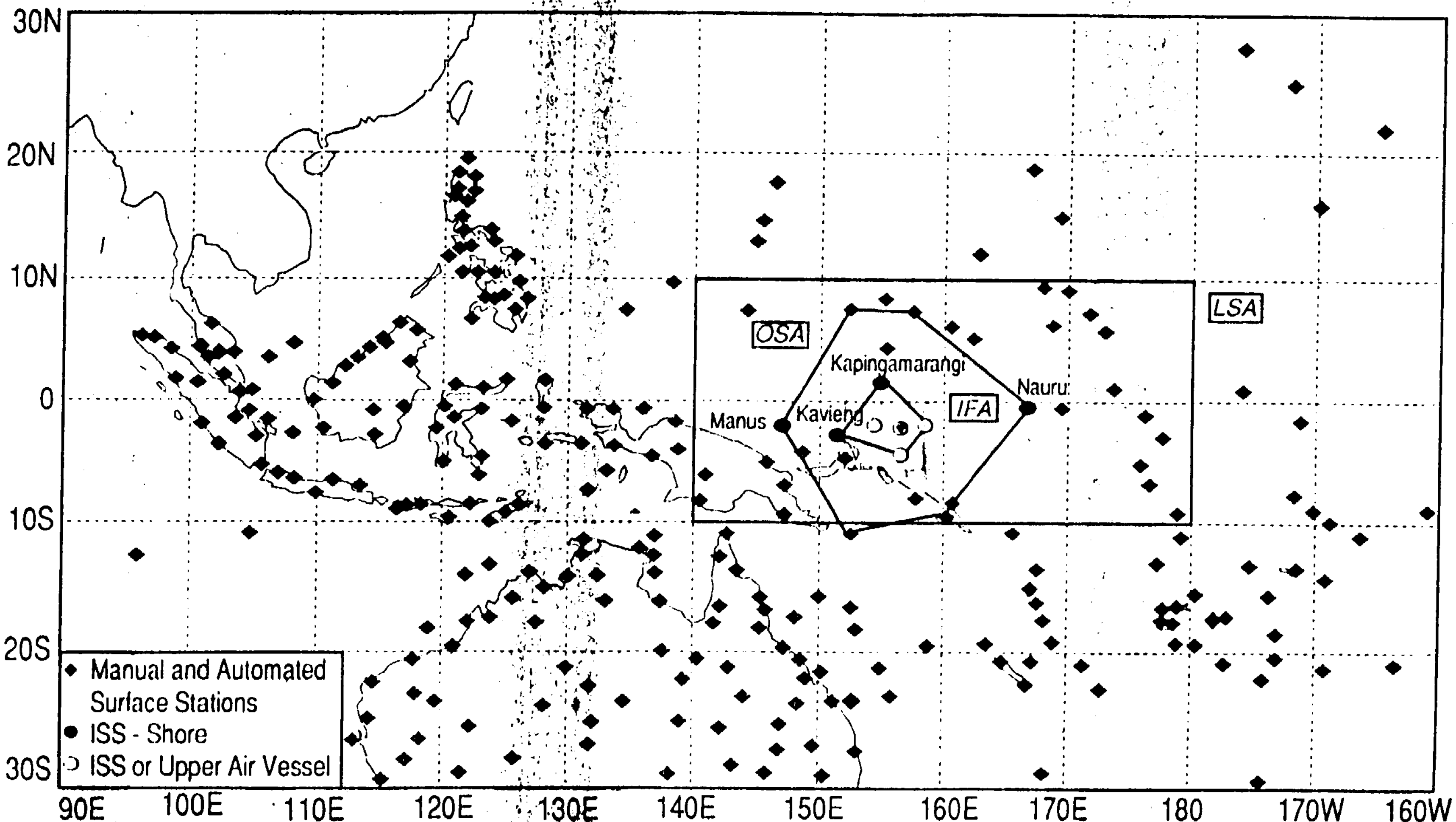
Rainfall maps generated by the TOGA and MIT radars were dramatically different, primarily because of range dependent affects. These differences are visible best in the COARE rain maps, particularly for the cruise two data. One of the possible explanations for the range dependencies are attenuation of the radar beam with range by rain. This would be a primary reason why range effects are less evident in cruise 3 because of lesser amounts of rain. Cruise 2 had about 50-80% more rainfall than cruise 3. RANGE DEPENDENT CORRECTIONS IMPROVED THE AGREEMENT BETWEEN TOGA & MIT RAINFALL MAPS.

FUTURE PLANS

The range correction based on this study are being incorporated in the 2nd version of the COARE rainfall maps.

Figure 1

Figure 5.3: ISS and Surface Meteorological Stations



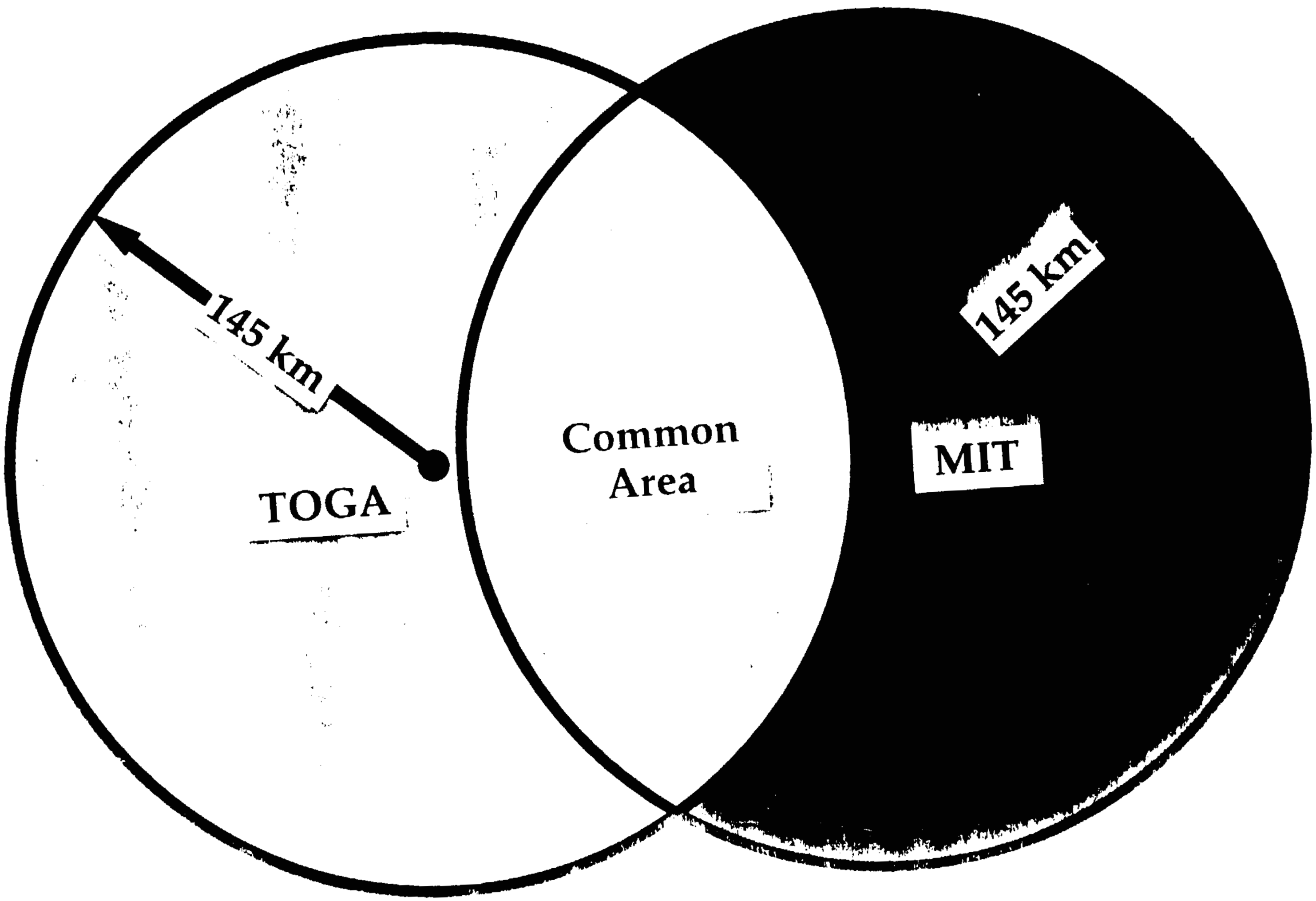


Figure 2

Figure 3

Radar Baseline Distances: 137 – 147 km

(5 x 5 km^2 area averages)

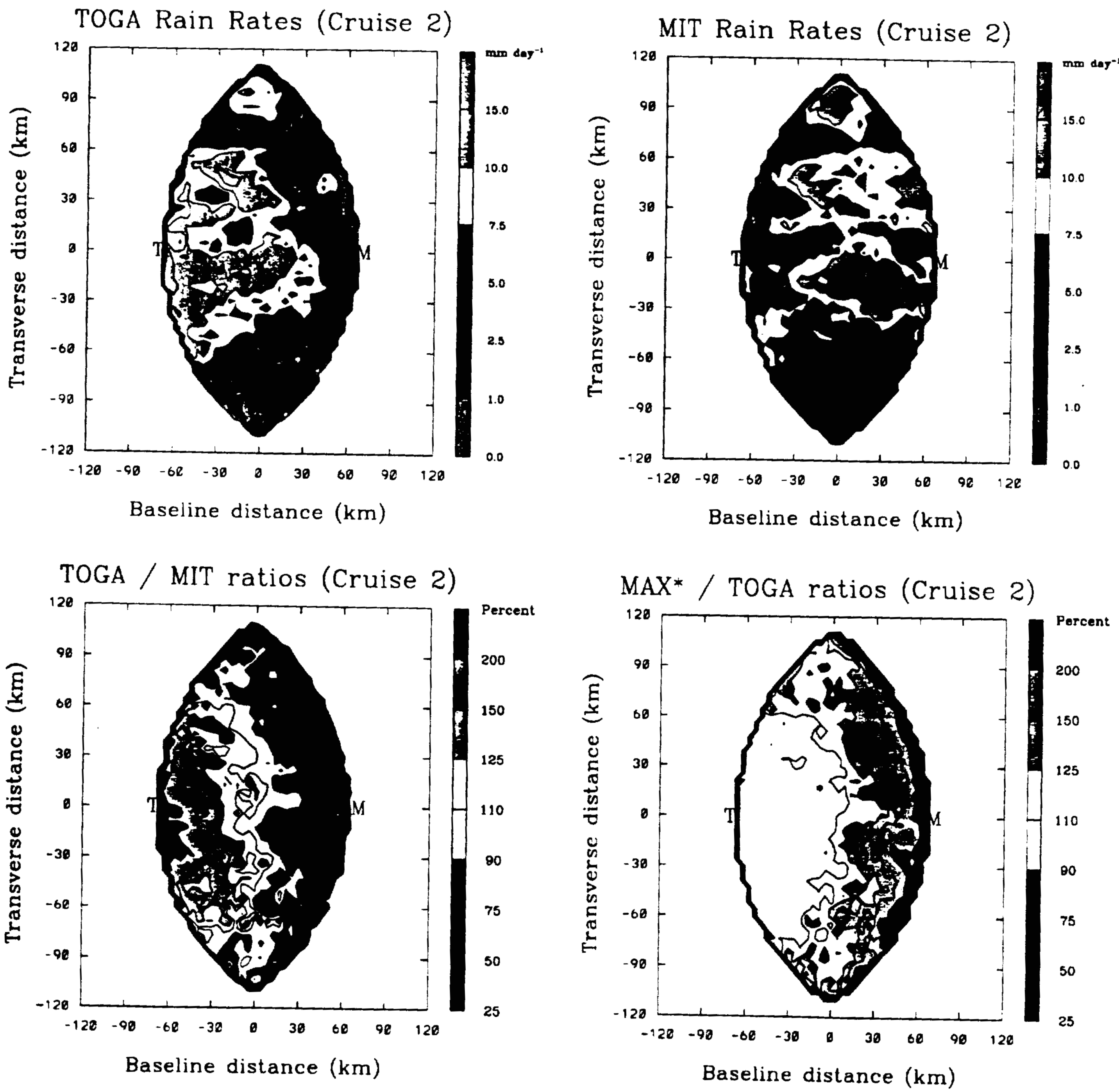
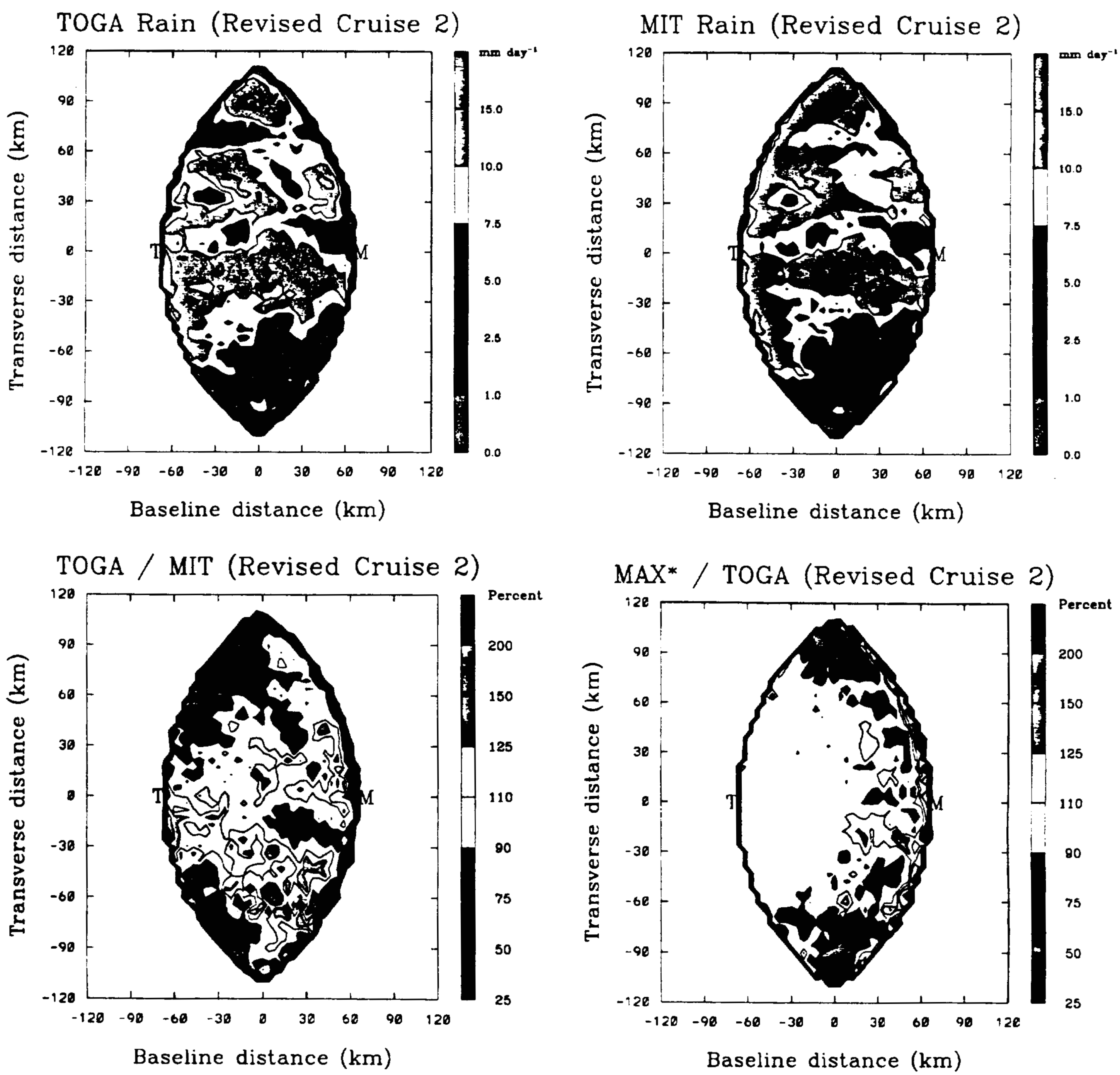


Figure 4

Results after applying range–dependent correction to Cruise 2 data



SUPPLYING OUR VEGETATED AREAS

BY: LOLITA T. CLAYBORN
SENIOR AT
CENTRAL STATE UNIVERSITY
DEPARTMENT OF
WATER RESOURCE MANAGEMENT

ADVISOR
DR. SUBRAMANIA I. SRITHARAN
EDUCATIOR AT
CENTRAL STATE UNIVERSITY

SIECA PROGRAM STUDENT
FOR THE SUMMER OF
1995

ABSTRACT:

This summer my project was to study the Simple Biosphere Model (SiB) for use within General Circulation Models (GCM) and then construct a program, that can be used along with the SiB and the GCM, that can accurately calculate, the percent of water that is needed in a specific agricultural area. To do this large task in such a short time I had to learn some other things to help me to understand the structure of each model. So before even learning about the model I had to take a class to learn FORTRAN programming. I also had to learn the computer operating system and its family of related utility programs Unix. Following the classes in FORTRAN I learned that there is another version of the SiB model and it is called SiB2 . This version is ran with several different FORTRAN compilers including Sun, Microsoft and Hp. The GCM versions of SiB2 use monthly vegetation index fields of FPAR (The fraction of PAR absorbed by the green canopy). This model was a lot easier to follow because a list of variables or parameters were made available. Inorder to run the model simply compile sib2.f and run in a directory that contains Data1, Data2, comsibc.h and pardif.h files. After tampering with the models and using the information that I had learned I was able to began trying to construct the program that I had set out to write.

SCOPE AND MAIN POINT OF PROJECT

The hydrological cycle shown in diagram 1 along with several other processes produces the Earth's climate, it serves also as a freshwater distillation system. It has two components: the first is associated with the movement of water and energy between oceans and continents due to the general, global circulation system of the atmosphere. The second is that water is being evaporated from the land surface and it is returned to it as rain or snowfall. The land generated hydrological cycle components, which are superimposed on the ocean continent advective cycle component.

Annually about 5.5×10^5 km^3 water is evaporated from the oceans and land surfaces. The energy necessary to convert this amount of water into vapor is approximately 36% of the solar radiation absorbed by the whole Earth. The evaporated water is transported by the general circulation of the atmosphere, which does included the vapors produced by transpiration. About 9% of the water evaporating from the oceans is transported by the global general circulation to the continent. As a global average it was estimated that only about 40% of the precipitation falling over continents returns to the oceans by river flow. The other part re-evaporates and falls back to the ground further downwind. On the average the water that is carried from the oceans is recycled and precipitated 2.7 times over land before it runs back to the ocean.

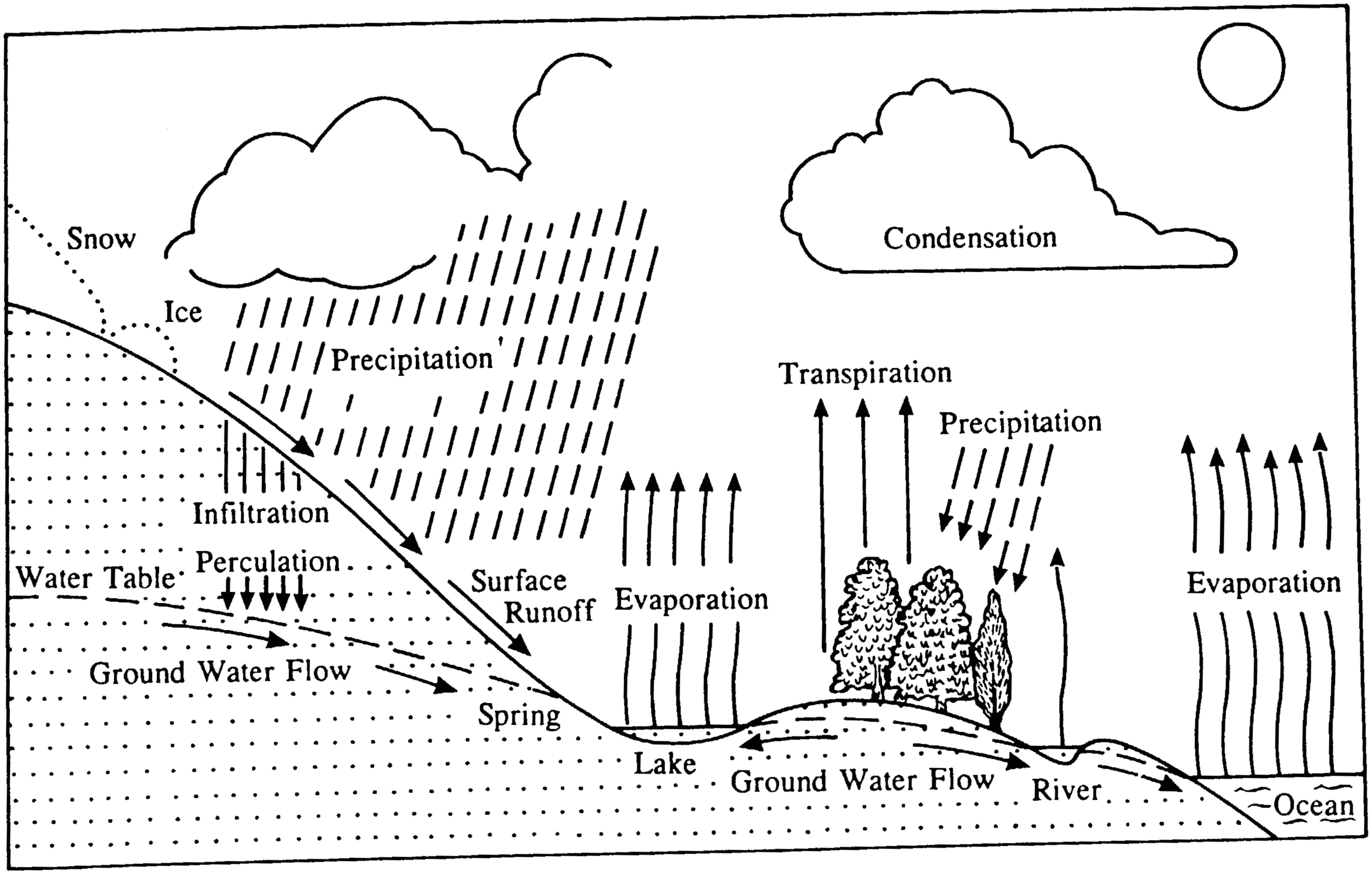
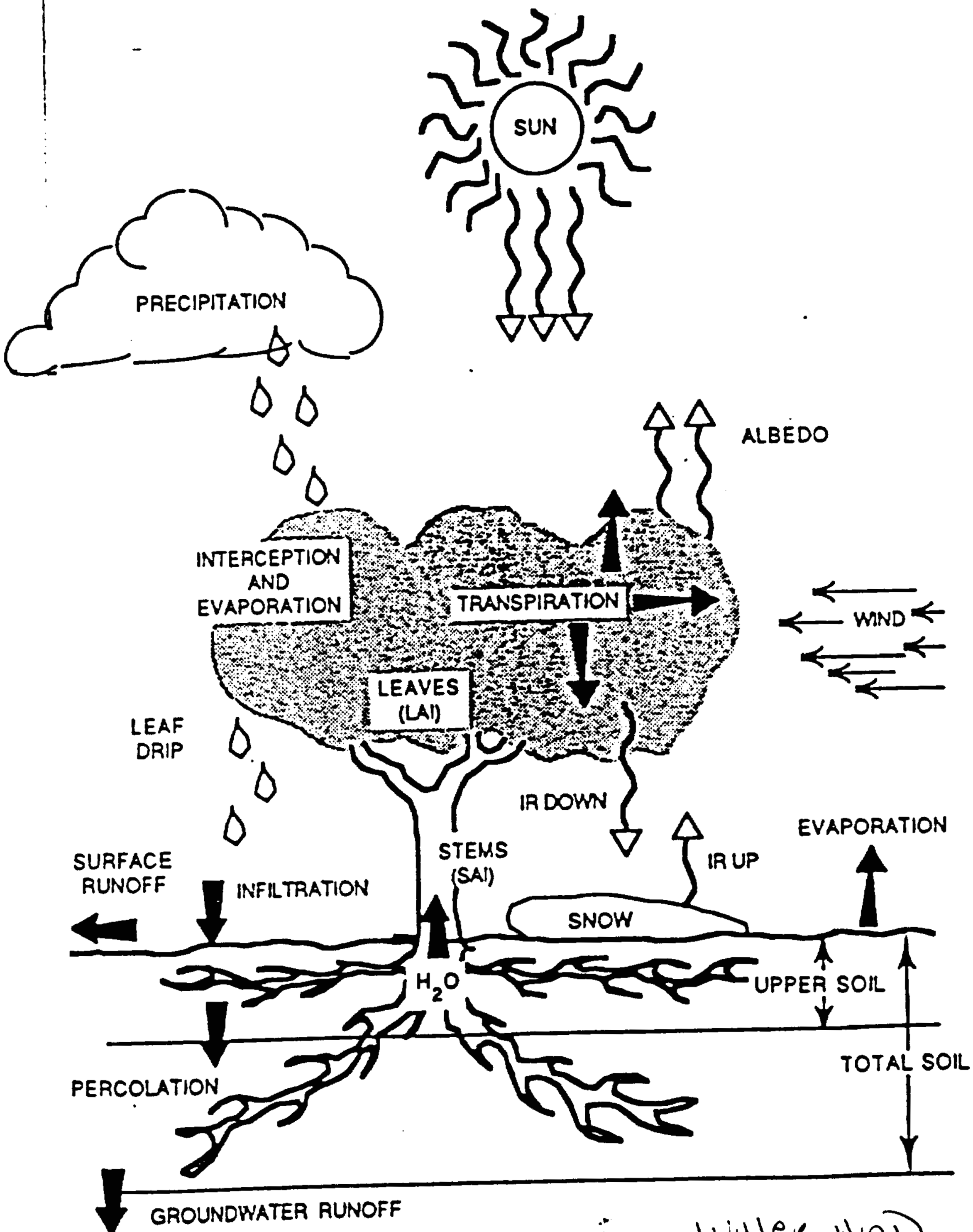


Fig. 1 Schematic representation of the hydrological cycle

HYDROLOGIC CYCLE



WATER AND
ENERGY EXCHANGE

Atmospheric processes change its precipitation to rain or snow, which supplies the vegetation, the soils and the subsurface aquifers with water. This import of water from the oceans is crucial for the existence of vegetation at the land surface. Without this transport the land would gradually dry out, exterminating life on its surface. This is one of the main reasons why we have interest in completing this project. To insure that if nature is incapable of supplying the vegetation with adequate amount of water our program can accurately determine how much water is needed so that it can be man delivered.

COMPONENTS INVOLVED:

The biosphere strongly interacts with the atmosphere by its controls on the return of water back to the atmosphere, as evaporation and transpiration. Therefore, changes in plant cover, due to either natural events or human activities, can have a significant impact on the hydrological cycle and on other components in the climatic system and on the Earth's vegetation.

Vegetation is the cover of plants in an area and can be characterized by different measures. The first is features, where the physiognomy of individual plants includes the attributes of leaf shape, growth form, phenology, and so on, and for assemblages of plants is more usually called vegetation structure. Vegetation structure has micrometeorological significance because it determines the magnitude and direction of the exchange of matter, momentum, and radiation between the earth's surface and the atmosphere (Running, 1986; Sellers and Dorman, 1987; Wilson, 1987; Taconet, 1986). Another characteristic of vegetation is its dynamics. Vegetation changes in space in response to climatic and landscape factors, and, at any one location, alter in time. These temporal changes include the rhythmic phenological changes of growth and flowering, as well as the irregular, episodic alterations of disturbance (Hobbs, 1987). Vegetation effluence's the exchange of energy, water, carbon and other substances at the land-surface, and atmosphere in many ways. Its color and structure affects the absorption of solar radiation. Leaves and branches increase the surface area for evaporation, they can intercept part of precipitation. Leaf litter can affect surface runoff and infiltration. The part of precipitation that is intercepted by the plants is more quickly re-evaporated. The water which reaches the soil might run off at the surface of infiltrate to deeper layers, where it replenishes soil moisture or recharges groundwater. Root growth and decay, and the decomposition of plant organic matter, by soil fauna and microbes, modify the soil texture and structure, affecting infiltration, percolation and drainage. The combined effects of these processes promote both water penetration and its return to the atmosphere. The presence of vegetation and the type of the plants determine to a large extent the partitioning of the water into different intermediate reservoirs (as shown in diagram 2).When evaporation occurs from the plants the biomass production due to photosynthesis causes water to move upwards from the soils through the roots and stems into the leaves. The intake of carbon dioxide through the stomata is accompanied by the loss of water. The heat consumed by the evaporation process cools the leaves and keeps the temperature within certain limits, which helps to maintain optimal conditions for biomass production. This latent heat returns a large amount of precipitated water to the atmosphere where it may later be released during condensation. The condensed water (clouds) is available for precipitation downwind. In addition to transpiration of plants, evaporation from soil has to be considered. It depends on the groundwater level, the structure of the soil in the unsaturated zone and the solar energy reaching the soil surface.

The proportion of precipitation falling as snow is important because, snowfall increases surface albedo. Water can be stored in snow for months, but may also leave the system within hours.

Research has indicated that vegetation strongly influences the amount and rate of water falling directly at the surface and also the possible runoff at the surface. If the vegetation is sparse, this water might erode large amounts of soil with its nutrients and organic detritus into rivers and into the ocean. During precipitation a few things had to be keep in mind, some of the factors that may affect radiation can always be a problem.

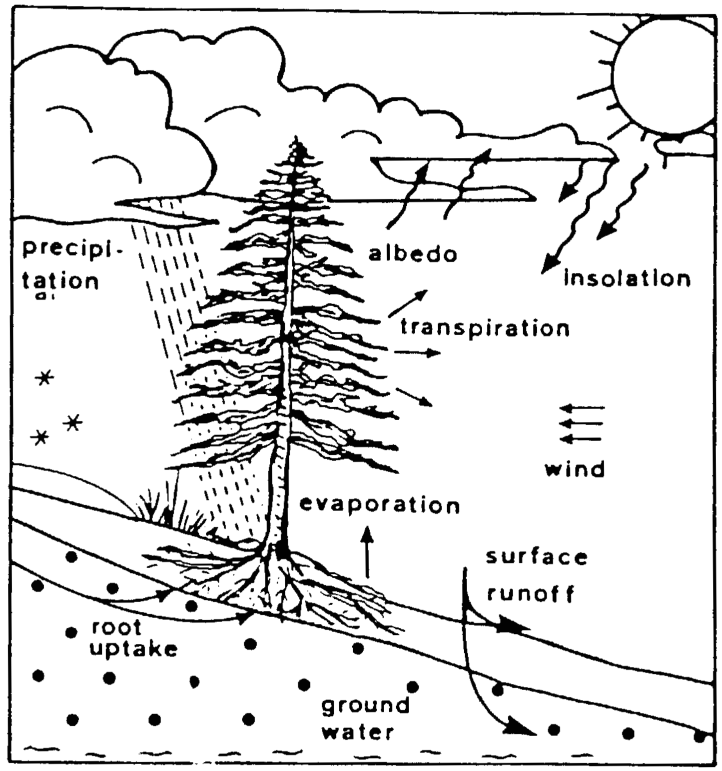


Fig. 2 Above and below-ground vegetation structure, soil properties, and other biological characteristics that strongly influence water and energy exchanges at the land surface

Several of the factors that may yield the quantity or the quality of reflected radiation from vegetation can be the way the leaves are angled on the vegetation and the location of a specific vegetation and so in this case the use of the (SiB) model and the (GCM) is greatly needed. Both of these models require a determination of the fluxes or radiation, water vapor, sensible heat momentum, whether used for numerical weather prediction or climate simulation. Because satellite remote sensing is the only data source that can be used to assess and monitor the vegetation areas accurately the models are very helpful because it is remote sensing factors that are implemented into the models.

THE SIMPLE BIOSPHERE MODEL (SiB) AND THE GENERAL CIRCULATION MODEL (GCM)

Land vegetation plays an important role in the cycles of the Earth. The transfer of CO_2 and H_2O between the soil and the atmosphere is now recognized as a key link between the biosphere and the atmosphere. The gas exchange between the Earth's surface and the atmosphere is determined to a considerable extent by the status of vegetation. This is the reasons why the simple biosphere model and the general circulation model was choose to complete the experiments because, it is not practical to measure these exchanges and all of our other components repeatedly and over large areas on the ground or from an aircraft platform, the availability of using the simple biosphere model and the general circulation model makes it a little more realistic in completing the project. Over the past decade it has become apparent that numerical weather models, from mesoscale to general circulation models, require a more accurate representation of the complex interactions occurring at the land surface. This is particularly important for hydrological and biophysical processes influencing the exchanges of heat, moisture, momentum, and trace gases through which the surface and atmosphere are coupled.

The simple biosphere model was developed for calculating the transfer of energy, mass and momentum between the atmosphere and the vegetated surface of the earth. The model is suitable to be used in the atmospheric general circulation model(GCM). The atmospheric GCM was developed at the NASA Goddard Laboratory for Atmospheres. It is a multi-level primitive equation model that uses a latitude-longitude coordinate in the horizontal and a standard sigma-coordinate in the vertical. Its prognostic variables are the two horizontal wind components, the potential temperature, and the water vapor mixing ratio (Koster, 1994). Scattering by clouds is computed using the delta-Eddington approximation(Koster, 1994).The model also includes a parametrization of large-scale condensation and a scheme to fill spurious negative water vapor produce by the dynamics.

A general circulation model (GCM) of the atmosphere provides a unique setting for the analysis of the atmospheric branch of the global hydrological cycle. The GCM's chief advantage is its ability to simulate moisture fluxes between all GCM reservoirs, allowing it to generate a complete set of global precipitation, evaporation, and the other hydrological data fields over most time scales of interest. GCMs, however, do have two important limitations in hydrological research. First, they cannot provide data at a scale smaller than the grid square, which typically covers an area of 400 x 400 km, this drastically limits their use for studying many hydrological processes at the land surface, such as the effect of storm structure on basin-scale runoff production. Second, and more important, GCMs are notorious for their inability to reproduce certain climatic features (local precipitation rates), which calls into question the hydrological insights they provide(Koster, 1994).

In SiB, the world's vegetation is divided into two morphological groups, trees or shrubs, which constitute the upper story canopy vegetation, and the ground cover, which consists of grasses and other herbaceous plants (Sellers, 1986). When the SiB was compared to the physiognomic classification of the natural vegetation types of the world given by Kuchler (1949,1983), it was found that of his 32 vegetation types 31 can be represented by SiB. So since we found that these models are accurate and can calculate what we need we used them for our project.

Methods and Equations:

This are a few methods that were used to try to come up with the model:

The Evaporation process equation was one of our tries:

$$Q_n = (1-2)R_s - R_l$$
$$Q_n = (1-2)R_s - (a_1 + b_1 c)\Sigma O T_{sk}{}^4$$
$$Q_n = net / radiation / on / surface$$
$$R_s = Incident / short / wave / radiation$$
$$R_l = Incident / long / wave / radiation$$
$$c = cloud / cover$$
$$T_{sk} = T_{sk} + 273 / Kelvin / scale$$

$$Ea = \Gamma Le(a + bu)(esu - ea)$$
$$Ea = \Gamma Le(a + bu)(1 - RH)$$
$$Ea = \Gamma Le(a + bu)(1 - \frac{ea}{esa})esa$$
$$Le = Latentheat$$
$$esa = 2.74894x10^8(\frac{-4278.6}{Ta + 242.79})$$
$$\Delta = \frac{2.7489x10^8 + 4278.6}{(Ta + 242.79)^2}$$
$$Le = 597.3 - 0.57Ta$$
$$u = windvelocity$$

Results and Conclusions:

In trying to come up with our model we failed to complete our project in the given time at NASA but we will continue our work at Central State Universities, Water Resource Department.

Even though my project was not completed I am very happy with what was accomplished in the short amount of time that was allowed for this project. I will return to Central State with a head with new ideas. During my summer I have learned not only about the simple biosphere model and the general circulation model, I have also learn the FORTRAN applications, and how to use the computer operating system UNIX and its family of related utility programs. This are not the only classes that I was fortunate to be able to attend at Goddard but, I also had the opportunity to be exposed to image processing using PCI software.

So at Goddard Space Flight Center I was given the opportunity to learn and be exposed to many new exciting and very interesting things and people. I hope that NASA will continue the summer programs so that other students can get a chance to experience what is offered at NASA.

ENHANCING THE USER FRIENDLINESS OF NSSDC'S MODEL ARCHIVE

Trena Covington

THE CODE

This summer I had the opportunity of working under code 630/Hughes. 630 is the code for the Space Science Data Operations Office (SSDOO) which was established to meet new challenges of processing, organizing, documenting, archiving, and disseminating data from many NASA space science missions. SSDOO accomplishes its responsibilities through the archiving, cataloging, and dissemination of space science data and provides many specialized science support services and systems. The data systems provide support for many of the laboratory research needs within the Space Sciences Directorate and the scientific needs of the space science research commmunity in general. SSDOO manages and operates three major data-intensive branches performing space data operations activities. It also serves the Space Science Directorate as a key interface for coordinating data management and archiving plans with the NASA Headquarters Information Systems Branch; Astrophysics Division; Space Physics Division; Planetary Physics Division; and Office of Aeronautics, Space, and Technology.

The National Space Science Data Center (NSSDC) is a multidiscipline archive, presently supporting astrophysics, solar and space plasma physics, lunar and planetary , and Earth science data. The NSSDC has accepted as one of its responsibilities the archiving and distribution of solar-terrestrial models. The goal of my project is to enhance the user friendliness of NSSDC's model archive and to focus on the development of callable Interactive Data Language (IDL) programs.

Presently, NSSDC's model holdings encompass more than 70 model software packages which are distributed on diskette, CD-ROM, and through anonymous FTP. All of the software packages relate to emperical models. In most cases emperical models consist of a set of mathematical functions and of the matrix of coefficients that were obtained by fitting the system of functions to the underlying database. Most importantly, NSSDC's model archive includes international standard models such as the COSPAR International Reference Atmosphere (CIRA), the International Reference Ionosphere (IRI), and the International Geomagnetic Reference Field (IGRF). These models are established and regularly improved by special working groups set up by the responsible specific unions. The World Data Center A for Rockets and Satellites is one of the centers that receives the newest editions of these international standard models. A few other often requested models are the Mass-Spectrometer-

Incoherent-Scatter (MSIS) thermosphere model, the SERF-EUV model of the solar EUV fluxes and the AE/AP radiation belt model. The NSSDC archives and distributes the latest versions of these models. An important aspect of NSSDC's activities related to models are efforts to improve the accessibility of model software. Guidebooks and catalogs are published to provide users with a good overview of what models and software are available for a specific region and parameter and how to best access those models. Interactive "frontends" were developed for several of the most frequently requested models. These driver programs prompt the user for the required input parameters/choices and then calculate and display tables of model parameters. With the help of interactive frontends, it is also possible to provide network users with the option to access and run the model programs online as part of NSSDC Online Data and Information Services (NODIS) account. Online capabilities include reading and copying information about specific models, obtaining source codes, running the model programs, and transferring model output to the user's home node.

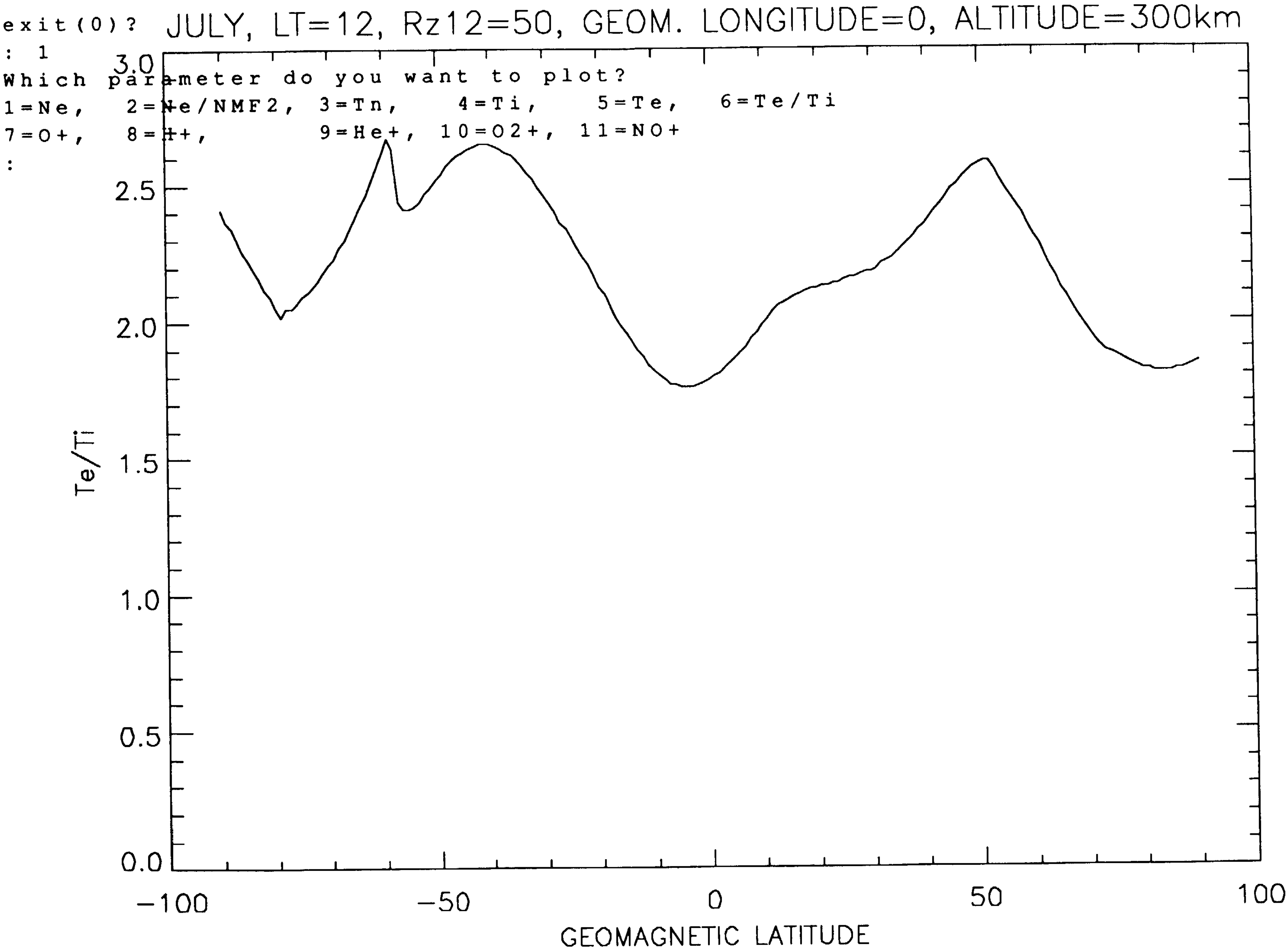
The first step to enhancing the user friendliness of the model is to use file compression software (PKZIP) to compress all files for a specific model into one self-extracting file. I focused on six emperical models: IRI-92, MSIS-86, CIRA-86, IGRF/91, SERF-EUV-92, and RADBELT/AE-8/AP-8. For all six models I followed a simple procedure. First, I made a directory of the model, then I copied the files of the model from a floppy disk onto the hard drive. I then used the file compression software (PKZIP) to "zip" all the files for a specific model into a single file. The command "zip2exe -j filename.zip" was used to put the files into one self-extracting file. I then formatted a blank floppy disk and copied the executable file onto that disk. Next, I copied the contents off the newly formatted floppy disk onto the hard drive and ran a test to make sure all the files were copied. The purpose of file compression is to simplify the electronic transfer of these software packages.

Feedback from the user community has shown that one of the most desired capabilities is the graphical display of model parameters. The main part of my project is to focus on the development of callable Interactive Data Language (IDL) programs that would allow the user to display the model parameters that he or she had created. NSSDC's Online Data and Information Services (NODIS) account makes available 4 different model programs: the IRI model, the MSIS model, the RADBELT model, and the IGRF model. I specificall y worked with the IRI model. To get to NODIS from NSI/DECnet node, first SET HOST NSSDCA and then designate username: NODIS. Next, enter display option (1,2,3), then select the menu-option (3) SPACE PHYSICS SERVICES, next select the menu-option (2) GEOPHYSICAL MODELS and from there follow the prompts and menus. The program prompts the user for the required input parameters/choices and then calculates and displays tables of model parameters. I chose different parameters for the IRI model and generated tables. There were six parameters to choose from: LATITUDE, LONGITUDE, RZ12 (solar sunspot number), MONTH, HOUR, and ALTITUDE. These parameters would be on the x-axis when plotted. After the tables were completed, I downloaded my data onto my mail account, then I extracted these tables into my personal directory in NSSDCA. The tables consist of one of the parameters listed above versus eleven variables which, when plotted, would be on the y-axis. The variables were neutral density (Ne/cm-3), density at the F2 peak (Ne/NMF2), neutral temperature (Tn/K), ionic temperature (Ti/K), electron temperature (Te/K), electron temperature/ ionic temperature (Te/Ti), percentage of Oxygen (O+), percentage of Hydrogen (H+), percentage of Helium (He+), percentage of Dioxide (O2+), and percentage of Nitrogen Oxide(NO+). I wrote a program in IDL that prompts the user to enter "exit(0)?". If the user enters anything other than 0 the program prompts "Which parameter do you want to plot?" (Graph 1A). All programs for the IRI model are very similiar.

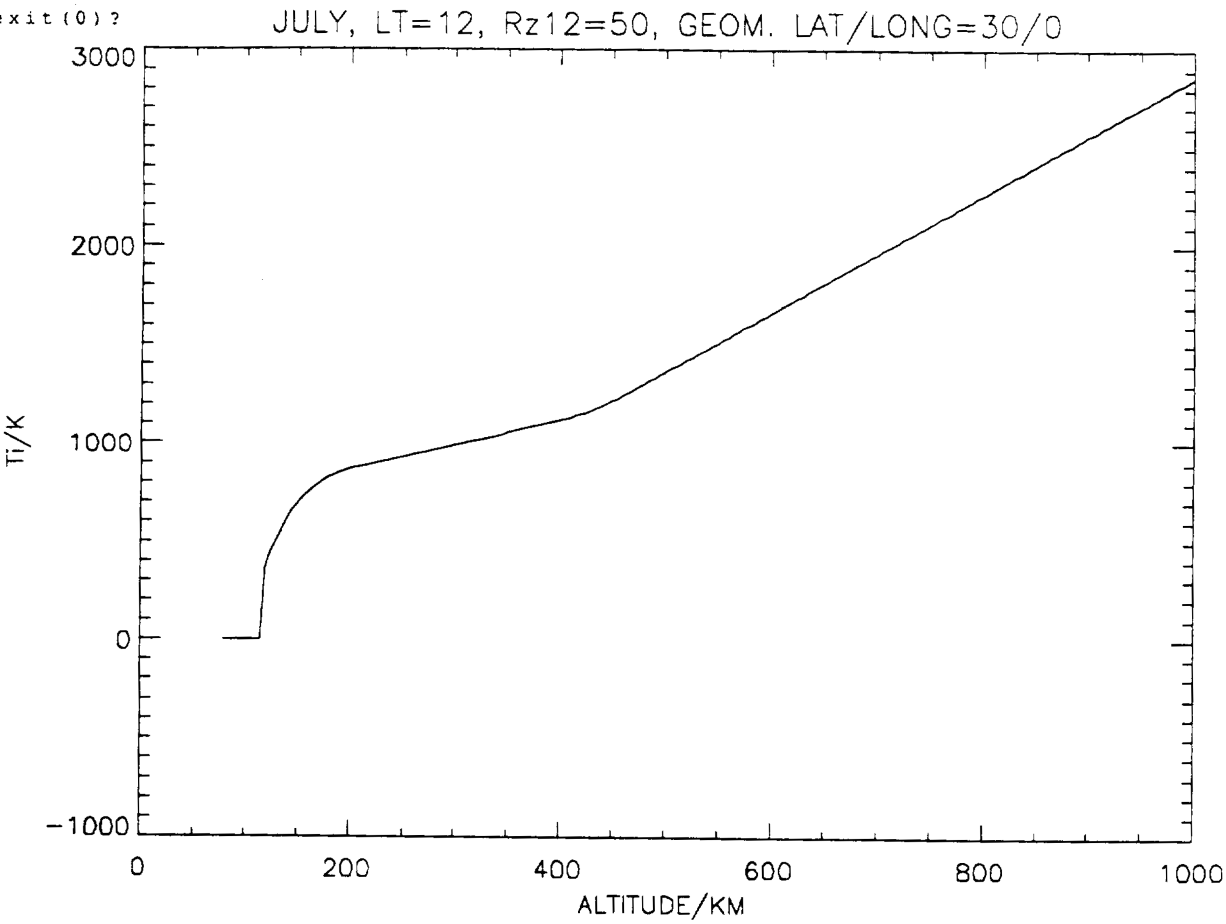
My NASA summer experience was very enjoyable. First, I was able to enhance the user friendliness of the model archive by making the electronic transfer of software packages easier. Before, a user had to transfer 30 or more files for a specific model, now they only have to transfer one self-extracting file which contains all the files for a specific model. Secondly, I made it more user friendly by allowing users to graphically display the model parameters they select. Lastly, I was able to work with both, Physics and Computer Science on a professional level and I really enjoyed it.

From: NCF::NODIS 26-JUL-1995 12:02:18.60
To: NSSDCA::BILITZA
CC:
Subj: NSSDC IRI Models Data

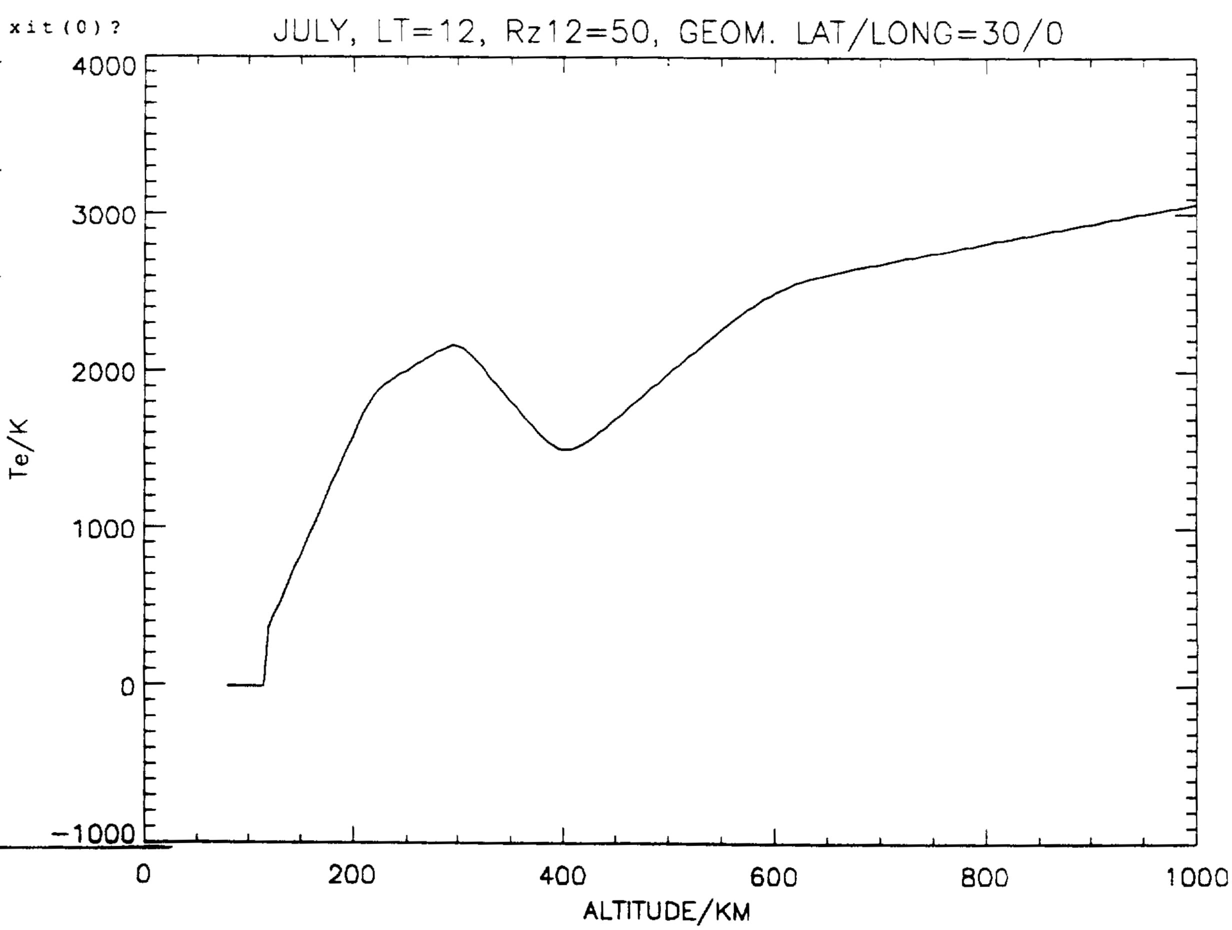
HOUR	ELECTRON DENSITY			TEMPERATURES			ION PERCENTAGE DENSITIES				
L.T.	NE/CM-3	NE/NMF2	TN/K	TI/K	TE/K	TE/TI	O+	H+	He+	O2+	NO+
0.0	384321	0.9732	779	779	785	1.01	77	20	2	0	0
0.5	364879	0.9789	779	779	785	1.01	77	20	2	0	0
1.0	345986	0.9792	777	777	778	1.00	77	20	2	0	0
1.5	326525	0.9751	774	774	774	1.00	77	20	2	0	0
2.0	305474	0.9683	769	769	770	1.00	77	20	2	0	0
2.5	283509	0.9630	763	769	769	1.00	77	20	2	0	0
3.0	263375	0.9649	756	770	770	1.00	77	20	2	0	0
3.5	248561	0.9762	750	774	783	1.01	77	20	2	0	0
4.0	241854	0.9914	746	782	881	1.13	77	20	2	0	0
4.5	245153	0.9999	745	795	1045	1.31	77	20	2	0	0
5.0	260302	0.9964	748	813	1278	1.57	77	20	2	0	0
5.5	285969	0.9773	755	835	1544	1.85	77	20	2	0	0
6.0	317429	0.9433	767	860	1762	2.05	100	0	0	0	0
6.5	353487	0.9152	782	883	1869	2.12	100	0	0	0	0
7.0	391007	0.9030	800	905	1889	2.09	100	0	0	0	0
7.5	424655	0.9054	819	923	1907	2.07	100	0	0	0	0
8.0	451187	0.9179	838	938	1978	2.11	100	0	0	0	0
8.5	471501	0.9357	856	950	2087	2.20	100	0	0	0	0
9.0	490469	0.9550	871	960	2154	2.24	100	0	0	0	0
9.5	514525	0.9730	884	968	2116	2.19	100	0	0	0	0
10.0	548134	0.9875	894	975	2004	2.06	100	0	0	0	0
10.5	591518	0.9964	902	980	1919	1.96	100	0	0	0	0
11.0	641197	0.9997	907	984	1933	1.96	100	0	0	0	0
11.5	691916	1.0000	912	988	2044	2.07	100	0	0	0	0
12.0	738133	0.9999	917	991	2165	2.18	100	0	0	0	0
12.5	775058	0.9999	923	995	2170	2.18	100	0	0	0	0
13.0	801251	1.0000	932	999	2014	2.02	100	0	0	0	0
13.5	818658	1.0000	942	1003	1775	1.77	100	0	0	0	0
14.0	830966	1.0000	954	1007	1562	1.55	100	0	0	0	0
14.5	841498	1.0000	966	1010	1419	1.41	100	0	0	0	0
15.0	851793	1.0000	978	1011	1333	1.32	100	0	0	0	0
15.5	861241	1.0000	988	1011	1272	1.26	100	0	0	0	0
16.0	867428	0.9997	994	1007	1223	1.21	100	0	0	0	0
16.5	867232	0.9985	996	999	1195	1.20	100	0	0	0	0
17.0	858349	0.9966	992	992	1206	1.22	100	0	0	0	0
17.5	840453	0.9949	982	982	1246	1.27	100	0	0	0	0
18.0	815081	0.9951	967	967	1274	1.32	100	0	0	0	0
18.5	783836	0.9978	947	947	1238	1.31	100	0	0	0	0
19.0	746408	0.9999	924	924	1122	1.21	77	20	2	0	0
19.5	703524	0.9985	898	898	968	1.08	77	20	2	0	0
20.0	656340	0.9908	873	873	874	1.00	77	20	2	0	0
20.5	606517	0.9753	848	848	849	1.00	77	20	2	0	0


exit(0)?
: 1
Which parameter do you want to plot?
1=Ne, 2=Ne/NMF2, 3=Tn, 4=Ti, 5=Te, 6=Te/Ti
7=O+, 8=H+, 9=He+, 10=O2+, 11=NO+
:
JULY, LT=12, Rz12=50, GEOM. LONGITUDE=0, ALTITUDE=300km
Te/Ti
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−100
−50
0
50
100
GEOMAGNETIC LATITUDE

exit(0)?
:


JULY, LT=12, Rz12=50, GEOM. LAT/LONG=30/0
Ti/K
3000
2000
1000
0
-1000
0
200
400
600
800
1000
ALTITUDE/KM

xit(0)?


JULY, LT=12, Rz12=50, GEOM. LAT/LONG=30/0
Te/K
4000
3000
2000
1000
0
-1000
0
200
400
600
800
1000
ALTITUDE/KM

National Aeronautics and Space Administration
Goddard Space Flight Center
Summer Institute in Engineering and Computer Applications

Summer Tasks:

Indoor Air Quality Evaluation of Building 18
Fire Pumps Performance Evaluation
Replacement of the Boilers at Building 24

Code 205.2 : Safety and Environmental Branch
Head : Ron Kaese
Mentor : Phil Nessler

Ricardo E. Diaz
SIECA-UG Student
August 2, 1995

Code 205.2 : Safety and Environmental Branch

Code 205.2 provides safety management services to the GSFC community. Among the responsibilities of this code is the review of facility designs and inspection of existing facilities for public safety. The code is responsible to have a emergency plan for life threatening situations. Also looks for keeping a comfortable environment in the workplace through building inspection and use of codes.

The indoor air quality inspection of Building 18 deals with keeping a comfortable environment in that workplace. The evaluation of the fire pumps performance plays a important role in fire emergencies. It is a method to test the reliability of the pumps in providing the necessary water flow and pressure to extinguish a fire. The replacement procedures of the boilers at Building 18 is a clear example of how the safety codes are implemented to provide GSFC public safety.

List of acronyms

AHU	Air Handler Unit
ASHRAE	American Society of Heating, Refrigerating and Air-Conditioning Engineers
ASME	American Society of Mechanical Engineers
CFM	Cubic Feet per Minute
CFR	Code of Federal Regulations
FMD	Facilities Management Division
GPM	Gallons per Minute
GSFC	Goddard Space Flight Center
NFPA	National Fire Protection Association
NFPC	National Fire Protection Codes
OSHA	Occupational Safety and Health Administration
PPM	Parts Per Million
PSI	Pounds per Square Inch
WFF	Wallops Flight Facility

Indoor Air Quality Evaluation of Building 18

Introduction

The occupants of Building 18 were complaining about the air conditioning system. Some of them felt temperature inconsistencies during the week, others complained about uncomfortable temperatures (air was too cold or too hot), others were upset about the excessive air blowing above their heads. The thermostats in the cold offices were set to the maximum but the office temperature was still cold. By setting the thermostats to the maximum temperature, the dampers in the air ducts closed and the supply air flow was interrumped. Other offices with manually operated vents had the vents closed or blocked with paper. Since little or no air was being supplied to those rooms, I was concerned that the occupants were not getting enough fresh air (oxygen rich air). These problems led me to evaluate the air quality in the building.

Building 18 has two AHU's. Both AHU have almost identical design characteristics. An AHU is mainly a huge fan that moves the supplied air through the building. The supply air is a mixture of return air (recycled) and outside air (fresh air). Note that air flow, heat load, supply air temperature, and other factors depend upon each other.

The indoor air quality of a building is evaluated in four aspects: ventilation, contaminants control, thermal comfort, and humidity comfort. This report includes what I found throughout the summer internship about the air quality in that building, and at the end of this report, I suggest corrections for the problems.

Ventilation / Contaminants Control

There was a possibility of excessive accumulation of CO_2 in offices. ASHRAE recommends a minimum of 20 CFM of fresh air per person in a closed building. A high concentration of CO_2 is one indicator of insufficient fresh air. A CO_2 meter was used in room 118. This meter generated the graph that appears as fig. A1. The maximum concentration of CO_2 measured was 1400 PPM . This concentration, although below the maximum concentration permitted by OSHA (5000 PPM), was above the recommended maximum concentration (1000 PPM).

A faster method to find the amount of outside fresh air for the whole building was to use the FMD air balancing report. After a long search, the reports were found, but the report was one year old, and I had some doubts in using that data with the present state of the system. Important to note was that the air supplied by AHU 1 was being balanced this summer by Comfort Control Inc. Therefore, I calculated the outside air supplied to the building by AHU 2 and waited for the AHU 1 balancing report.

The method used to calculate the outside air was the estimation of the percentage of outside air. There is a relationship between air temperatures:

$$\% \, OA = [T_{RA} - T_{MA}] / [T_{RA} - T_{OA}] * 100$$

where T = temperature (^{0}F) RA = return air

MA = mixed (supply) air OA = outside air

After making the proper calculations with the data provided by Bob Earnest of the refrigeration shop (see fig. A2) the results were 32.14% for AHU 1 and 3.29% for AHU 2.

The next step was to find the total amount of air supplied to the building by AHU 2. A balometer was used to read the air flow on every diffuser belonging to AHU 2 (see fig. A3). The total flow was 13480 CFM, but adjusting for an air leakage of 5000 CFM from the previous balancing report, the total flow results in 18480 CFM. Using the percent of outside air that was calculated for AHU 2, the outside air amount became approximately 600 CFM. This amount supplies just enough for 30 people with 20 CFM of outside air. Considering that the building was designed for 300 people, this amount is insignificant.

The report from Comfort Control Inc. has not been finished. But based on the above calculated results, AHU 1 is overloaded for supplying the building with fresh air while AHU 2 provides almost no fresh air.

I personally checked the 2nd floor mechanical room and found enormous air leaks in the air ducts due to holes, loose bolts, loose connections, improperly fitted cooling tubes, etc. The air ducts leak, occupants unbalance the system by blocking the diffusers, diffusers that are not blocked blow too much air, AHU 2 is supplying practically no fresh air, and AHU 1 is overloaded trying to cool down more fresh air than it was designed to cool (fresh/outside air has a temperature ranging from 85 to 100^0F). These factors complicate the problem with the air conditioning system in Building 18.

Thermal / Humidity Comfort

The second part of this project was to see whether the temperature and humidity in the building rooms were stable or not and to find reasons for the uncomfortable air temperature. A hygrometer was used to find the stability of the air temperature. This instrument measures temperature and relative humidity for long periods of time. This device was placed in room 173 for a week.

The generated graph (see fig. A4) shows temperatures ranging from 71^0F to 74^0 F duing the weekdays which is not a big fluctuation. The relative humidity ranges between 44% to 50% which is an acceptable range. My personal preference for temperature is a close range of 74^0 F to 75^0F for that range of humidity. The readings were a little below that range. Another important consideration is the Announcement 95-07 for energy conservation which sets office temperature ranges during summer to be 76^0 - 80^0F. The majority of the people that were interviewed at GSFC felt that this temperature range was unreasonable.

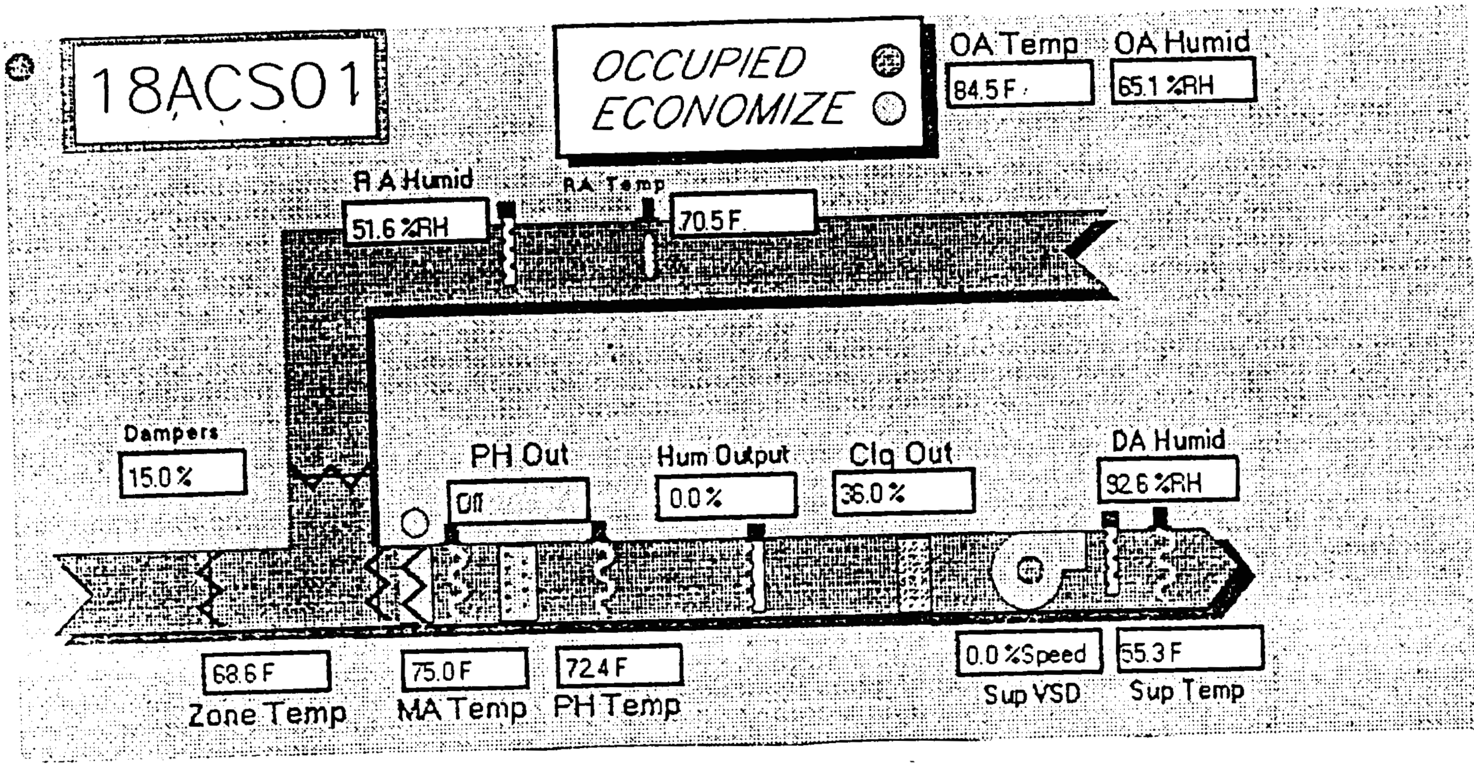
Occupants in other offices felt the temperature to be too cold, especially the ones working near or inside computer rooms. Blocked diffusers (vents) were found in these offices. In offices with vents blowing air just above the occupants heads, flow deflectors were placed to redirect the air flow to unoccupied places.

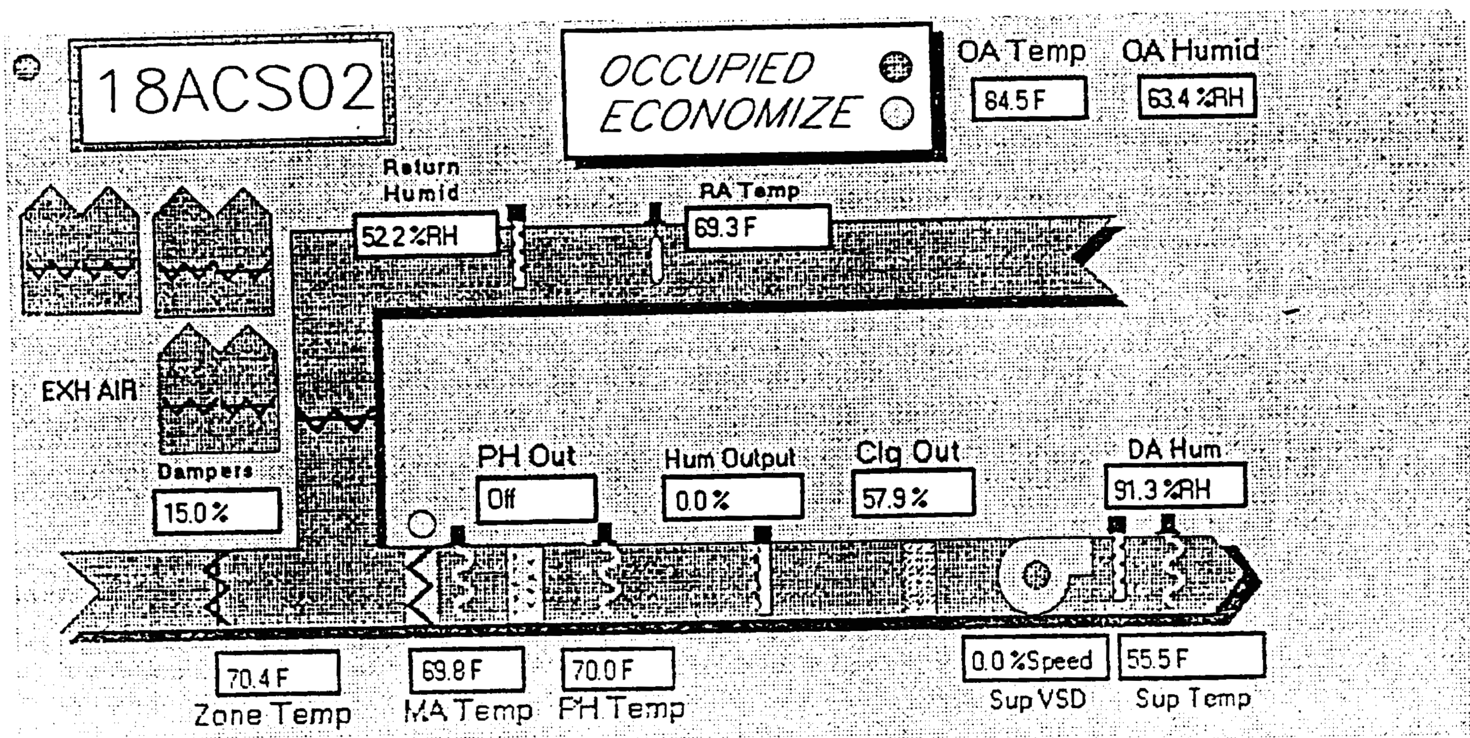
Suggestions

In order to fix the problems with the air conditioning system of Building 18 and other buildings facing similar problems, I am presenting some short term and long term suggestions.

Short Term Suggestions:

1) Fix air leaks in main air ducts.
2) Update the mechanical drawings (drawing GF-18-20699, GF-18-20700)
   - Ex. : Rooms 123/127, 173, 200.
     - Room 127 actually was not included in the latest drawing of 1993.
     - Room 173 is not drawn with the actual partition of the room.
     - Room 200 has walls not proportional to the size of the actual walls.
   - Assign the numbers of the actual zones in the zone drawings.
3) Balance the air supplied by AHU 2.
4) Balance both systems to have equal loads of fresh air.
   - Total amount must reach or surpass the 6000 CFM minimum (20 CFM * 300 occupants)
5) Install air returns in the offices surrounding computer room 110 to extract the cold air.
6) Build deflectors similar to that in room 159. Install them to diffusers that blow air directly to the occupants.


18ACS01
OCCUPIED
ECONOMIZE
OA Temp
84.5 F
OA Humid
65.1 %RH
R A Humid
51.6 %RH
RA Temp
70.5 F
Dampers
15.0 %
PH Out
Off
Hum Output
0.0 %
Clg Out
36.0 %
DA Humid
92.6 %RH
68.6 F
Zone Temp
75.0 F
MA Temp
72.4 F
PH Temp
0.0 %Speed
Sup VSD
55.3 F
Sup Temp


18ACS02
OCCUPIED
ECONOMIZE
OA Temp
84.5 F
OA Humid
63.4 %RH
Return Humid
52.2 %RH
RA Temp
69.3 F
EXH AIR
Dampers
15.0 %
PH Out
Off
Hum Output
0.0 %
Clg Out
57.9 %
DA Hum
91.3 %RH
70.4 F
Zone Temp
69.8 F
MA Temp
70.0 F
PH Temp
0.0 %Speed
Sup VSD
55.5 F
Sup Temp

ACTUAL AIR FLOW DATA OF 18-AC-02

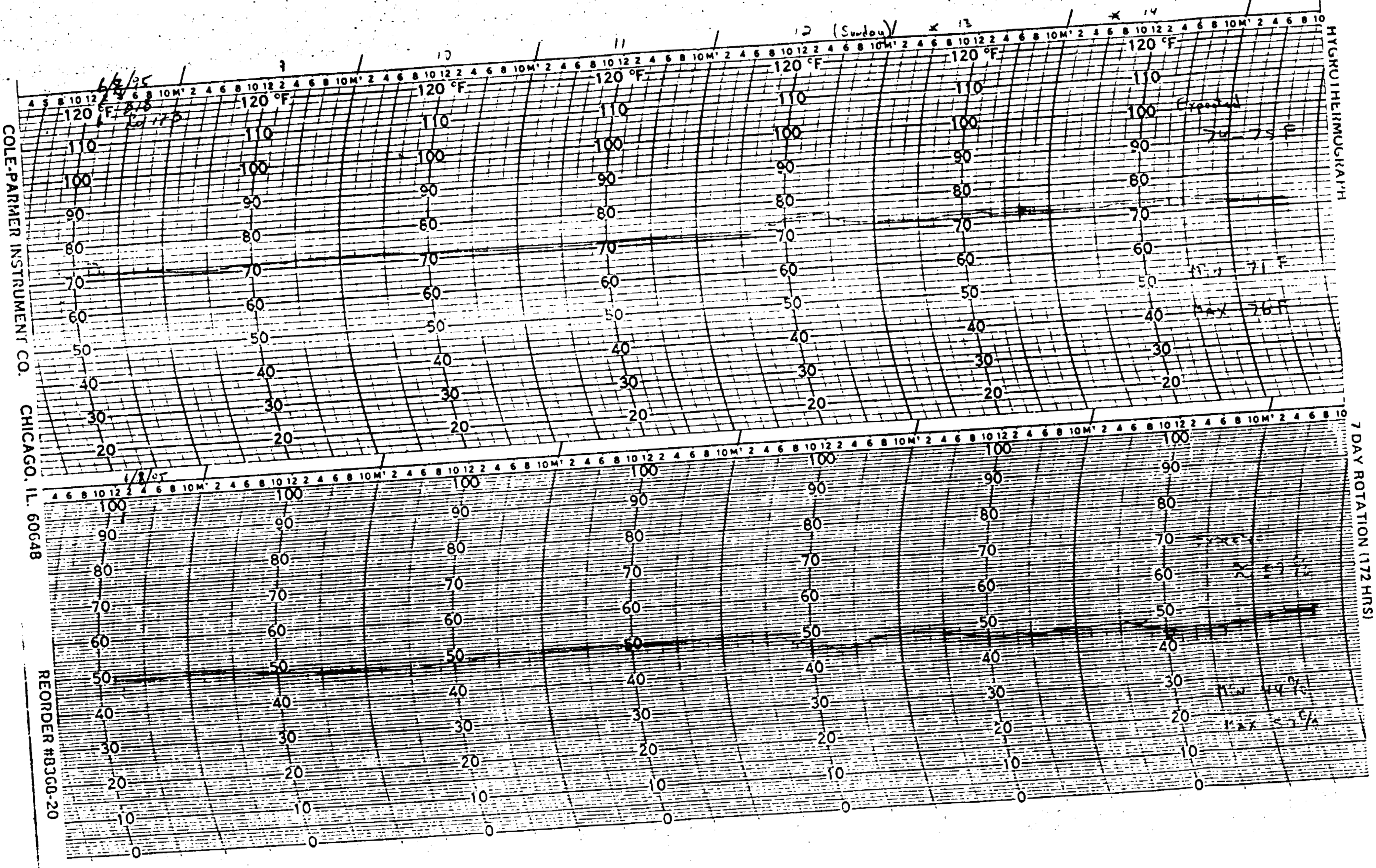
AREA SERVED	OUTLET NUMBER	AIR FLOW (CFM)
136B	1	130
136	2	140
136A	3	100
136A	4	70
142	5	70
142B	6	180
159A	7	70
159A	8	165
159A	9	150
159A	10	150
159B	11	195
159	12	280
159	13	280
159	14	305
159	15	205
159	16	295
159	17	205
159	18	410
159	19	185
159C	20	205
ENTRY	21	130
118	22	285
118	22A	0
118	22B	110
118	23	130
118	23A	0
114	24	290
114	25	0
199	26	290
199	27	230
199	28	250
199	29	225
199	30	305
274	31	210
	SUB TOTAL	6245

AREA SERVED	OUTLET NUMBER	AIR FLOW (CFM)
274	32	270
274	33	290
200	34	220
200	35	230
200	36	260
200	37	240
274	38	90
274	39	85
274	40	410
274	41	410
274	42	335
274	43	390
200	44	220
200	45	340
200	46	210
200	47	305
200	48	320
200	49	270
200	50	300
200	51	305
200F	52	65
200F	53	70
200F	54	70
200F	55	75
234	56	285
234	57	270
242	58	285
255	59	0
242	60	140
255	61	55
255	62	130
255	63	130
255	64	160
	SUB TOTAL	7235

TOTAL AIR FLOW 13480

**** Rooms number as in drawings GF-18-20699 and GF-18-20700*

A. 4


COLE-PARMER INSTRUMENT CO.
CHICAGO. IL. 60648
REORDER #83G0-20
HYGROTHERMOGRAPH
7 DAY ROTATION (172 HRS)
12 (Sunday)
13
14
Expected
74-75 F
Min 71 F
Max 76 F
Min 44 %
Max 53 %

Fire Pumps Performance Evaluation

Introduction

The performance of a fire pump is defined as the effective pressure the pump can provide to a certain amount of water flow. As years pass, the performance of the pumps starts to decrease. Some potential reasons for the decrease of performance can be due to electrical problems, material wear, lubrication loss, etc.

To keep track of the pump performance, every year the pump must be tested to a performance test. In this test, water flow and pressure are measured at predetermined points. The net pressure is compared to the original acceptance test data by means of the percentage of loss in pressure. This percentage of loss is then compared with the NFPC criteria (NFPA 25:5-3) and with technical booklets provided by the industry. For this evaluation the ***Simplified Water Supply Testing*** booklet was used. The maximum allowed pressure loss is 10 percent at any water flow. If the pressure loss is greater than 10%, further investigation is needed.

There are four fire pumps at WFF that have not been tested to NFPA criteria since they were installed more than ten years ago. A performance test was done on March 13, 1994. The goal in this task was to apply the procedures discused before to determine if the state of the pumps were acceptable. Also performance curves were generated from the test data of the fire pumps using mathematical analysis.

Mathematical Analysis

A typical performance curve looks like figure B.0. The pattern is one of a quadratic curve. The equation for this curve is defined as:

$$P = a + bQ + cQ^2$$

where Q = flow rate (GPM) P = net pressure (PSI)

a, b, c = constants

and the best way to find the value of this constants is solving simultaneous equations by matrix:

$$\begin{bmatrix} 1 & Q_1 & (Q_1)^2 \\ 1 & Q_2 & (Q_2)^2 \\ 1 & Q_3 & (Q_3)^2 \end{bmatrix} * \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

where (Q_1, P_1), (Q_1, P_1), and (Q_1, P_1) are pairs of data. After having the constants the equation is ready to use and a nice, smooth graph can be generated showing the relationship of the net pressure and the flow rate of a pump. Using the data at 0%, 100%, and 150% of the rated flow rate of the pump and the proper calculations, the curves equations for the original data were as follow:

Pump 85-65115	$P = 121 + 2.667 * 10^{-3} Q - 5.333 * 10^{-6} Q^2$
Pump 85-65116	$P = 119 + 3.00 * 10^{-3} Q - 5.0 * 10^{-6} Q^2$
Pump 85-65117	$P = 120 + 3.667 * 10^{-3} Q - 5.333 * 10^{-6} Q^2$
Pump 85-65118	$P = 118 + 5.167 * 10^{-3} Q - 5.833 * 10^{-6} Q^2$

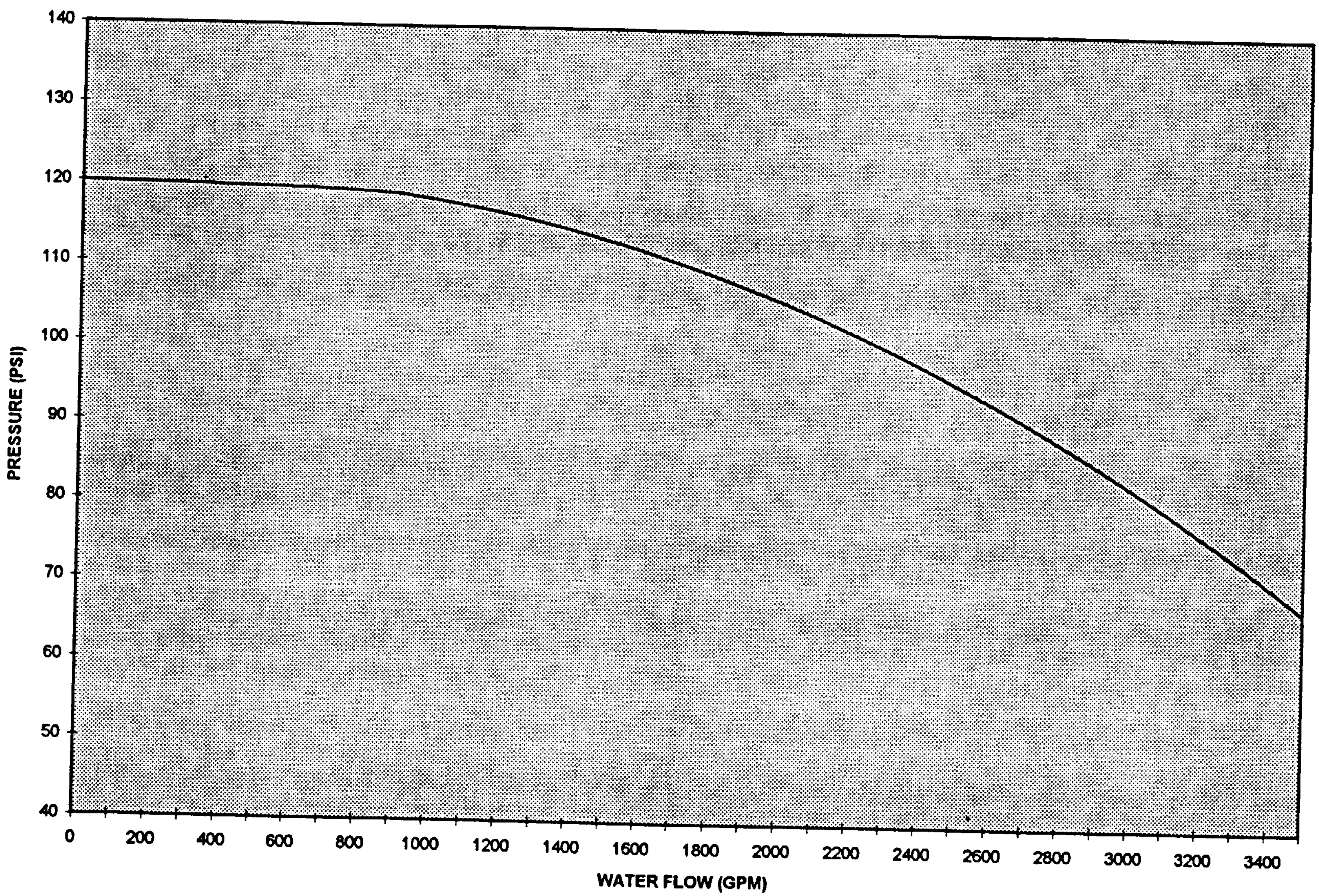
and for the test data are as follow:

Pump 85-65115	$P = 135 - 2.25 * 10^{-2} Q + 2.5 * 10^{-6} Q^2$
Pump 85-65116	$P = 120 - 5.00 * 10^{-3} Q - 5.0 * 10^{-6} Q^2$
Pump 85-65117	$P = 130 - 1.167 * 10^{-2} Q - 1.167 * 10^{-6} Q^2$
Pump 85-65118	$P = 130 - 1.30 * 10^{-2} Q - 2.0 * 10^{-6} Q^2$

All charts and graphs are shown on figures B.1 - B.12 . Also a curve using all the data points of the performance test is given for each pump to compare their aspect to those of the generated curves.

Evaluation of the Fire Pumps

Looking at the chart of the fire pumps (see fig. B. 13) it can be say that pumps 82-65116 and 82-65118 have to be checked in more details since their POD (percent of difference) are way beyond the limit. On the other hand, pump 82-65117 has an almost acceptable POD, but taking a close look at the flow of 3500 GPM (175% of rated flow rate) it is shown that the POD is almost at the limit so its good performance can be questionable. That pump should be investigated also. Finally the pump 82-65115 is a little bit beyond the limit but knowing that with time the percent of difference will grow, the best thing to do is to make further investigations and testings.


140
130
120
110
100
90
80
70
60
50
40
PRESSURE (PSI)
0
200
400
600
800
1000
1200
1400
1600
1800
2000
2200
2400
2600
2800
3000
3200
3400
WATER FLOW (GPM)

B. 0

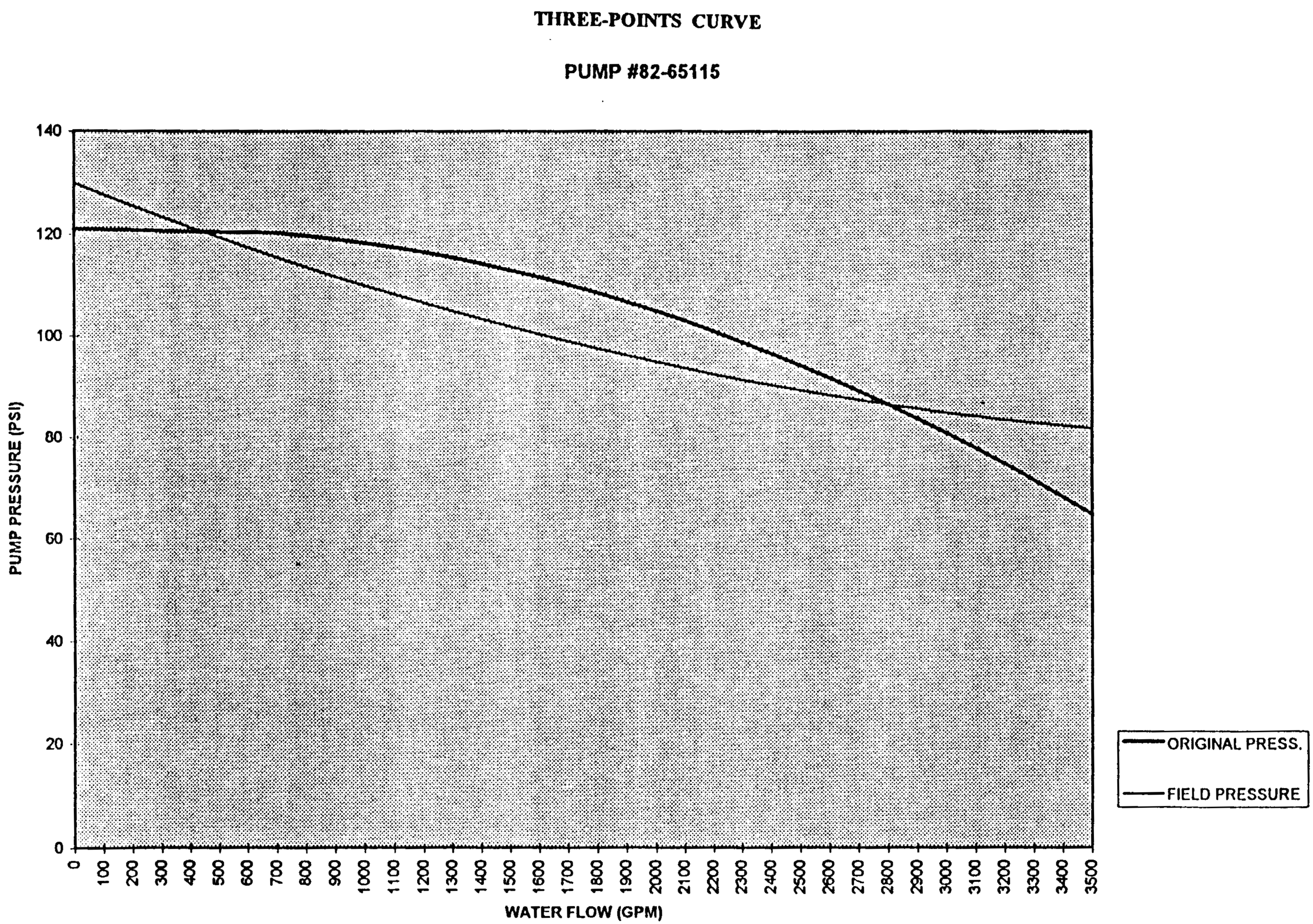
FIRE PUMP : 82-65115

MATHEMATICALLY GENERATED DATA

GPM	ORIGINAL PRESS.	CALCULATED PRESS.
0	121	130
100	120.91	127.78
200	120.82	125.60
300	120.65	123.48
400	120.51	121.40
500	120.40	119.38
600	120.38	117.40
700	120.25	115.48
800	119.72	113.60
900	119.08	111.78
1000	118.33	110.00
1100	117.48	108.28
1200	116.52	106.60
1300	115.45	104.98
1400	114.28	103.40
1500	113.00	101.88
1600	111.61	100.40
1700	110.12	98.98
1800	108.52	97.60
1900	106.82	96.28
2000	105.00	95.00
2100	103.08	93.78
2200	101.06	92.60
2300	98.92	91.48
2400	96.68	90.40
2500	94.34	89.38
2600	91.88	88.40
2700	89.32	87.48
2800	86.66	86.60
2900	83.88	85.78
3000	81.00	85.00
3100	78.02	84.28
3200	74.92	83.60
3300	71.72	82.98
3400	68.42	82.40
3500	65.01	81.88

TEST DATA

GPM	ORIGINAL PRESS.	FIELD PRESSURE
0	121	135
1000	116	
2000	105	100
2500	94	
3000	81	90
3500	63	60

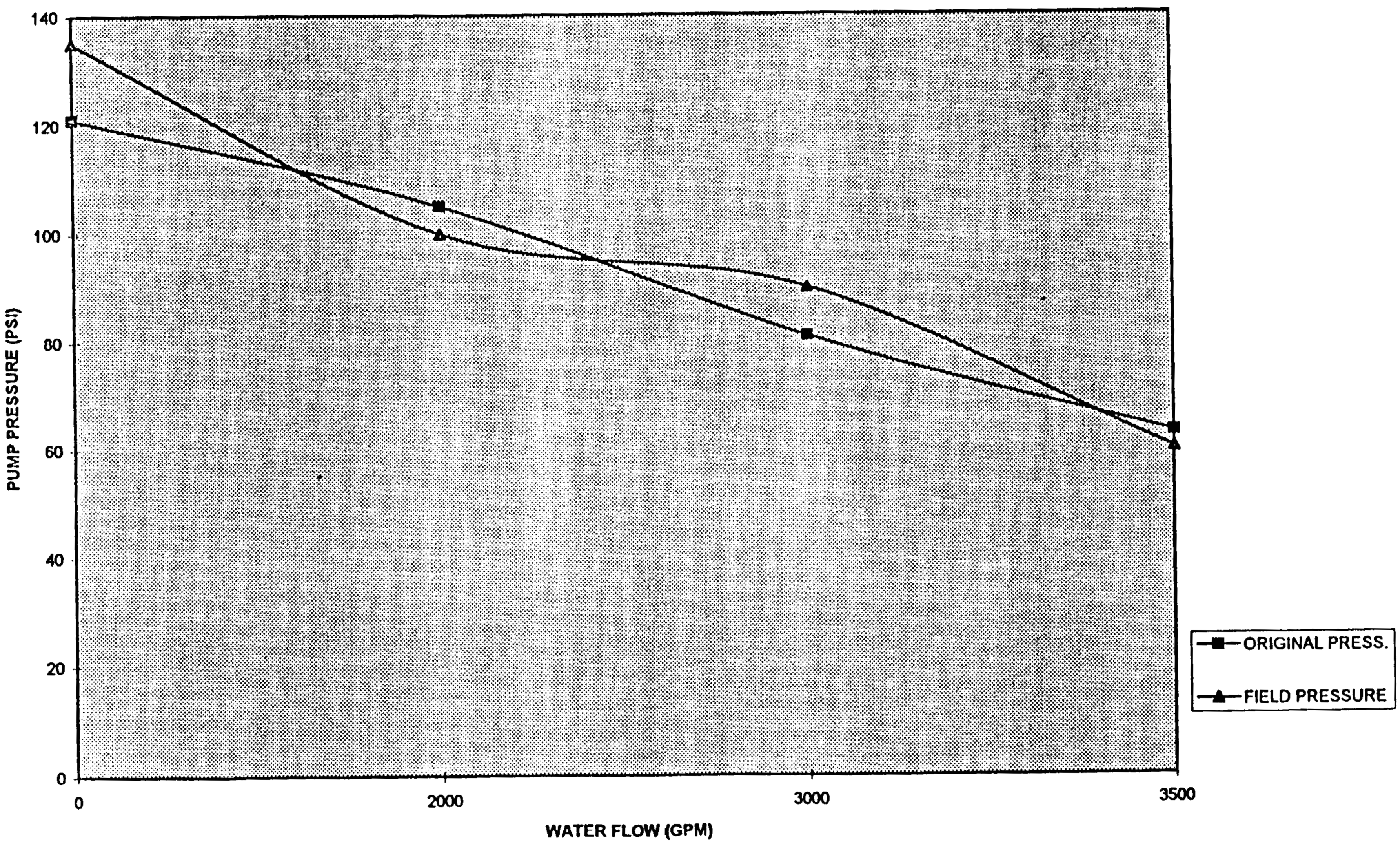


THREE-POINTS CURVE
PUMP #82-65115
PUMP PRESSURE (PSI)
140
120
100
80
60
40
20
0
0
100
200
300
400
500
600
700
800
900
1000
1100
1200
1300
1400
1500
1600
1700
1800
1900
2000
2100
2200
2300
2400
2500
2600
2700
2800
2900
3000
3100
3200
3300
3400
3500
WATER FLOW (GPM)
ORIGINAL PRESS.
FIELD PRESSURE

B. 2

ALL-POINTS CURVE

PUMP #82-65115



FIRE PUMP : 82-65116

MATHEMATICALLY GENERATED DATA

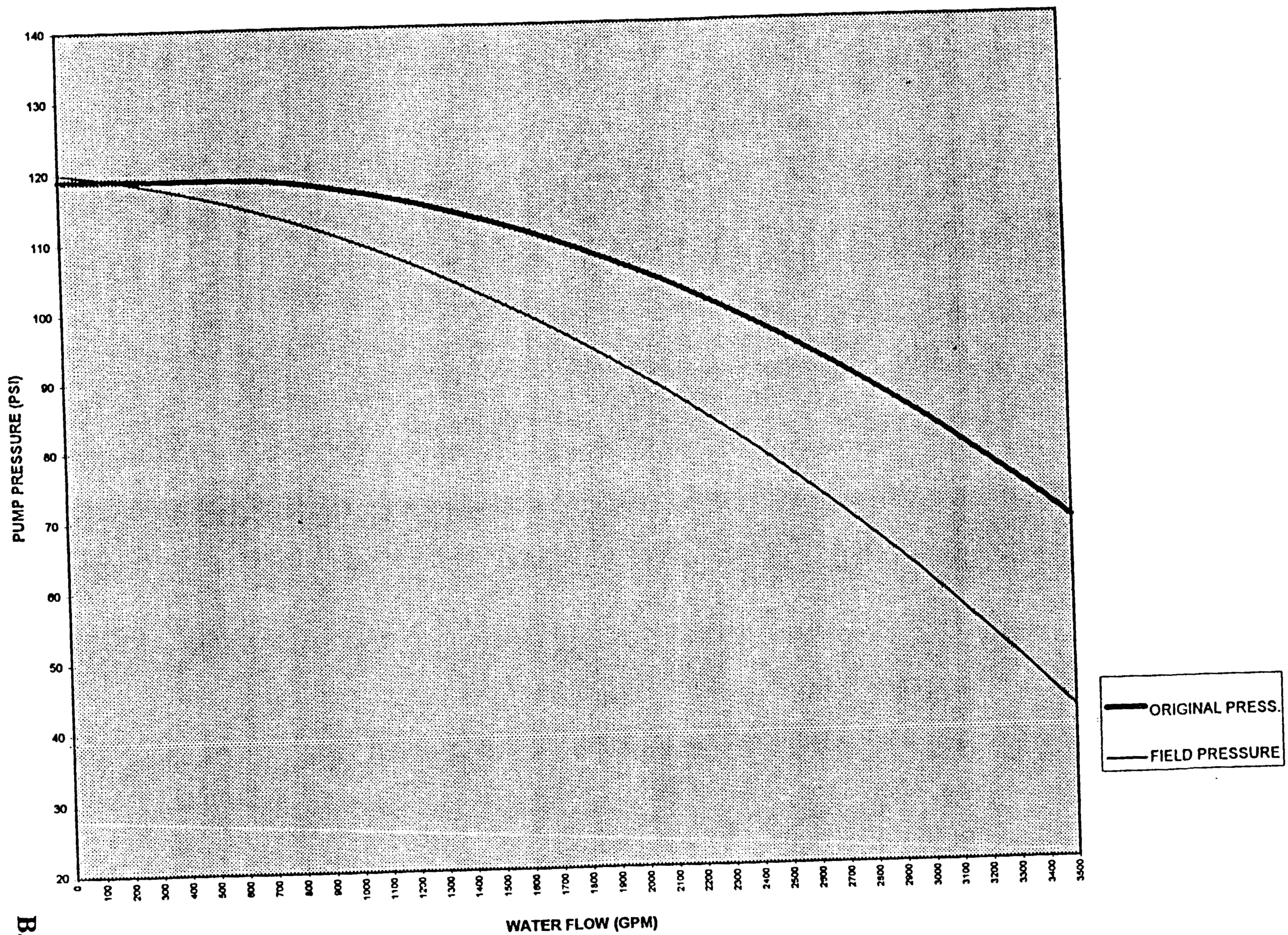
GPM	ORIGINAL PRESS.	CALCULATED PRESS.
0	119	120
100	118.95	119.45
200	118.93	118.8
300	118.89	118.05
400	118.84	117.2
500	118.79	116.25
600	118.73	115.2
700	118.65	114.05
800	118.2	112.8
900	117.65	111.45
1000	117	110
1100	116.25	108.45
1200	115.4	106.8
1300	114.45	105.05
1400	113.4	103.2
1500	112.25	101.25
1600	111	99.2
1700	109.65	97.05
1800	108.2	94.8
1900	106.65	92.45
2000	105	90
2100	103.25	87.45
2200	101.4	84.8
2300	99.45	82.05
2400	97.4	79.2
2500	95.25	76.25
2600	93	73.2
2700	90.65	70.05
2800	88.2	66.8
2900	85.65	63.45
3000	83	60
3100	80.25	56.45
3200	77.4	52.8
3300	74.45	49.05
3400	71.4	45.2
3500	68.25	41.25

TEST DATA

GPM	ORIGINAL PRESS.	FIELD PRESSURE
0	119	120
1000	114	
2000	105	90
2500	94	80
3000	83	60
3500	56	45

THREE-POINTS CURVE

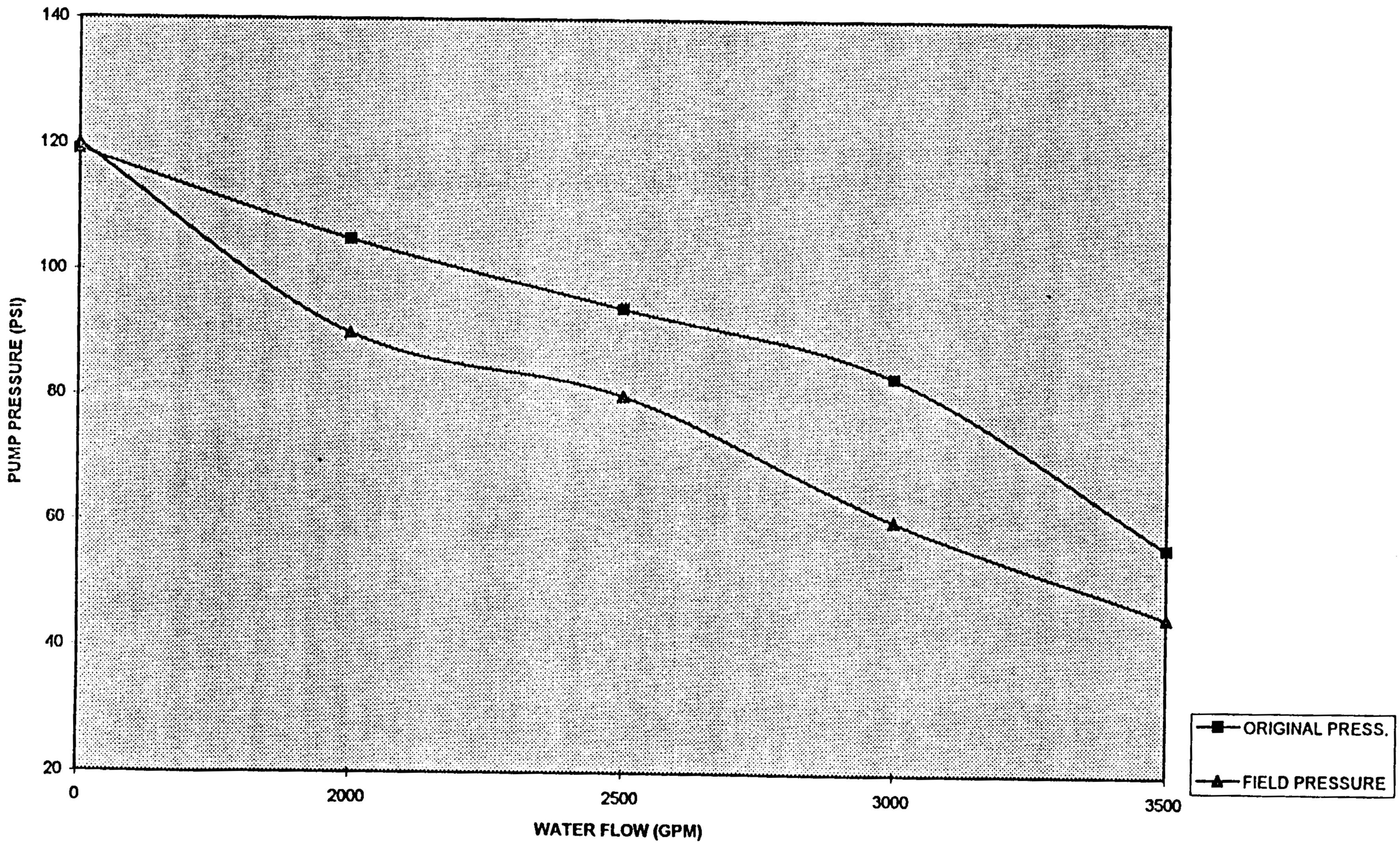
PUMP # 82-65116



B.5

ALL-POINTS CURVE

PUMP #82-65116



B. 6

FIRE PUMP : 82-65117

MATHEMATICALLY GENERATED DATA

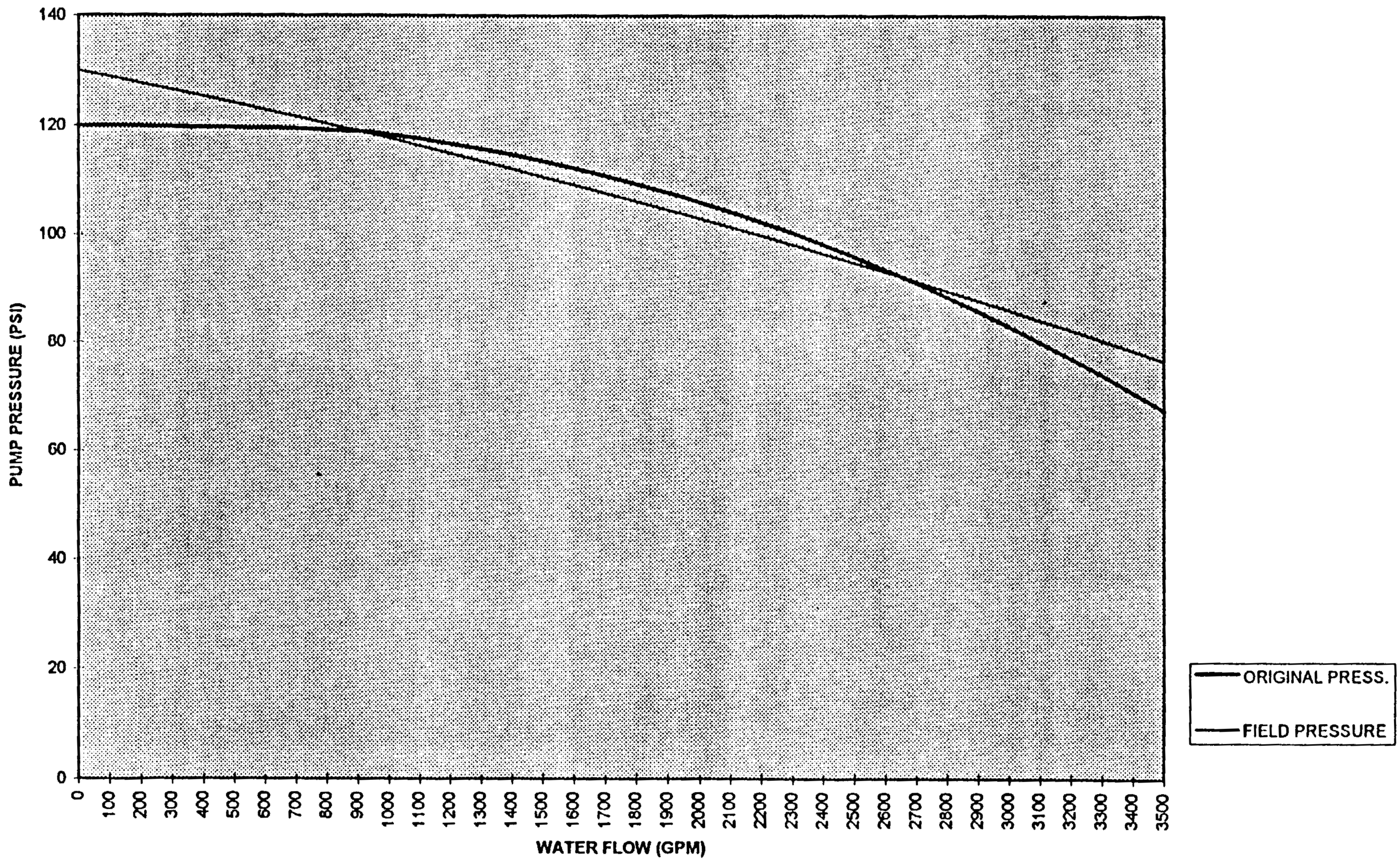
GPM	ORIGINAL PRESS.	CALCULATED PRESS.
0	120	130
100	119.95	128.87
200	119.92	127.72
300	119.85	126.54
400	119.71	125.35
500	119.64	124.12
600	119.58	122.88
700	119.45	121.61
800	119.22	120.32
900	118.98	119.00
1000	118.33	117.67
1100	117.58	116.30
1200	116.72	114.92
1300	115.75	113.51
1400	114.68	112.08
1500	113.50	110.62
1600	112.21	109.15
1700	110.82	107.64
1800	109.32	106.12
1900	107.72	104.57
2000	106.00	103.00
2100	104.18	101.40
2200	102.26	99.78
2300	100.22	98.14
2400	98.08	96.48
2500	95.84	94.79
2600	93.48	93.08
2700	91.02	91.34
2800	88.46	89.58
2900	85.78	87.80
3000	83.00	86.00
3100	80.12	84.17
3200	77.12	82.32
3300	74.02	80.44
3400	70.82	78.54
3500	67.51	76.62

TEST DATA

GPM	ORIGINAL PRESS.	FIELD PRESSURE
0	120	130
1000	116	
2000	106	100
2500	96	95
3000	83	80
3500	66	60

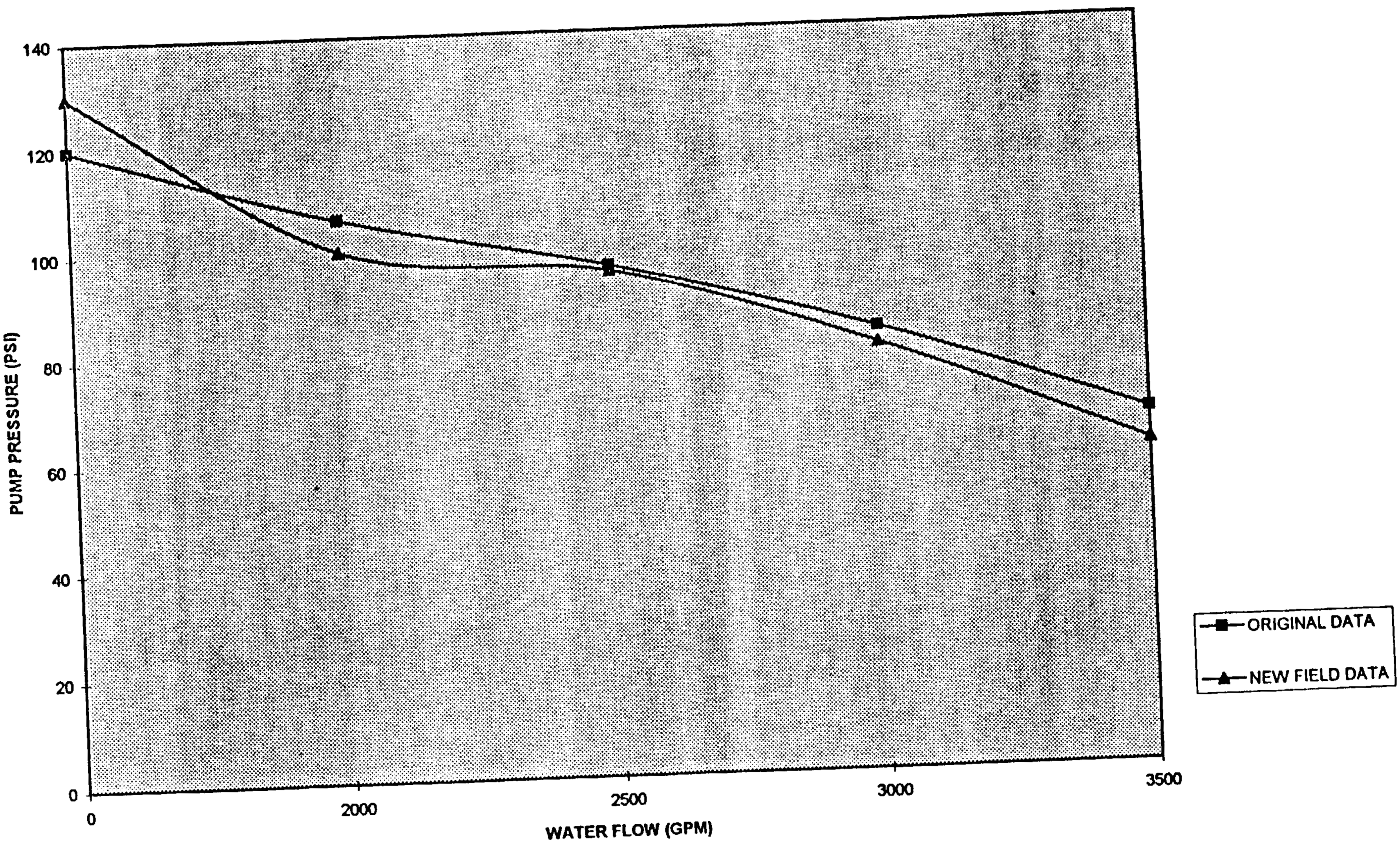
THREE-POINTS CURVE

PUMP #82-65117



ALL-POINTS CURVE

PUMP #82-65117



FIRE PUMP : 82-65118

MATHEMATICALLY GENERATED DATA

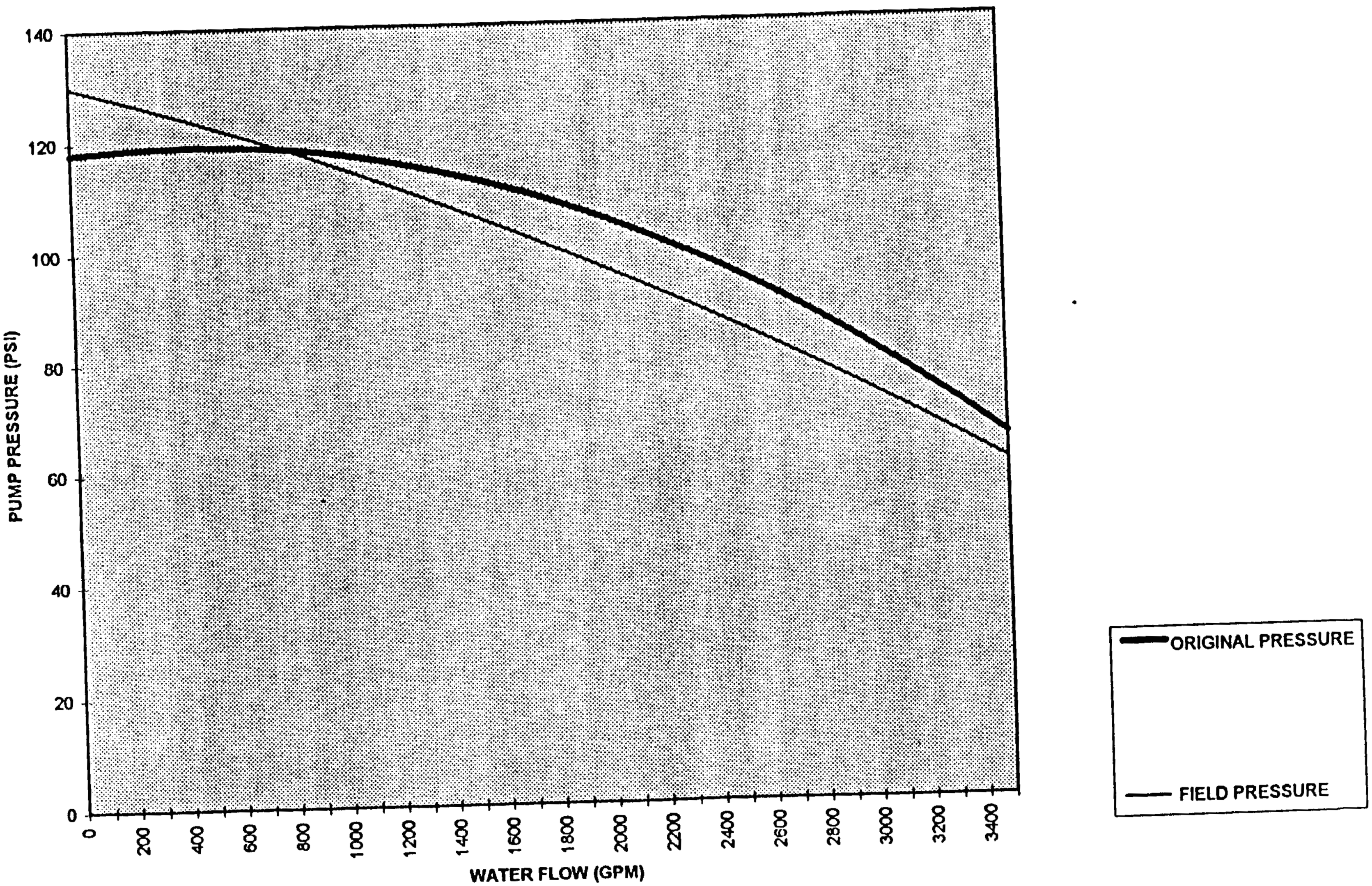
GPM	ORIGINAL PRESSURE	CALCULATED PRESS.
0	118	130
100	118.46	128.68
200	118.80	127.32
300	119.03	125.92
400	119.13	124.48
500	119.13	123.00
600	119.00	121.48
700	118.76	119.92
800	118.40	118.32
900	117.93	116.68
1000	117.33	115.00
1100	116.63	113.28
1200	115.80	111.52
1300	114.86	109.72
1400	113.80	107.88
1500	112.63	106.00
1600	111.33	104.08
1700	109.93	102.12
1800	108.40	100.12
1900	106.76	98.08
2000	105.00	96.00
2100	103.13	93.88
2200	101.14	91.72
2300	99.03	89.52
2400	96.80	87.28
2500	94.46	85.00
2600	92.00	82.68
2700	89.43	80.32
2800	86.74	77.92
2900	83.93	75.48
3000	81.00	73.00
3100	77.96	70.48
3200	74.80	67.92
3300	71.53	65.32
3400	68.14	62.68
3500	64.63	60.00

TEST DATA

GPM	ORIGINAL PRESS.	FIELD PRESSURE
0	118	130
2000	105	85
2500	95	85
3000	81	80
3500	60	60

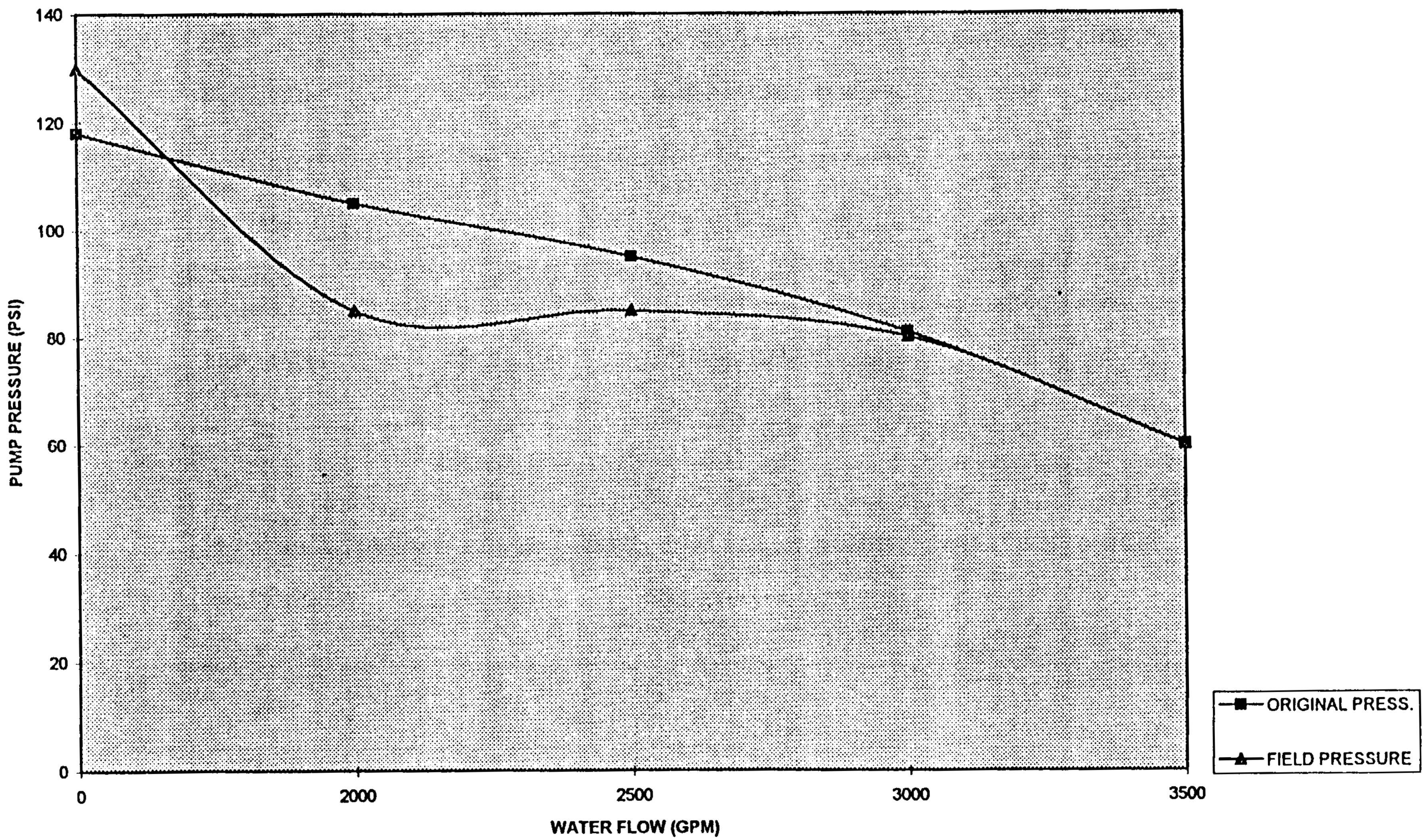
THREE-POINTS CURVE

PUMP #82-65118



ALL-POINTS CURVE

PUMP #82-65118



FIRE PUMP : 82-65115 RATED FLOW: 2000GPM

	TOTAL FLOW (GPM)	ORIGINAL NET PRESSURE (PSI)	TEST NET PRESSURE (PSI)	DIFFERENCE (PSI)	% OF DIFFERENCE (%)
0%	0	121	135	14	11.57
100%	2000	105	100	-5	-4.76
150%	3000	81	90	9	11.11

FIRE PUMP : 82-65116 RATED FLOW: 2000GPM

	TOTAL FLOW (GPM)	ORIGINAL NET PRESSURE (PSI)	TEST NET PRESSURE (PSI)	DIFFERENCE (PSI)	% OF DIFFERENCE (%)
0%	0	119	120	1	0.84
100%	2000	105	90	-15	-14.29
150%	3000	83	60	-23	-27.71

FIRE PUMP : 82-65117 RATED FLOW: 2000GPM

	TOTAL FLOW (GPM)	ORIGINAL NET PRESSURE (PSI)	TEST NET PRESSURE (PSI)	DIFFERENCE (PSI)	% OF DIFFERENCE (%)
0%	0	120	130	10	8.33
100%	2000	106	100	-6	-5.66
150%	3000	83	80	-3	-3.61
175%	3500	66	60	-6	-9.09

FIRE PUMP : 82-65118 RATED FLOW: 2000GPM

	TOTAL FLOW (GPM)	ORIGINAL NET PRESSURE (PSI)	TEST NET PRESSURE (PSI)	DIFFERENCE (PSI)	% OF DIFFERENCE (%)
0%	0	118	130	12	10.17
100%	2000	105	85	-20	-19.05
150%	3000	81	80	-1	-1.23

Replacement of the Boilers at Building 24

Due to the lack of time, I was able to check only the Part 3 of Section 15554 of the Division 15 of the specification book of the Replacement of the Central Plant Steam, Building 24. This portion deals with the erection, installation, inspection and operation of the new boilers. I compare this portion of the specification book with the codes:

ASME B31.1

ASME BPVC - Section I

29 CFR 1910 - Subpart Q

NFPA 8501 A

and every aspect is in accordance to this codes.

Goddard Space Flight Center

Use of 3D Computer Modelling in Spacecraft Development

Francisco Fernandez

Summer Institute in Engineering and Computer Applications (SIECA)

GSFC/NASA Code 735.3 8/1/95

Introduction

The development of software for controlling unmanned spacecraft is a challenging problem. In particular, embedded systems whose functions are to allow the spacecraft to autonomously control some part of itself can be difficult to produce. This is due mainly to the fact that this software may require the spacecraft to be fully assembled and functional before it can be used to prove that it actually does what it is supposed to do. However, software developers rarely until much later in the spacecraft development cycle have such an opportunity to test their work. In order to aid these developers to test their software, there are many methods for simulating working spacecraft conditions. One such method is the use of 3-dimensional computer modeling of a spacecraft. An accurate 3D model capable of displaying the results of a software system command can be a valuable tool to the developer of an embedded system program. My project for the summer under mentor Tim Leath has been to take part in the development of a 3D computer model for the X-Ray Timing Explorer (XTE).

Project Background

One of the spacecraft developed here at Goddard and set for launch in August of this year is XTE. XTE's mission is to monitor space for high energy x-ray emissions from compact stellar objects, such as suspected black holes. XTE will provide near-continuous communications to ground controllers by making use of NASA's Tracking and Data Relay Satellite System (TDRSS). This feature of XTE is accomplished through very sophisticated pieces of software and hardware.

The section I worked for this summer was code 735.3, the Flight Software section. In the Flight Software Section, developers work to provide unmanned spacecraft with the software needed to make them autonomous. One spacecraft with projects currently in Flight Software is the XTE. Developers in Flight Software Section have been working since the beginning of the XTE project to create programs to manage spacecraft communications and control the spacecraft attitude. They have also assisted in the development of ground station software that provides ground control systems with easy to understand interfaces. Unfortunately, all of these programs are designed to be used with an actual, working spacecraft, so testing them presents an interesting challenge for even the most experienced among these developers.

One of the software developers in the Flight Software Section is Tim Leath, who has acted as my mentor for the past two summers here at Goddard. Among his many other accomplishments here, Tim is responsible for developing the Antenna Manager (AM) software task for XTE. One of the most complex tasks on XTE, AM is the software that manages all spacecraft communications to the ground. It autonomously controls the spacecraft high gain antennas for tracking, as well as transmitting to and receiving messages from the TDRSS. This software, in order to function, is dependent upon information from a different system on the spacecraft, the Attitude Control System (ACS). During the early development of AM, ACS software did not exist. Therefore, it was necessary to find a way to simulate the ACS in order for Mr. Leath to test his AM software.

By that time, a 3D model of XTE had already been developed by Gurpartap S. Sandhoo, a contract employee from Advanced Technology and Research, Corp. Using a simulation development application called IGRIP, Mr. Sandhoo developed the simulation in order to perform various studies on the XTE spacecraft, including:

- Blockage of instrument and sensor fields-of-view (FOV) by the spacecraft body.
- Mission scenarios for high gain antenna management.
- Visualization of possible mechanical interference.

Through the use of this model, XTE instrument managers and the ACS team were able to determine that the high gain antennas and the FOV of some of the instruments were blocked by other parts of the spacecraft itself. Thanks to this model, these teams were able to detect and correct problems with the spacecraft early in the development.

The usefulness of this model extended far beyond simply studying the spacecraft. This model was also used by Mr. Leath to test the AM software. As the simulated XTE went about its orbit, the simulation program would extract information about the spacecraft's position and attitude and pass it to the AM software, thus acting as the ACS. AM in turn would take this information, analyze it, and send back to the simulation commands for the High Gain Antennas (HGAs). These commands would then be interpreted and the appropriate action would be displayed by the simulation software. This model proved to be so useful to Mr. Leath in his work that he became determined to develop the concept further. It has been my assignment to participate in this further development of the XTE model.

The purpose of my project this summer is to add to the existing simulation of XTE the ability to connect to a mission ground control system. With this ability, the simulation is able to obtain real data, such as XTE's position and attitude relative to the earth, and use this information to accurately depict what is occurring with the spacecraft during its mission. This project can be broken down into two parts:

- Create an internet socket connection between the simulation and the ground system.
- Modify the existing simulation program code to make use of the real data being fed to it.

The process I underwent to study and solve the two problems listed above can be found in the next section.

Project Research and Development

Modifying the existing 3D model of XTE required solving two main problems. The first of these was how to connect the model to a ground system. This connection had to be general enough to allow the simulation to connect to any machine running ground system software. The second problem was to learn enough of the simulation language to be able to modify the existing code. The difficulty was to change the program to take advantage of a ground system connection without adversely affecting the simulation. Each of these problems was addressed separately.

My first task was determine how to connect to the ground system. Developers of the ground systems had already devised ground system software, the Advanced Spacecraft Integration and Systems Test (ASIST) system. ASIST provides a windowed environment for monitoring and commanding a spacecraft such as XTE. Among its many other facilities, ASIST provides a method for software developers to connect to it and obtain telemetry. This facility is known as the Telemetry Stream Decommutated Server (TSDS). TSDS allows programmers to connect to ASIST and request specific pieces of telemetry data simply by listing the mnemonic identifiers associated with the information desired. This allows a programmer to get any piece of telemetry data needed. TSDS not only allows controllers on the ground system workstation to view telemetry data, but also allows other machines on the internet to connect to the ground system workstation and request telemetry. These connections are accomplished through the use of UNIX sockets.

To understand how the ground system connection is established, it is useful to understand what a socket is. A socket is a UNIX mechanism of communicating commands or streams of information between a client and a server. The client and server may be on the same machine or different machines connected over a network. An individual socket connects only two processes, although a process may connect to multiple processes by establishing multiple socket connections. The connection process is simple. A server process must be running. This server process opens a socket on a particular port (see **Fig. 1**). It then listens to this socket and waits for a connection request from a client process on the same port. When such a request is made, the connection is then established between the client and server (see **Fig. 2** and **Fig. 3**), and now the two processes may pass information, such as commands or data, to each other over this connection.

Socket Connection Configurations

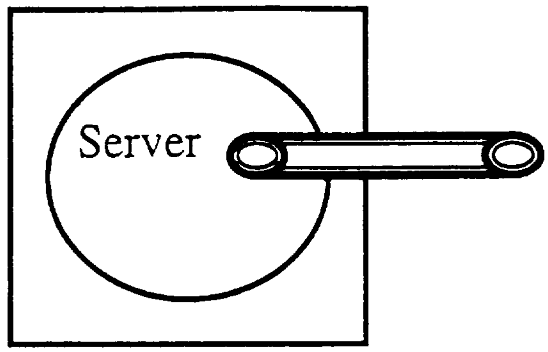


Fig. 1: Server process awaiting connect request

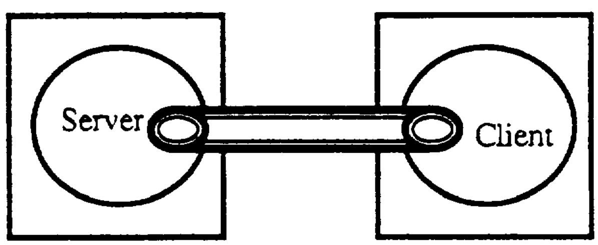


Fig. 2: Inter-machine socket connection

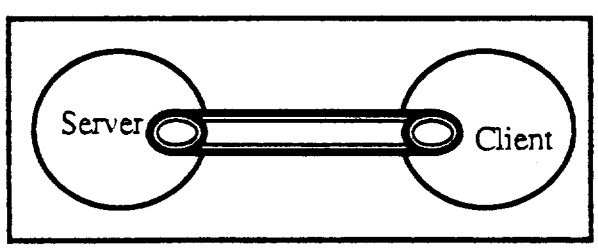


Fig. 3: Inter-process socket connection
on the same machine.

As stated above, ASIST provides a socket server process know as TSDS. When the ASIST software is started, the TSDS server is automatically begun. It opens port number 4202 and waits to service connection requests from client processes. My task was to create the client process to connect to TSDS. My first major difficulty was encountered when I tried to directly connect the TSDS and XTE simulation (see **Fig. 4**).

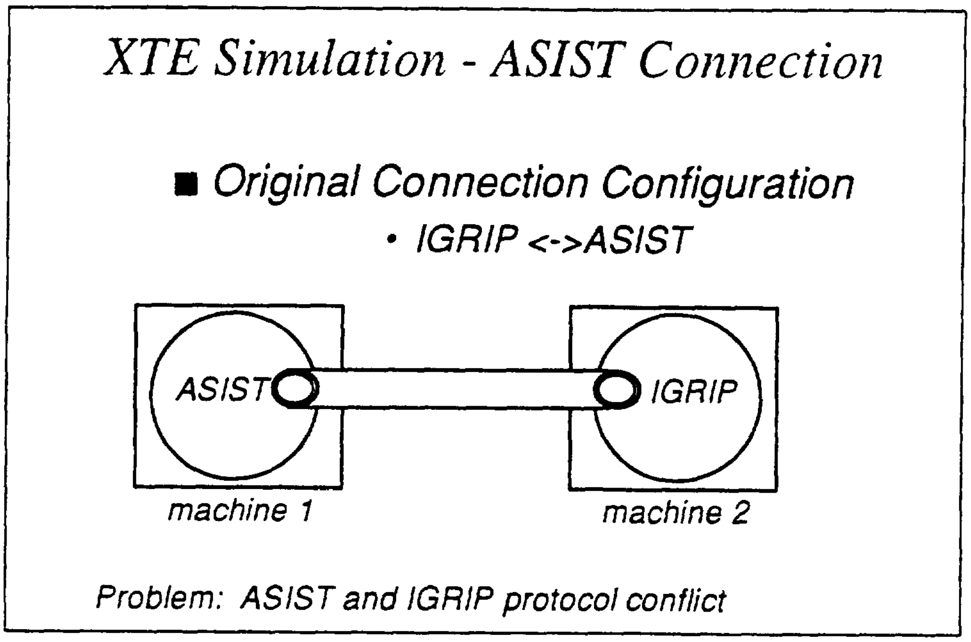


Fig. 4

The XTE simulation was developed in a simulation package called IGRIP. IGRIP provides methods for establishing socket connections itself, in either the server or client modes. In my initial attempt to connect the simulation and the ground system, I attempted to establish the simulation as a client requesting a connection to the TSDS server. This attempt failed due to a minor protocol difference. The TSDS server requires messages passed to it and sent from it to be in a specific format, as follows:

- **UUUU** (four U's in a row), as a synchronization pattern.

- *XXXX* bytes of message **length**, represented as an integer.
- ***Length*** bytes of ASCII text message.

Messages sent to the TSDS server not in this format are automatically rejected. Also, messages sent from TSDS are automatically sent in this format, with no way to change it. The IGRIP socket utilities also expect messages to be in a particular format, which is different from the one used by TSDS. The only difference is that IGRIP does not use a sync pattern in message passing. Unfortunately, this difference is enough to make the two processes incapable of directly communicating with each other successfully. In order to get around this problem I developed a client process to act as a socket relay between IGRIP and TSDS (see **Fig. 5**).

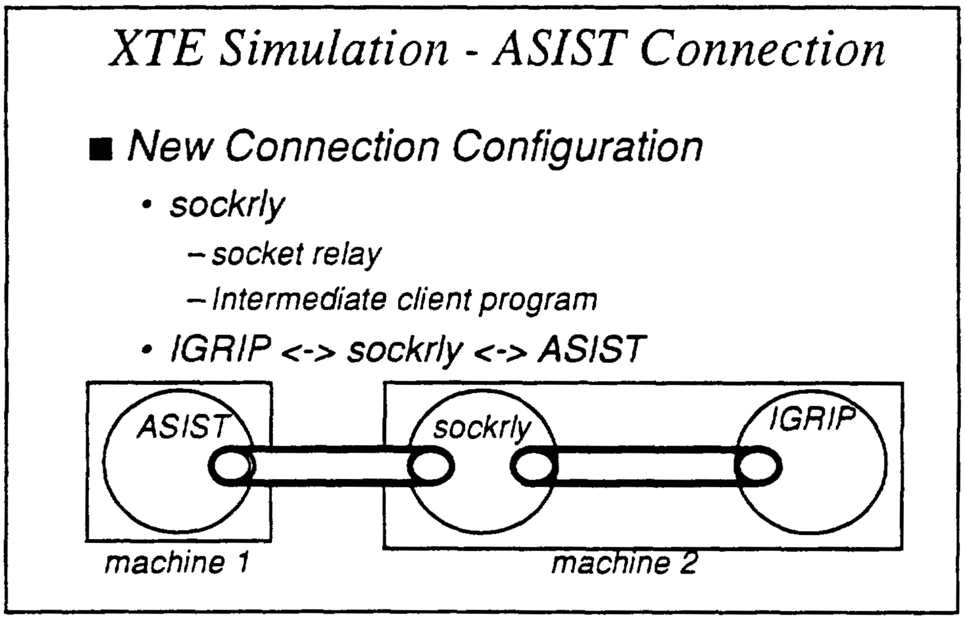


Fig. 5

The client program I developed to allow IGRIP and TSDS to communicate is called **"SOCKRLY"** (short for socket relay). **"SOCKRLY"** is a bi-directional client program written in the C programming language. It is an executable program that is run independently of IGRIP, but is used to connect IGRIP and TSDS. The program acts as a client to both the TSDS server and to IGRIP in server mode. It allows communication between IGRIP and TSDS by taking a message from one server and reformatting it to be sent to the other server. In other words, "SOCKRLY" intercepts messages from TSDS intended for IGRIP, reformats them to a form IGRIP can understand, then passes the newly formatted message to IGRIP. It undergoes the same process for messages from IGRIP to TSDS. In order to receive telemetry, however, TSDS must be provided with a list of mnemonic identifiers for the telemetry desired. "SOCKRLY" allows a user to change the mnemonics requested through the use of a script file. This script file contains the list of initializing commands to TSDS, including the list of mnemonic identifiers to request. With this intermediate program, IGRIP is able to request telemetry from TSDS, and TSDS is able to send telemetry to IGRIP.

With "SOCKRLY" developed and functional, the next problem was how to modify the existing simulation code to make use of telemetry data. As it turned out, this part was easily accomplished. The simulation was originally written as a to-scale model. Therefore, the exact telemetry values provided could be directly used in the simulation without having to undergo modifications, such as scaling. It was a simple matter of finding which telemetry values were needed as input to the simulation, then replacing the assignment statements in the original simulation with assignments to telemetry values.

Summary

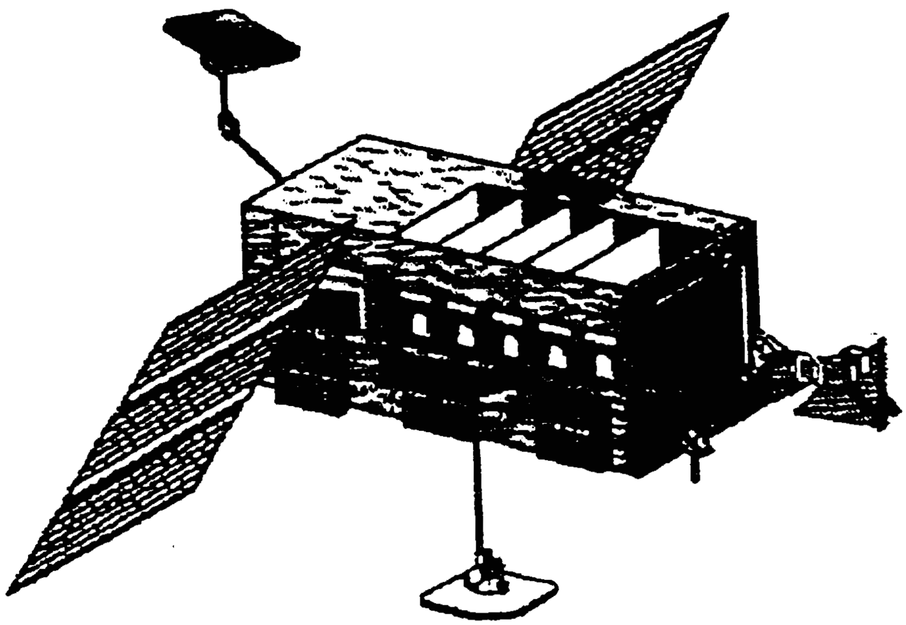
When applying for this program, the application requested that I provide an idea of the field of study I wished to examine this summer. In response to that question, I stated that I wanted to study the software engineering process. I wanted to learn the steps that occur from the time an idea is presented up to the time a working computer program for the idea is completed. My project has given me the opportunity to see these steps in great detail. I first had the opportunity to see what it is like to maintain a working piece of software whose requirements have changed. The original purpose of the XTE simulation in IGRIP was to perform studies on the spacecraft. At that time, the requirements were fairly simple. The model had to provide a to-scale 3D picture of the XTE spacecraft, show the movements of the HGA and other spacecraft instruments, and run with only a fixed orbit and attitude. Now, the users of the simulation need for the spacecraft to not be fixed in its position relative to the earth or its attitude. Additionally, the simulation should no longer run autonomously, but now needs to connect to some type of ground system to obtain real orbital data. These new requirements, though seemingly simple, require a non-trivial assessment and alteration of the original program, which is often the case with modern software, where the context of a software system can change frequently.

In addition to the maintenance of an existing software package, I had the opportunity to work on the creation of a new software package, specifically the "**SOCKRLY**" program. This program, though not as complex as most software created today, still required proceeding through the steps in the software development cycle. I had to determine the requirements for the program, develop a design, and implement the design. Naturally, testing of the program to validate and verify its correctness with respect to the requirements was also performed. Lastly, but certainly just as important as any other step, documenting of the software was needed so that future maintenance of the program would be more manageable.

The result of my work experience is not just an increase in my knowledge of the software engineering process, but also the development of a useful tool for developers of embedded spacecraft systems. With this simulation package, developers can run simulations where they can visually monitor what their software does. They can more easily find problems with their code earlier in the development process, and avoid costly bug fixes at later dates in the mission development. Additionally, this tool can be used during the spacecraft's actual mission so that ground controllers are able to more easily monitor a spacecraft's flight. They are no longer confined to trying to interpret a series of cryptic numbers, because now they have an actual picture of the spacecraft and what it's doing. Most importantly for both of the above potential users, this software is modifiable for any spacecraft mission, not just XTE. If a developer is provided with the spacecraft dimension specifications, he or she can easily model the new spacecraft in IGRIP. Then, provided the new spacecraft uses the ASIST ground station software, the model can access and use real telemetry for its display.

GSE.BASE FOR THE XTE SATELLITE.

(Ground Support Equipment For The X-Ray Timing Explorer Satellite.)



By Stacy A. Flowe

August 2, 1995

S.I.E.C.A Participant

Being a summer intern at NASA Goddard Space Flight Center, I worked hands on with advanced technology in my field of study, computer science. Dr. Jean Swank, who is an astrophysicist, is my assigned mentor. Dr. Swank and I worked out of code 666, the Laboratory for High Energy Astrophysics (LHEA). The majority of my summer was spent assisting my mentor in producing a program that ran all test data, recorded from the testing of the PCA's, in the last past three years. The running of this data will produce a data summary for scientific use.

In the beginning weeks, I became familiar with my new work environment. I took several classes at the learning center and read various books that pertained to my project. I applied the learned material by running small sample programs in the languages of C and Fortran 77. After becoming fairly comfortable at my Sun computer, I began to learn the objectives of my project. This so happened to be the general information of my mentors headed project, the (X)-ray (T)iming (E)xplorer Satellite.

The objective of the XTE satellite is to time a broad-band spectra of x-ray sources from 2-200 kev (XTE: Taking the pulse of the universe, Pg. 9). The XTE is considered to be highly maneuverable and the PCA/HEXTE can be pointed to any position in the sky on any day of the year provided the angle to the sun is 30 degrees (Pg. 20). The primary targets of study with the XTE, are systems containing compact objects. The targeting galactic objects are basically binary systems with a neutron star, white dwarf, or possibly a black-hole. The extragalactic objects of target are Seyferts, quasars, and BL-Lac type objects (Pg. 24). It has been shown that x-rays give very direct information about these bodies, making the XTE satellite ideal for this type of study.

The team for the XTE satellite consist of 8 institutions. They are organized into a Science Working group (SWG), a Science Operations Center (SOC) team and three instrument teams. The three instrument groups are the High-Energy X-ray Timing Experiment (HEXTE) supported by the University of California at San Diego (Fig. 3.2), the All Sky Monitor (ASM) supported by Massachusetts Institute of Technology (Fig. 3.1) and the Proportional Counter Array (PCA) supported by Goddard Space Flight Center (Fig. 3.3).

The Proportional Counter Array consist of five large proportional detectors with anticoincidence features that provide a very low back-ground. The PCA is sensitive to x-rays ranging from 2 to 60 keV (kilo electron volts). This instrument is supported by the Experimental Data Systems (EDS). The EDS is a microprocessor-driven data system used for on board processing of the PCA and ASM data. This system will process count rates from the PCA up to 500,000 x-rays per second with minimum loss of data.

Each PCA detector is filled with xenon and methane gas. Once a X-ray particle enters a detector, it interacts with the gases sending signals to the EDS. The EDS

Fig. 3.1

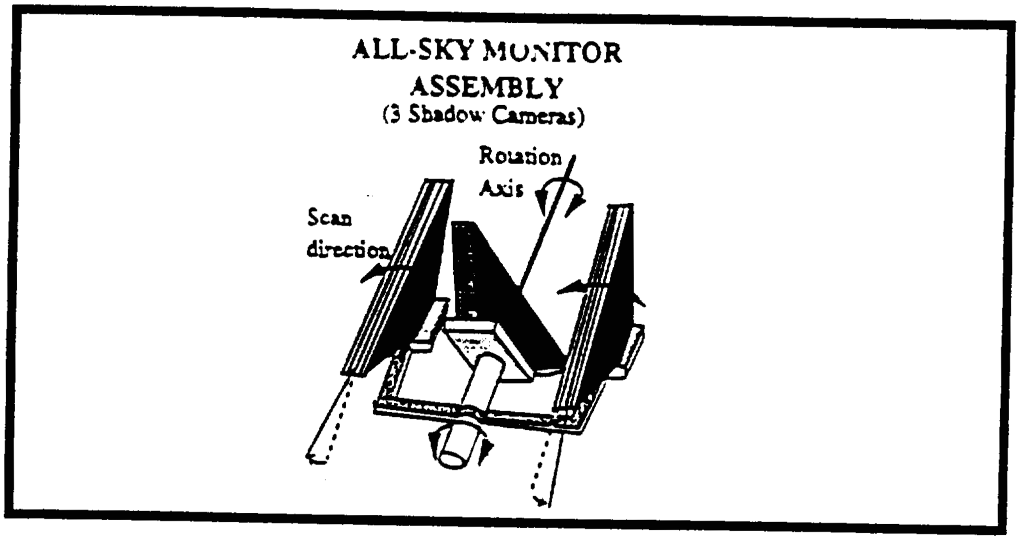


Fig. 3.2

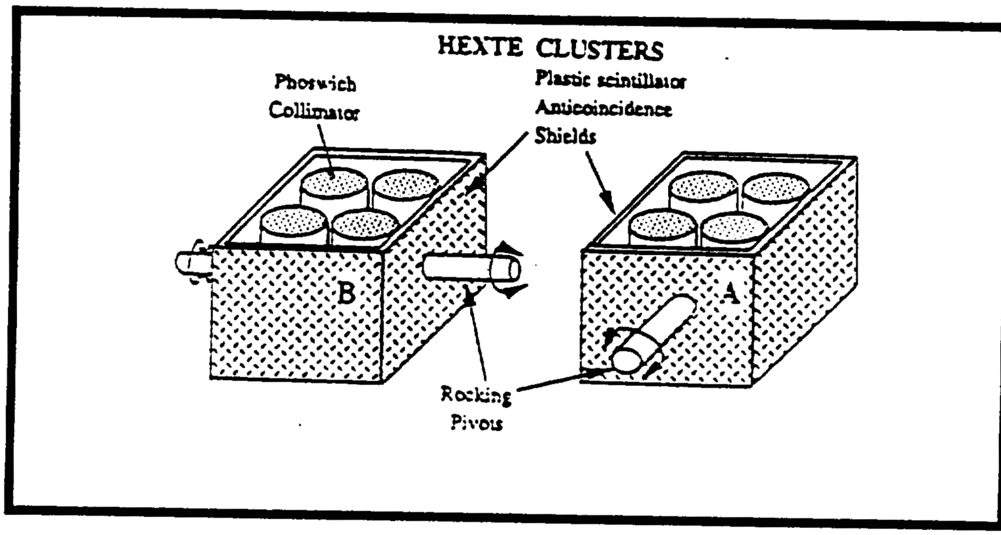
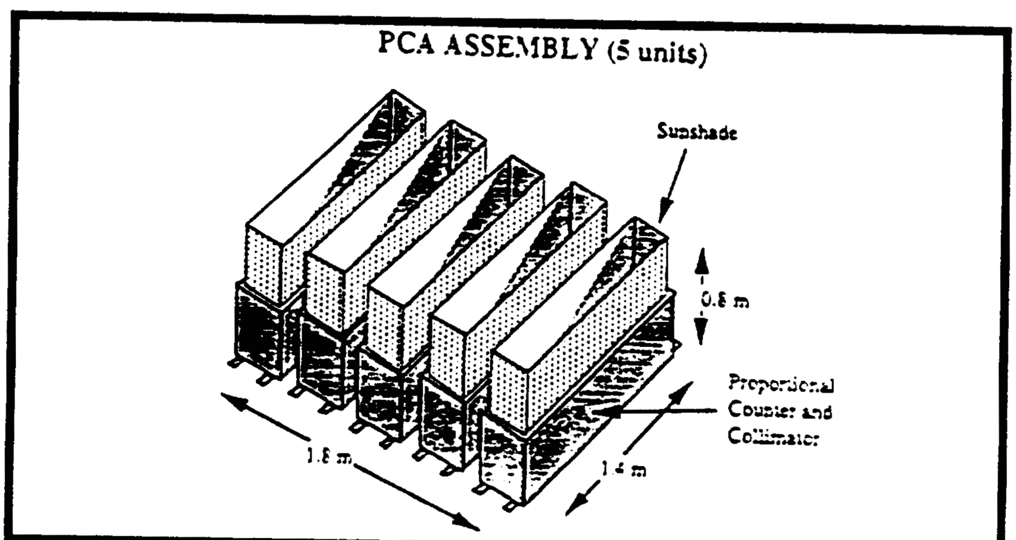


Fig. 3.3



transforms the signals into codes. The EDS sends the codes to a space-craft which sends the coding to the Trans---- Digital Relay Satellite System (TDRSS). This satellite then sends the code to NASA spaceflight center at Goddard to be translated into a computer readable code and here it is stored. The noise caught in transfering the data is coded out. The Data is received and recorded, on the ground, slight differently.

Each individual detector was placed in a vacuum for testing. Inside of the vacuum, the detector was exposed to a radioactive source emitting x-ray particles. The x-rays and the gases interact as they would in space. The signals are transferred by wiring, to a computer which has a simulator board placed in it. This simulator board would be a replica of the EDS on board the satellite and this board codes the signals before storing the data into files of the computer.

Every 600 kilo bytes, a new file is created and each file is named with a number sectioned into three parts. The first part is the: month, day, year, the second section is the: hour, minute, second and the last section is the file number which increases by five. This data stored from PCA testing, is the data that our program must run and record in a form readable to LHEA scientists.

The program is named gse.base and it is an acronym for (G)round (S)upport (E)quipment.Data(B)ase. The entire program is based on formulae written with Paw, a programming tool for physicist (These formulae were constructed by my mentor and her co-worker). The primary Computer language used is Fortran. The program is written to ask the user to enter a file number. Once the number is read correctly, the original program would list the number of files processed, the total number of events processed, the number of time tags encountered, and the number of (V)ery (L)arge (E)vents (Fig 6.1). Because the PCA's test data recorded more significant information than the original program printed out, we made modifications.

To assure proper usage of the test data, each layer of the detector was assigned two variables, one for reading the number of VLE's in each layer and the other for reading the rate at which these events occurred in each layer. By putting the layer format statements in a programing loop, a layer chart was produced for each PCU that recorded data in a file. Histogram formulae were added for each PCU but their output printed to a separate file. By dividing the number of events for each detector by the number of seconds, we calculated the rate for each layer. These modifications allowed the program to list more in-depth data facts. The updated program now prints out the previous information, the overall dead time (time ital), the very large events dead time, the number of events for each layer in each PCU and the rate in each layer of each individual PCU (Fig. 6.2). For detailed viewing of the final program, refer to Appendix A.

Currently, we are in the process of loading the test data files into a database on

The Original Program Sample Output

```
 Total number     of events processed:  144501
 Total number of time tags encountered:    14219.000000000
 Total number of VLE  events  received:  164
nll=  64050  nvll=  64050
 PCU=  0          # of Events=  0
 PCU=  1          # of Events=  0
 PCU=  2          # of Events=  0
 PCU=  3          # of Events=  0
 PCU=  4          # of Events=  0
 PCU=  5          # of Events=  144501
 PCU=  6          # of Events=  0
 PCU=  7          # of Events=  0
```

Fig. 6.1

The Final Program Sample Output

```
940528_TV_H9_QCM/052894_112015_1.arc
 Total number of files processed:    229
 Total number of events processed:  19544266
 Total number of time tags encountered: 16643894.
 Total number of seconds of data:     17043.
 PCU=  4            # of Events=  1158500352
 Total rate (c/s) =      439.174
 Dead time =        7
 VLE dead time=    136
              L1         R1         L2         R2         L3         R3         VP
 No.=       540738     292442     344897     212151     323252     211111     354067
 Rate=        31.7       17.2       20.2       12.4       19.0       12.4       20.8
 PCU=  5            # of Events=  1167108096
 Total rate (c/s) =      707.565
 Dead time =        7
 VLE dead time=    145
              L1         R1         L2         R2         L3         R3         VP
 No.=      1687816    1401335    1117799     958973    1012527     877599    1038094
 Rate=        99.0       82.2       65.6       56.3       59.4       51.5       60.9
```

Fig. 6.2

HEARSAC. This database will be available through browse of the databases. By loading test files into the database, the data will be more flexible to the individual user. The user will no longer be bound to the structure of the program. For example, the user will be able to compare and contrast various pcu's by their individual number of events. Through the database, the user will also be able to create plots of information. These to, will also be flexible to the individual scientist's interest as well as making the test data easier to comprehend.

The test data is based on x-rays from our atmosphere and known x-ray sources. The sole purpose of the test data was to check if the detectors were functioning properly. However, the test data can also be used to be compared with the data from the actual XTE satellite. For example, one layer of a PCU may show high levels of x-ray detections, the test data may be used to tell whether this is normal or possible for this layer of a PCU. The test data can also be used for scientific research and will be used as a summary for the PCA project testing.

The XTE satellite is currently passing all ground test. The launch date is set for the end of August. Until then, the test data, recorded in the last past three years, will be available for browsing in the PCAGSE DATABASE, under HEARSARC soon. The test data has been found to be very informative to the study of x-rays as they pertain to the universe.

It has been an experience working with this XTE satellite team. I think the entire project will be very fruitful. My summer with NASA has been filled with bursts of helpful information. My mentor and her colleagues were friendly and gave me several educational challenges that I feel can only better me. I received opportunities to learn and practice skills that were not available to me before (See Appendix B).

Not only within the S.I.E.C.A program, but throughout the facility, I met distinguished individuals having interest parallel to mine. My summer with NASA at Goddard Space Flight Center is one worth recommending to any college level student striving for advancements in Computer applications.

Works Cited

XTE Satellite Team. "Taking the Pulse of the Universe," (X-Ray Timing Explorer 1992), "Brochure."

APPENDIX A

THE FINAL GSE.BASE PROGRAM

```
      program gse_base
c     Version for multiple detectors
      implicit none

      integer HMAX
      parameter (HMAX=1200000)
      real h
      common /pawc/ H(HMAX)

      integer*4 MAXWD
      parameter (MAXWD=158722)
      integer*4 in(MAXWD)
      integer*4 ipcu                    ! PCU id, 7 being time tag rollover.
      integer*4 iflag                   ! event ID flag (0 to 511).
      integer*4 vle                     ! vle flag, either 1 or 0.
      integer*4 vxpha                   ! VX layer PHA.
      integer*4 xepha                   ! Xenon layer PHA.
      integer*4 tbits                   ! the 10-bit time of the current event.
      real*8 ntag,Nsec
      real*8 iclk                       ! clock reading of current event.


      real   vecl1(256)
      real   vtdif(1024),vtvle(1024), chi,chimin, rdt
      real   Trate, rl1,rr1,rl2,rr2,rl3,rr3,rvp,rvle
```

yer Rate – Variables

```
      integer*4 i,j,k

      real*8 iclk_last(7), iclk_vle(7)
      real   tdif(7), vle_tdif(7)

      integer*4 detID(0:7)

      integer*4 IMM,IDAY,IYR,IMIN,IMAX,JULDAY
      integer*4 nfile, nevent

      integer*4 nvle(7), nbits,
     &          nvl1, nvr1, nvl2, nvr2,nvl3, nvr3, nvvp
```

ayer Event – Variables

```
      integer nwd, nrd, rd_err

      integer jdt,jdtvle

      parameter(nwd=200000)
      integer in(nwd)
      character dname*80

      call hlimit(hmax)

      write(0,*) ' Enter directory name:'
      read(5,'(A)') dname
```

] File Number Read

```
      nfile = 0
      nevent = 0
      ntag = 0
      call hbook1(98,'Julian Day', 1, 0.0, 1.0, 0.0)

c     Initialization for each detector

      do k=0,6

      detID(k) = 0
      nvle(k)=0
      nvl1=0
      nvr1=0
```

```
      nvl2=0
      nvr2=0
      nvl3=0
      nvr3=0
      nvvp=0
```

Paw Formulae for file Out Put

```
      call hbook1(4+k*10,'Time Intervals',1024,0.0,1024.0,0.0)
      call hbook1(5+k*10,'Time Intervals from VLE',1024,0.0,1024.0,0.0)
      call hbook1(6+k*10,'Flags      ', 512, -0.5,511.5, 0.0)
      call hbook1(7+k*10,'Number of Flags', 10, -0.5, 9.5, 0.)
      call hbook1(101+k*10,'L1 PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(102+k*10,'R1 PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(103+k*10,'L2 PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(104+k*10,'R2 PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(105+k*10,'L3 PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(106+k*10,'R3 PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(107+k*10,'VP PHA Spectrum', 256, -0.5, 255.5, 0.0)

      call hbook1(201+k*10,'L1+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(202+k*10,'R1+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(203+k*10,'L2+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(204+k*10,'R2+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(205+k*10,'L3+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(206+k*10,'R3+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      call hbook1(207+k*10,'VP+Alpha PHA Spectrum', 256, -0.5, 255.5, 0.0)
      enddo

 100  continue
c     Read files and get data and process into histograms

      call rd_dir(dname,nwd,in,nrd,nfile,rd_err)
        if(rd_err.LT.0) then
           write(0,*) ' Error in rd_dir:  rd_err=', rd_err
           stop
        endif
      do i = 1, nrd
         call parser(in(i),ipcu,iflag,vle,vxpha,xepha,tbits)
         if(ipcu.eq.7) then       ! a time tag roll over
           if(nevent .GT. 0) ntag = ntag + 1.000
           goto 200
         endif

         DetID(ipcu) = DetID(ipcu) + 1

         nevent = nevent + 1
         iclk  = tbits + (ntag-1)*1024
         if(vle.NE.0) then
            nvle(ipcu) = nvle(ipcu) + 1
            iclk_vle(ipcu) = iclk
         endif

         tdif(ipcu) = iclk-iclk_last(ipcu)
         vle_tdif(ipcu) = iclk-iclk_vle(ipcu)
         iclk_last(ipcu)=iclk
```

Paw Formulae Calculations

```
         call hf1(4+ipcu*10,tdif(ipcu),1.0)
         call hf1(5+ipcu*10,vle_tdif(ipcu),1.0)
         call hf1(6+ipcu*10,real(iflag),1.0)
         call hf1(7+ipcu*10,real(nbits(iflag)),1.0)

         if(iflag.EQ.1 ) call hf1(101+ipcu*10,real(xepha),1.0)
         if(iflag.EQ.2 ) call hf1(102+ipcu*10,real(xepha),1.0)
```

-Additions

```
      if(iflag.EQ.4 ) call hf1(103+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.8 ) call hf1(104+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.16) call hf1(105+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.32) call hf1(106+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.64) call hf1(107+ipcu*10,real(xepha),1.0)

      if(iflag.EQ.129) call hf1(201+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.130) call hf1(202+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.132) call hf1(203+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.136) call hf1(204+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.144) call hf1(205+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.160) call hf1(206+ipcu*10,real(xepha),1.0)
      if(iflag.EQ.192) call hf1(207+ipcu*10,real(xepha),1.0)

 200     continue
      enddo                          ! end of i loop: processing one buffer.
      if(rd_err.NE.0) goto 100       ! It's the end if rd_err=0.

c     write out histograms in file for paw
      call hrput(0,'gse_base.hst','N')

      open(10,file='.tmp')
      read(10,'(A)') Dname
      close(10)
      DO I=80,1,-1
         if(Dname(I:I).EQ.'/') then
            IMIN = I
            GOTO 342
         endif
      ENDDO
 342  CONTINUE
      IMIN = IMIN+1
      IMAX = IMIN+6
      read(Dname(IMIN:IMAX),'(3I2)') IMM,IDAY,IYR
      IYR=IYR+1900
      call hf1(98,0.5,real(JulDay(IMM,IDAY,IYR)-2400000))

      Nsec = ntag*1024.0E-6

c     write out general header results

      write(6,'(A)')Dname
      write(6,*) ' Total number of files processed: ', nfile
      write(6,*) ' Total number of events processed:', nevent
      write(6,340) 'Total number of time tags encountered:', ntag
 340  format(A39,F10.0)
      write(6,345) 'Total number of seconds of data:', Nsec
 345  format(A33,F10.0)

c     calculate and write out results by detector

      do k=0,6
      If(detID(k).ne.0)then

         call hunpak(101+k*10,vecl1,'HIST',0)
         do j=1,256
         nvl1=nvl1+vecl1(j)
         enddo

         call hunpak(102,vecl1, 'HIST',0)
         do j=1,256
         nvr1=nvr1+vecl1(j)
         enddo

         call hunpak(103,vecl1, 'HIST',0)
         do j=1,256
```

Histogram Formation Done With Paw

PCU 1

PCU 2

PCU

```
nvl2=nvl2+vecl1(j)
enddo
```

3

```
call hunpak(104,vecl1, 'HIST',0)
do j=1,256
nvr2=nvr2+vecl1(j)
enddo
```

P.C.U 4

```
call hunpak(105,vecl1, 'HIST',0)
do j=1,256
nvl3=nvl3+vecl1(j)
enddo
```

P.C.U 5

```
call hunpak(106,vecl1, 'HIST',0)
do j=1,256
nvr3=nvr3+vecl1(j)
enddo
```

P.C.U 6

```
call hunpak(107,vecl1, 'HIST',0)
do j=1,256
nvvp=nvvp+vecl1(j)
enddo
```

P.C.U 7

These two P.C.U's are located at the [illegible] of the Satellite

```
call hunpak(94,vtdif,'HIST',0)

call hunpak(95,vtvle,'HIST',0)
```

```
Trate = detID(k)/Nsec
rl1 = nvl1/Nsec
rr1 = nvr1/Nsec
rl2 = nvl2/Nsec
rr2 = nvr2/Nsec
rl3 = nvl3/Nsec
rr3 = nvr3/Nsec
rvp = nvvp/Nsec
rvle = nvle(k)/Nsec
```

Calculations for each Layers Peak

```
rdt=Trate*1.0e-6

        jdt=0
        chimin=1.0e+20
        do j=5,40
                chi=0.0
                do i=1,j
                chi=chi+vtdif(i)**2
                enddo
                do i=j+1,60
                chi=chi+(detID(k)*rdt*exp(-(i-j)*rdt)-vtdif(i))**2
                enddo
        if(chi.lt.chimin)then
        chimin=chi
        jdt=j
        endif
        enddo


        jdtvle=0
        chimin=1.0e+20
        do j=10,600
                chi=0.0
                do i=1,j
                chi=chi+vtvle(i)**2
                enddo
                do i=j+1,900
```

```
                    chi=chi+(nvle(k)*rdt-vtvle(i))**2
                    enddo
              If(chi.lt.chimin)then
              chimin=chi
              jdtvle=j
              endif
              enddo


        write(6,*) ' PCU=',k,'              # of Events=', DetID(i)

        write(6,*) ' Dead time =     ', jdt

        write(6,*) ' VLE dead time= ',jdtvle

        write(6,350)'L1','R1','L2','R2','L3','R3','VP','VLE'      Output
350         format(5X,7(9x,a2),8x,A3)
        write(6,360)' No.= ',nvl1,nvr1,nvl2,nvr2,                   Format
     *                  nvl3,nvr3,nvvp,nvle(k)
360         format(A6,8i11)

        write(6,370)' Rate=',rl1,rr1,rl2,rr2,rl3,
     *                  rr3,rvp,rvle
370          format(A6,8f11.1)

        endif

        enddo          End Loop

        call exit(0)
        end
```

APPENDIX B

ACCOMPLISHMENTS

Summer Internship Accomplishment Fact Sheet

Name: Stacy Alicia Flowe

School: Bennett College, rising junior
900 E. Washington St.
Greensboro, NC 20745

Major: Computer Science

Program: Summer Institute in Engineering and Computer Applications, undergraduate

Skills Accomplished for Project

- General use of the Sun computer
- General use of the Unix system
- Basic programming in the C language
- Basic programming in the Fortran language
- Basic use of a Database
- FrameMaker

Courses Completed at the Learning Center

- ANSI C Language for Programmers
- Introduction to the Unix World
- SQL TUTOR
- UNIX Software Tools for Programmers

Awards

Sister program mentor (Plaque received)

World Wide Web Experience

- Netscape
- Mosaic

NASA
Goddard Space Flight Center
Greenbelt, Maryland

Final Report

Roberto Gómez Báez
[redacted]
SIECA (UG)
Mentor: Dr. MacDowall and Dr. Hess
Code 695
August 2, 1995

Pattern Recognition

The project that I have been assigned deals with the daily data received from the spacecraft Ulysses. Ulysses is a joint mission between the European Space Agency (ESA) and NASA. To understand the Sun's environment and its influence on the Earth, the Sun not only has to be studied around the ecliptic plane, which is the plane where the Earth and other planets of our solar system orbit around the Sun, but also around the solar poles. This is the goal of the Ulysses mission, the first spacecraft to navigate over the solar poles of the Sun.

Several types of observations are made aboard Ulysses, including those of the Unified Radio and Plasma Wave Investigation (URAP). This is an experiment to detect both distant radio emissions (via remote sensing), as well as locally generated plasma waves. Dr. Robert MacDowall is the principal investigator of URAP.

The low band of the radio receiver of the URAP experiment measures 64 frequencies in the range from 1 to 48 KHz. The data are averaged over some time period to produce a series of spectra that are used to analyze and display the data. These spectra are represented on a plot as a function of time against frequency. The time ranges from 0 to 24 hours and the frequencies from 0 to 48 KHz (see Fig. 1). These plots are a representation of a two dimensional array with 64 lines and 600 columns. This array contains the data recorded from the radio receiver in Ulysses for a single day.

One important quantity that can be determined from the radio spectra is usually the "plasma frequency", (fp) which is the natural period of oscillation of a plasma. Once this number is obtained, a scientist can determine the density of the plasma which is a fundamental parameter. However, determination of fp requires careful analysis of the radio spectra.

The data received from Ulysses not only contains the quasi-thermal noise, which is the factor that creates the spectra of the plasma, (see Fig. 2) but it also contains different types of interference that inhibit accurate measurements of the fp. One is Type III Solar Bursts from the Sun, which appear as emissions coming from high frequencies. Another one is Jovian radio emissions, (produced by Jupiter). Changes in operational mode in the spacecraft create interference in the data. These three types of interference, among others, "pollute" the data making it very difficult to create a computer program to recognize the plasma line to obtain fp. Without any interference in the data; recognizing the pattern of the plasma line and obtaining fp would be easy.

To find fp for every day of data it was decided to write a computer program to determine fp from the URAP spectra. Dr. MacDowall developed a simple computer program that worked most of the time but failed when interference, such as the ones mentioned before, is present in the data. The program has been coded in IDL to have easy access to plots and visual analysis of data.

IDL or Interactive Data Language is a complete computing environment for the interactive analysis and visualization of data. It is an array oriented language which provides easier computations with arrays. IDL also provides easy and fast image processing, as well as graphical displays. There is a variety of pre-defined functions and procedures to handle image processing in IDL such as smoothing, translations, and median filtering among others. An easy to learn syntax allows the user to concentrate on something specific rather than waste time in system design and program development. For this and other reasons IDL was the language chosen to code the program. On the other hand the disadvantage of IDL is that if a program is not coded properly, execution can be very slow. Sometimes even while coded properly is slow. This is due to IDL having an interpreter instead of having a compiler.

The primary stage of the program started by scanning each spectrum starting at the highest frequency (48 KHz) going all the way down to the lowest frequency (1 KHz) or until a peak that satisfied the criteria established in the program was found. The peak in each spectrum (see Fig. 2) is fp in that specific time. To decide if something is a peak or not the program has to go through certain tests such as how high the peak is, how smooth it is, how much tolerance is going to be allowed, and whether or not it is due to interference. All these things have to be carefully coded and validated on a daily basis to ensure accurate interpretation of the data.

Eleven different problematic days were chosen to test the program and determine its limitations. Dr. MacDowall's approach of scanning from high to low frequencies failed many times when interferences such as Jovian Emissions and Type III Solar Bursts, (which are present on high frequencies or above the plasma line) were present. These interference emissions are very common and the problems they caused had to be overcome. The program was not smart enough to decide if the right peak was being calculated; as a consequence, peaks due to interference were being obtained instead of fp.

The new approach to improve fp calculation was to scan each spectrum from low to high as well as from high to low frequencies. This idea was based on the fact that most interference originates above the plasma line even though there is sometimes some interference present below it. The results obtained were very promising. In some cases the high scan, (H) would get the right plasma frequency; in others the low scan, (L) and sometimes both, H and L, would get the same fp. At this stage the problem was to code certain instructions in the program so it could decide whether H or L was right and pick fp.

With this in mind the idea of the 3-way approach was developed. The 3-way approach decides whether H or L scanned the right fp based on a previous fp. The previous fp must be correct in order for the 3-way approach to work. Once a starting fp is obtained in a spectrum the next spectrum is scanned by H and L. If both H and L have the same value that fp is chosen. If they have different values H and L are subtracted from the previous fp. Then the absolute value is determined for H and L. Doing this, the distance of H and L from previous is deduced and the one closer to previous is chosen as fp.

With this change the output of the program improved; however, the results were not as good as hoped. For example, sometimes L would not find a peak, (this is sometimes due to the low intensity of the peak) but H would find one and if it was an interference peak it would be chosen even though it is not fp. To make things worse this incorrect fp would then be used when scanning the next spectrum. The scan from there on would be in error making the 3-way approach by itself unreliable. This is where the idea of setting a limit came into work.

The plasma line is normally a very smooth line. From one spectrum of the array to the other the plasma frequency does not change dramatically except in the cases of shocks which do not occur often. To set a limit, observations were made to deduce approximately how far apart in frequency fp in one spectrum is in relation to fp in the next spectrum. After a series of tests the limit was set to be 1.2358.

The setting of a limit for the plasma line to move around from spectrum to spectrum made the program "smarter". For example, if it was scanning from high to low and it found a peak at 42.50 KHz and the previous peak was at 17.50 KHz, the peak at 42.50 KHz would be set to be a non-detection denoted by -1. The scanning would start again from 42.50 downward looking for the fp peak. From a previous value of 17.50 KHz the highest it can expand is 21.63 KHz, (previous * limit) and the lowest is 14.16 KHz, (previous ÷ limit). As a result the fp for the next spectrum should be between the range of $14.16 \leq fp \leq 21.63$. If a number between this range is not found fp for that spectrum is set to be -1, which means fp could not be found.

Sometimes during execution the program still got confused and picked interference instead of staying on the path of the line. It is what I called "The Ladder Effect". If interference was too close to the plasma line, as sometimes happened, (see Fig. 3) the scanning would catch this interference and the previous value would be set incorrectly again and instead of following the plasma line it would follow the interference. To cope with this problem instead of setting only one previous value two previous values were set. This was found to eliminate "The Ladder Effect". The original value set for limit was still good enough to compare the range of separation of three consecutive spectrums.

The program still needs to be further improved to work for conditions that were observed in the data such as the plasma line going into a Type III Burst and when more than one shock wave occurs in a day. When the plasma line runs together with the Type III, which is not very usual, the scanning follows the Type III instead of the plasma line giving erroneous results. At this moment, the program is not set to outsmart this situation. The effects of shock waves make the frequency change from one spectrum to the other by twice the previous value making the limit approach inadequate in these conditions. A quick solution to this problem is to set the limit to 2.0; however, this limit would catch too much interference and in some cases would lead to the wrong path. An algorithm was modified to deal with the problem of the shock waves. When a shock wave is detected the limit is increased to 2.0 in order to get the value after the previous value. Once this value is obtained the limit is again set to its original value. This algorithm needs to be improved because it only works for data that contains one shock and there are many days that have more than one shock. This becomes a problem when consecutive days are run and the execution depends on two previous values. Once the previous values are incorrect the rest of the days will most probably be mistaken also (see Fig. 4).

In an attempt to reduced the interference in the data to make the scanning easier and more reliable Dr. Roger Hess created a new binary data set. The new data set produces a spectrum every 122 seconds instead of every 144 seconds. This data set will create a two dimensional array of 64 lines and 675 columns, which is 75 more columns than the original array. Then a pre-defined function in IDL called "smooth" was used to smooth the data and remove some of the interference. Smoothing the array column by column could damage the plasma line. That is why the array was smoothed row by row, smoothing three points at a time this way the plasma line would not be altered. The smooth function takes the average of the three points. Using the eleven experimental days, test runs were made on the new data set. Some days changed very noticeably, improving the scan of the plasma line, while others barely changed,(see Fig. 5).

To see the behavior of the plasma line and interference together in a single plot a two- dimensional binary array was created. This array contains only zeroes,(0) and ones,(1). A one indicates the program found a peak and a zero means no peak was found. This way, all the interference and the plasma line could be observed closer for further investigation. The idea behind this is to find a way to create an algorithm to make the computer see where the plasma line is. A solution such as running a histogram of the array might tell where the line is; however, to this date it has not been tested. There is a certain pattern that the ones follow to form the plasma line and another to form the interference. If this can be expressed in an algorithm together with the program to obtain fp the whole problem of determining the right fp could be solved.

After my internship Dr. MacDowall will be working on an efficient way to detect where the plasma line lies in the plots to overcome the problems the program is facing. A possible and very promising approach will be the use of artificial intelligence (neural nets) to identify where is the line in the plots. Since the pattern of the plasma line can be determined easily by eye, some people believe that feeding the data into the neural network will identify in most of the cases where the plasma line lies. Later the program can be used to get fp.

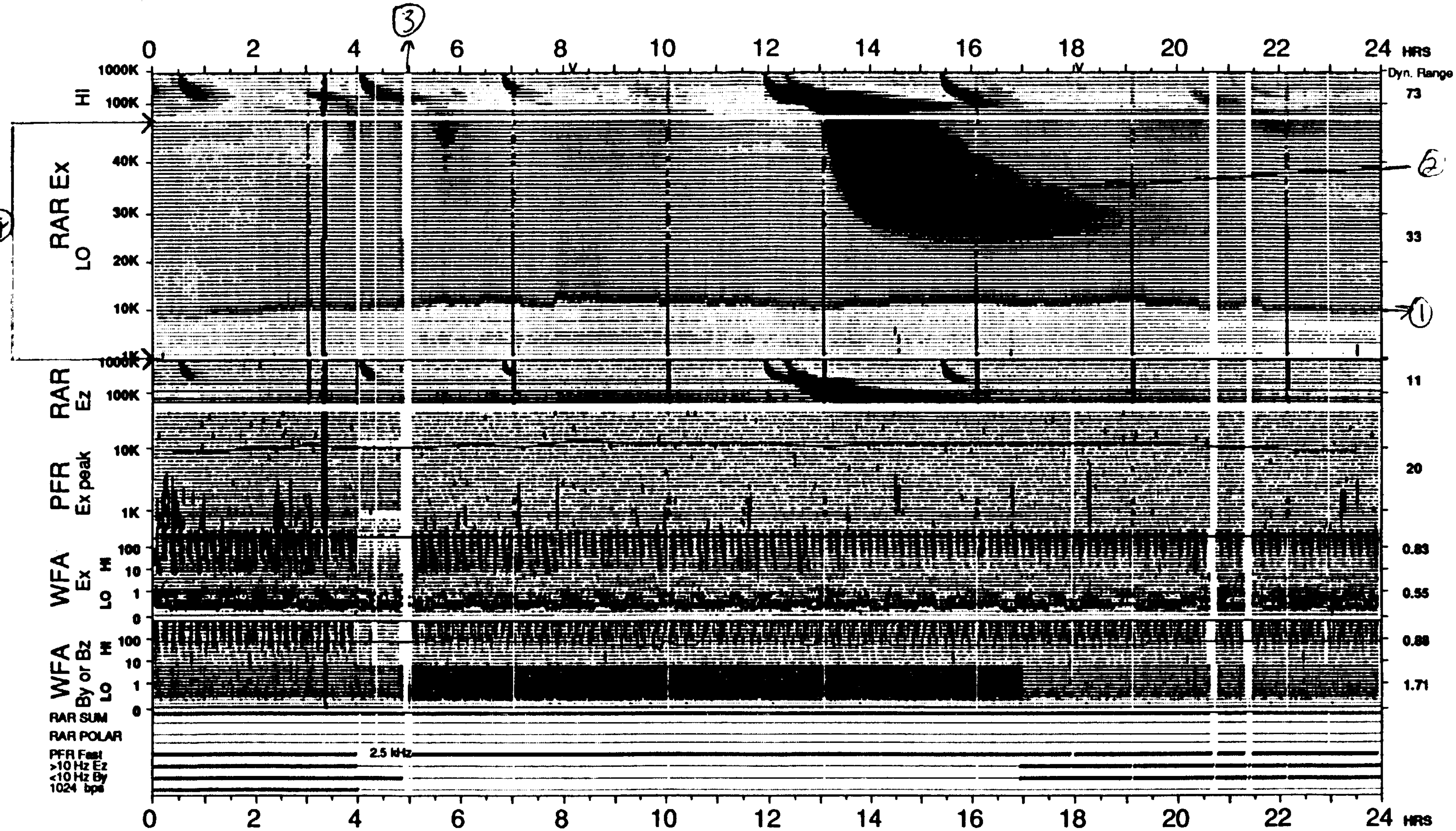
Figure 1 explanation

1) Plasma line as represented in the plots.

2) Type III Solar Burst

3) Changes in operational mode of the spacecraft

4) Frequency range



1991-FEB- 1 DAY- 32
ULYSSES URAP SUMMARY PLOT v2.28
3-MAR-1995 20:48 /91026_91035./DECOM v02.12

RAR Hi: M #11 -- M #11
RAR Lo: L -- L
RAR Bkg Type/Offset: 1 / 2. 3. 1.
RAR Lo min,max: 70. , 112.

U-S = 1.93AU U-J = 3.48AU U-E = 1.05AU
E-S-U (deg)= 18.9 S-U-E (deg)= 17.8
J-S-U (deg)= 16.1 E-U-J (deg)= 137.4
LON_H(deg)= 125.7 LAT_H (deg)= -2.3
LON_E(deg)= 112.8 LAT_E (deg)= 2.0

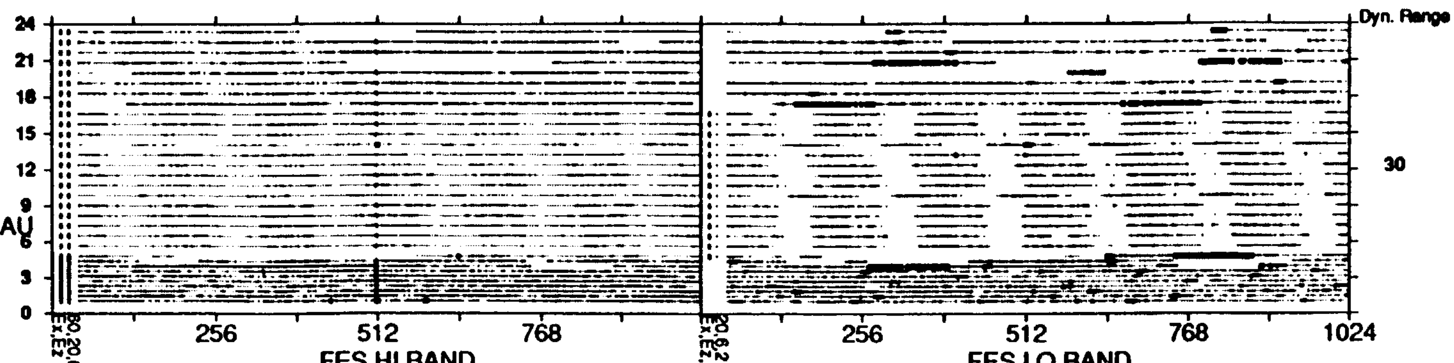


Figure 2 explanation

1) This is the quasi-thermal noise spectrum. The spectrum is somehow different from the interference peak (2) in intensity, slope, and symmetry. However, if the quasi-thermal noise spectrum could be set apart from interference fp would be easier to find.

2) interference peak

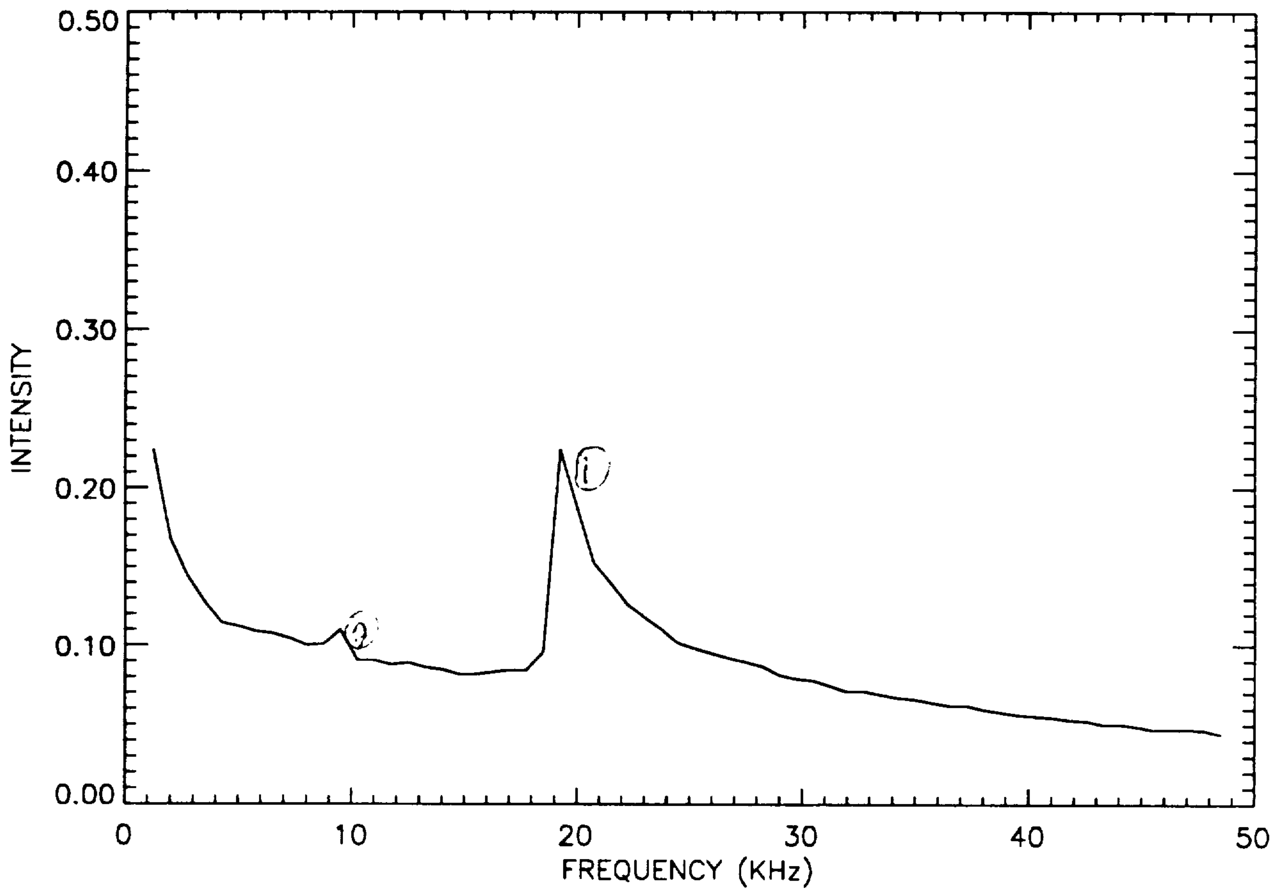

0.50
0.40
0.30
0.20
0.10
0.00
INTENSITY
0
10
20
30
40
50
FREQUENCY (KHz)
i

Figure 3 explanation

The plus signs (+) that are highlighted represent the plasma line. The rest represent interference peaks for this specific day. This graph shows a two-dimensional binary array of the 144 seconds interval data set. A one means a peak was found (which is represented in the graph as a plus sign) and a zero does not graph anything. The points in the graph mean the peak was found with L and the plus signs with H.

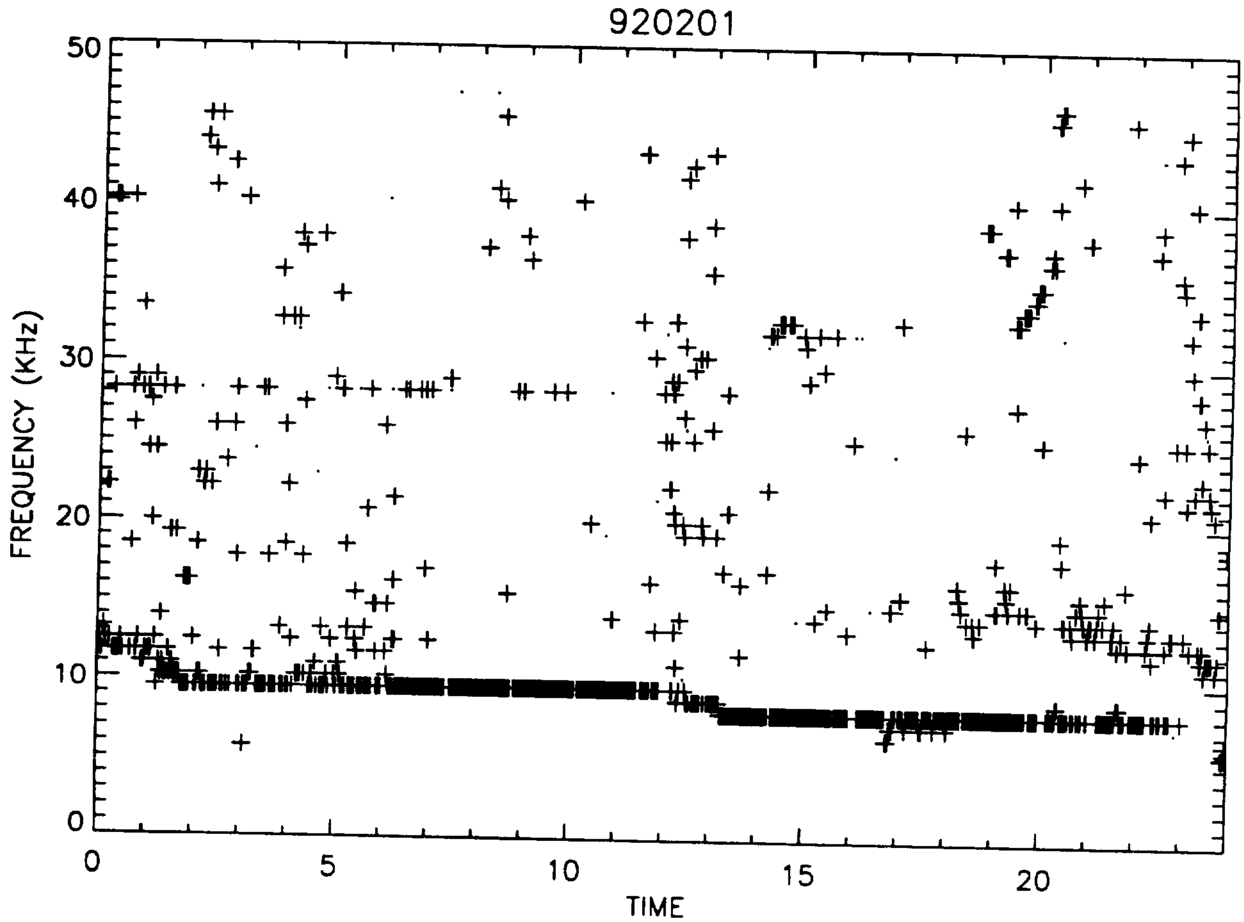
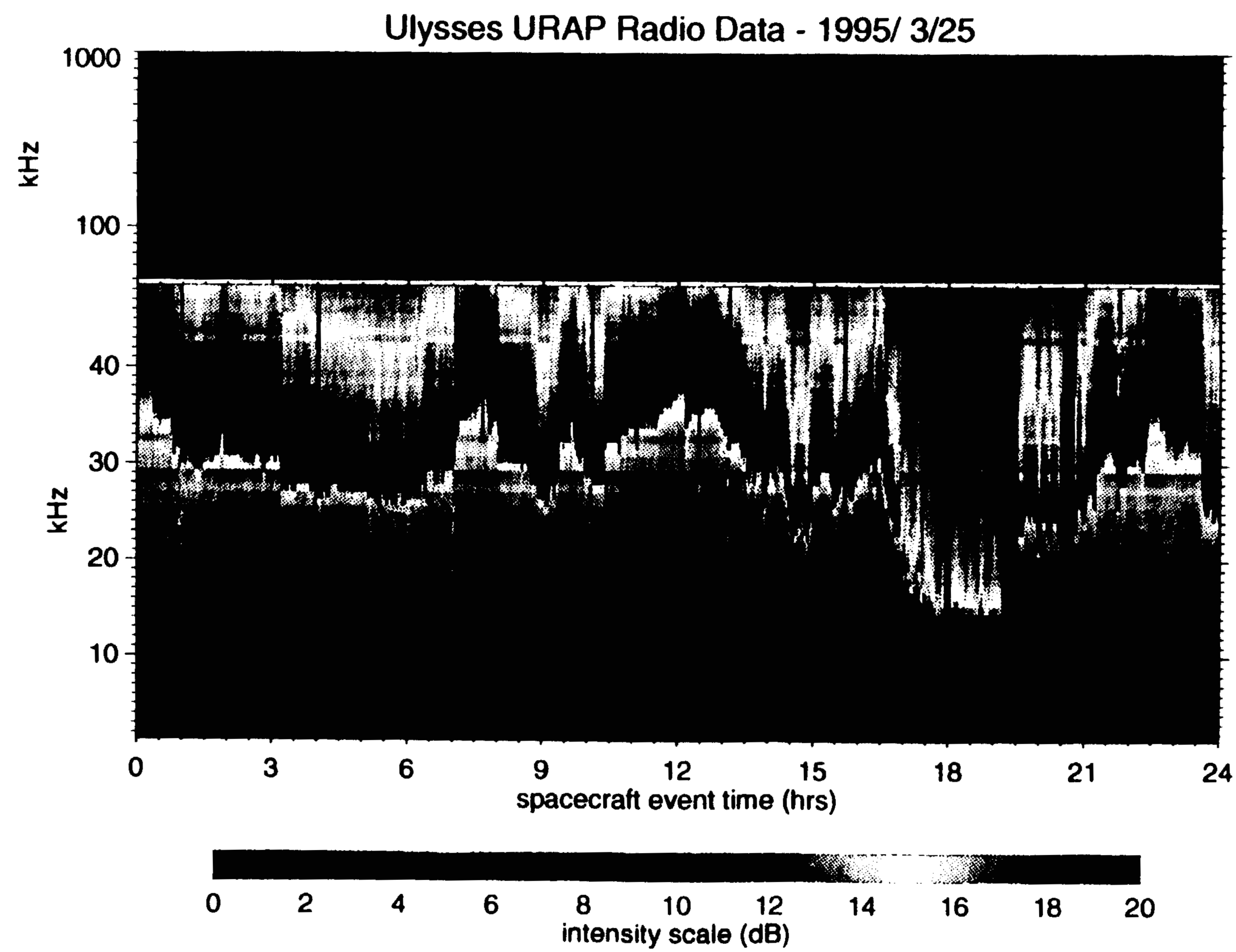

920201
FREQUENCY (KHz)
TIME
0
5
10
15
20
10
20
30
40
50

Figure 4 explanation

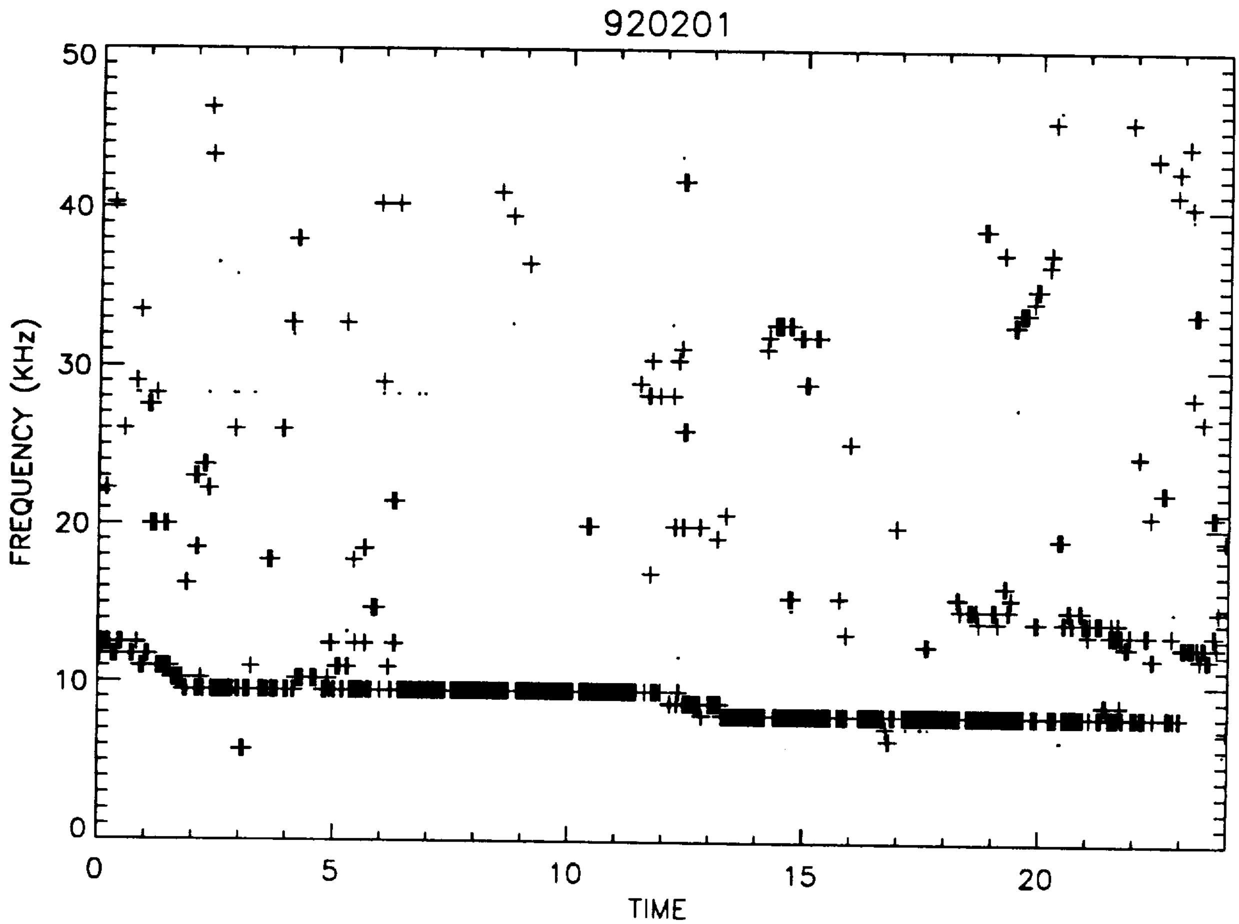
This is a color plot for the day 03/25/95. With the help of color tables the intensity of the plasma line can be observed clearly. In this case the intensity of the line is quite visible and strong (in red). At approximately 19:30 hours a shock wave can be seen. From a frequency of approximately 16 KHz suddenly the frequency changed almost 2X to about 28 KHz. When a shock wave is in the plasma line more than once, execution of the program is halted. Enhancing of the algorithm to handle one shock will most probably handle this situation.



Ulysses URAP Radio Data - 1995/ 3/25
1000
100
40
30
20
10
kHz
kHz
0
3
6
9
12
15
18
21
24
spacecraft event time (hrs)
0
2
4
6
8
10
12
14
16
18
20
intensity scale (dB)

Figure 5 explanation

Figure 5 is the same day as figure 3; however figure 5 is a binary representation of the 122 seconds interval data set smoothed with the smooth function. A comparison of figure 3 and 5 shows that there is less interference around the plasma line in figure 5. This improves the scanning of the fp.


920201
FREQUENCY (KHz)
TIME
0
10
20
30
40
50
5
15

Electromagnetic Interference (EMI) and Hewlett's Packard's Visual Engineering Environment (HP VEE)

Marvin Jackson

National Aeronautics and Space Administration (NASA)

Summer Institute in Engineering & Computer Applications (SIECA)

Electromagnetic Interference (EMI) and Hewlett Packard's Visual Engineering Environment (HP VEE)

Marvin Jackson
National Aeronautics and Space Administration (NASA)
Summer Institute in Engineering & Computer Applications (SIECA)

ABSTRACT

Electromagnetic Compatibility is the ability of systems, subsystems, circuits, and components to function as designed, without malfunction or unacceptable degradation of performance due to electromagnetic interference (EMI), within their intended operational environment. Electromagnetic Interference (EMI) consists of any unwanted, spurious, conducted, and/or radiated signals of electrical origin that can cause unacceptable degradation of systems or equipment performance.

Two types of tests that are performed to test for incompatibility of systems are: Emissions and Susceptibility. Conducted and Radiated Emissions Tests are used to 1.) measure the levels of narrowband and broadband conducted emissions which may exist on the D.C. power lines and return leads of the test sample, and 2.) demonstrate that the levels of electric and magnetic field emissions do not exceed the specified limits from the test sample and interconnecting cable. Conducted and Radiated Susceptibility Tests are used to 1.) demonstrate the ability of test samples to survive a(n) conducted A.C. sinusoidal ripple appearing on the input power lines, and 2.) ensure that the test sample does not exhibit any degradation of performance, malfunction, or undesirable effects when immersed in an electric field.

With the use of an 8904A Multifunctional Synthesizer, a General Purpose Interface Bus (GPIB), and Hewlett Packard's Visual Engineering Environment (HP VEE) software package, automation of radiated susceptibility tests will facilitate the laborious, manual testing of test engineers. HP VEE, an iconic programming language, is optimized for instrument control to simplify test procedure setup. The objective is to implement HP VEE in the testing environment to provide the capabilities of easy collection, analysis and presentation of data in other programs for further processing.

Key Words: Electromagnetic Interference (EMI), Electromagnetic Compatibility (EMC), HP VEE (Hewlett Packard's Visual Engineering Environment), Radiated Susceptibility

Acknowledgments

The author would like to extend his appreciation to the people who assisted in the compilation of this report. Throughout this summer experience these individuals have aided in increasing my acumen of engineering. Be a participant in Summer Institute in Engineering and Computer Applications (SIECA) has allowed me to take note to a talent that was prominent.

The author would first like to thank his mentor: Angelo Wade, who has not only been great in assisting me with my summer project but has me consciously aware of the advantage and the opportunities available to me in life. You have advised me wisely about my career choices and made me cognizant of the potential in which I possess. Mr. Wade I thank you and may God bless you.

The author also would like to thank the following individuals for their assistance, no matter how small: Ted Dyer, for his advice along the way through my project; Mary Kitterman, thanks for the parallel explanations of EMI, good luck and I wish you well in your recovery; Gerry Taylor, thanks for the morning and afternoon comments to keep my day interesting; Vaughn Nelson and Bob Bender, now that I'm gone your instruments are safe.

And lastly, I would like to thank the program coordinators: Dan Krieger, Joan Langdon, and Mary Lampe, for a job well done. Keep up the good work.

Introduction

The growth of the electrical industry has led to the utilization of almost the entire electromagnetic spectrum to perform many of our tasks. The proliferation of solid-state electronic products has created an electromagnetic environment wherein the probability of mutual electromagnetic interference (EMI) from man-made interference sources increase. Additional to man-made sources, effects due to lightning and other natural interference sources must be considered.

The existence of these interference sources within the same environment may cause adverse effects on electronic systems, subsystems and circuits. There is a requirement for electromagnetic compatibility (EMC) among electronic products in order for them to operate as so properly defined without malfunction. Conducting both emissions and susceptibility tests on these devices will provide information on whether an device is capable to produce electromagnetic waves capable of causing a degradation of performance by another system, and/or determine if the operation of the device is capable of being affected by electromagnetic interference.

While conducting such test manually becomes laborious to human operators, implementation of software programs may prove to facilitate the procedure. Automation of EMI tests will give the operator more control over the test in regards to rate, instrument control and data analysis. For example, Hewlett Packard's Visual Engineering Environment (HP VEE) is an iconic programming language which increases productivity by shortening the time it take to solve engineering problems Incorporation of software programs such as HP VEE will provide the test engineer with accuracy and manipulation of data.

This paper will provide the following:

A definition and background of both EMC and EMI.

A discussion of the test setup of an radiated susceptibility test (RSO3).

A description of HP VEE's function and capabilities.

Applications employed by HP VEE.

EMC and EMI

Electrical, electromechanical, and electronic equipment all must comply with specifications intended to assure electromagnetic compatibility (EMC), which is the capability of electronic systems to operate in the intended electromagnetic environment at designed levels of performance and efficiency. To obtain electromagnetic compatibility one or a combination of the following is required:

Sources of EMI have been sufficiently suppressed

Coupling Paths have been severed or sufficiently reduced

Victims have been sufficiently hardened

The italicized words above are the three essential elements for an EMI situation to exist. The noise source emission can be either a conducted voltage or current, or an electric or magnetic field propagated through space. The sources of EMI can be classified as man-made or natural. Some examples are: motor operations, digital circuit switching, lightning, and antenna excitation. It can be so noted that some equipment or systems can serve as both sources or victims. Victims are those electrical devices which are have a potential of being susceptible to electromagnetic waves in their environment. Methods of coupling between sources and victims can be divided into two basic groups: radiation or field coupling by electromagnetic wave propagation through space or materials, and coupling via conducting paths through which current can flow. The first page of the collection of picture shows a basic illustration of the three elements.

Electromagnetic Interference (EMI) is an unwanted electromagnetic disturbance. There are two forms of EMI: Conducted and Radiated. Conducted EMI is a conducted interfering signal defined by an undesirable voltage or current coupled into a signal, power, or pertinent conductor. Radiated EMI is a radiated interfering signal defined by an electric and/or magnetic field amplitude and frequency spectrum intentionally or unintentionally radiated into space by an electrical source.

All EMI problems can be broken down into the following two categories: Emissions and Susceptibility. Emissions are the conducted or radiated energy emanating from equipment or system. Susceptibility is when equipment or system is being affected adversely by conducted or radiated energy. Susceptibility can also be defined as an anomalous behavior in the system. Test are performed to characterize the electromagnetic emissions and susceptibility of the unit under test to determine if it will be a source of electromagnetic interference or be susceptible to EMI when integrated into a spacecraft.

Radiated Susceptibility

The second page of the collection of pictures is a outline of the small EMI facility with the test set up of a radiated susceptibility test. The test setup is as follows: the unit under test is grounded to a facility copper ground. The unit's external cables are intact (shielded, wrapped, etc.) and a radiating antenna is placed one to three meters away from the unit. A radio frequency source is used to produce radiated fields (through the antenna) to illuminate the unit. The response of the unit to the radiated fields is monitored. This test simulates environments where broadcast or other radiated radio frequency fields can be a source of interference. If the unit is susceptible, frequency and field strength thresholds are determined.

Focusing again on the diagram of the test set, the specific test used is RSO3. This test determines susceptibility to electric fields in the frequency range of 14kHz to 18Ghz. The main components to focus on in the diagram are: Signal Generator, Leveling Pre-Amp, E-Field Monitor, E-Field Probe, and Transmitting Antenna.

Hewlett Packard's Visual Engineering Environment (HP VEE)

HP VEE is an iconic programming language optimized for instrument control and produces dramatic reductions in test development time. With HP VEE you create programs by connecting icons together using the mouse; with a textual language you use keywords following rules of syntax. The result in HP VEE resembles a data-flow diagram, which is easier to use and understand than traditional lines of code. Pages three (3) and four (4) illustrate the set up of both a textual programming language and HP VEE working environment. The following is a summarize, detailed description of the available functions and capabilities of HP VEE.

Data Collection and Generation
HP VEE allows you to collect data from files, standard input, and external programs and applications. HP VEE has data generators to simulate the data which normally would be gathered externally.

Data Analysis and Handling
Objects are available for commonly used math operations, as well as for calculus, regression analysis, data filtering, probability, statistics, and much more. For solutions which require long mathematical formulas, a formula object is also provided. With this object, you can specify the number of inputs and type in the formula, rather than constructing it from individual math objects.

Data Storage and Presentation
The graphical objects are highly customizable, allowing you to define the number of traces, trace colors, grids, axis, scaling, and markers for easy analysis. With HP VEE, the data gathered or generated through analysis may be printed, stored, or sent to other programs or packages for further processing.

Flow and User Interfaces
HP VEE allows you to control the flow of data where complex solution require repetitions, conditionals, and flow control objects. A common test development method, particularly for manufacturing applications, is to specify a series of tests with pass/fail

parameters, and execution sequencing based on the results of each test. HP VEE includes a built-in sequencer object which allows you to specify test parameters, pass/fail conditions, and results- based sequencing much as you would lay out a test plan on a sheet of paper. This object lets you define execution sequence (continue, retest, stop, goto, call a sub-test, etc.) depending on whether the test passed or failed a limit, range, or tolerance. HP VEE's sequencer includes the ability to automatically log the executed test sequence and test results, making the collection of data as easy as pushing a button. Two different views, Detail and Panel views, provides an easy way for a developer to build a user interface. The Detail view is an editing environment in which applications are built. The Panel view allows an end user to focus only on those objects that are of interest by removing extraneous from the screen.

Modularity and Code Re-use and Memory Management
UserObjects are objects or systems created for use within HP VEE. They can aid in modularity and to simplify the task of debugging. These objects can be registered and become UserFunctions. A UserFunction lets you create a master object that you can access from many places throughout your program. The Global Variable capability allows you to define single values, arrays, waveforms, etc. in a single place to be accessed or modified from many different places in your program.

Platforms and Compatibility
HP VEE offers a high level of integration and access to the host platform/operating system. Access features: Shared data with other applications, Call external subprograms, Execute external programs or OS commands. Applications developed on one platform may be loaded and executed on any of the other platforms with no modifications if the application is entirely self-contained.

Debugging and Documentation
The Line Probe feature shows which objects are connected by a specific line, in addition to the type and value further examination at the specified points. Show Data Flow and Show Execution Flow help you verify the correctness of your solution by allowing you to see the data flow or the sequence in which objects are executed. A Note Pad is available for writing descriptions, instructions, or comments about sections of your application. Show Description allows you to attach comments and descriptions to individual objects. ALL HP VEE programs are saved as ASCII files that can be read by others.

Applications

There were two applications which employed the use of HP VEE. The first application requested that a program be created to calculate the scan time of a radiated susceptibility test based upon MIL-STD-462D specifications. The second application of HP VEE was to create a program that functions like the Leveling Pre-Amplifier in the figure on page two(2).

The first applicaton of HP VEE created a program that allowed user input of: ending and beginning frequency, step size factor, scan rate factor. The program is located on the fifth page in the collection of pictures. The scan rate and step size factors are determined by the table located in the upper right corner. HP VEE's formula objects allowed the capability of writing the formulas for our calculations. Located in the lower right corner is a calculation of the time needed to run a RSO3 test over the frequency range of 14 kHz to 18 GHz. Based upon the calculations of HP VEE and formulas in the table, it take approximately five hours to run the test.

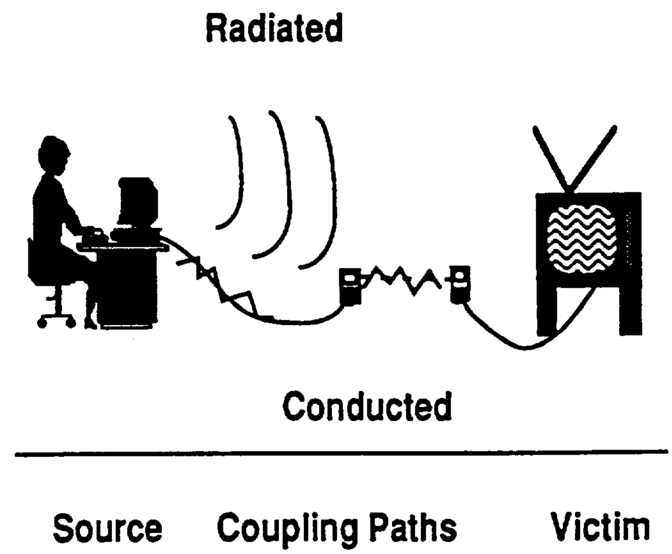
The second application of HP VEE created a program that would replace the function of the Leveling Pre-Amplifier. The objective of this device is to read recorded data from the E-Field Probe and the E-Field Monitor. The data then is tested against a preset E-field range. It the data falls within the range it allows the signal generator to step to the next frequency. If the data is outside the minimum and maximum bounds of the preset range, the amplitude is adjusted coming from the signal generator until the E-field about the test sample is within the preset range.

range, the amplitude is adjusted coming from the signal generator until the E-field about the test sample is within the preset range.

The Radiated Susceptibility Leveler program is located on the sixth page of the collection of pictures. This program carries on the same function as described in the above paragraph. However, this program gives the capability to control the rate at which the test run. It also provides the ability to continue the test beyond the 1 GHz limit of the Leveling Pre-Amplifier. This program will provide the test engineer with better accuracy and the capability to place the data into a file and transport the data either into a spreadsheet or a database.

Interference Situation

"Nuisance" Interference



Probable Annoyance

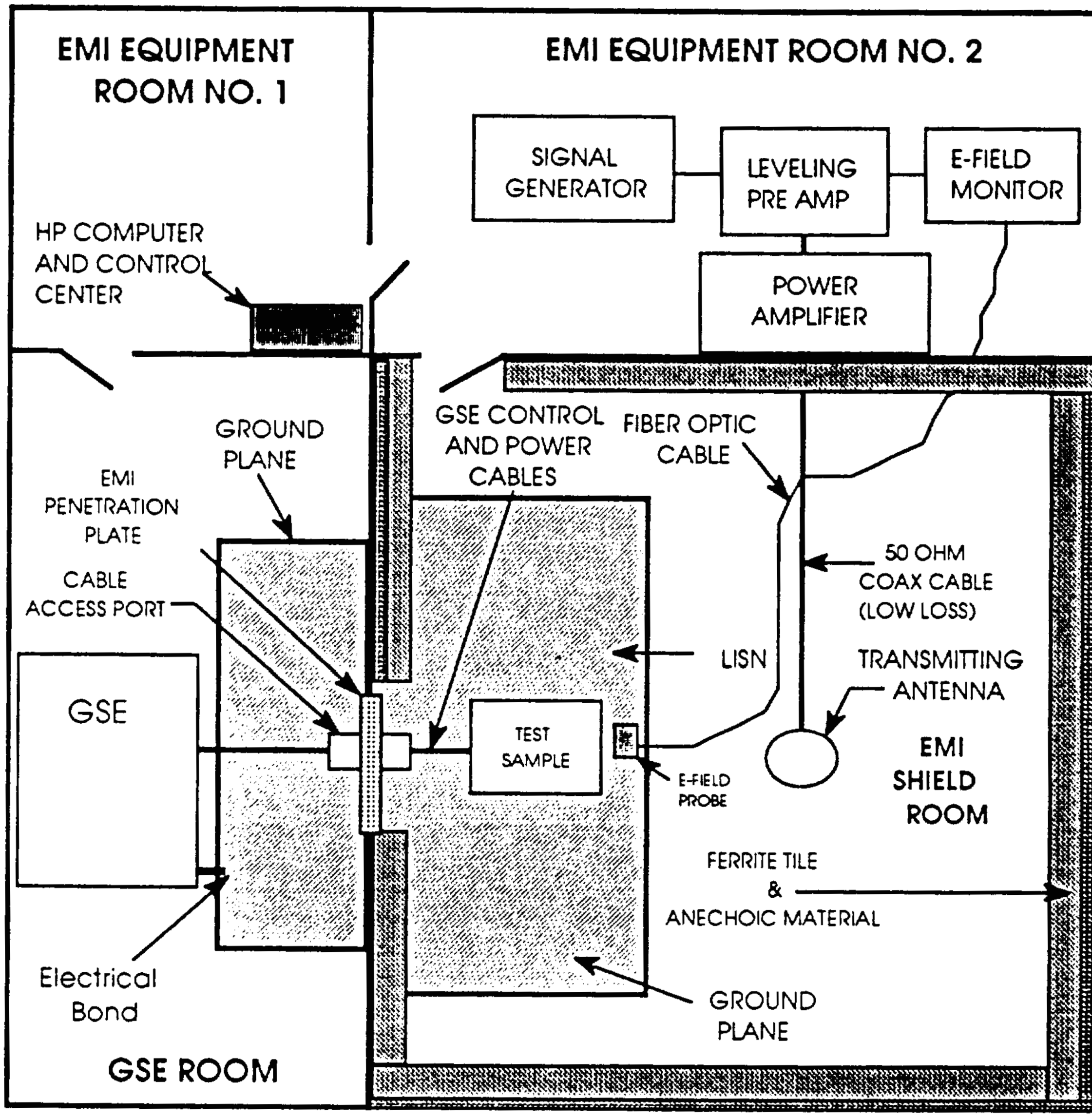
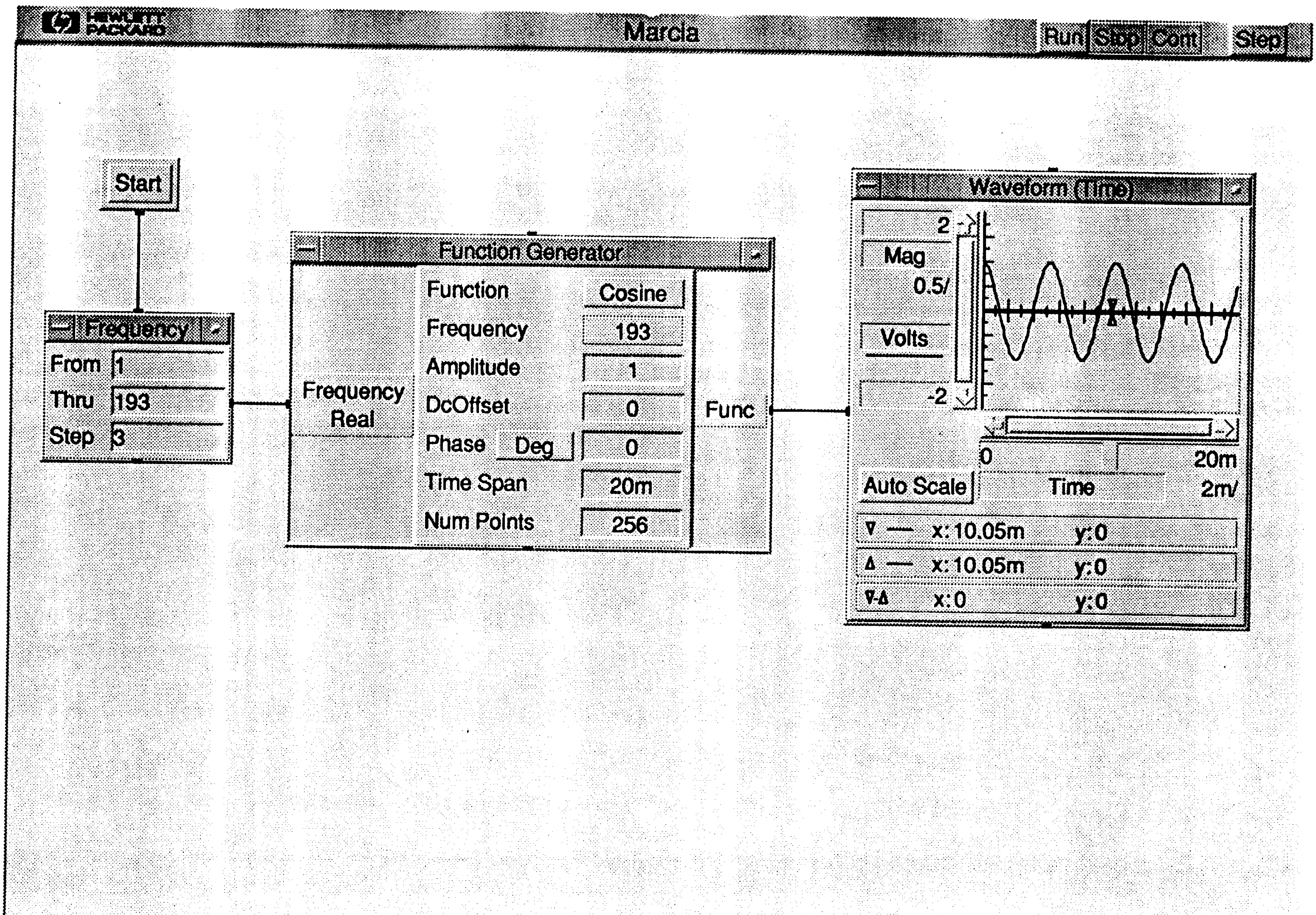
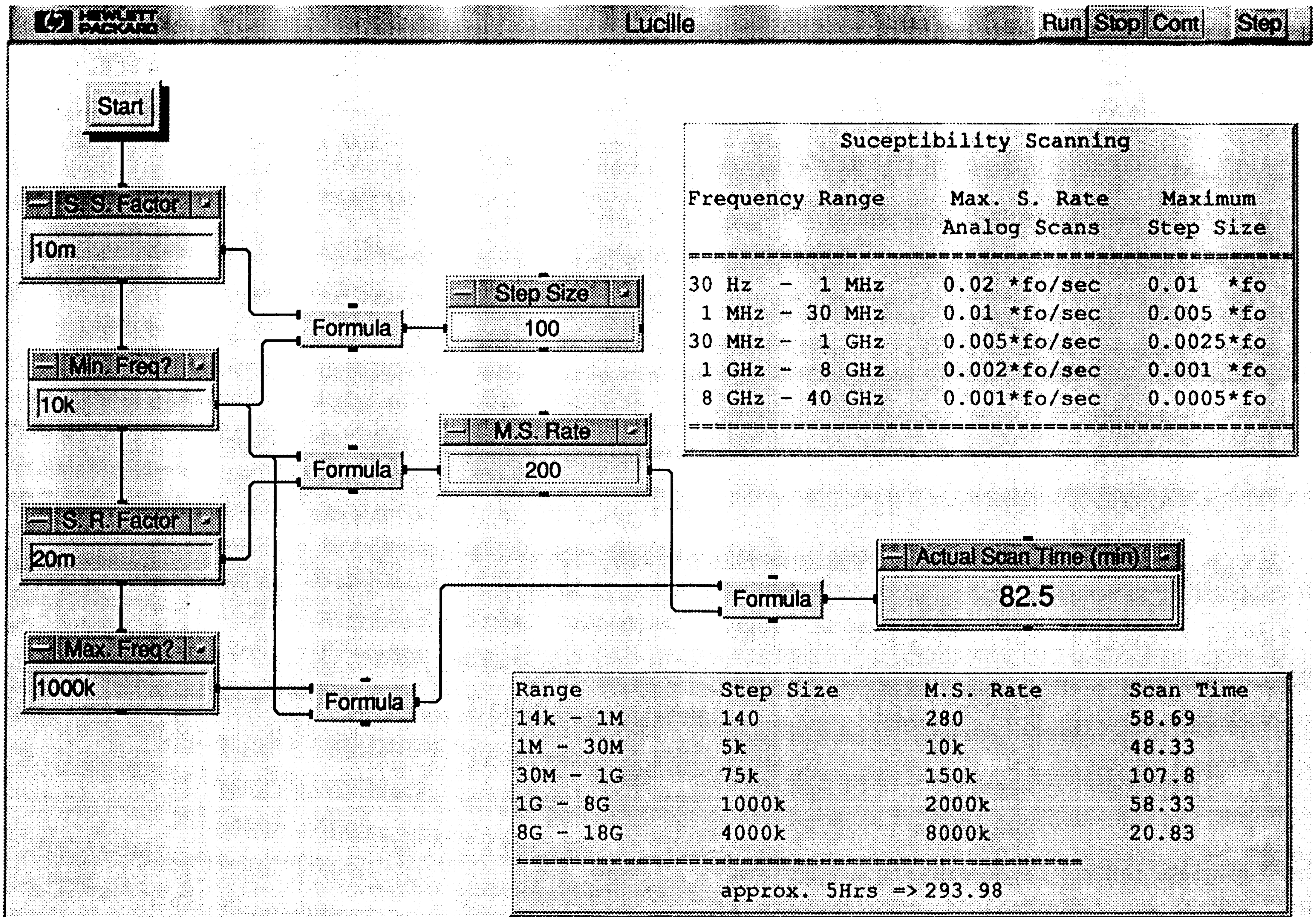


Figure 18: RS03 Test Setup (14 kHz to 18 GHz)

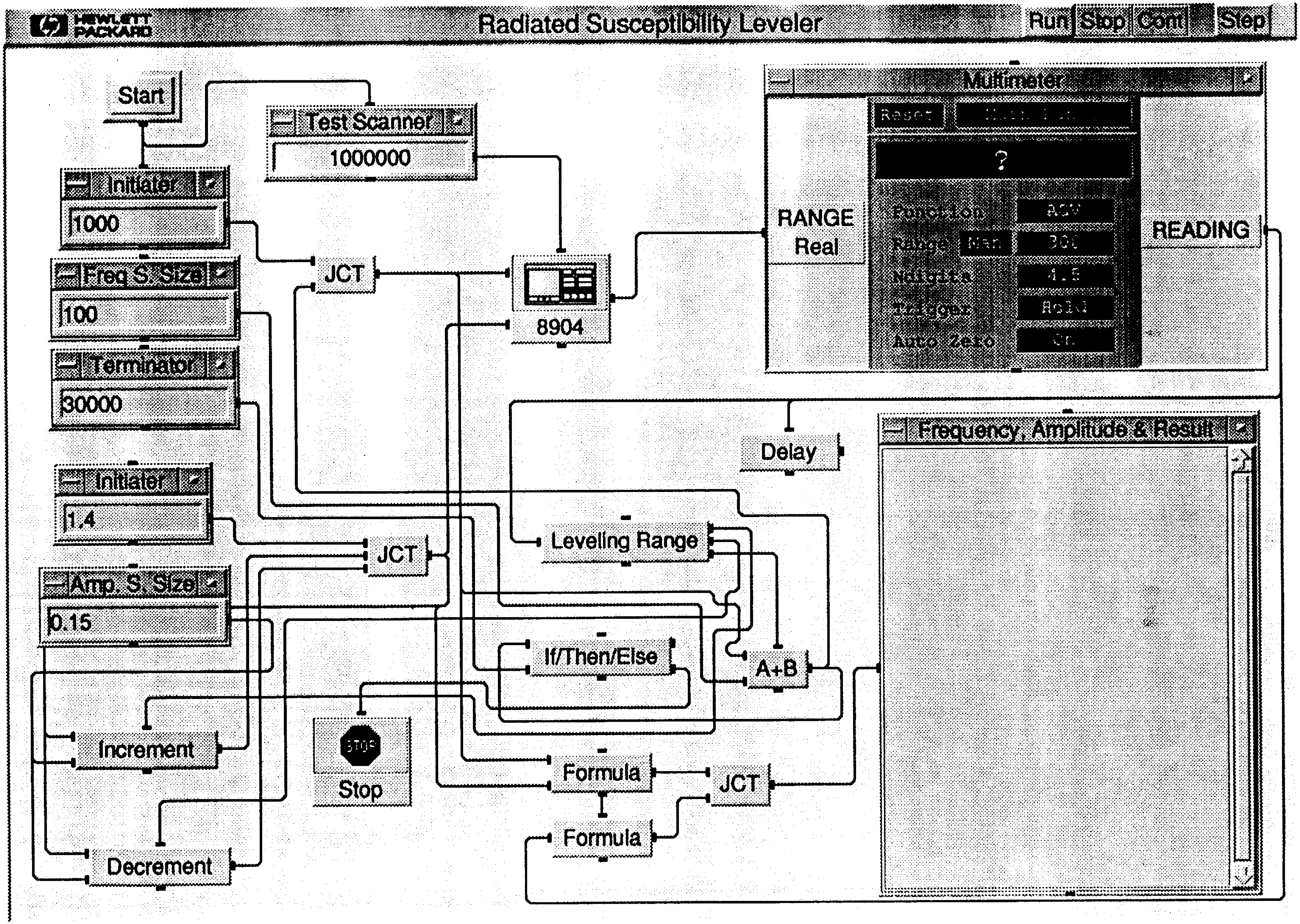
```
/* Program to find maximum element in array */

#include <math.h>
main( )
{
 double num[10], max;
 int i;

   for (i=0; i<10; i++){
    num[i]=(double) rand( )/pow(2.0,15.0);
    printf("%f\n",num[i]);
   }
   max=num[0];
   for (i=1; i<10; i++){
    if (num[i] > max) max=num[i];
   }
   printf("\nmax: %f\n",max);
}
```

Figure 1: ANSI C Program


HEWLETT PACKARD
Marcia
Run
Stop
Cont
Step
Start
Frequency
From 1
Thru 193
Step 3
Function Generator
Frequency
Real
Function Cosine
Frequency 193
Amplitude 1
DcOffset 0
Phase Deg 0
Time Span 20m
Num Points 256
Func
Waveform (Time)
2
Mag
0.5/
Volts
-2
0
20m
Auto Scale
Time
2m/
x:10.05m y:0
x:10.05m y:0
x:0 y:0



Lucille
Run
Stop
Cont
Step
Start
S. S. Factor
10m
Min. Freq?
10k
S. R. Factor
20m
Max. Freq?
1000k
Formula
Step Size
100
Formula
M.S. Rate
200
Formula
Formula
Actual Scan Time (min)
82.5
Suceptibility Scanning
Frequency Range | Max. S. Rate Analog Scans | Maximum Step Size
30 Hz - 1 MHz | 0.02 *fo/sec | 0.01 *fo
1 MHz - 30 MHz | 0.01 *fo/sec | 0.005 *fo
30 MHz - 1 GHz | 0.005*fo/sec | 0.0025*fo
1 GHz - 8 GHz | 0.002*fo/sec | 0.001 *fo
8 GHz - 40 GHz | 0.001*fo/sec | 0.0005*fo
Range | Step Size | M.S. Rate | Scan Time
14k - 1M | 140 | 280 | 58.69
1M - 30M | 5k | 10k | 48.33
30M - 1G | 75k | 150k | 107.8
1G - 8G | 1000k | 2000k | 58.33
8G - 18G | 4000k | 8000k | 20.83
approx. 5Hrs => 293.98


HEWLETT PACKARD
Radiated Susceptibility Leveler
Run
Stop
Cont
Step
Start
Test Scanner
1000000
Initiater
1000
Freq S. Size
100
Terminator
30000
Initiater
1.4
Amp. S. Size
0.15
JCT
8904
Multimeter
RANGE
Real
READING
Function
ACV
Range
Man
Ndigits
4.5
Trigger
Hold
Auto Zero
On
Delay
Frequency, Amplitude & Result
Leveling Range
If/Then/Else
A+B
Increment
Decrement
STOP
Stop
Formula
Formula
JCT

Conclusion

Electromagnetic interference will continue to pose a possible threat to the operation of electrical systems. It is the goal of EMI test engineers to conduct tests on electrical devices to ensure their compatibility within a viable environment. Essentially, any equipment or system should not adversely affect the operation of any other equipment or system as a result of radiated or conducted EMI, and, in turn, it should not be affected by the same.

Automation of EMI tests, will facilitate the manual, laborious procedures of the test engineers. Incorporation of software programs, such as HP VEE, will provide better accuracy of the test run and allow control of the rate of the test. The objective for implementing HP VEE in the testing environment is to provide the capabilities of easy collection, analysis, and presentation of data in other in other programs for further processing.

References

1. Helsel, Robert *Cutting Your Test Development Time With HP VEE.* Englewood Cliffs, New Jersey: P T R Prentice Hall, 1994.

2. Violette, J.L. Norman, and White, Donald, and Violette, Michael *Electromagnetic Compatibility Handbook.* New York: Van Nostrand Reinhold Company, 1987.

3. Violette, J.L. Norman *A Workshop: Electrical Noise Control in High Performance Electronic Circuits and Systems*, 1993

ANALYSIS OF GSPAR COMPLIANCE WITH ISO 9000

- PRESENTED BY:
  MICHELLE JONES
- CODE 300, OFA
- MENTOR:
  BOYD PEARSON
- ASSISTED BY:
  T. HAMMER/
  G. COGER

Table of Contents

MY SUMMER INTERNSHIP EXPERIENCE

Coming into this internship, I did not know what to expect. After all, this is Goddard Space Flight Center, one of the top areas for engineering and science. However, my worries were soon stopped when I saw the laid back atmosphere in which I would be working. The first day I met my mentor Boyd Pearson in Building 6, the Office of Flight Assurance. Roger Evans, a contractor from Unisys was also there. The two gave me a brief overview of my project for the summer. They told me that I would be "comparing the GSPAR and CODE 500 QA Plan with the ISO 9000 family to see what changes should be made so that the two will fit ISO standards." After introducing me to several other employees, they showed me to my office and I began reading up on the GSPAR, CODE 500 QA Plan, and the ISO 9001.

Boyd suggested that I make a comparison matrixes between the ISO 9001 & GSPAR and the ISO 9001 & CODE 500 QA Plan. At first glance, all of the terminology seemed so overwhelming but as time went on, I began to understand the GSPAR more so than the CODE 500 QA Plan, probably because the layout of the GSPAR is similar to that of the ISO 9001. I completed all that I could and took my findings to Boyd. We set appointments with Ted Hammer and Ralph Viehman, both systems assurance managers, to discuss any input they may have for the matrixes. Ted's focus was the ISO 9001 & the GSPAR while Ralph gave his observations on the ISO 9001 & the CODE 500 QA Plan. Ted suggested that I form a make shift trace matrix that will show proposed sections of the GSPAR that may be correlated to the ISO 9001 and also try to list all of the paragraphs in the GSPAR that are over and above those in the ISO 9001. Ralph told me it would take a couple of days before he could give me any type of feedback.

Because of time constraints, Boyd and I decided to concentrate on the ISO 9001 & GSPAR matrix. Ralph understood completely and said that this was probably a good thing because everything I found he totally agreed with and the entire thing was not worth considering while working on the project. Later, Ted and I met to discuss my analysis of the two tasks he assigned me to do. He made a few corrections and told me that I needed to get a copy of the ISO 9000-3 to compare it with Section 3 of the GSPAR because both pertain to software assurance. He also told me that I may want to start tying any together information that will be used to write a ten page report on my finding and suggestions. Shortly thereafter, I began using EXCEL to construct a comparison matrix between the ISO 9000-3 and the GSPAR (Section 3). After drawing all of my data together from each of my comparisons (three in all), Ted and I again met to discuss an outline of the paper I would be writing to be presented at the end of the project. I took all of the suggestions that he gave and began writing my paper. Unfortunately, the paper only covered seven pages of text.

Back to Ted's office I went where he critiqued my paper and made some wonderful suggestions to lengthen it. However, these suggestions only looked good on paper because after making my corrections, I still only had seven pages. There was nothing more I could add and we both knew it. I took my finished product to Boyd and he seemed to be very pleased with the outcome. I added my attachments (pie charts included) and was completely done!!

ANALYSIS OF GSPAR COMPLIANCE WITH ISO 9000

Michelle Antoinette Jones

Bldg 6, Code 300

SIECA Summer Program Participant

August 1, 1995

INTRODUCTION

During the 1995 Summer Institute in Engineering and Computer Applications (SIECA) Program at Goddard Space Flight Center (GSFC), I have been working with the staff in Code 300, Office of Flight Assurance. More closely I have been the shadow of my mentor, Boyd Pearson, in the Assurance Management Office. OFA defines assurance policies for GSFC projects, implements project assurance activities, conducts an assurance technology program, and manages the independent assessment activity for the Center. More importantly, this office manages all aspects of performance assurance, from negotiating resources to garnering support from other offices, while constantly assessing hardware and software quality and status and providing feedback both for the project and the office. This office also provides feedback to the Director of OFA on the effectiveness of assurance programs.

The new hot spot in technology is being ISO 9000 certified. As a result, it has become very important to GSFC to receive certification to ensure that they will remain at the top, technologically speaking. However, GSFC now uses the GSPAR to deliver desired information to contractors. At first glance, it would seem as if the ISO and the GSPAR have no correlation. However, after examining both documents, several links can be made between the two. My task is to determine what those links are and exactly how close the GSPAR fits the ISO criteria. Because this project is essentially research and data analysis, the majority of the work was done in my office with occasional trips to the library.

ABSTRACT

The International Organization for Standardization , commonly known as ISO, is a worldwide federation of national standards bodies established to provide guidance for quality management and models for quality assurance. By December 1994, eighty countries formally adopted ISO 9000 with over 70,000 registrations worldwide. ISO 9000 has become such a vital tool in quality management, especially to the customer because ISO 9000 registration means an International Standard for Quality has been defined and measured, a method that helps customers choose between competitors' service offerings has been established, and a 'Freedom from fear' attitude that the customer's service provider will be 'inconsistent' in performance has been created.

The Guidelines for Ground Systems Performance Assurance Requirements, or GSPAR, is designed to address hardware and software assurance requirements for ground system developers. This manual represents a baseline picture that can be tailored to form the requirements for an assurance program for any Goddard Space Flight Center (GSFC) ground system project.

The purpose of this project is to determine whether or not the GSPAR complies with the standards presented in the ISO 9000 manual. In examining the GSPAR with the ISO 9001 and ISO 9000-3 (for software), the focus will be to determine what requirements of the GSPAR, if any, need to be modified in order to comply with the standards of the ISO 9000 family.

ANALYSIS:

The first table, "ISO 9001 vs Guidelines for Ground Systems Performance Assurance Requirements," (See Attachment A), is a comparison matrix between the ISO 9001 and the GSPAR. The purpose of the table is to show any and all connections between the two documents and exactly how close the two are in comparison in order to complete ISO compliance. There are twenty ISO requirements that must be examined, from Management Responsibility to Statistical Techniques. The table is designed as follows:

ISO 9001 - QUALITY SYSTEM REQUIREMENTS	GUIDELINES FOR GROUND SYSTEMS PERFORMANCE ASSURANCE REQUIREMENTS	COMMENT
4.2 Quality System	**1.1 Description of Overall Requirements**	**The GSPAR provides more details on how the program should be implemented whereas the ISO 9001 states that a program should be established. No details for implementation are provided in the ISO 9001.**

The first column lists each of the twenty requirements cited in the ISO 9001 with their section number and title. The second column lists any information found in the GSPAR. The third and final column provides brief comments on comparisons and differences between the ISO 9001 and the GSPAR. There are three conditions that may occur. The first is the case where there is a match between the ISO 9001 requirement and the GSPAR section. If this condition is met, the paragraph number and title of the

GSPAR are recorded in the second column and any amplifying notes are placed in the third column. The second case is where there is concern whether or not the designated section of the GSPAR is the correct match to the ISO 9001 requirement. In order to identify this concern, the statement "This correlation needs verification." will be placed in parentheses after the section number and title in the GSPAR column because futher research is needed to comfirm the match (ex 1). The third case includes all ISO 9001 requirements that are not met by the GSPAR. If this situation occurs, the statement "This requirement is not addıessed in GSPAR" will be placed in all spaces under the GSPAR column where this case is evident (ex 2)..

EX 1:

4.1 Management Responsibility	**1.2 Organization (This correlation needs verification.)**	**The GSPAR only details who is responsible for QA GSPAR activities.**

EX 2:

4.20 Statistical Techniques	**This requirement is not addressed in the GSPAR.**	

In analysing the collected data, the comparisons between the ISO 9001 and the GSPAR would tend to suggest that the hardware section of the GSPAR is over 90% compliant with the ISO 9001. Overall , three requirement correlations need further verification and only one requirement is not addressed in the GSPAR.

Because the ISO 9001 does not include the requirements for software applications, it is necessary to review the ISO 9000-3, written for software, and its relation to Section 3 of the GSPAR. The ISO 9000-3 does two things: 1) identifies specific software requirements, and 2) identifies general ISO 9001 requirements by references. As a result the second table, "ISO 9000-3 vs GSPAR (Section 3)," (See Attachment B), has been created. The same ISO requirements for 9001 hold for 9000-3. The outline is designed as follows (example included):

ISO 9000-3 Requirement		**GSPAR (Section 3) Paragraph**
4.3 Contract Review		**1.5 Surveillance of the Contractor**

The first column contains the ISO 9000-3 requirement and the second column contains the corresponding paragraph from Section 3 of the GSPAR. One minor difference between the ISO 9001 and 9000-3 tables is that there are no comments made on the 9000-3 table. Because the ISO 9000-3 only reviews software, several requirements are not discussed as the same information that applies to hardware also applies to software. To identify these requirements the statement "*See Note*" , which states that the pargraph points back to ISO 9001 for the requirement in this section, follows the number and title in the ISO 9000-3 column while the sentence "This requirement is not addressed in the GSPAR" is placed in the GSPAR column (ex 3). While 9000-3 makes several references to 9001, it provides an expanded set of software requirements that are not addressed by Section 3 of the GSPAR. In this case, the space in the GSPAR column is noted with the latter sentence stated above (ex 4).

EX 3:

4.9 Process Control ***See Note***	**This requirement is not addressed in the GSPAR.**

EX 4:

4.18 Training	**This requirement is not addressed in the GSPAR.**

As compared to the correlation between the ISO 9001 and the GSPAR, the comparisons between the ISO 9000-3 and the GSPAR (Section 3) are not as compatible. Out of the twenty requirements, eight requirements are not addressed in the GSPAR. However, these eight requirements include both the requirements labeled with "*See Note*" and those that are simply not addressed in the GSPAR at all.

The final phase is to identify those requirements that are addressed in the GSPAR and not in the ISO 9001 and/or 9000-3 thereby establishing the third table "Comparisons Between the GSPAR and the ISO 9001/9000-3" (See Attachment C). When looking at this comparison, the listing of paragraphs over and above in the GSPAR are placed in the order in which they appear, similar to the table of contents of the GSPAR so that the sections are easier to locate.

RECOMMENDATIONS:

After reviewing each of the tables, several recommendations can be made to ensure that any necessary changes and additional analysis is taken to make the GSPAR ISO 9000 compliant. From both Table 1 and Table 2, it can be concluded that one of two recommendations can take effect. Foremost, if there is a match, the present text in the GSPAR should be deleted and references to ISO 9001 or 9000-3 be developed. However if there is no match between the ISO to the GSPAR, indicating that either the ISO 9001 or 9000-3 has more requirements than the GSPAR, ISO 9000 requirements over and above the GSPAR should be used and incorporated by reference. The final condition, pertaining to questionable matches (Table 1) and unique GSPAR requirements (Table 3), need additional review to determine if the requirement should be modified or deleted in light of ISO 9001 or 9000-3 philosophy. In order to help in the execution of the recommendations, a schedule (See Attachment E) has been developed through the consultation with a GSFC/OFA engineer to outline the time each phase of review has before being completed.

ISO 9001 vs GUIDELINES FOR GROUND SYSTEMS PERFORMANCE ASSURANCE REQUIREMENTS

ISO 9001 - QUALITY SYSTEM REQUIREMENTS	GUIDELINES FOR GROUND SYSTEMS PERFORMANCE ASSURANCE REQUIREMENTS (GSPAR)	COMMENT
4.1 Management Responsibility	1.2 Organization (This correlation needs verification.)	The GSPAR only details who is responsible for QA GSPAR activities.
4.2 Quality System	1.1 Description of Overall Requirements	The GSPAR provides more details on how the program should be implemented whereas the ISO 9001 states that a program should be established. No details for implementation are provided in the ISO 9001.
4.3 Contract Review	1.5 Surveillance of the Contractor	The GSPAR tells exactly who has the ability to conduct the reviewing process and what they are looking to find. The ISO 9001 just gives a listing containing brief descriptions of what is being examined (e.g. records, amendments.)
4.4 Design Control	2.2 Configuration Control & Verification (This correlation needs verification.)	The GSPAR states that all existing policies, plans and procedures are to be used to the maximum extent. No outlined design plan is given.
4.5 Document and Data Control	2.2 Configuration Control and Verification (Data Control is not addressed.)	The GSPAR requires that all documents and revisions be controlled in accordance with the Project Configuration Management Plan. The ISO 9001 simply states that procedures should be established and maintained

		documented procedures to ensure product requirements are reached.
4.6 Purchasing	1.8 Procurement Requirements	The GSPAR provides very limited details on the evaluation of subcontractors and no information on purchasing data or contractor/customer verification. The ISO 9001 incorporates each topic in their own section giving more information (e.g. purchasing data, customer verification.)
4.7 Control of Customer-Supplied Product	1.9 Contractor Receiving Inspection Section 4 Nonconformance Reporting	The GSPAR requires more input for overall documentation as well as nonconformance documentation. The ISO 9001 simply states that documented procedures should be established and maintained and nonconformance products should be recorded and reported to the customer.
4.8 Product Identification and Traceability	2.3 Identification and Traceability Requirements	The GSPAR specifically states that the product identification should include a unique part or type number that is consistent with the configuration management system. The ISO 9001only states that the contractor should establish documented procedures for identification.
4.9 Process Control	2.4 Control of Fabrication Activities	The GSPAR divides this requirement into sub-requirements including Fabrication and Inspection Requirements, Training and Certification for Manufacturing and Inspection Personnel, and the Evaluation and Control of Process Specifications and Procedures, with each stating specific details in establishing the requested criteria. The ISO

		9001 only states that controlled conditions should be followed.
4.10 Inspection and Testing ------------------------------------ 4.10.3 In-process Inspection and Testing	2.7 Control of Assembly and Inspection/Test Activities - 2.7.2 Inspection and In-Process Test Procedures & 2.7.3 Inspection Activity ------------------------------------ 2.7.4 Inspection and Test of Stored Stock Hardware	The GSPAR states detailed steps for the in-process inspection, the final inspection, and the end-item inspection. Although the ISO 9001 divides the topic into subsets, it briefly discusses each subset. ------------------------------------ The GSPAR specifies which standards are to be used. ISO simply requires that standards be used.
4.11 Control of Inspection, Measuring, and Test Equipment	2.8 Metrology	Although both the GSPAR and the ISO 9001 both require execution of documented procedures the ISO 9001 also requires that test software/hardware are checked for approval before being used in production, installation, or servicing and are consequently rechecked at necessary intervals.
4.12 Inspection and Test Status	2.7 Control of Assembly and Inspection/Test Activities - 2.7.5 Records of Inspections and Tests	The GSPAR states that each component, subsystem, and system be covered in the records. The ISO 9001 states that the product shall simply be identified by suitable means.
4.13 Control of Nonconforming Product	4.1 Nonconformance System Definition	The GSPAR states in more detail what criteria is to be met by the system. The ISO 9001 states that a procedure must be established and gives brief descriptions what is to be maintained .
4.14 Corrective and Preventive Action	4.2 Nonconformance and Corrective Action Reporting (Preventive Action is not addressed.)	Both the GSPAR and the ISO 9001 state complementary procedures except that the GSPAR does not address the effective

		handling of customer complaints.
4.15 Handling, Storage, Packaging, Preservation, and Delivery	1.10 Handling, Preservation, Marking, Packaging, Packing, and Transportation	The GSPAR states a detailed listing of the procedure included in expediting each process. ISO 9001 simply states that procedures should be established and maintained, but does not give detailed descriptions on how to execute them.
4.16 Control of Quality Records	2.9 Stamp Control System	The GSPAR uses a stamping system to identify the product's information but does not state that the information can be made open to the customer. The ISO 9001 does not state what particular system could be used for identification.
4.17 Internal Quality Audits	1.6 Audits and Reports	The GSPAR states what the audit schedule should be based on in order to produce a fully documented audit. The ISO 9001 simply states that the supplier shall establish and maintain documented procedures for planning and implementing internal quality audits.
4.18 Training	2.4.2 Training and Certification for Manufacturing Inspection Personnel	Training for both the GSPAR and ISO 9001 are compatible except that the GSPAR also details certification after training has been completed.
4.19 Servicing	1.1 Description of Overall Requirements (This correlation needs verification.)	
4.20 Statistical Techniques	This requirement is not addressed in the GSPAR.	

* The GSPAR uses the term contractor while the ISO 9001 uses the term supplier. However, both terms are being used in the same text. *

ISO 9000-3	GSPAR (SECTION 3)
4.1 Management Responsibility	3.1 General Requirements
4.2 Quality System	3.2 Verification and Validation
4.3 Contract Review	1.5 Surveillance of the Contractor
4.4 Design Control *See Note*	This requirement is not addressed in the GSPAR.
4.5 Document and Data Control	3.3.1 Standards/ 3.1.1 Documentation
4.6 Purchasing *See Note*	This requirement is not addressed in the GSPAR.
4.7 Control of Customer-Supplied Product	1.9 Contractor Receiving Inspection/ Section 4 Nonconformance Reporting
4.8 Product Identification and Traceability	This requirement is not addressed in the GSPAR.
4.9 Process Control *See Note*	This requirement is not addressed in the GSPAR.
4.10 Inspection and Testing	3.2.2 Software Test Procedures/ 3.2.3 Software Test Reports
4.11 Control of Inspection, Measuring, and Test Equipment	3.4 Software Configuration Management
4.12 Inspection and Test Status	3.3.2 Assurance Function
4.13 Control of Nonconforming Product	3.5 Software Nonconformance reporting and Corrective Action
4.14 Corrective and Preventive Action	3.5 Software Nonconformance Reporting and Corrective Action
4.15 Handling, Storage, Packaging, Preservation, and Delivery	1.10 Handling, Preservation, Marking, Packaging, Packing, and Transportation
4.16 Control of Quality Records	This requirement is not addressed in the GSPAR.
4.17 Internal Quality Audits	3.2.4 Walkthroughs or Inspections
4.18 Training	This requirement is not addressed in the GSPAR.
4.19 Servicing	This requirement is not addressed in the GSPAR.
4.20 Statistical Techniques	This requirement is not addressed in the GSPAR.

The paragraph points back to ISO 9001 for the requirement in this section.

COMPARISON BETWEEN THE GSPAR AND THE ISO 9001

The following paragraphs of the GSPAR are over and above those in either the ISO 9001/9003 and **are not** discussed in the ISO 9001 or 9000-3:

1.4 Assurance Status Reports
1.7 Quality Support To Design Reviews
1.8 Procurement Requirements
1.8.2 Procurement Review by the Government
1.11 Government Property Control
1.12 Contract Deliverables
1.14 Acronyms

2.1 General Requirements
2.4 Control of Fabrication Activites
    2.4.3 Evaluation and Control of Process Specifications and Procedures
2.5 Electrostatic Discharge Control
2.6 Alert Information
2.7 Control of Assembly and Inspection/Test Activities
2.10 Electromagnetic Compatibility Control

3.1.2 GFE, Existing and Purchased Software
3.2.1 Software Test Plan
3.2.5 Software Reviews

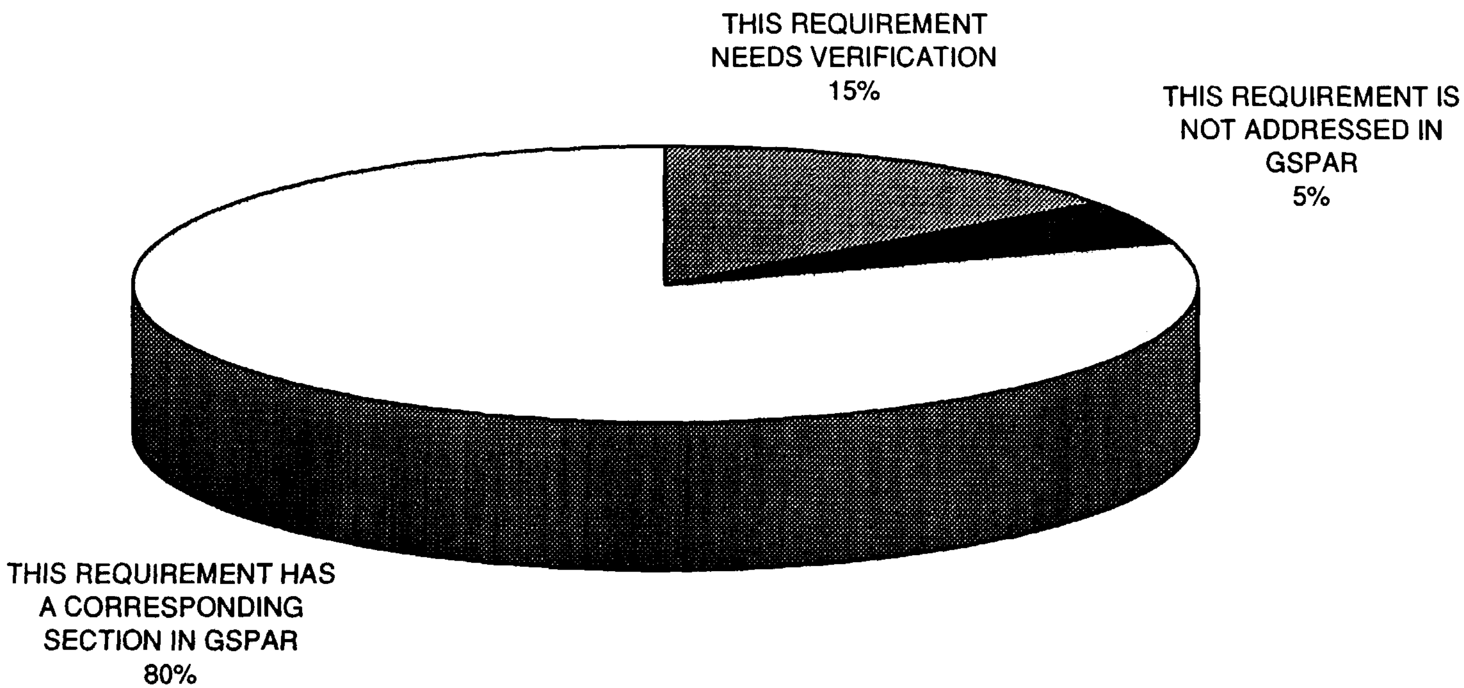
4.3 Information To Be Submitted
4.4 Initial Review Disposition
4.5 Review Boards

5.1 QA Activities During the Integration and Test Phase
5.2 Pre-Government Acceptance Activity

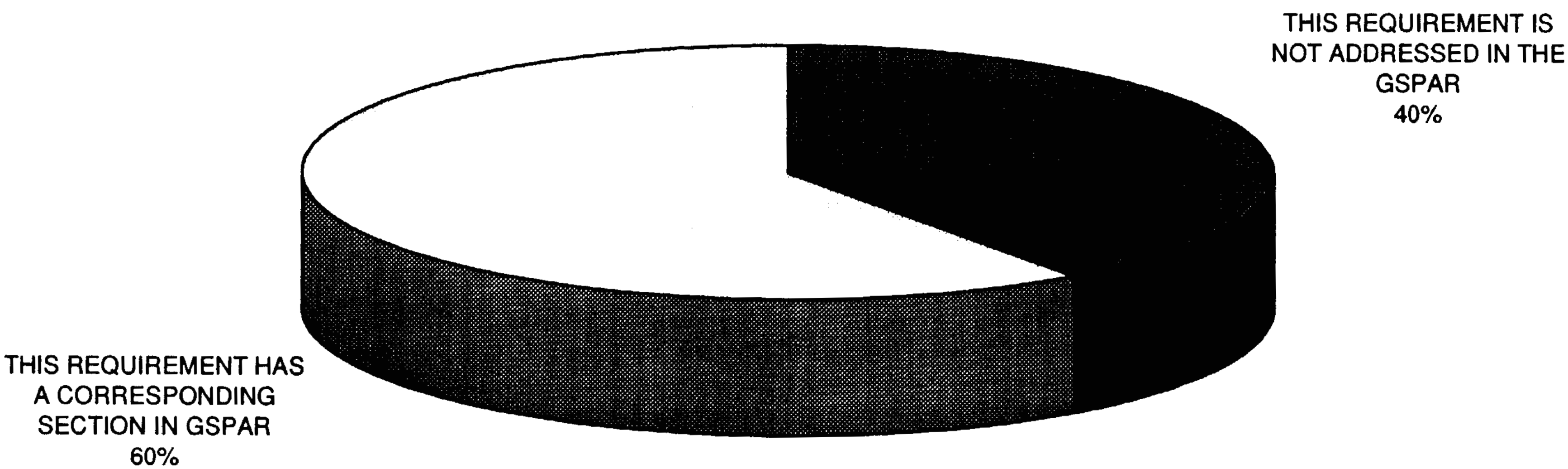
6.1 Government Acceptance
6.2 End-to-End Compatibility Test
    6.2.1 General
    6.2.2 End-to End-Compatibility Testing
    6.2.3 End-to-End Compatibility Test Plan
6.3 Test Procedures

7.1 General
7.2 Reliability Allocations
7.3 Reliability Prediction
7.4 Failure Mode Effects and Criticality Analyses (FMC)
7.5 Parts Stress Analyses

ISO 9001 VS GSPAR



ISO 9000-3 VS GSPAR (PT. 3)



SCHEDULE

1) Analysis of Requirements

2 man-month

2) Update Document

1 man-month

3) Review Cycle/Comment Resolution

1 month (2week review/2 comment resolution)

4) Publish Final

1/2 month

Schedule based on experience with previous document development/update efforts.

Biospheric Processes in the General Circulation Model

Dana Murph
Senior, Water Resources Management
Central State University
Wilberforce, Ohio

(SIECA) Program
NASA-Goddard Space Flight Center
Greenbelt, Maryland
August, 1995

Mentor: Dr. Forrest G. Hall
Biospheric Sciences Division
NASA-Goddard Space Flight Center
Greenbelt, Maryland

Advisor: Dr. Subramania I. Sritharan
International Center for Water Resources Management
Central State University
Wilberforce, Ohio

INTRODUCTION

Technology in the study of the Earth's environment has taken steps forward in remote sensing methods, and climate modeling. Remote sensing information is gathered by NASA satellites and aircraft. Second generation sensors provide crisp images with very high accuracy. This technology advances the investigation of land and vegetation properties. General Circulation Models use this advanced data to estimate the amount of moisture in the atmosphere at three different phases liquid, solid, and gas. (GCM's) have reached a level of sophistication where it has become very important to treat all variables and land surface properties in a very realistic manner. The important areas of study are land cover, soil type, solar radiation, latent heat, and relative humidity. All factors combine to create the atmosphere.

GENERAL CIRCULATION MODELS (GCM's)

The Earth and it's atmosphere are constantly in motion so all climatic parameters change values frequently. Constantly changing variables make the global climatic models very complex. Years of analysis have gone into each component of the (GCM). The Biospheric component of the model focuses on evaporation, transpiration, and carbon dioxide gas exchange between the earth and its atmosphere. Vegetation has a large effect on the atmosphere due to photosynthesis and transpiration. Photosynthesis occurs in the leaves of green plants. This process is under physiological control by stomatal pores which regulate the influx of carbon dioxide and the release of water in the photosynthesis process. Plants absorb and scatter solar radiation, alter wind movement, and release oxygen to the atmosphere. Solar radiation is the most important part of the biosphereic processes. The amount of water transpired by the plants will depend on the energy from the solar radiation. The Radiation budget is used for accounting net shortwave radiation, and net long-wave radiation. The budget takes factors into account such as albedo, and surface roughness. Albedo is the net amount of reflected radiation divided by the incoming radiation. Albedo is measured by sensing instruments called radiometers. Radiation is measured above and below plant canopies and then the information is averaged appropriately. Several parameters affect the radiation budget such as cloudiness, temperature, time of the year, latitude of the area and albedo.

PLANT PARAMETERS IN RADIATION BUDGET

Plant foliage receives amounts of radiation depending also on leaf angle. Some radiation will be absorbed by the foliage while some is reflected and scattered. Plant canopies increase their trapping of light by arranging there leaves randomly, and by inclining them some what vertically. Surface roughness effects scattering of solar radiation due to vegetation density. The more dense the vegetation the more scattering will occur. The roughness of the terrain also effects the wind velocity. Roughness length is the velocity of the wind related to the height above the surface. In the plant kingdom there are several types of leaf orientations that very from species. Leaf orientations are plants means of adjusting to the environment to receive optimal photosynthetic activity. Wind activity causes the leaves, stems, and flowers to flutter around and this changes the angle of inclination. Leaf orientation affects the net radiation received by the plants. Leaf orientation can be described by leaf angle distribution functions which can be done in different ways. A fundamental way to describe this function is to ascribe a probability distribution function for leaf angle variation considering the physiology of the plant stands. A distribution function that can encapture well the different possibilities of leaf angle distributions is the beta distribution function.

SIMPLE BIOSPHERIC MODEL (SiB)

The Simple biosphieric model is effective at estimating evapotranspiration, carbon dioxide consumption, root zone moisture, and sensible heat flux. Evapotranspiration is the combination of evaporation and transpiration moisture released to the atmosphere. Carbon dioxide consumption is an estimate of the amount of carbon dioxide used by plant canopy foliage in the photosynthesis process. The root zone is the area of soil where the root systems are located. The moisture in this soil can be estimated by the (SiB) model. Sensible heat flux is the heat released by surface soil and gained by the air or vise versa. The Simple biospheric model inputs must take into consideration solar radiation, relative humidity, wind velocity and temperature.

IMPROVEMENTS IN EVAPOTRANSPIRATION ESTIMATION

In the traditional method of estimating evapotranspiration, we do not describe these details. We will assume a lumped value for the albedo of the surface and obtain the net radiation received by the plants. Comparing the traditional method of estimating evapo-transpiration by hydrologists to the (SiB) model we suspect a difference in evaporation amounts. The traditional method is thought to over-estimate the amount of moisture released. The (SiB) model will be modified with an application of the Beta Distribution. Beta distribution provides density for plant foliage at an interval of finite length. This information was computed with a FORTRAN 77 program. Values obtained were then graphed with IDL Basics, Version 3.6.

An important plant parameter in the (SiB) model is the ZMEW factor which depends on the leaf angle distribution. This factor is called in this paper as Mu bar. This function directly involves leaf angles and incoming radiation angles. A FORTRAN 77 program was also written to compute these values for more general beta distributed leaf angles. Using the MU bar values for beta distributed leaf angles, we will see if the simple biospheric model is applicable to agricultural lands.

A method of comparing our data to see if the modifications to the (SiB) will be effective is the Goudriaan's Function for (SiB). The Goudriaan's function was used to specify the values of leaf angle distribution. This function is thought to validate the transmission of radiation through a vegetation canopy. We will compare the MU bar values obtained by using Goudriaan's functions and those obtained using beta distribution.

CONCLUSIONS

In conclusion, the beta distribution function is more direct in the leaf angle distribution properties than the Goudriaan's function in accounting the bulk amount of solar radiation that strikes vegetation canopies. Further research and study will be needed to apply this information to the NASA Simple biospheric model. Once added, the model should be more accurate in estimating moisture and carbon dioxide exchange between vegetation and the atmosphere. The modifications to the (SiB) should make this climate model appropriate for use over agricultural lands.

Traditional Method of estimating evapotransporation:

Energy Budget Method:

$$E_{(cm/day)} = \frac{Q_n + Q_v - Q_\theta}{\rho L_e (1+R)}$$

$\rho = 1 \frac{gm}{cm^3}$ Density of Water

Q_n = Net Radiation absorbed

Q_v = advected energy of Inflow and outflow

Q_θ = increase in energy stored

L_e = latent heat of vaporatsation (cal/g)

R = (Bowen Ratio) used to Measure sensible Heat transfer

$$* \Rightarrow R = .66\left(\frac{T_s - T_a}{e_s - e_a}\right)\left(\frac{P}{1000}\right) = \gamma \frac{T_s - T_a}{e_s - e_a}$$

P = atmospheric pressure (mb)

T_a = Air temp (°C)

T_s = Water surface Temp

e_a = Vapor pressure of air (mb).

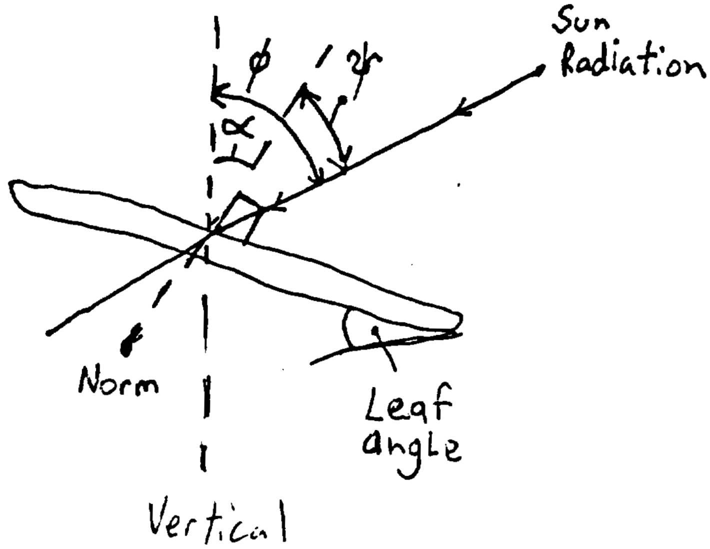
e_s = Saturation vapor pressure at surface water temp. (mb)

γ = The psychometric constant $0.66\ P/1000$

[redacted] Evapotranspiration:

$$ET = \underbrace{K_c}_{\text{(Crop Coefficient)}} \underbrace{K_s}_{\text{(Soil Moisture Factor)}} E$$ — evaporation (Energy Budget)

Calculation of Mubar



$$\text{Mu bar } \bar{\mu} = \int_0^1 \frac{\mu' d\mu'}{G(\mu')}$$

$$\bar{\mu} = \cos\phi$$

$$G(\mu) = \cos\psi$$

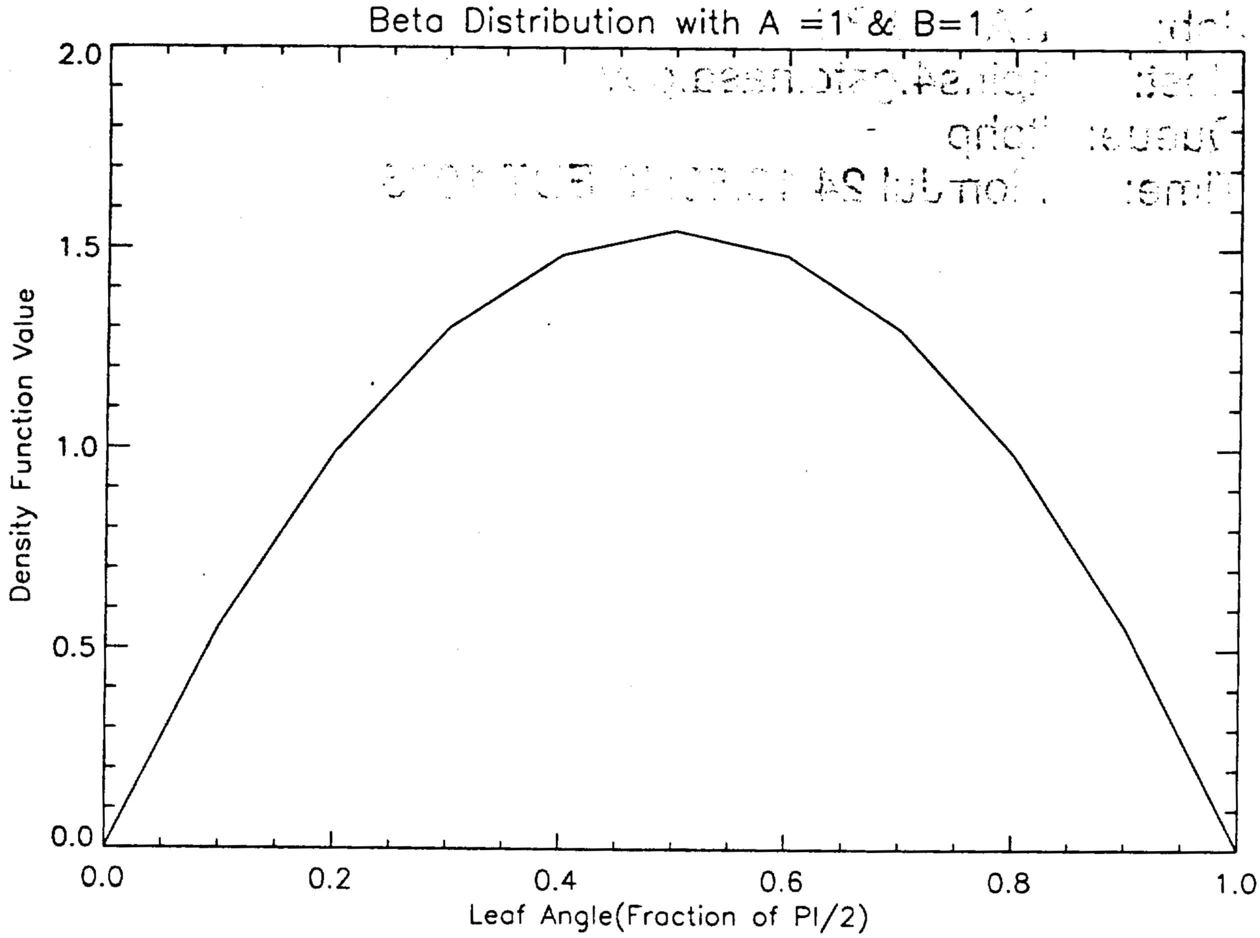
$$= \cos(\phi - \alpha)$$

Thus:

$$\bar{\mu} = \int_0^1 \frac{\mu' d\mu'}{(\cos\phi \cos\alpha + \sqrt{1-\mu'^2}\sin\alpha)}$$

$$\bar{\mu} = \int_0^1 \frac{\mu' d\mu'}{\mu'\cos\alpha + \sqrt{1-\mu'^2}\sin\alpha}$$

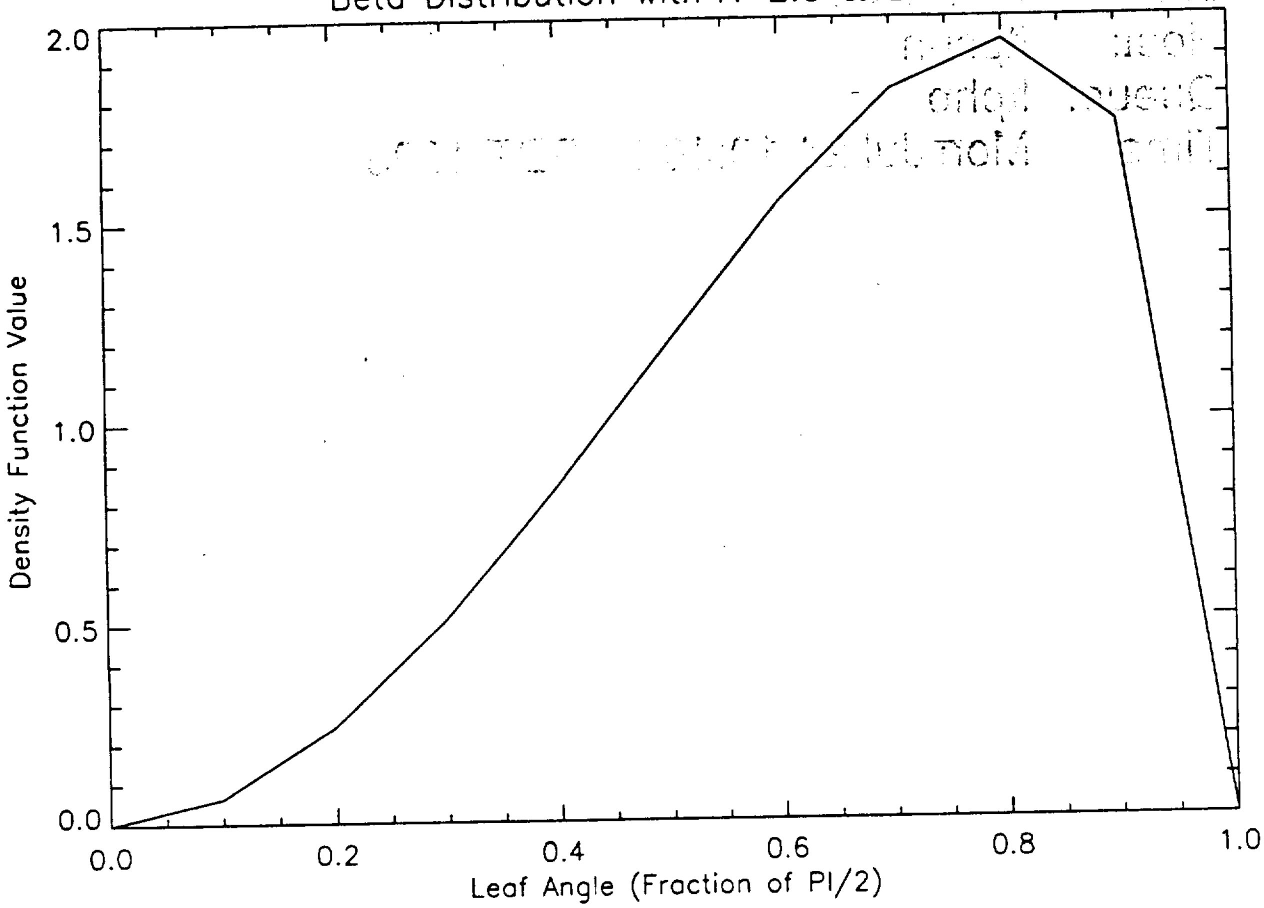
Graphs and Results of Beta Distribution



$\bar{\bar{\mu}} = .4744$

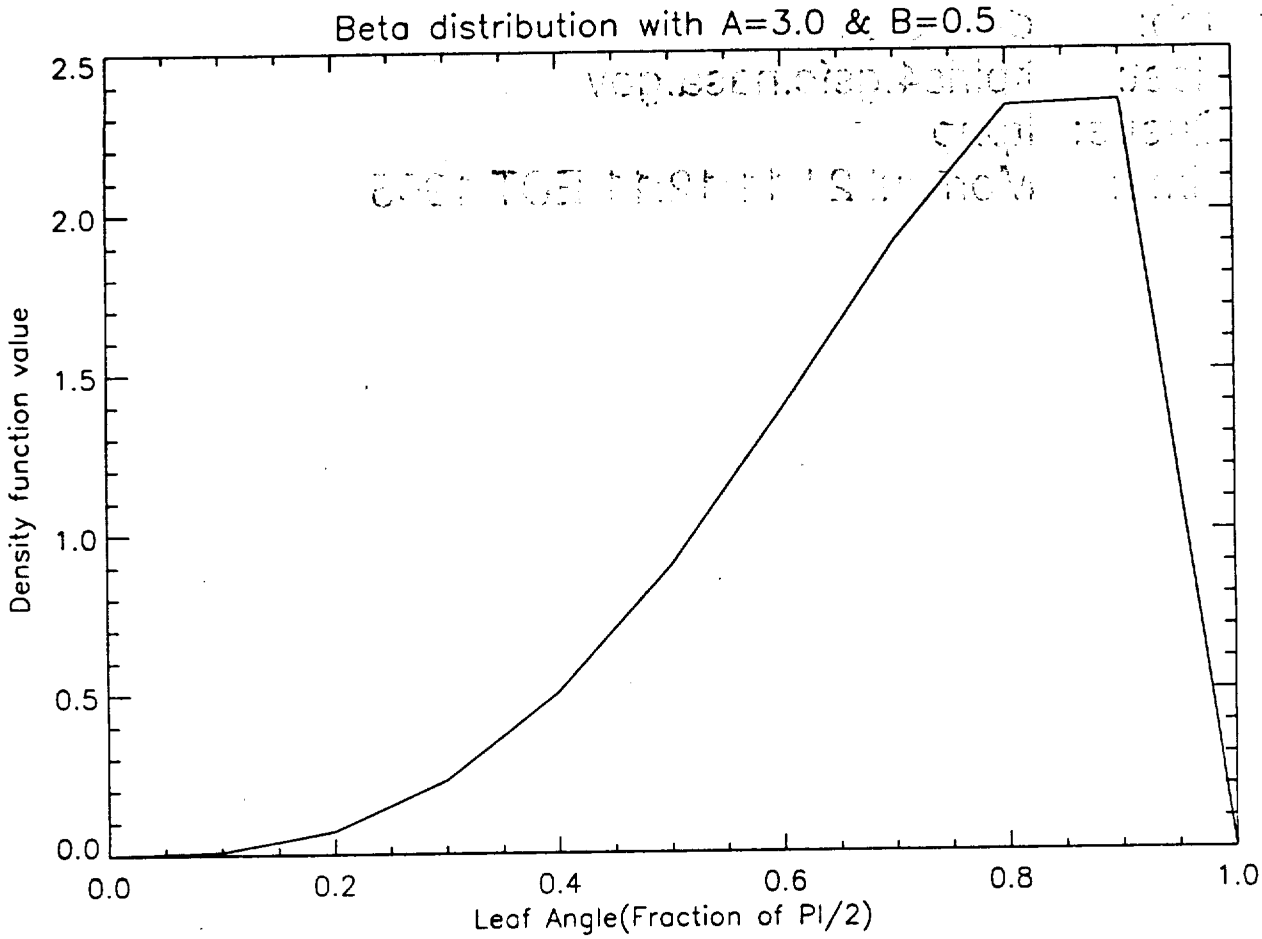
6#

Beta Distribution with A=2.0 & B=0.5



$\bar{\mu} = .6805$

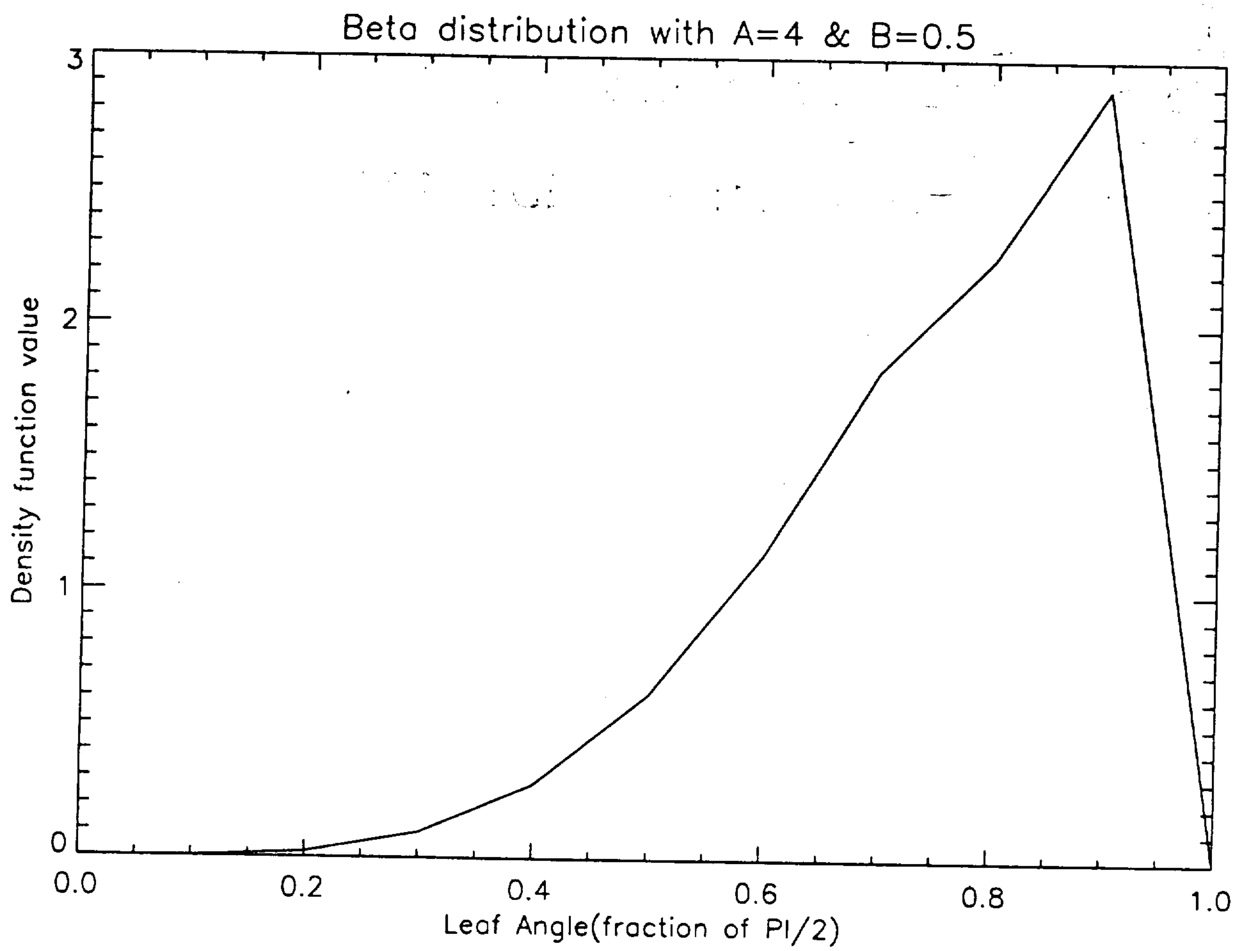
2#



Beta distribution with A=3.0 & B=0.5
Density function value
Leaf Angle(Fraction of PI/2)
0.0
0.5
1.0
1.5
2.0
2.5
0.0
0.2
0.4
0.6
0.8
1.0

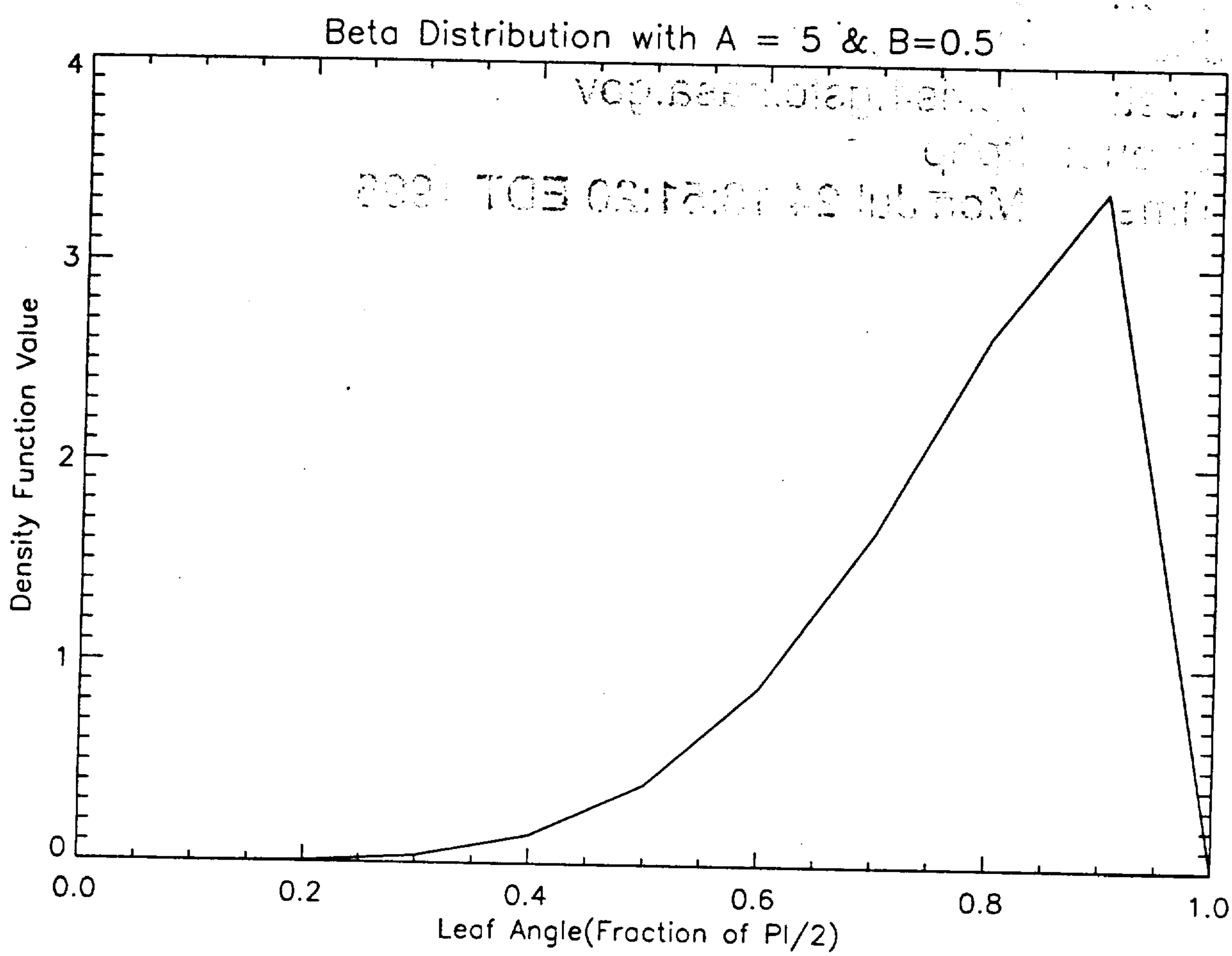
$\bar{\mu} = .7628$

3#



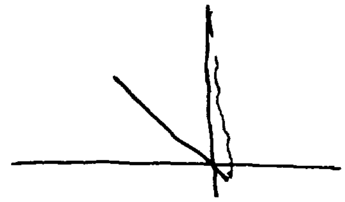
$\bar{\mu} = .8278$

4/II



$\bar{\bar{\mu}} = .8810$

5#



RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 1.000000

VALUE OF B = 0.5000000

VALUE OF C0 = 3.898459

x	y
0.0000000000000000E+00	0.0000000000000000E+00
0.100000000000000	0.3698402901278955
0.200000000000000	0.6973775389478267
0.300000000000000	0.9785054324903880
0.400000000000000	1.207893329514980
0.500000000000000	1.378313382573698
0.600000000000000	1.479361160511582
0.700000000000000	1.494691737437568
0.7999999999999999	1.394755077895654
0.8999999999999999	1.109520870383687
0.9999999999999999	4.1076939778736159E-08
~~1.100000000000000~~	~~nan~~

d

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 2.000000

VALUE OF B = 0.5000000

VALUE OF C0 = 6.780156

.0643

	0.0000000000000000E+00	0.0000000000000000E+00
	0.100000000000000	6.4322204316121706E-02
	0.200000000000000	0.2425742232149135
	0.300000000000000	0.5105416691998304
	0.400000000000000	0.8403017584295679
	0.500000000000000	1.198573510945654
	0.600000000000000	1.543732903586921
	0.700000000000000	1.819685656862365
	0.7999999999999999	1.940593785719307
.9	0.8999999999999999	1.736699516535286
1.0	0.9999999999999999	7.1440548357108454E-08
	~~1.100000000000000~~	~~nan~~

0

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 3.000000

VALUE OF B = 0.5000000

VALUE OF C0 = 10.15035

0.0000000000000000E+00	0.0000000000000000E+00
0.100000000000000	9.6294716746999805E-03
0.200000000000000	7.2630023684524120E-02
0.300000000000000	0.2292946857737714
0.400000000000000	0.5031955647061068
0.500000000000000	0.8971730521682455
0.600000000000000	1.386643921156797
0.700000000000000	1.906935275794874
0.7999999999999999	2.324160757904771
0.8999999999999999	2.339961616952094
0.9999999999999999	1.0695136215308251E-07
~~1.100000000000000~~	nan

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 4.000000

VALUE OF B = 0.5000000

VALUE OF C0 = 13.94487

0.000000000000000E+00	0.000000000000000E+00
0.1000000000000000	1.3229261638913845E-03
0.2000000000000000	1.9956267978596350E-02
0.3000000000000000	9.4503608079046605E-02
0.4000000000000000	0.2765221605471099
0.5000000000000000	0.6162818399321173
0.6000000000000000	1.143008205602156
0.7000000000000000	1.833864056194496
0.7999999999999999	2.554402301260332
0.8999999999999999	2.893239520430457
0.9999999999999999	1.4693304060271193E-07
~~1.100000000000000~~	~~nan~~

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 5.000000

VALUE OF B = 0.5000000

VALUE OF C0 = 18.12027

0.000000000000000E+00	0.000000000000000E+00
0.1000000000000000	1.7190399101552417E-04
0.2000000000000000	5.1863243844312949E-03
0.3000000000000000	3.6840031978131266E-02
0.4000000000000000	0.1437276374138942
0.5000000000000000	0.4004052182439845
0.6000000000000000	0.8911502894244768
0.7000000000000000	1.668074841899821
0.7999999999999999	2.655398084828822
0.8999999999999999	3.383586255158561
0.9999999999999999	1.9092808639716350E-07
1.100000000000000	~~nan~~ 0

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 6.000000

VALUE OF B = 0.5000000

VALUE OF C0 = 22.64442

0.000000000000000E+00	0.000000000000000E+00
0.1000000000000000	2.1482380020870439E-05
0.2000000000000000	1.2962420555762257E-03
0.3000000000000000	1.3811399530516402E-02
0.4000000000000000	7.1845027173297382E-02
0.5000000000000000	0.2501878231518210
0.6000000000000000	0.6681879481691537
0.7000000000000000	1.459183827651592
0.7999999999999999	2.654703729820109
0.8999999999999999	3.805539173557134
0.9999999999999999	2.3859770121748058E-07
~~1.100000000000000~~	~~nan~~

```
RESULTS OF BETA DISTRIBUTION
*****************************

VALUE OF A =   7.000000

VALUE OF B =  0.5000000

VALUE OF C0 =   27.49220

  0.0000000000000000E+00  0.0000000000000000E+00
  0.1000000000000000      2.6081393583881447E-06
  0.2000000000000000      3.1474910320568122E-04
  0.3000000000000000      5.0304558445061626E-03
  0.4000000000000000      3.4890332056894728E-02
  0.5000000000000000      0.1518743984413692
  0.6000000000000000      0.4867413996198289
  0.7000000000000000       1.240099252423132
  0.7999999999999999       2.578424653460939
  0.8999999999999999       4.158216566283460
  0.9999999999999999      2.8967742622636618E-07
   1.100000000000000    nan

RESULTS OF BETA DISTRIBUTION
*****************************

VALUE OF A =   8.000000

VALUE OF B =  0.5000000

VALUE OF C0 =   32.64333

  0.0000000000000000E+00  0.0000000000000000E+00
  0.1000000000000000      3.0968185222534009E-07
  0.2000000000000000      7.4744537674734354E-05
  0.3000000000000000      1.7918991312188353E-03
  0.4000000000000000      1.6571051115633409E-02
  0.5000000000000000      9.0165322001810502E-02
  0.6000000000000000      0.3467643959382110
  0.7000000000000000       1.030717022612224
  0.7999999999999999       2.449229010525693
  0.8999999999999999       4.443596097169144
  0.9999999999999999      3.4395340729450391E-07
   1.100000000000000    nan

RESULTS OF BETA DISTRIBUTION
*****************************

VALUE OF A =   9.000000

VALUE OF B =  0.5000000

VALUE OF C0 =   38.08085

  0.0000000000000000E+00  0.0000000000000000E+00
  0.1000000000000000      3.6126668750381594E-08
  0.2000000000000000      1.7439001569331172E-05
  0.3000000000000000      6.2711469092352316E-04
  0.4000000000000000      7.7325406083468742E-03
  0.5000000000000000      5.2592244222962631E-02
  0.6000000000000000      0.2427157234393636
  0.7000000000000000      0.8416851206682087
  0.7999999999999999       2.285764813695373
  0.8999999999999999       4.665403891071281
  0.9999999999999999      4.0124698046083508E-07
   1.100000000000000    nan
```

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A =   10.00000

VALUE OF B = 0.5000000

VALUE OF C0 =   43.79040

0.000000000000000E+00	0.000000000000000E+00
0.100000000000000	4.154321885989415 3E-09
0.200000000000000	4.010733809411144 4E-06
0.300000000000000	2.163418086120848 4E-04
0.400000000000000	3.556758903569598 6E-03
0.500000000000000	3.023875695796699 3E-02
0.600000000000000	0.1674639722125918
0.700000000000000	0.6775165624720527
0.7999999999999999	2.102779607468548
0.8999999999999999	4.828408249600259
0.9999999999999999	4.614068139338219 9E-07
1.100000000000000	nan

==

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A =   1.000000

VALUE OF B =   1.000000

VALUE OF C0 =   6.184451

0.000000000000000E+00	0.000000000000000E+00
0.100000000000000	0.5566005563735962
0.200000000000000	0.9895121002197267
0.300000000000000	1.298734631538391
0.400000000000000	1.484268150329590
0.500000000000000	1.546112656593323
0.600000000000000	1.484268150329590
0.700000000000000	1.298734631538391
0.7999999999999999	0.9895121002197268
0.8999999999999999	0.5566005563735966
0.9999999999999999	6.866119480057098 1E-16
~~1.100000000000000~~	-0.~~6802895689010~~611

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A =   2.000000

VALUE OF B =   1.000000

VALUE OF C0 =   12.28518

0.000000000000000E+00	0.000000000000000E+00
0.100000000000000	0.1105666122436524
0.200000000000000	0.3931257324218751
0.300000000000000	0.7739662857055667
0.400000000000000	1.179377197265625
0.500000000000000	1.535647392272949
0.600000000000000	1.769065795898437
0.700000000000000	1.805921333312988
0.7999999999999999	1.572502929687500
0.8999999999999999	0.9950995101928719

0.9999999999999999	1.3639288740856059E-15
1.100000000000000	-1.486506675720213

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 3.000000

VALUE OF B = 1.000000

VALUE OF C0 = 20.42979

0.0000000000000000E+00	0.0000000000000000E+00
0.1000000000000000	1.8386808013916019E-02
0.2000000000000000	0.1307506347656250
0.3000000000000000	0.3861229682922365
0.4000000000000000	0.7845038085937501
0.5000000000000000	1.276861667633057
0.6000000000000000	1.765133569335937
0.7000000000000000	2.102225049591064
0.7999999999999999	2.092010156250001
0.8999999999999999	1.489331449127198
0.9999999999999999	2.2681619562679888E-15
1.100000000000000	-2.719204607391353

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 4.000000

VALUE OF B = 1.000000

VALUE OF C0 = 30.61424

0.0000000000000000E+00	0.0000000000000000E+00
0.1000000000000000	2.7552818298339850E-03
0.2000000000000000	3.9186230468750007E-02
0.3000000000000000	0.1735827552795411
0.4000000000000000	0.4702347656250001
0.5000000000000000	0.9566950798034668
0.6000000000000000	1.587042333984375
0.7000000000000000	2.205143891143799
0.7999999999999999	2.507918750000000
0.8999999999999999	2.008600453948976
0.9999999999999999	3.3988636964589118E-15
1.100000000000000	-4.482231252288810

RESULTS OF BETA DISTRIBUTION
****

VALUE OF A = 5.000000

VALUE OF B = 1.000000

VALUE OF C0 = 42.83726

0.0000000000000000E+00	0.0000000000000000E+00
0.1000000000000000	3.8553535079956068E-04
0.2000000000000000	1.0966338867187503E-02
0.3000000000000000	7.2866181301116995E-02
0.4000000000000000	0.2631921328125000
0.5000000000000000	0.6693322062492371
0.6000000000000000	1.332410172363281
0.7000000000000000	2.159897546962738
0.7999999999999999	2.807382750000000
0.8999999999999999	2.529497436595918

```
  0.9999999999999999       4.7558913696067629E-15
   1.100000000000000       -6.898983753513322

 RESULTS OF BETA DISTRIBUTION
 *****************************

 VALUE OF A =    6.000000

 VALUE OF B =    1.000000

 VALUE OF C0 =    57.09830

  0.0000000000000000E+00  0.0000000000000000E+00
  0.1000000000000000       5.1388474273681665E-05
  0.2000000000000000       2.9234332031250012E-03
  0.3000000000000000       2.9137264913177516E-02
  0.4000000000000000       0.1403247937500000
  0.5000000000000000       0.4460805058479309
  0.6000000000000000        1.065591402539062
  0.7000000000000000        2.015267536608123
  0.7999999999999999        2.993595599999999
  0.8999999999999999        3.034438017386628
  0.9999999999999999       6.3391852598887600E-15
   1.100000000000000       -10.11531298586195

 RESULTS OF BETA DISTRIBUTION
 *****************************

 VALUE OF A =    7.000000

 VALUE OF B =    1.000000

 VALUE OF C0 =    73.39715

  0.0000000000000000E+00  0.0000000000000000E+00
  0.1000000000000000       6.6057433319091822E-06
  0.2000000000000000       7.5158679687500028E-04
  0.3000000000000000       1.1236369407577525E-02
  0.4000000000000000       7.2152332500000027E-02
  0.5000000000000000       0.2867076098918915
  0.6000000000000000       0.8218601623828123
  0.7000000000000000        1.813371226930160
  0.7999999999999999        3.078499519999999
  0.8999999999999999        3.510562842053147
  0.9999999999999999       8.1487203798329590E-15
   1.100000000000000       -14.30302776566929

 RESULTS OF BETA DISTRIBUTION
 *****************************

 VALUE OF A =    8.000000

 VALUE OF B =    1.000000

 VALUE OF C0 =    91.73372

  0.0000000000000000E+00  0.0000000000000000E+00
  0.1000000000000000       8.2560346984863328E-07
  0.2000000000000000       1.8787065625000012E-04
  0.3000000000000000       4.2130545066375773E-03
  0.4000000000000000       3.6071166000000016E-02
  0.5000000000000000       0.1791674196720123
  0.6000000000000000       0.6163096878281249
  0.7000000000000000        1.586479902862289
  0.7999999999999999        3.078072831999999
  0.8999999999999999        3.948835802578446
```

```
   0.9999999999999999      1.0184488682626361E-14
    1.100000000000000      -19.66393732738553

 RESULTS OF BETA DISTRIBUTION
 *****************************

 VALUE OF A =   9.000000

 VALUE OF B =   1.000000

 VALUE OF C0 =   112.1079

   0.0000000000000000E+00  0.0000000000000000E+00
   0.1000000000000000      1.0089707107543951E-07
   0.2000000000000000      4.5919378125000026E-05
   0.3000000000000000      1.5446332610939044E-03
   0.4000000000000000      1.7633041200000008E-02
   0.5000000000000000      0.1094803288578987
   0.6000000000000000      0.4519155598171873
   0.7000000000000000       1.357186917876450
   0.7999999999999999       3.009372364799999
   0.8999999999999999       4.343288068301612
   0.9999999999999999      1.2446472380577075E-14
    1.100000000000000      -26.43444619677719

 RESULTS OF BETA DISTRIBUTION
 *****************************

 VALUE OF A =   10.00000

 VALUE OF B =   1.000000

 VALUE OF C0 =   134.5195

   0.0000000000000000E+00  0.0000000000000000E+00
   0.1000000000000000      1.2106755065917977E-08
   0.2000000000000000      1.1019837500000007E-05
   0.3000000000000000      5.5602693991241531E-04
   0.4000000000000000      8.4632352000000039E-03
   0.5000000000000000      6.5683349967002869E-02
   0.6000000000000000      0.3253551922687499
   0.7000000000000000       1.139952883942396
   0.7999999999999999       2.888784281599998
   0.8999999999999999       4.690404967841165
   0.9999999999999999      1.4934664697421522E-14
    1.100000000000000      -34.89089407611348
 ============================================
```

Fortran 77 Program To find $\bar{\mu}$.
& Goudrian's Function for (GCM)

```
      PROGRAM SUM
       DOUBLE PRECISION X(8),W(8),S,SUM1,SUM2
       REAL GAMMA
       EXTERNAL GAMMA

C      DATA FOR WEIGHTS & X POINTS
          X(1)= 0.0446339553
          X(2)= 0.1443662570
          X(3)= 0.2868247571
          X(4)= 0.4548133152
          X(5)= 0.6280678354
          X(6)= 0.7856915206
          X(7)= 0.9086763921
          X(8)= 0.9822200849

          W(1)= 0.0032951914
          W(2)= 0.0178429027
          W(3)= 0.0454393195
          W(4)= 0.0791995995
          W(5)= 0.1060473594
          W(6)= 0.1125057995
          W(7)= 0.0911190236
          W(8)= 0.0445508044
          XPI = 3.14159
C        CALCULATION OF BETA FUNCTION VALUES
         PRINT *, " THIS PROGRAM WILL CALCULATE THE MU BAR VALUES "
         PRINT *, "                              FOR "
         PRINT *,  "GAMMA DISTRIBUTED LEAF ANGLES "
         PRINT *, "  Give Values of A and B"
         read *, A,B

          A1=1.0 -A
          B1=1.0  -B
           C1=A1+B1+1.0
       GAFA=GAMMA(A1)

       GAFB=GAMMA(B1)
       GAFC= GAMMA(C1)
                 C0 = GAFC*(XPI/2)/(GAFA*GAFB)
       PRINT *, "VALUES OF GAMMA'S"
        PRINT *, GAFA,GAFB, GAFC
        PRINT *, "MEAN ANGLE FOR LEAF DISTRIBUTION"
        XML1= A1/C1
        XMLA= (XPI/2)*XML1
        PRINT *,XML1
         SUM2=0.0
         DO 20 J= 1,8
           THETA =XPI*X(J)/2.0

         SUM1 = 0.0
         DO 10 I= 1,8


C         ******(FUNCTION)******

       S =  W(I)/(X(I)*COS(THETA)+(SQRT(1-X(I)**2))*SIN(THETA))
       SUM1 = SUM1 + S
   10 CONTINUE

      SUM2=SUM2+W(J)*(X(J)**A1)*((1.0-X(J))**B1)*SUM1



  20  CONTINUE
```

```
      SUM2=SUM2*C0
      PRINT *, " VALUE OF MU BAR "
      PRINT *, SUM2

      STOP
      END
```

```
C         ***********************************
C         * Daily average of MU bar using   *
C         * GOUDRIAN'S Functions for (GCM)  *
C         ***********************************
          OPEN(8,FILE="DANA.DAT",STATUS="old")

          DO 100 I= 1,41

C         *********GIVENS********************

          XL= -0.2+(I-1)*0.01
          XFI= 0.5-(0.633*XL)-(0.33*(XL**2))
          XFI2= 0.877*(1-(2.0*XFI))
          XMUBAR= (1-(XFI/XFI2)*ALOG((XFI+XFI2)/XFI))/(XFI2)
          WRITE(8,*), XL, XMUBAR
  100     CONTINUE
          CLOSE (8)
          STOP
          END
```

TROPICAL RAINFALL MEASURING MISSION (TRMM)

THE VIBRATIONAL EFFECT OF SOLAR ARRAY MOTION ON TRMM.

BY

NNAEMEKA H. NWOSU
(SUMMER INTERN & AN UNDERGRADUATE MECHANICAL ENGINEERING STUDENT AT UNIVERSITY OF MARYLAND, COLLEGE PARK, MD.)
CODE 712.3

MENTORS

DR. APRILLE ERICSSON-JACKSON
(ANALYST, NATIONAL AERONAUTICS AND SPACE ADMINISTRATION (NASA))
CODE 712.3

DR. DARRELL ZIMBELMAN
(ANALYST, NATIONAL AERONAUTICS AND SPACE ADMINISTRATION (NASA))
CODE 745

CONTENTS

ABSTRACT

Most satellites require solar arrays to power them during their lifetime in orbit. Solar arrays are efficiently positioned so that they can experience maximum sun exposure. In the case of the Tropical Rainfall Measuring mission (TRMM) satellite, its solar panels quickly re-position as the satellite emerges from the dark phase to sunlight phase.

During the dark phase, the solar panels are in a "feathered" position. This minimizes the effect of drag on the satellite. As TRMM emerges from this phase, its solar panels begin tracking and rotating itself to the correct angular position, to receive the maximum solar illumination from the sun. The re-positioning of the solar panels to the desired position requires the use of Harmonic drives (H.d.) and Stepper motors. As in any mechanical system, vibrations arise from the movement of the Stepper motor and H.d. .

In this paper, the vibrational effect of these flexible solar array appendages (magnified by the Step motor and H.d.) on the entire satellite are examined. Using a FORTRAN based software called TREETOPS, the TRMM satellite can be modeled and simulated similar to the real life system. The main FORTRAN program in TREETOPS is first modified to accept various cases of angular displacements and angular rates for the solar panel. An input file containing the exact specifications of the different components of the satellite, is made. Through the incorporation of all of the above and running TREETOPS, one can generate sampled data output.

These results are plotted using MATLAB routines. By analyzing these time response plots one can effectively analyze the vibrational effect of the solar arrays motion on the spacecraft.

INTRODUCTION

As the need to understand the effect of global warming and other atmospheric conditions on earth grew, the National Aeronautics and Space Administration (NASA) has devised a program called the Mission to Planet Earth to respond to these needs[1]. One aspect of the Mission to Planet Earth, is the study of tropical rain and how it affects the Earth's climate. Tropical rainfall is critical in the re-distribution of water across the earth's surface. The energy released in the process determines the nature of the weather and climatic conditions of the earth. Droughts and floods may often times a result of the energy released from tropical rainfall.

To understand the above phenomenon, a workshop sponsored by both the National Aeronautical and Space Administration (NASA) and their Japanese counterpart National Space Development Agency (NASDA), was convened in 1985. A decision was reached to perform a feasibility study on the Tropical Rainfall Measuring Mission (TRMM). Upon the completion of the study in 1988, the objective of TRMM were stated as follows:

1. To obtain and study multi-year science data sets of tropical and subtropical rainfall measurements.
2. To understand how interactions between the sea, air, and land masses produce changes in global rainfall and climate.

3. To help improve modeling of tropical rainfall processes and their influence on global circulation in order to predict rainfall and variability at various time scale intervals.
4. To test, evaluate, and improve the performance of satellite rainfall estimate measurements and techniques[2].

The degree of accuracy needed by the TRMM project requires the use of specialized facilities and machinery. In addition, a satellite was necessary to provide more accurate rainfall measurements. State of the art rain measuring equipment carried by the TRMM satellite include the following:

A. The TRMM Microwave Imager (TMI) which measures the rate of rainfall detecting radiation emitted by clouds during heavy rainfall.
B. The Precipitation Radar (PR) which provides accurate 3-dimensional rainfall structures over the ocean and land.
C. The Visible Infrared Scanner (VIRS) which measures the variation of rain-water at different altitudes and also provides a means of determining condensation heat release.
D. The Clouds and Earth's Radiant Energy System (CERES) which measures the radiation emitted by both the earth and the clouds in relation to precipitation.

E. The Lighting Imaging Sensor (LIS) which investigates the rate and distribution of lighting over clouds.

(A Global Eye on Tropical Rainfall: The Tropical Rainfall Measuring Mission (TRMM))

The above instruments will be orbiting the earth at an altitude of 217.5 miles with an inclination of 35°. At this orbit, it is required that the mechanical components of TRMM (such as the stepper motor) not interfere with the functioning of the instruments on board TRMM. Since both the instruments and the stepper motor are frequency dependent, it is probable that both components may have similar natural frequencies resulting in resonance. Therefore, the objective of this paper is to study the motion of the stepper motor at a sampling rate of 2Hz and to determine the effect the associated vibrational frequencies have on the satellite, ultimately preventing resonance. A second objective is to study the effect of jitters on the spectral contents of both the drive frequency and motor resonance. These jitters are a result of the incremental motion of the stepper motor.

THE STEPPER MOTOR

Stepper motors are brushless DC motors which are capable of converting digital signals into fixed incremental shaft rotations. The rotation of the shaft attached to the solar arrays is achieved by aligning both the stationary and rotating parts of the motor magnetically[3]. Thus, a simple open loop system can be used to control the stepper motor. In addition, stepper motors have the ability to hold the last position of the solar arrays without dissipating power. These qualities make the stepper motor a more feasible choice over other types of motors.

There are primarily three types of stepper motors; variable reluctance, permanent magnet and hybrid types. The stepper motor utilized on the TRMM satellite is the permanent magnet type; it has a step size of 1.5°, a harmonic drive reduction gear of 200:1 and output step size of 0.0075°[3]. More specifically , a permanent magnet stepper motor package called the Gimbal and Solar Array Control Electronics (GSACE) stepper motor, is implemented and incorporated into the TRMM hardware configuration.

In order to receive maximum solar energy exposure during the light phase, the GSACE stepper motor tracks and keeps the major surface of the solar array panels normal to the sun. It also rotates the arrays to a "feathered" position during the dark phase in order to reduce the aerodynamic drag. To perform this tasks the stepper motor has to be guided by a commanded angle equation. The stepper motor depends on these equations of motion derived in the next section.

Stepper Motor Dynamic Equations of Motion

As in any mechanical system, the stepper motor can be described by a set of dynamic equations of motion. These equations relates the rotation of the gears to the rotation of the shaft. A brief derivation of these equations of motion is shown below.

Nomenclature:

GR = Speed ratio

J_1 = Moment of inertia of S/C (kg-m^2)

J_2 = Moment of inertia of motor rotor (kg-m^2)

θ_1 = Angular displacement of S/C (rads)

θ_2 = Angular displacement of motor rotor (rads)

θ_1 = Angular velocity of S/C (rads/sec)

θ_2 = Angular velocity of motor rotor (rads/sec)

θ_1 = Angular acceleration of S/C (rads/sec^2)

θ_2 = Angular acceleration of motor rotor (rads/sec^2)

K_{sa} = Stiffness coefficient (N-m/rads)

C_{sa} = Damping coefficient (N-m/rads-sec)

T_{flange} = Displacement of Solar array (rads)

DT_{flange} = Velocity of Solar array (rads/sec)

T_{rotor} = Solar array command (rads)

Cmd ang = Commanded angular displacement (degrees)

ξ = Damping ratio

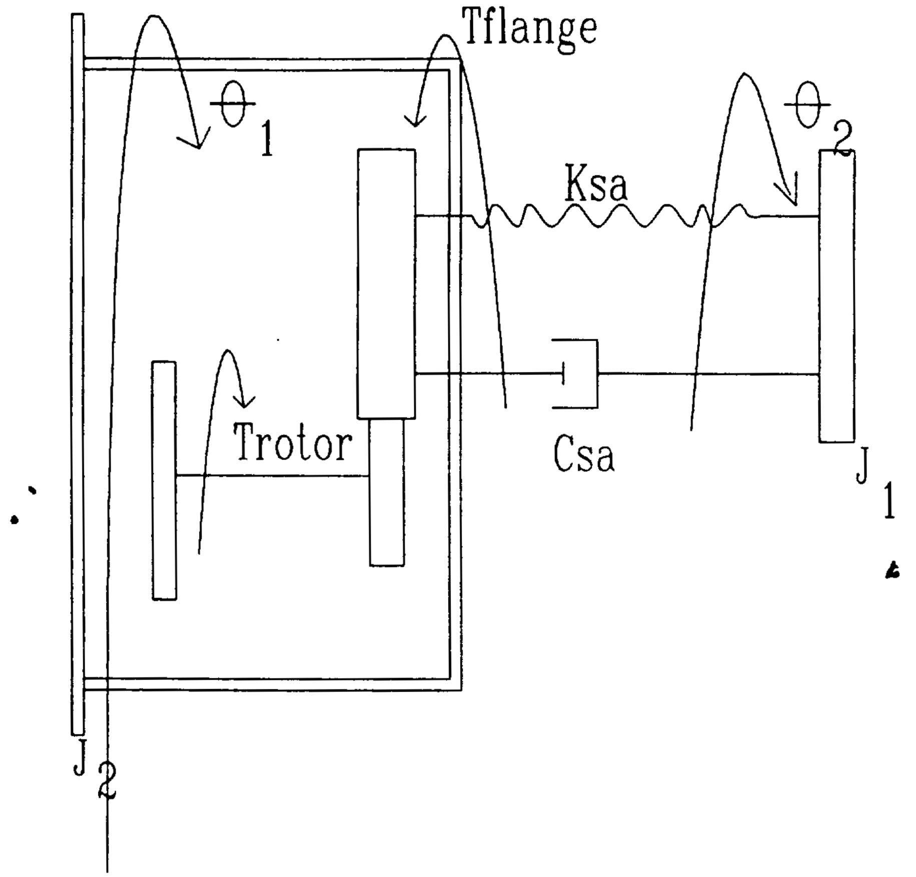


Fig.1 Schematics of the stepper motor.

Using Newton's Equations of motion

(1)

$$J_1\theta_1 = K_{sa}\ \{T_{flange} - (\theta_1-\theta_2)\} + Csa\ \{T_{flange} - (\theta_1-\theta_2)\}$$

(2)

$$J_2\theta_2 =-K_{sa}\ \{T_{flange} - (\theta_1-\theta_2)\} - Csa\ \{T_{flange} - (\theta_1-\theta_2)\}$$

Solving equations 1 and 2 for the acceleration terms

(3)

$$\theta_1 = [K_{sa}\ \{T_{flange} - (\theta_1-\theta_2)\} + Csa\ \{T_{flange} - (\theta_1-\theta_2)\}]/J_1$$

(4)

$$\theta_2 = [-K_{sa} (T_{flange} - (\theta_1-\theta_2)) - Csa (T_{flange} - (\theta_1-\theta_2))]/J_2$$

The rotor angle is given by

(5)

$$T_{rotor} = (\text{Cmd ang}) * GR$$

To this rotor angle, the solar array responds with an angular displacement and velocity given by the equations below

(6)

$$T_{flange} = T_{rotor} / GR + G_{error}$$

(7)

$$DT_{flange} = d/dt(\text{cmd ang}) + GR*d/dt(\text{cmd ang}) \{2*Fc2*\cos(2*T_{rotor}) + 4*Fc4*\cos(4*T_{rotor}) + 9*Fc9*\cos(9*T_{rotor})\}$$

G_{error} is given by

(8)

$$G_{error} = Fc2*\sin(2*T_{rotor}) + Fc4*\sin(4*T_{rotor}) + Fc9*\sin(9*T_{rotor})$$

The values of the constants used in the above equations are given in the table below. The inertia values were supplied by NASTRAN data output, the stiffness and the damping coefficients were derived by a first order approximation:

$\varpi = (K/J)^{1/2}$

$c = 2*\zeta*\varpi*J$

Table of Properties

Constants	Numerical Value
J_1	43.76 $Kg\text{-}m^2$
J_2	10254 $Kg\text{-}m^2$
K_{sa}	1741.9ads
C_{sa}	11.0435ds-sec
GR	200

Software used in Investigation

In this investigation, a simulation based software, TREETOPS and an analytical software, MATLAB, were used. TREETOPS was used to simulate the motor and solar array while MATLAB was used to analyze the result of the simulation.

TREETOPS:

TREETOPS is a software designed to simulate the motion of complex multi-body flexible structures with active control elements, over time[5]. Amongst other options TREETOPS consists of:

1. A program file written in FORTRAN.
2. An input file.
3. An output file.

Modifying the Program File

The program file consists of a main program linked with subsidiary subroutines, which contain equations that define the motion of the object(s) being modeled. This is where the equations of motion of the motor and the solar arrays are also defined. In addition, the program file was also developed to accept angular displacements and angular velocities. These modifications are shown in appendix A.

Creating an Input file

An input file contains the constants and properties such as the moment of inertia, relevant to defining the components and orbital motion of the system. TREETOPS generally models all systems as a collection of bodies inter-linked by hinges. This implies that for each modeled body, their individual properties and their relative position has to be defined accordingly. Other quantities needed for a complete description of the system include the sensors and actuators used, the controller, the duration of the simulation and also how the bodies are inter-connected. Below is a breakdown of some of the quantities needed to define a system.

Body:

1. Moment of inertia.
2. Product of moment of inertia.
3. Mass.
4. Number of nodes on body.
5. Positions of nodes.

Hinge:

1. Number of degrees of freedoms.

2. Initial translation.

3. Initial velocity.

4. Initial transaction stiffness.

5. Initial translation damping.

Sensor:

1. Mounting point.

2. Type.

Actuator:

1. Type.

2. Location.

3. Mounting point.

Controller:

1. Type.

2. Sample time.

3. Number of inputs.

4. Number of outputs.

Interconnection

1. Source type.

2. Destination type.

Simulation Control

1. Time duration of simulation.

2. Step size.

For a more detailed description of the content of the input file, refer to appendix B.

Output File

TREETOPS creates an output file in ASCII format after every simulation. This files are then loaded into MATLAB where plots are generated.

MATLAB:

MATLAB is an analytical software used in numerical and graphical computation[6]. For our purposes, the output file generated by running TREETOPS is loaded into MATLAB, and plots of important variables are generated. An m-file shown in appendix C is used to generate these plots.

PROCEDURE

Apparatus Used: A workstation or a personal computer with TREETOPS and MATLAB.

STEPS:

i. Divide the given commanded angular position into sections.

ii. Store these sections as different data files.

iii. Generate the input file in TREETOPS by inputting the necessary constants and variables required to define the system.

iv. Modify the Main TREETOPS subroutine to accept your stepper motor equations of motion.

v. Run TREETOPS by using the following steps:

a. Compile your main subroutine file by typing the command <MAKEL>.

b. Run the simulation by using the command TREETOPS <INPUT FILENAME>.

vi. Plot the output file using MATLAB.

RESULTS AND ANALYSIS

The first plot generated using MATLAB represents the commanded path the solar array has to follow.

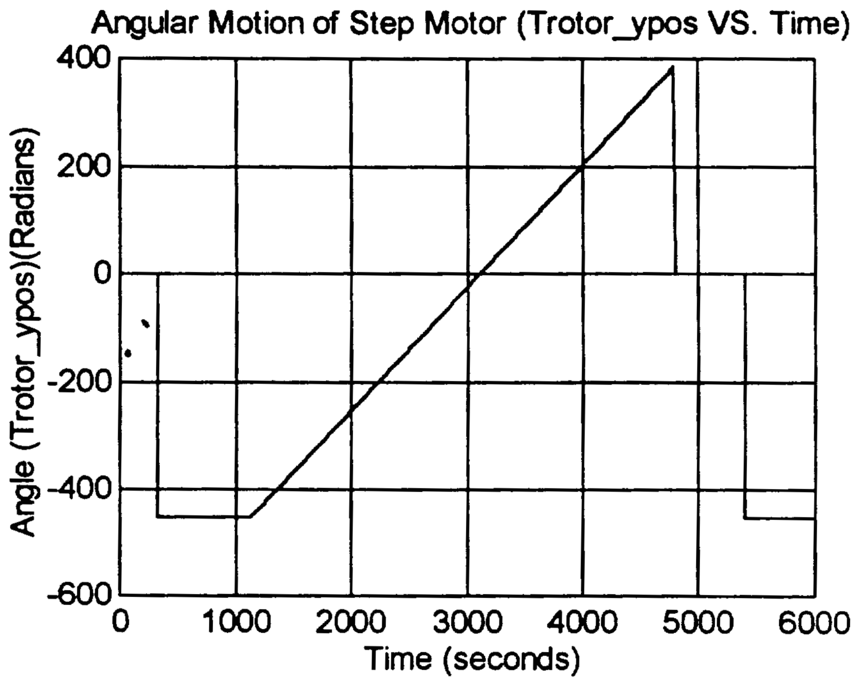


Fig. 2 Commanded path of the solar array.

The solar arrays are initially kept constant at an angular displacement of 0 rads for 323 sec and then kept constant at -450 rads for about 1000 sec. The angular displacement is then increased linearly for about 3000 secs and then kept at constant angular displacements for the rest of the path.

The ideal response of the solar array flanges to the commanded path should look very similar to the commanded path. It differs from the commanded path by a magnitude of 200 which is the corresponding gear reduction ratio for this motion.

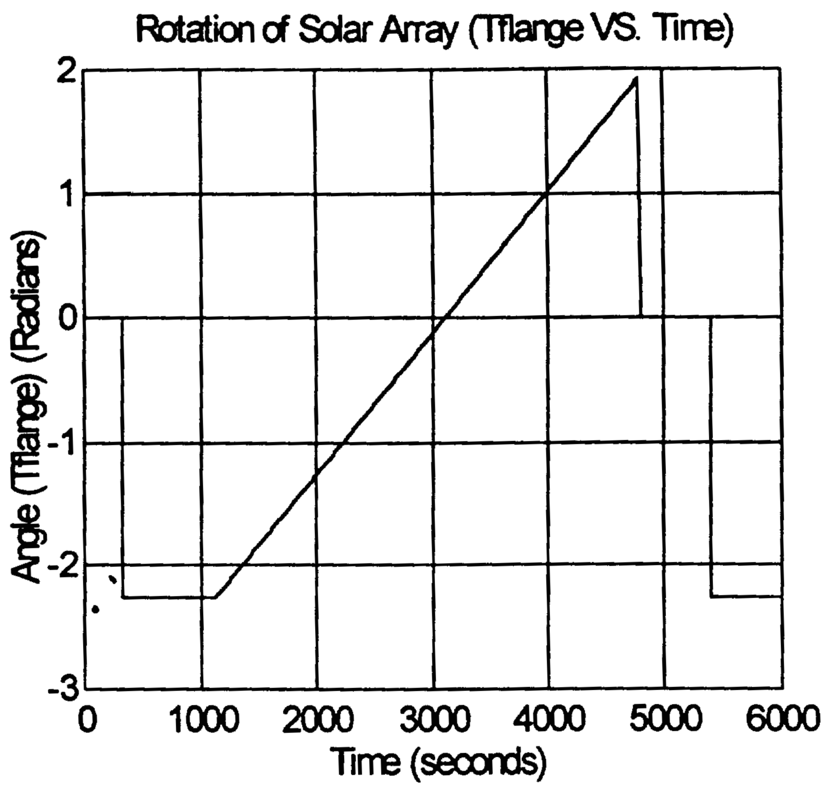


Fig. 3 Ideal solar array response to commanded path.

The above graph represents the response of the solar array without any errors. When the errors shown Fig. 4 are introduced into the system, the solar array motion overshoots its commanded path.

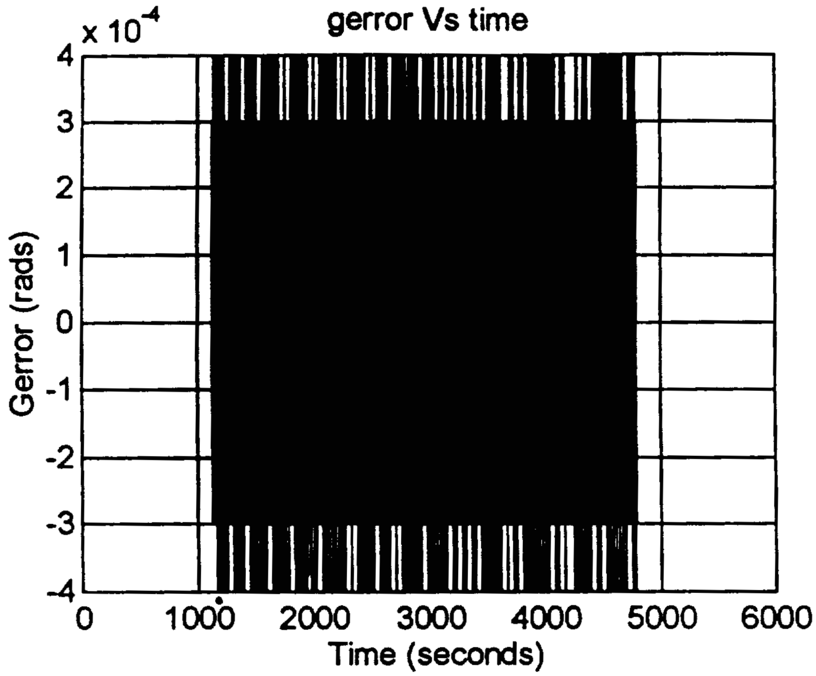


Fig. 4 Error introduced by the motion of the solar array.

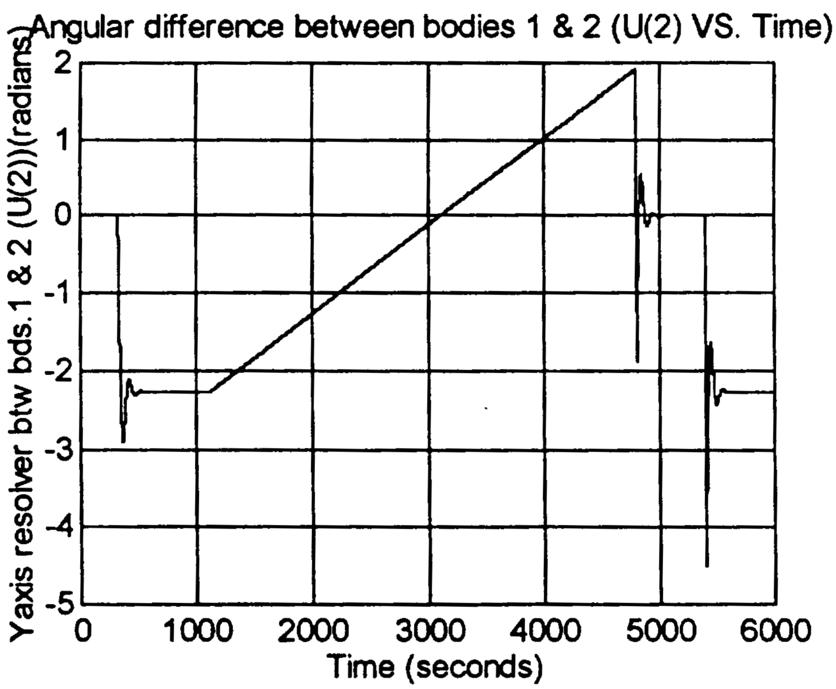


Fig. 5 Actual solar array response to commanded path.

The overshoots occur at instances where the commanded path is suddenly changed. This is the jittering effect mentioned earlier. Their occurrence is due to the under-damping of the system. Also as can be seen from the above plot, the change in commanded paths are not as horizontal or vertical as it was in the ideal case. This is because the stepper motor gradually gets to the next commanded path incrementally instead of jumping rapidly to that command. Although, the solar array overshoots , it still returns to the commanded path. This suggests that the stepper motor tracks very well. The effect of overshooting is shown on the next graph.

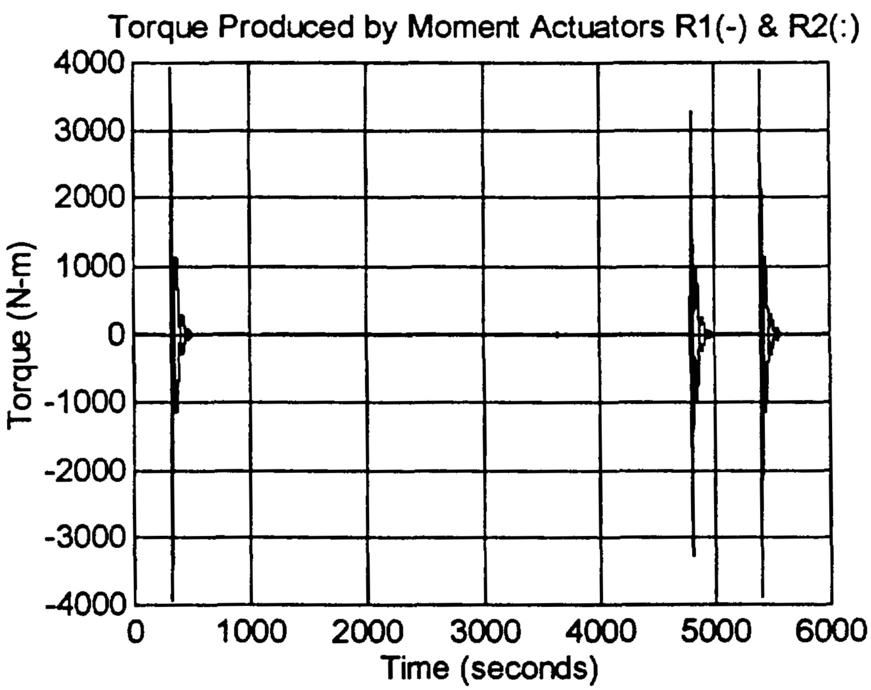


Fig 6. Torque generated by motion of the solar array.

As mentioned earlier, overshooting occurs at instances where the command changes. From the above graph, it can be seen that an equal and oppose vibrations are generated, thus canceling the jittering effect.

CONCLUSION

This investigation has shown that:

1. The jittering effect caused by changing the path of the solar array is insignificant because the effect is canceled out by an equal and opposite reaction.

2. The stepper motor corrects the overshoot caused by the change of path and effectively keeps the solar arrays within the commanded path.

3. These results have displayed no resonance effect that would critically damage the spacecraft, while the stepper motor is being sampled at 2 Hz.

Other areas for future studies would include:

1. An analytical study of the vibrational effect due to the motion of the negative solar array panel and the high gain antenna. This study is essential since only the positive solar array panels have been studied.

2. An investigation which incorporates the closed-loop control logic by integrating sensors and actuators.

REFERENCES

1. Reber, Carl "Mission to Planet Earth." Lecture at Goddard Space Flight Center (GSFC) ,Greenbelt, Md., 5 July, 1995.

2. NASAFacts. Tropical Rainfall Measuring Mission. Maryland: GPO, Feb. 1995.

3. Mcglew, Dave. "TRMM Solar Array Drive Actuator", NASA, May 28, 1991: 1-7.

4. Thomson, William T., Theory of Vibration with Applications. 4th ed. Englewood Cliffs, New Jersey: Prentice Hall, 1993.

5. Users Manual For Treetops. Maryland: NASA, 1995.

6. MATLAB Reference Guide. 1993 ed.

APPENDIX

APPENDIX A: PROGRAM FILE

```
C     U(1) = Y AXIS RESOLVER BETWEEN BODIES 0 & 1
C     U(2) = Y AXIS RESOLVER BETWEEN BODIES 1 & 2
C     U(3) = Y AXIS INTEGRATING RATE GYRO ON BODY 2
C     R(1) = BODY 1 MOMENT ACTUATOR
C     R(2) = BODY 2 MOMENT ACTUATOR
C
      SUBROUTINE USDC(TIME,U,R)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      REAL*8 U(3), R(2)
      save
     logical init
     integer i, iplot, nplot, itime,itcnt
     real*8 outvar(12000)
     REAL*8 torq(2), ncols
     data ncols / 12 /
     data init / .true. /
     data itime / 100 /
     data itcnt / -1 /
     itcnt = itcnt + 1
      if (itcnt.eq.itime) then
        itcnt = 0
        write(6,*) ' Sim. Time =',time
      endif
C
C  Initial variables
      if (init) then
        init = .false.
        nplot = 100
        iplot = nplot - 1
        open (unit=14, file='data.asc', status='unknown')
      endif                        ! End of initialization
C
C     Step Motor
C
     call stepmotor(time,u,ang_ypos, ang_yneg, torq,
     &   trotor_ypos, tflange_ypos, dtflange_ypos,
     &   ddtheta1_ypos, ddtheta2_ypos,Gerror_ypos)
     r(1)=torq(1)
     r(2)=torq(2)
C
C  Output data for plotting
C
     iplot = iplot + 1
     if (iplot .eq. nplot) then
        iplot = 0
        outvar(1) = time
        outvar(2) = u(1)
        outvar(3) = u(2)
```

```
          outvar(4) = u(3)
          outvar(5) = r(1)
          outvar(6) = r(2)
          outvar(7) = trotor_ypos
          outvar(8) = tflange_ypos
          outvar(9) = dtflange_ypos
          outvar(10) = ddtheta1_ypos
          outvar(11) = ddtheta2_ypos
          outvar(12) = Gerror_ypos
        write(14,4) (outvar(i),i=1,ncols)
4       format(12(2x,f30.4))
       endif
        END                        ! End of main subroutine.


        SUBROUTINE stepmotor(time,u,ang_ypos, ang_yneg,
     &  torq, trotor_ypos, tflange_ypos, dtflange_ypos,
     &  ddtheta1_ypos, ddtheta2_ypos,Gerror_ypos)
C
C Subroutine to model 2-DOF stepper motor, Y-axis
C Incorporates gear error and gear reduction
C   into dynamic eqns. of motion
C
        REAL*8 ang_ypos, ang_yneg, dtang_ypos, dtang_yneg,
     & torq(2), ksa, csa, Jsc, Jsa_pos, ddtheta1_ypos,
     & ddtheta2_ypos,u(3), uprev, dtU2
        REAL*8 time,time1,deltime,pi, d2r, gr, trotor_ypos,
               fc2, fc4
        REAL*8 fc9, gerror_ypos, dtflange_ypos, tflange_ypos
        INTEGER counter,n,deltr,delts,deltg
        SAVE
        COMMON /count/ counter
        COMMON uprev
        LOGICAL INIT
        DATA INIT /.TRUE./
        IF (INIT) THEN
        OPEN(4194, file='sacmds.dat', status='old')
        INIT = .FALSE.
        END IF
C       Reading in data at 0.5 sec
        delts=100
        deltg=2
        deltr=delts/deltg
        n=n+1
        if(n.eq.deltr)then
          n=0
          write(*,*) counter
          counter = counter+1
          READ(4194,*) ang_ypos, ang_yneg, dtang_ypos,
                   dtang_yneg
           print*, ang_ypos, ang_yneg, dtang_ypos, dtang_yneg
        endif
C
C  rotor angle in radians
```

```
      pi = 4.0d0*datan(1.0d0)
      d2r = pi/180.0d0     ! degrees to radians
      GR = 200.0d0            ! Gear reduction
      trotor_ypos = ang_ypos*GR*d2r
C
C  Gear error
C
      Fc2 = 0.00025d0
      Fc4 = 0.5d0*Fc2
      Fc9 = 0.2d0*Fc2
      Gerror_ypos = Fc2*dsin(2.0d0*trotor_ypos)
      &            + Fc4*dsin(4.0d0*trotor_ypos)
      &            + Fc9*dsin(9.0d0*trotor_ypos)
C
C  flange angle in radians
C
      tflange_ypos = ang_ypos*d2r + Gerror_ypos
C
C  calculate 1st derivative of flange angle
C
      dtflange_ypos = dtang_ypos*d2r
      &            + 2.0d0*GR*dtang_ypos*d2r
      &            *Fc2*dcos(2.0d0*trotor_ypos)
      &            + 4.0d0*GR*dtang_ypos*d2r
      &            *Fc4*dcos(4.0d0*trotor_ypos)
      &            + 9.0d0*GR*dtang_ypos*d2r
      &            *Fc9*dcos(9.0d0*trotor_ypos)
C
C  Coefficients and loading inertias
C
      ksa = 1741.9      ! stiffness coefficient -SA- N-m/rad
      csa = 11.0435     ! damping coefficient -SA- N-m/rad/sec
      Jsa_pos = 4.376D1       ! Rotational inertia for +Y SA
                                                      kg-m^2
      Jsc     = 1.0254D4      ! Rotational inertia for  S/C
                                                       kg-m^2
C
C  Rate of change of the relative angular difference between
C   bodies 1 & 2
C
       deltime=time-time1
       time1=time
       if(time.gt.0.0d0) dtU2=(u(2)-uprev)/deltime
C
C  Dynamic Equations of Motion
C
      ddtheta1_ypos = (ksa*(tflange_ypos-u(2))
      &            + csa*(dtflange_ypos-dtU2))/Jsa_pos
C
      ddtheta2_ypos = (-ksa*(tflange_ypos-u(2))
      &            - csa*(dtflange_ypos-dtU2))/Jsc
C
C  Torque for +Y Solar Array and Spacecraft
C
```

```
torq(2) = ddtheta1_ypos*Jsa_pos
torq(1) = ddtheta2_ypos*Jsc
uprev = u(2)
 return
 end
```

APPENDIX B: INPUT FILE

SIM CONTROL

```
   1 SI    0 Title
             MOTOR SIMULATION DEGREE-2
   2 SI    0 Simulation stop time
             6000
   3 SI    0 Plot data interval
             .5
   4 SI    0 Integration type (R,S or U)
             R
   5 SI    0 Step size (sec)
             .01
   6 SI    0 Sandia integration absolute and relative error
   7 SI    0 Linearization option  (L,Z or N)
             N
   8 SI    0 Restart option        (Y/N)
             N
   9 SI    0 Contact force computation option     (Y/N)
             N
  10 SI    0 Constraint force computation option  (Y/N)
             N
  11 SI    0 Small angle speedup option
             (All,Bypass,First,Nth)  ALL
  12 SI    0 Mass matrix speedup option
             (All,Bypass,First,Nth)  A
  13 SI    0 Non-Linear  speedup option
             (All,Bypass,First,Nth)  A
  14 SI    0 Constraint  speedup option
             (All,Bypass,First,Nth)  A
  15 SI    0 Constraint stabilization option  (Y/N)
             N
  16 SI    0 Stabilization epsilon
```

BODY

```
  17 BO    1 Body ID number
             1
  18 BO    1 Type (Rigid,Flexible,NASTRAN)
             R
  19 BO    1 Number of modes
  20 BO    1 Modal calculation option (0, 1 or 2)
  21 BO    1 Foreshortening option (Y/N)
  22 BO    1 Model reduction method (NO,MS,MC,CC,QM,CV)
  23 BO    1 NASTRAN data file FORTRAN unit number (40 - 60)
```

```
24 BO    1 Number of augmented nodes (0 if none)
25 BO    1 Damping matrix option (NS,CD,HL,SD)
26 BO    1 Constant damping ratio
27 BO    1 Low frequency, High frequency ratios
28 BO    1 Mode ID number, damping ratio
29 BO    1 Conversion factors: Length,Mass,Force
30 BO    1 Inertia reference node (0=Bdy Ref Frm; 1=mass
            cen) 0
31 BO    1 Moments of inertia (kg-m2) Ixx,Iyy,Izz
           2527.3 10254 9694.3
32 BO    1 Products of inertia (kg-m2) Ixy,Ixz,Iyz
           -28.5 -200.7 24.7
33 BO    1 Mass (kg)
           3289.1
34 BO    1 Number of Nodes
3
35 BO    1 Node ID, Node coord. (meters) x,y,z
           1 1.3 -.0111 -.0423
36 BO    1 Node ID, Node coord. (meters) x,y,z
           2 0 0 0
37 BO    1 Node ID, Node coord. (meters) x,y,z
           3 .64 1.180 0
38 BO    1 Node ID, Node structual joint ID

39 BO    2 Body ID number
           2
40 BO    2 Type (Rigid,Flexible,NASTRAN)
           R
41 BO    2 Number of modes
42 BO    2 Modal calculation option (0, 1 or 2)
43 BO    2 Foreshortening option (Y/N)
44 BO    2 Model reduction method (NO,MS,MC,CC,QM,CV)
45 BO    2 NASTRAN data file FORTRAN unit number (40 - 60)
46 BO    2 Number of augmented nodes (0 if none)
47 BO    2 Damping matrix option (NS,CD,HL,SD)
48 BO    2 Constant damping ratio
49 BO    2 Low frequency, High frequency ratios
50 BO    2 Mode ID number, damping ratio
51 BO    2 Conversion factors: Length,Mass,Force
52 BO    2 Inertia reference node (0=Bdy Ref Frm; 1=mass
           cen) 0
53 BO    2 Moments of inertia (kg-m2) Ixx,Iyy,Izz
           1339.3 43.763 1335.2
54 BO    2 Products of inertia (kg-m2) Ixy,Ixz,Iyz
           -39.636 0 0
55 BO    2 Mass (kg)
           99.69
56 BO    2 Number of Nodes
           2
57 BO    2 Node ID, Node coord. (meters) x,y,z
           1 0 3.178 0
58 BO    2 Node ID, Node coord. (meters) x,y,z
           2 0 0 0
59 BO    2 Node ID, Node structual joint ID
```

HINGE

```
60 HI   1 Hinge ID number
          1
61 HI   1 Inboard body ID, Outboard body ID
          0 1
62 HI   1 "p" node ID, "q" node ID
          0 1
63 HI   1 Number of rotation DOFs, Rotation option (F or
          G)  1 F
64 HI   1 L1 unit vector in inboard body coord. x,y,z
          0 1 0
65 HI   1 L1 unit vector in outboard body coord. x,y,z
          0 1 0
66 HI   1 L2 unit vector in inboard body coord. x,y,z
67 HI   1 L2 unit vector in outboard body coord. x,y,z
68 HI   1 L3 unit vector in inboard body coord. x,y,z
          0 0 1
69 HI   1 L3 unit vector in outboard body coord. x,y,z
          0 0 1
70 HI   1 Initial rotation angles (deg)
          0 0 0
71 HI   1 Initial rotation rates (deg/sec)
          0
72 HI   1 Rotation stiffness (newton-meters/rad)
          0
73 HI   1 Rotation damping (newton-meters/rad/sec)
          0
74 HI   1 Null torque angles (deg)
          0
75 HI   1 Number of translation DOFs
          0
76 HI   1 First translation unit vector   g1
          1 0 0
77 HI   1 Second translation unit vector  g2
          0 1 0
78 HI   1 Third translation unit vector   g3
          0 0 1
79 HI   1 Initial translation (meters)
          0 0 0
80 HI   1 Initial translation velocity (meters/sec)
81 HI   1 Translation stiffness (newtons/meters)
82 HI   1 Translation damping (newtons/meter/sec)
83 HI   1 Null force translations

84 HI   2 Hinge ID number
          2
85 HI   2 Inboard body ID, Outboard body ID
          1 2
86 HI   2 "p" node ID, "q" node ID
          3 2
87 HI   2  No of rotation DOFs, Hinge 1 rotation
```

```
            option(F/G) 1
 88 HI    2 L1 unit vector in inboard body coord. x,y,z
            0 1 0
 89 HI    2 L1 unit vector in outboard body coord. x,y,z
            0 1 0
 90 HI    2 L2 unit vector in inboard body coord. x,y,z
 91 HI    2 L2 unit vector in outboard body coord. x,y,z
 92 HI    2 L3 unit vector in inboard body coord. x,y,z
            0 0 1
 93 HI    2 L3 unit vector in outboard body coord. x,y,z
            0 0 1
 94 HI    2 Initial rotation angles (deg)
            0 0 0
 95 HI    2 Initial rotation rates (deg/sec)
            0
 96 HI    2 Rotation stiffness (newton-meters/rad)
            0
 97 HI    2 Rotation damping (newton-meters/rad/sec)
            0.
 98 HI    2 Null torque angles (deg)
            0
 99 HI    2 Number of translation DOFs
            0
100 HI    2 First translation unit vector   g1
            1 0 0
101 HI    2 Second translation unit vector  g2
            0 1 0
102 HI    2 Third translation unit vector   g3
            0 0 1
103 HI    2 Initial translation (meters)
            0 0 0
104 HI    2 Initial translation velocity (meters/sec)
105 HI    2 Translation stiffness (newtons/meters)
106 HI    2 Translation damping (newtons/meter/sec)
107 HI    2 Null force translations

            SENSOR


108 SE    1 Sensor ID number
            1
109 SE    1 Type (G,R,AN,V,P,AC,T,I,SU,ST,IM,P3,V3,CR,CT)
            R
110 SE    1 Mounting point body ID, Mounting point node ID
111 SE    1 Second mounting point body ID, Second node ID
112 SE    1 Input axis unit vector (IA) x,y,z
113 SE    1 Mounting point Hinge index, Axis index
            1 1
114 SE    1 First focal plane unit vector  (Fp1) x,y,z
115 SE    1 Second focal plane unit vector (Fp2) x,y,z
116 SE    1 Sun/Star unit vector (Us) x,y,z
117 SE    1 Euler Angle Sequence (1-6)
118 SE    1 CMG ID number and Gimbal number
```

```
119 SE    2 Sensor ID number
            2
120 SE    2 Type (G,R,AN,V,P,AC,T,I,SU,ST,IM,P3,V3,CR,CT)
            R
121 SE    2 Mounting point body ID, Mounting point node ID
122 SE    2 Second mounting point body ID, Second node ID
123 SE    2 Input axis unit vector (IA) x,y,z
124 SE    2 Mounting point Hinge index, Axis index
            2 1
125 SE    2 First focal plane unit vector  (Fp1) x,y,z
126 SE    2 Second focal plane unit vector (Fp2) x,y,z
127 SE    2 Sun/Star unit vector (Us) x,y,z
128 SE    2 Euler Angle Sequence (1-6)
129 SE    2 CMG ID number and Gimbal number

130 SE    3 Sensor ID number
            3
131 SE    3 Type (G,R,AN,V,P,AC,T,I,SU,ST,IM,P3,V3,CR,CT)
            I
132 SE    3 Mounting point body ID, Mounting point node ID
            2 1
133 SE    3 Second mounting point body ID, Second node ID
134 SE    3 Input axis unit vector (IA) x,y,z
            0 1 0
135 SE    3 Mounting point Hinge index, Axis index
136 SE    3 First focal plane unit vector  (Fp1) x,y,z
137 SE    3 Second focal plane unit vector (Fp2) x,y,z
138 SE    3 Sun/Star unit vector (Us) x,y,z
139 SE    3 Euler Angle Sequence (1-6)
140 SE    3 CMG ID number and Gimbal number

            ACTR


141 AC    1 Actuator ID number
            1
142 AC    1 Type(J,H,MO,T,B,MA,SG,DG,W,L,M1-M7)
            MO
143 AC    1 Actuator location; Node or Hinge (N or H)
144 AC    1 Mounting point body ID number, node ID number
            1 3
145 AC    1 Second mounting point body ID, second node ID
146 AC    1 Output axis unit vector x,y,z
            0 1 0
147 AC    1 Mounting point Hinge index, Axis index
148 AC    1 Rotor spin axis unit vector x,y,z
149 AC    1 Initial rotor momentum, H
150 AC    1 Outer gimbal-
            angle(deg),inertia,friction(D,S,B,N)
151 AC    1 Outer gimbal axis unit vector x,y,z
152 AC    1 Out gim fric
            (Tfi,Tgfo,GAM)/(Tfi,M,D,Kf)/(m,M,B,k)
153 AC    1 Inner gimbal-
            angle(deg),inertia,friction(D,S,B,N)
```

```
154 AC    1 Inner gimbal axis unit vector x,y,z
155 AC    1 In gim fric
            (Tfi,Tgfo,GAM)/(Tfi,M,D,Kf)/(m,M,B,k)
156 AC    1 Initial length and rate, y(to) and ydot(to)
157 AC    1 Constants; K1 or wo, n or zeta, Kg, Jm
158 AC    1 Non-linearities; TLim, Tco, Dz

159 AC    2 Actuator ID number
            2
160 AC    2 Type(J,H,MO,T,B,MA,SG,DG,W,L,M1-M7)
            MO
161 AC    2 Actuator location; Node or Hinge (N or H)
162 AC    2 Mounting point body ID number, node ID number
            2 2
163 AC    2 Second mounting point body ID, second node ID
164 AC    2 Output axis unit vector x,y,z
            0 1 0
165 AC    2 Mounting point Hinge index, Axis index
166 AC    2 Rotor spin axis unit vector x,y,z
167 AC    2 Initial rotor momentum, H
168 AC    2 Outer gimbal-
            angle(deg),inertia,friction(D,S,B,N)
169 AC    2 Outer gimbal axis unit vector x,y,z
170 AC    2 Out gim fric
            (Tfi,Tgfo,GAM)/(Tfi,M,D,Kf)/(m,M,B,k)
171 AC    2 Inner gimbal-
            angle(deg),inertia,friction(D,S,B,N)
172 AC    2 Inner gimbal axis unit vector x,y,z
173 AC    2 In gim fric
            (Tfi,Tgfo,GAM)/(Tfi,M,D,Kf)/(m,M,B,k)
174 AC    2 Initial length and rate, y(to) and ydot(to)
175 AC    2 Constants; K1 or wo, n or zeta, Kg, Jm
176 AC    2 Non-linearities; TLim, Tco, Dz

            CONTROLLER

177 CO    1 Controller ID number
            1
178 CO    1 Controller type (CB,CM,DB,DM,UC,UD)
            UD
179 CO    1 Sample time (sec)
            .01
180 CO    1 Number of inputs, Number of outputs
            3 2
181 CO    1 Number of states
182 CO    1 Output No., Input type (I,S,T), Input ID, Gain

            INTERCONNECT

183 IN    1 Interconnect ID number
            1
184 IN    1 Source type(S,C, or F),Source ID,Source row #
```

```
              S 1 1
185 IN      1 Destination type(A or C),Dest ID,Dest row #
              C 1 1
186 IN      1 Gain
              1

187 IN      2 Interconnect ID number
              2
188 IN      2 Source type(S,C, or F),Source ID,Source row #
              C 1 1
189 IN      2 Destination type(A or C),Dest ID,Dest row #
              A 1 1
190 IN      2 Gain
              1

191 IN      3 Interconnect ID number
              3
192 IN      3 Source type(S,C, or F),Source ID,Source row #
              C 1 2
193 IN      3 Destination type(A or C),Dest ID,Dest row #
              A 2 1
194 IN      3 Gain
              1

195 IN      4 Interconnect ID number
              4
196 IN      4 Source type(S,C, or F),Source ID,Source row #
              S 2 1
197 IN      4 Destination type(A or C),Dest ID,Dest row #
              C 1 2
198 IN      4 Gain
              1

199 IN      5 Interconnect ID number
              5
200 IN      5 Source type(S,C, or F),Source ID,Source row #
              S 3 1
201 IN      5 Destination type(A or C),Dest ID,Dest row #
              C 1 3
202 IN      5 Gain
              1
```

APPENDIX C: MATLAB GRAPHING PROGRAM

```
%  Plot motor simulation data
%t=c(:,1); U1=c(:,2); U2=c(:,3); U3=c(:,4);
%R1=c(:,5); R2=c(:,6);
%trotor_ypos=c(:,7); tflange_ypos=c(:,8);
%dtflange_ypos=c(:,9),ddtheta1_ypos=c(:,10),
ddtheta2_ypos=c(:,1);
%
load data.asc
%
% Rename variables
c=data;
t=c(:,1);
U1=c(:,2); U2=c(:,3); U3=c(:,4);
R1=c(:,5); R2=c(:,6);
trotor_ypos=c(:,7); tflange_ypos=c(:,8);
dtflange_ypos=c(:,9);ddtheta1_ypos=c(:,10);
ddtheta2_ypos=c(:,11);
%
clear c
save motorsim
%
% Plot Data
plot(t, U1, '-')
title(' Angular difference between bodies 0 & 1 (U(1) VS.
Time)')
xlabel(' Time (seconds)'); ylabel('Yaxis resolver btw bds.1
& 2 (U(1))(radians)');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph1'])
print
pause

plot(t, U2, '-')
title(' Angular difference between bodies 1 & 2 (U(2) VS.
Time)')
xlabel(' Time (seconds)'); ylabel('Yaxis resolver btw bds.1
& 2 (U(2))(radians)');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph2'])
print
pause

plot(t, U2,'-.',t,tflange_ypos, '-')
title('t vs U2(:) & tflange(-)')
xlabel(' Time (seconds)'); ylabel(' U(2) & Tflange
(radians)');
```

```
grid
%eval(['print -deps graph3'])
print
pause

plot(t,R1,'-',t,R2,'-.')
title(' Torque Produced by Moment Actuators R1(-) & R2(:)')
xlabel(' Time (seconds)'); ylabel('Torque (N-m)');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph4'])
print
pause

plot(t, trotor_ypos, '-')
title(' Angular Motion of Step Motor (Trotor_ypos VS.
Time)')
xlabel(' Time (seconds)'); ylabel('Angle
(Trotor_ypos)(Radians) ');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph5'])
print
pause

plot(t, tflange_ypos, '-')
title(' Rotation of Solar Array (Tflange VS. Time)')
xlabel(' Time (seconds)'); ylabel('Angle (Tflange)
(Radians)');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph6'])
print
pause

plot(t, dtflange_ypos, '-')
title(' Rate of rotation of Solar Array (Dtflange VS.
Time)')
xlabel(' Time (seconds)'); ylabel('Angle (Dtflange)
(Radians/Sec)');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph7'])
print
pause

plot(t, ddtheta1_ypos, '-', t, ddtheta2_ypos, '-.')
title(' Angular Accel.1(ddtheta1 (-)) & Accel.2 (ddtheta2 (-
.))')
xlabel(' Time (seconds)'); ylabel('Acceleration
(degrees/sec^2)');
grid
%gtext('Data Range: 0 - 646')
%eval(['print -deps graph8'])
```

```
print

plot(t, gerror)
title('gerror Vs time')
xlabel(' Time (seconds)'); ylabel('Gerror (rads)');
grid
%eval(['print -deps graph8'])
print
```

THE DESIGN AND IMPLEMENTATION OF THE CCSDS SIMULATOR CHIP

(A SIECA SUMMER EXPERIENCE)

Miguel Angel Ordaz

The University of Texas at El Paso

Summer Institute in Engineering

and Computer Applications

Summer 1995

Code 521.2

TABLE OF CONTENTS

INTRODUCTION

For the last ten weeks I have been working at NASA/Goddard Space Flight Center as an undergraduate-intern in the SIECA program. I am a third-year undergraduate student at the University of Texas at El Paso, where I expect to receive my Bachelor of Science degree in Electrical Engineering in December of 1996. Currently at Goddard, I am working in the Systems Applications Section of the Microelectronic Systems Branch of the Data Systems Technology Division, also known as Code 521.2.

The System Applications Section is responsible for the near term application of custom and commercial VLSI technology to NASA communications data systems. The focus of 521 is to design and develop custom components and systems not met by commercial sources and the integration of these custom components and systems with commercial components. The primary goal of 521 is to identify NASA Communications functions, subsystems, and systems for the application of VLSI technology and to design, develop, test and fabricate prototype systems. State-of-the-art Cadence Software on the SUN workstations in the VLSI Design Laboratory is used to design and simulate subsystem components. These efforts are focused in cooperative inter-organizational projects and Directorate testbed activities intended to demonstrate and evaluate technology use and application.

My summer project entails the design and implementation of the CCSDS Simulator Chip.The CCSDS Simulator Chip will be capable of generating instrument packets given the packet header. These packets will then be formatted into frames as recommended by the Consultative Committee for Space Data Systems(CCSDS). This project intends to reduce both the complexity and size of current telemetry simulator technology.

General Description and Rationale

Future Earth Observing System (EOS) spacecraft carrying high data rate instruments and employing the Consultative Committee for Space Data Systems (CCSDS) telemetry standards have created a need for a low cost, high rate CCSDS data simulation capability for flight and ground data system testing and verification. The Data Systems Technology Division at Goddard is developing a high performance telemetry data simulator chip in support of EOS. The CCSDS Simulator Chip is capable of outputting simulated satellite telemetry data in the CCSDS AOS format at any rate up to 150Mbps. By using the CCSDS approach to satellite telemetry, specific simulations and analysis scenarios may be implemented for system testing according to mission requirements. The CCSDS Simulator Chip will replace the current simulator technology thereby reducing size, cost and complexity.

The CCSDS recommends that satellite telemetry be formatted using two data structures. The first is the source packet, which provides protocol data formatting services so that data may be exchanged between a source application process in space and its associated user application process on the ground. A source packet encapsulates a block of observational and ancillary application data which is to be transmitted and permits future versions of the data structure to be defined if required (See Figure 1 for data structures).

The second data structure is the VCDU, which facilitates movement of the packetized or segmented source data through the spacecraft-to-ground communications path. The VCDU also provides a mechanism to time-share the data link between sources by creating logical virtual channels.

The CCSDS Simulator Chip is a VLSI chip designed to simulate satellite telemetry following the source packet and VCDU formats (See figure 1 for data structures). Functionally, the chip will conform to these formats allowing it to be used in various systems requiring simulated telemetry data. Packet headers and the associated virtual channels are supplied via a CPU to the FIFO interface of the CCSDS Simulator Chip. Source packets will be generated and supplied to the MPDU blocker. The MPDU's will then be formatted into the VCDU data unit zone. Virtual channel sequence counts will be maintained in hardware. These VCDU's will then be output as a CCSDS return link data stream.

SOURCE PACKET DIAGRAM

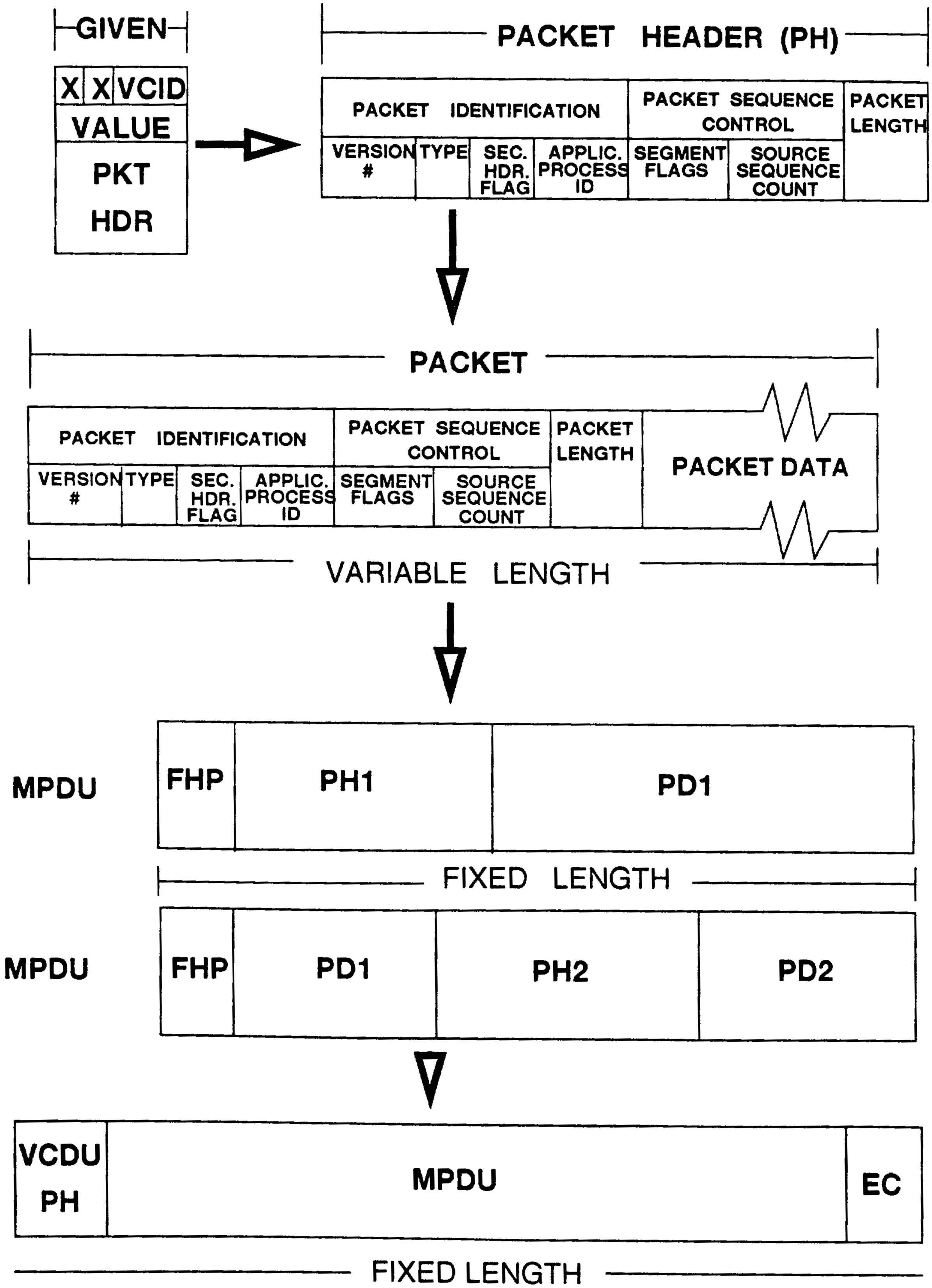


Figure 1

Design Process

The design process used by the Systems Applications Section is completed in several steps. The designer must first gather and meet all requirements and recommendations by the CCSDS and Code 521. The requirements include but are not limited to asserting the functional tasks that are to be performed and also determining the interface specifications.

After completing the first step a functional design must be developed; it includes block diagrams, flow charts and communication protocols. A Preliminary Design Review is scheduled after the initial functional design is completed. The Preliminary Design Review is held in a conference room and it is where the functional design is presented to engineers in the section, which gives them the opportunity to offer constructive criticism on the functional design and change requirements and interface specifications if needed.

The next step involves the use of Cadence software to perform schematic capture and simulation. The schematics are based on the functional design and the block diagram developed from the requirements and interface specifications. The simulation allows the designer to verify the functions of the schematics by analyzing timing diagrams. Once the circuit schematics and simulations are completed a Critical Design Review is scheduled.

The Critical Design Review (CDR) is set up just like the Preliminary Design Review except that now the review is highly technical and permits the reviewing engineers to critique the design as a whole, which includes schematics, timing diagrams and architecture. Layout of the printed circuit board is then performed by a layout engineer. The printed circuit board is fabricated while the design engineer prepares the hardware documentation and software development using the VXWORKS operating system. Once the board is fabricated and returned to the designer, the prototype is debugged and integrated into an operational system.

Figure 2 demonstrates the design process in block diagram form. The designer must be aware at all times of the necessary steps to take and complete while completing a design and uses the design process to meet schedules and deadlines. Thus, the design engineers at code 521 foment their design techniques by employing the use of the code 521 design process.

DESIGN PROCESS DIAGRAM

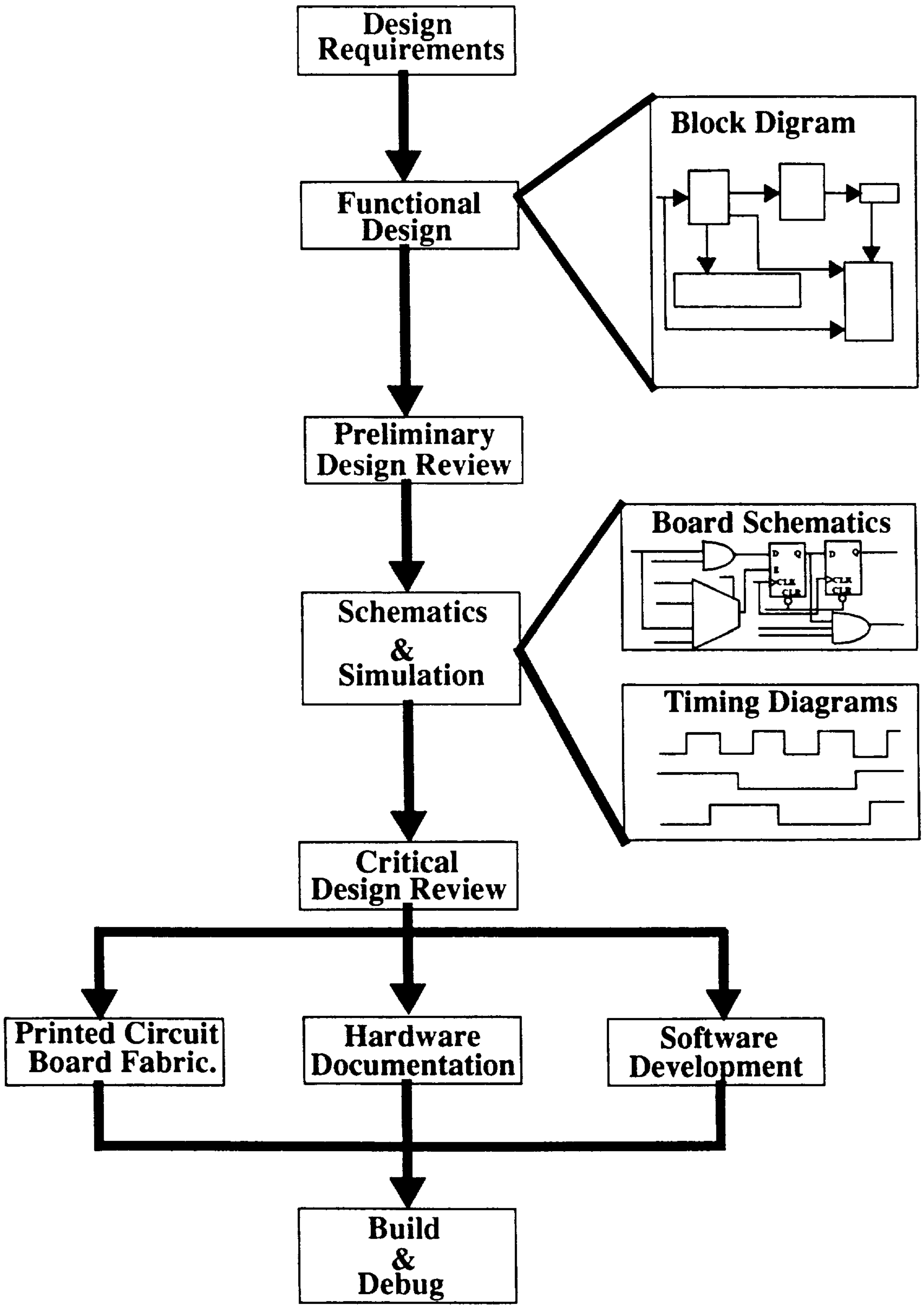


Figure 2

CCSDS Simulator Chip

These are the basic requirements of CCSDS Simulator Chip, as recommended by Code 521:

a. Be capable of outputting simulated satellite telemetry data in the CCSDS AOS format at any rate, up to 150Mbps.

1. To generate a source packet given the packet header

2. To generate the sequence counters for each of the virtual channels.

b. To maintain all existing functions of the current simulator technology thereby reducing size, cost and complexity.

c. To provide a CCSDS return link data stream, to be used for specific simulations and analysis.

d. Accept external input clock for synchronization of output from packet generator to MCDU blocker and to the frame formatter and TLU.

Figure 3 gives the general functional block diagram of the CCSDS Simulator Chip. The FIFO accepts data provided through the CPU, which serves as the interface, along with the FIFO, to the CCSDS Simulator Chip. The packet generator will generate a source packet given the packet header via the FIFO. The FIFO's read-enable is generated from the packet generator. The packet generator will then provide the VCID to the MPDU blocker and the frame formatter. MPDU's will then be formatted in the MPDU blocker, into the VCDU data unit zone. The virtual channel sequence counts will then be generated and maintained. Finally, the VCDU will be outputted as a CCSDS return link data stream.

CCSDS SIMULATOR CHIP BLOCK DIAGRAM

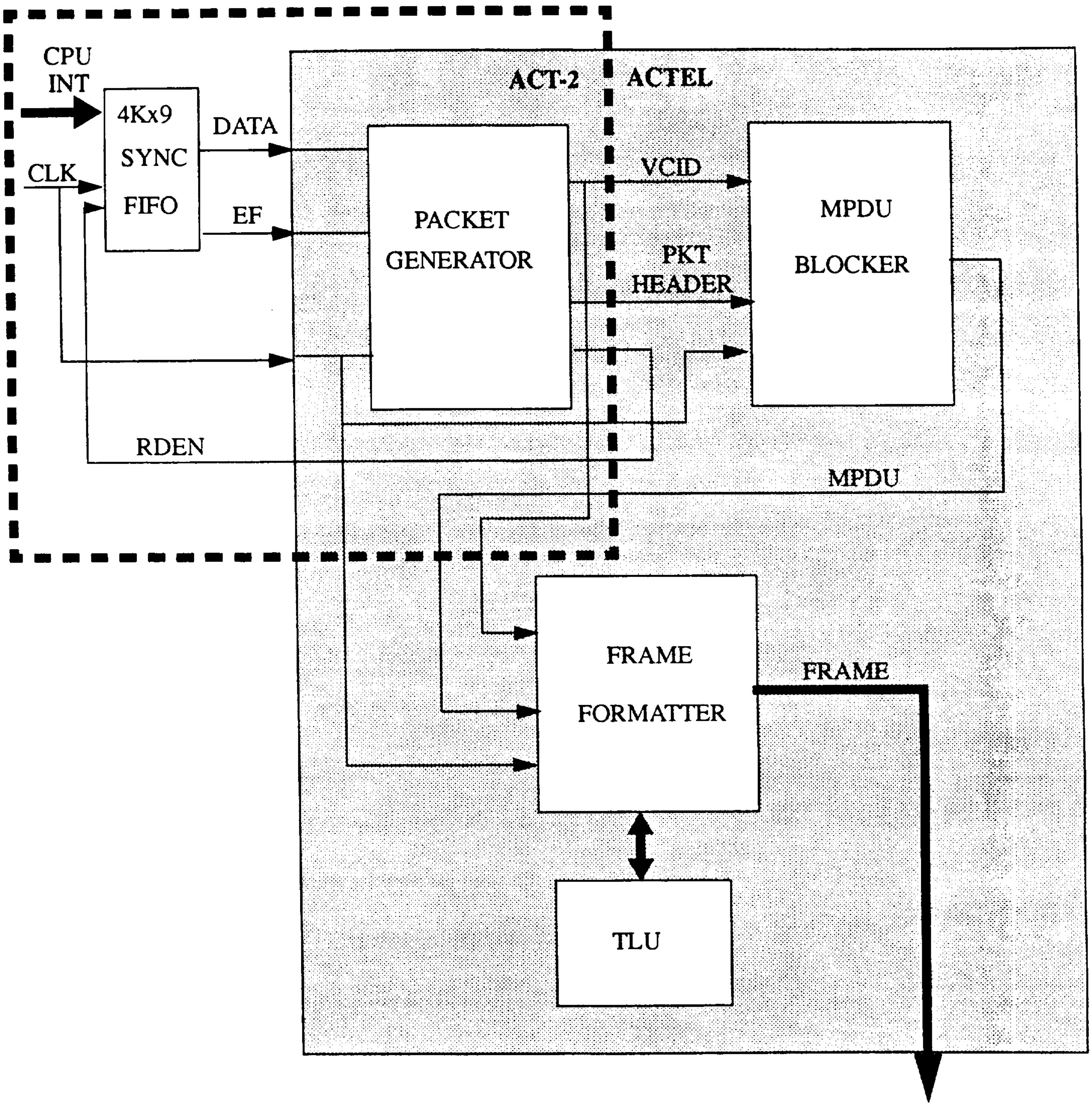


FIGURE 3

Packet Generator and FIFO Interface

The packet generator has been designed to meet specific requirements according to CCSDS standards and FIFO interface control. In order for the CCSDS Simulator Chip to function properly the packet generator must provide, within the chip, the following packet source requirements:

a. Accept data input from the FIFO.

b. Generate the required control signals to manipulate the data received into the proper CCSDS standards packet source format.

c. Provide the REDN signal for the FIFO.

d. Load a 16-bit down counter with 16 bits of data at the same time.

e. Accommodate the data into packet header format.

f. Provide the signals defining the packet header into its components as defined by CCSDS packet source standards.

The packet generator diagram (Figure 4) provides a block overview of the circuit. The packet generator is only part of the Act-2 Actel FPGA (See figure 3). As seen in figure 4 the data is inputted to packet generator via the FIFO and it is bused to the packet counter and data control components. The data is loaded to the packet counter using the load signal provided from the shift register of the FIFO Control.

The shift register provides several other signals including the REDN signal which enables the FIFO to output data into the packet generator. The packet counter accepts data and uses 8-bit register to hold data for one clock cycle and then latches 16-bits of data in the next clock cycle into the 16-bit down counter. Data is also bused to the data control component from the FIFO and from the packet counter. Once the data goes to the control component it is latched at different clock cycles into a 32-bit multiplexer (The MX4x8 component of figure 4). Finally the data is in the of the packet header and VCID form necessary to interface with the next component the MPDU Blocker of the frame simulator as seen in figure 3.

PACKET GENERATOR DIAGRAM

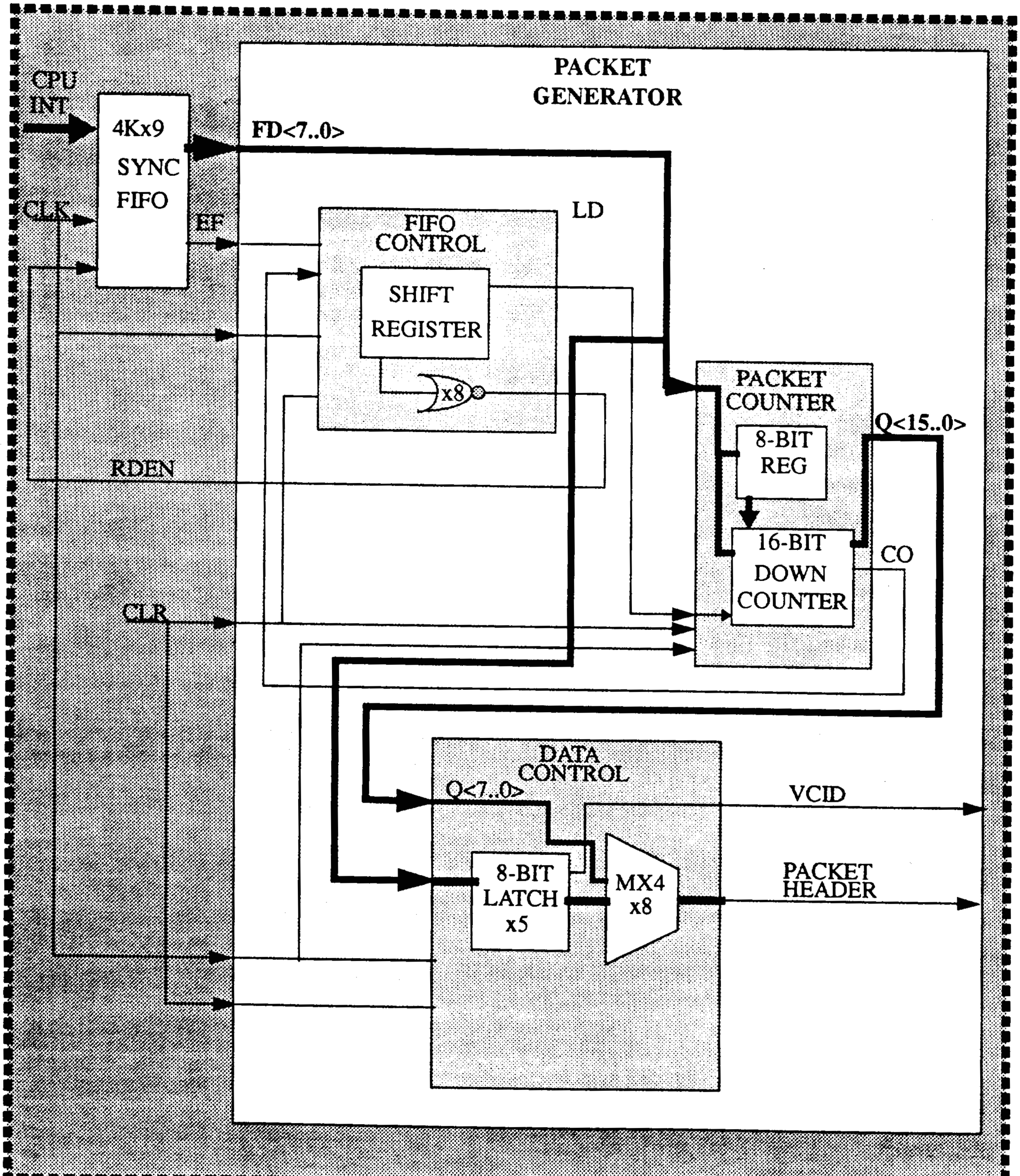


FIGURE 4

Conclusion

During the summer I have been able to initiate the design work for the CCSDS Simulator Chip. I have completed the design specifications for the packet generator part of the chip. The completion of the design specifications included the simulation schematics of the packet generator interface and its timing diagrams.

My summer project has helped me to gain understanding of data processing using digital hardware to route it. The use of the state-of-the-art Cadence Software enabled me to appreciate its functions and its applications which I used for the simulation of the packet generator circuit. Also, I was able to gain design and problem-solving skills using wave patterns as a perspective of what a certain circuit is expected to do when completed.

It would be an understatement to say that my summer experience at Goddard encompassed the understanding of logic design theory and some of its applications through the design simulations which I have completed; my summer experience has been much more than that. Even though understanding logic design theory was one of my main objectives for the summer, my learning only began there.

Through the Office of Public Affairs here at Goddard I was able to attend several colloquiums and social functions that have greatly improved my communication skills as well as professional presentation etiquette. The professionals who introduced the colloquiums and those who presented them offered to me much more than the material they were presenting. Since I was exposed to a professional environment both at the colloquiums and at the work site I was able to appreciate the way professionals handle themselves and their work in the day-in and day-out activities they were responsible for.

My mentor provided the backbone to my learning experience as a whole. Through the various meetings I had with him and his corrections and suggestions to my problems I was able to appreciate learning from a truly dedicated professional on a one-to-one basis. It was exposure to such a professional environment which permitted me to learn not only about theoretical applications and problem solving skills but also about what is expected of a professional as a person and as a post-scholar in a professional environment.

I was asked in a self-evaluation form whether my summer experience at Goddard had changed my mind on the career I currently pursue; it did not change my career goal it did much more than that; my summer experience reinforced my will and charged me with motivation to continue the ladder of knowledge which I have chosen. If I were asked to sum up my summer experience in a phrase I would have to reply, "I have experienced the SIECA effect."

ACRONYMS AND ABBREVIATIONS

Term	Defenition
AOS	Advanced Orbiting Systems
CCSDS	Consultative Committe of Space and Data Systems
EOS	Earth Orbiting System
FIFO	First-in First-out
MPDU	Multiplexer Data Unit
NASA	National Aeronautics and Space Administration
REDN	Read-enable
SIECA	Summer Institute for Engineering and Computer Appl.
VCDU	Virtual Channel Data Unit
VCID	Virtual Channel Identification
VLSI	Very Large Scale Integration

HIGH RESOLUTION

PENETRATION DEPTH

THERMOMETER TESTING

Patsy Polston
Tuskegee University
SIECA-UG

CRYOGENICS, BUILDING 7
GODDARD SPACE FLIGHT CENTER
PETER SHIRRON, MENTOR
MICHAEL DIPIRRO, MENTOR

HIGH RESOLUTION PENETRATION DEPTH THERMOMETER TESTING

Cryogenics is a branch of physics that deals with the production and effects of very low temperatures. Within the area of Cryogenics, many studies are done which involve the use of temperatures at and below 2.0 Kelvin. There are also a great deal of projects that utilize the superfluid helium. With these studies being done more and more frequently and to greater precision, greater emphasis is placed on accurately determining the temperature at which these tests are being conducted. Hence, the development and study of a high resolution Penetration Depth Thermometer (PDT) is underway.

The PDT is based on the temperature dependence of the magnetic penetration in a superconductor. The PDT consists of a thin superconducting film deposited on a substrate with two coils in close proximity acting as a primary and secondary. A current in one coil will produce a magnetic field. The primary coil is connected to a current source which provides a sine wave output at various frequencies and currents. The secondary is connected to a lock-in amplifier with which amplitude and phase are read. The substrate used in this project was sapphire. Sapphire was chosen for two reasons: (1) it has high thermal conductivity which means that it conducts heat well, and (2) it has a low heat capacity which means that it only take a small amount of heat to warm it up, and it is only necessary to extract a small amount of heat to cool it down. The temperature is determined by a calibrated germanium resistance thermometer (GRT). The PDT also has a very simple geometry, and can be easily integrated with an experiment. For example, with our project, it was in intimate contact with the liquid helium. The active element in the sensor is the thin film of superconductor. The superconductor used was aluminum. Thin aluminum films can be easily deposited. It has also been shown that for thinner films of aluminum, higher transition temperatures can be obtained. Our goal for the development of the PDT is to achieve a transition temperature of 2.2 K. Hence, we set out to find the aluminum film thickness that would give us this transition temperature.

In order to effectively assess the abilities of the PDT under various conditions and to determine at what film thickness we would achieve our desired transition temperature, it was necessary for us to move through three main stages - the deposition, the actual running of the test, and the analysis of the data we obtained. In conducting these tests we used two different geometric forms of sapphire - a 5mil x 1" diameter disk and a 1.5" x .5 diameter solenoid (mandrel). Within stage one, it must first be determined how thick of an aluminum film we need to deposit onto the substrate. After this determination the mount had to be cleaned thoroughly. When the sapphire mandrel is being deposited on, it too must be cleaned. It was necessary to rid these pieces of as much dirt, oil and dust as possible in order to obtain a smooth even coating. Some of the cleaning methods that we used included acetone and distilled water, as well as the Ultrasonic Cleaner. The Ultrasonic utilized ultrasonic vibrations to loosen or remove all particle from the object to be cleaned. After cleaning, the sapphire is then connected to its mount.

In the case of the sapphire mandrel, there is a small, metal, cylindrical mount into which the stem of the mandrel would be inserted. On the other hand, with the disk, it was set on a flat metal mount. With the sharpened end of a cotton swab, varnish was placed on the edge of the disk, being sure to avoid the surface facing upward. The surface tension between the disk and the mount sucked the varnish underneath, and once dry, would adhere the disk securely to the mount. The mount and its substrate, either the mandrel or the disk, are then put onto the shaft of the bell jar of the evaporator. Pellets of 99.9999% pure aluminum are then placed in the tungsten boat that rests inside the bell jar. Once the substrate and aluminum are in place, suspended from the shaft, it is necessary to rid the bell jar of any dust particle or dirt by using the Microduster.

The actual deposition of the aluminum is quite brief. The pressure in the bell jar was lowered to approximately 5.0×10^{-7} torr. If the film coating is being applied to the mandrel, the shaft is connected to its motor which is then turned on. This motor allows the shaft to rotate in order to obtain an even coating on all the sides of the mandrel. A large current is then put through the clamps that hold the tungsten boat and aluminum pellets. This current causes the boat to heat and melts the pellets. The crystal monitor begins to record the rate at which the aluminum is evaporating. After the rate has been steadied by adjusting the amount of current, the shutter is then opened allowing the aluminum to evaporate onto the substrate. Once the thickness monitor has reached the desired amount the shutter is closed and the current flow is cut off. The deposition is complete. The bell jar is then vented by opening the air release. The mount and the coated sapphire are then removed. The sapphire is placed in a fiberglass tube that is wrapped with superconducting wire. When the disk is coated and is in its tube, it is secured with a spacer and a pancake coil on both sides. The spacer is to avoid having the disk in direct contact with the coil. This tube is then fastened to the cryostat with tie wraps that are designed to work well in low temperatures. The coils - the two pancakes coils and the outer coil outside the tube - are then connected to their respective leads, as noted in the wiring scheme for the PDT.

The dewar is the device that was designed to insulate the liquid helium from warm temperatures. Any heat that is conducted or radiated into the dewar will boil the liquid helium away. The cryostat holds the thermometer and was designed to fit inside the dewar. The other instruments that will be used to ensure accurate data are the universal source and the lock-in-amplifier. The universal source is used to input the desired currents and frequency The sensitivity can be adjusted by use of the lock-in-amplifier. The purpose of the lock-in-amplifier is to keep unwanted temperature signals from interfering with the data being taken.

The cryostat is placed into the dewar and the testing process begins. Liquid helium is transferred from its tank into the dewar by using a transfer line. This is done by placing one end of the line into the helium tank and the other into the dewar. Once the percentage of helium reaches 100%, meaning the experiment is fully immersed in liquid, the transfer is complete. The valves are then opened to allow the pump down of the helium bath. This process lowers the temperature. The temperature drops from 4.2 Kelvin to 1.0 Kelvin. The purpose of the testing is to measure the mutual inductance between the primary and the secondary coils. The universal source applies a sinusoidal current to the primary at a given frequency. While the temperature is above the transition temperature of the film, the current is changed to the desired value on the universal source. The lock-in-amplifier is checked to make sure the sensitivity corresponds to the current flowing. We make sure the lock-in output will not rise above 10V. If this happens the sensitivity is reduced. While all of this is occurring the computer is storing the data.

While the computer is storing data, change all input values so that the desired currents can be analyzed. This requires a cool-down and warm-up process. The first step, cool-down, is to open the main valve which allows a vacuum pump to lower the pressure above the helium bath, thus lowering its temperature. When the temperature is low enough the main valve is closed and the warm-up process begins. When the voltage reading is steady, this indicates the temperature is above the transition temperature of the film, and the test is complete. Different currents can then be used and the test repeated. Finally, the testing is finished, the data file is crunched, and the data analysis begins.

Over the (10) week period numerous runs were executed. Six different tests using a 35Å, 37Å, 40Å, 45Å, 47Å, and 50Å film thickness were analyzed. With each different film thickness at least four different currents (10 μA , 20 μA, 50 μA, and 1 mA) were tested. The sensitivity and frequencies were changed to correspond with the currents. The graphs that were used were plotted by using the mutual inductance (Y-axis) and the temperature (X-axis). The mutual inductance was calculated from the two coils being used, the primary and the secondary, by this formula:

M=(X-voltage*sensitity*2*sqrt(2))
(2*pi*frequency*current)

Once everything was computed and plotted, the graphs taken were compared to what was expected. The transition temperature was determined as the point where the mutual inductance reached a steady value.

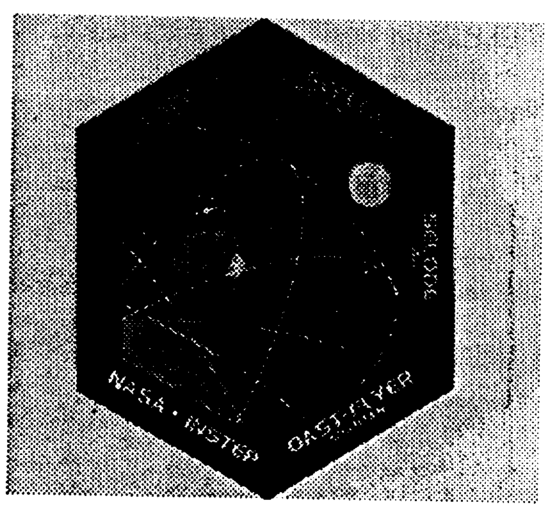
As we progressed with our research and testing of the effects of different film thicknesses, currents, sensitivities, and frequencies, we came upon an unexpected trend. A trend that contradicted all past research and related information. We should have found that the transition temperature increased steadily with decresing film thickness. However, this was not observed. We instead noticed that the film thickness increased, so did the transition temperature. Future work will focus on seeing whether this result is due to the way in which the films were made, or are due to contamination during the deposition process.

This summer internship at NASA/Goddard Space Flight Center has been a very enlightening experience. I had know idea what my work would require, and I was not familar with the word cryogenics. But, over the (10) ten week period I was able to experience things I might not had a chance to if it were not for this program. I learned about cryogenics and the importance of research. I also learned that when you take research a lot of time is spent waiting for certain results. We came across some problems that hendered are testing, but we had to be patient and figure out other methods of solving the same problem. It was a good experience for me to work with other people to reach the same common goal. I know that this experience will help me in completing my studies in Electrical Engineering and Physics at Tuskegee University.

My Experiences with the Spartan 206 Spacecraft Project

By:
Marcellus A. Proctor
SIECA Intern

August 2, 1995



My experiences with the Spartan 206 Spacecraft Project

By: Marcellus Proctor (SIECA)

Code 743-Instrumentation Branch

Mentors :

Mr. Robert Stone

Mr. Tom Gostomski

Ms. Cindi Lewis

1.0 Code 743

Code 743, which is known as the Instrumentation Branch, is located in building 5 and is headed by Mr. Robert W. Stone. The Instrumentation Branch provides engineering design, procurement, fabrication, integration and testing of instrumentation electronic boxes which include data handling and data storage systems, power systems, ground stations, and components for payloads. The Instrumentation Branch also performs the integration and test of the entire spacecraft. This effort is in support of both Shuttle and Expendable Launch Vehicle payloads such as Hitchhiker, Spartan, and Small Explorers. The Instrumentation Branch also provides applied engineering and systems updating to keep the services of the branch aware of state-of-the-art developments.

2.0 OAST-Flyer

The Office of Advanced Science and Technology (OAST)-Flyer, Spartan 206, is currently manifested for the STS-72 Space Shuttle mission to fly in November 1995. OAST-Flyer, the seventh Spartan to launch, is composed of four experiments: REFLEX, GADACS, SELODE, and SPRE. Three of the four experiments are sponsored by the Office of Space Access and Technology

(OSAT). The fourth experiment, SPRE, is a volunteer effort comprised of University of Maryland students, area engineers, and space industry contractors. A picture of the STS-72 Shuttle Mission can be found on page D1.

2.1.0 The Spartan Project

For the past ten weeks, I have been working on the Spartan 206 Spacecraft which is schedule to launch in November 1995. The Spartan Project is designed to provide easy and inexpensive access to Earth's orbit via the Space Shuttle for science experiments that need to make precise measurements in orbit but away from the shuttle. The Spartan spacecraft is a small, rectangular, free-flying vehicle, measuring 1 x 1.25 x 1.5 meters. It is released from the shuttle and picked up after several days, usually 40-45 hours, of conducting its experiments. Spartan missions support stellar, solar, or Earth fine-pointing experiments; experiments requiring microgravity; and experiments requiring space environments away from the Space Shuttle. The Spartan Project provides the hardware, systems management, and all support activities associated with the integration of the Spartan with the Space Shuttle. With the reusable carriers flexibility, an unprecedented six Spartan missions were manifested to launch in a 26 month period beginning in September 1994 and ending in November 1996.

There are four (4) experiments that will be done by the Spartan 206 Spacecraft. They are the Return Flux Experiment (REFLEX), Global Attitude Determination and Control Experiment (GADACS), Solar Exposure to Laser Ordnance Device (SELODE), and the Spartan Packet Radio Experiment (SPRE). A picture of were all the experiments are located on the spacecraft can be found on pages D2-D4.

2.1.1 Return Flux Experiment (REFLEX)

A Spacecraft can be disabled by exposure to the space environment in several ways. One way is when the lenses, sensors, and instruments get coated with tiny particles or dirt. This dirt can cause failure of the mechanical systems on the spacecraft. The main objective of REFLEX is to investigate the Molecular Backscattering or " return flux ", associated with on-orbit spacecraft. This phenomenon occurs when the spacecraft gives off tiny particles of dirt into the atmosphere which then collide with other particles and bounce back to the spacecraft. REFLEX will also study the erosion of the spacecraft surface coatings as a result of particles chemically reacting with the atmosphere.

The REFLEX experiment interacts with the residual atmosphere by blowing inert gas (Argon and Krypton gas) in three different directions with respect to the spacecraft. The REFLEX instruments can determine the amount and type of dirt in the residual atmosphere and will measure the molecules of Argon and Krypton gas that bounce back to the spacecraft due to return flux. REFLEX is supported by Goddard and the University of Minnesota.

2.1.2 Global Positioning System (GPS) Attitude Determination and Control Experiment (GADACS)

A spacecraft turns in a very precise way. In order to turn a spacecraft, you need to know what direction the spacecraft is headed plus how fast it is turning. In the past, spacecraft measured turns using gyroscopes, startrackers, and/or the Earth. For the first time ever, GADACS will use GPS to gather this information in order to control OAST-Flyer's turns. GADACS' objectives are to determine if the space environment will impact the ability to use the GPS to control the spacecraft. The Global positioning System (GPS) is a group of 24 satellites orbiting the Earth that allow anyone to determine where they are, how fast they are moving, in what direction they are moving. GADACS will allow the experimenter to control a spacecraft using GPS.

GADACS will use this way and then compare the data with that of the GPS for two-thirds of the OAST-Flyer mission. For the last third of the mission, GADACS will control the spacecraft's turns and the starting and stopping of the turns solely using GPS data. GADACS is supported by Goddard and Stanford University.

2.1.3 <u>Solar Exposure to Laser Ordnance Device (SELODE)</u>

When parts are to move in space, they are latched until they are require to move. In the past, the release has been done by one-time contained miniature electrical explosive device (EED). These devices are set off with electricity (just as one sets off a large explosives with a detonator in the movies).

SELODE will test a new way of setting off these devices using light, a laser, instead of electricity. Scientists have to be certain that these devices won't set themselves off. SELODE will test whether sunlight or exposure to the space environment will trigger the laser operated devices.

To start the experiment, once OAST-Flyer begins its free-flying mission, a small door to expose the samples will be opened using one of these laser operated devices. The samples will be exposed in different ways to sunlight but all of them will be exposed to space. SELODE is supported by Johnson Space Center.

2.1.4 Spartan Packet Radio Experiment (SPRE)

This experiment is constructed by integrating several specific subsystems. Some of these subsystems were designed and built by students, while others subsystems were purchased as a unit, such as the transmitters.

SPRE will perform a primary Amateur Radio Experiment related to LEO digital communication techniques. In addition to Amateur Radio Experiments, the telemetry subsystem will forward to Earth a sampling of real-time telemetry for two of OAST-Flyer's experiments: REFLEX and GADACS. The telemetry and command subsystem will have a limited and protected ground command capability with REFLEX. The telemetry and command subsystem will also gather several internal status words such as temperature, main power voltage, and current. This data will be stored onboard, and used later in generating a subsystem profile.

The results generated from SPRE will be used later to study the behavior of electronic components in the space environment. This study will result in a better understanding of the effect of space on electronics and will eventually lead to more efficient designs with improved performance. The knowledge collected from both the experimental hand-off and the status data will broaden the methods of transmission available to amateur radio. SPRE is supported by the University of Maryland College Park and several students from DuVal High School.

3.0 My involvement with the Spartan 206 Spacecraft

I started work on the Spartan 206 Spacecraft Summer 1994. During that time I built the harness for the Spartan 206 Spacecraft. A harness is a group of wires and connectors which allows the spacecraft to talk to the different onboard systems. A harness can be compared to the human nervous system. When you move your arms or legs your brain send electrical impulses to your arm and leg muscles, telling them to contract or expand. The way the electrical impulses get from your brain to your muscles is through your nervous system. The nervous system acts like the bridge or communication system between your brain and all the vital functions of your body. The harness does the same thing as the nervous system. It allows the electrical impulses of one system to interact or communicates to another system. An example illustration of a harness is in Figure (1).

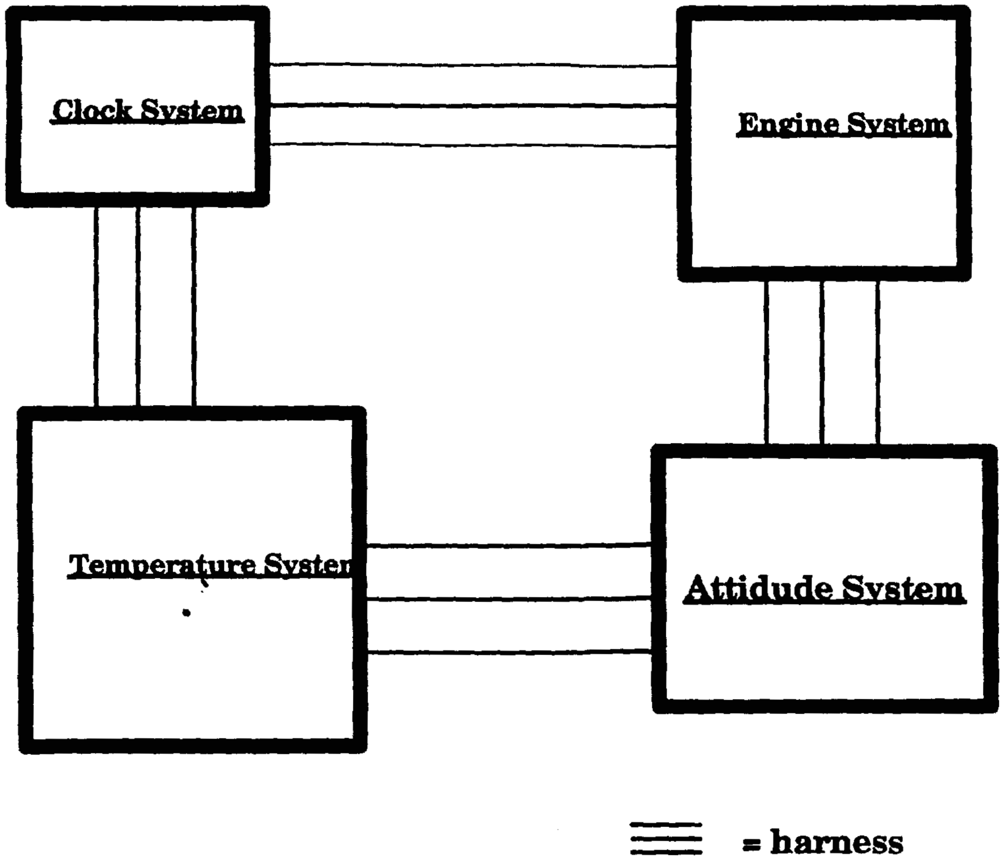


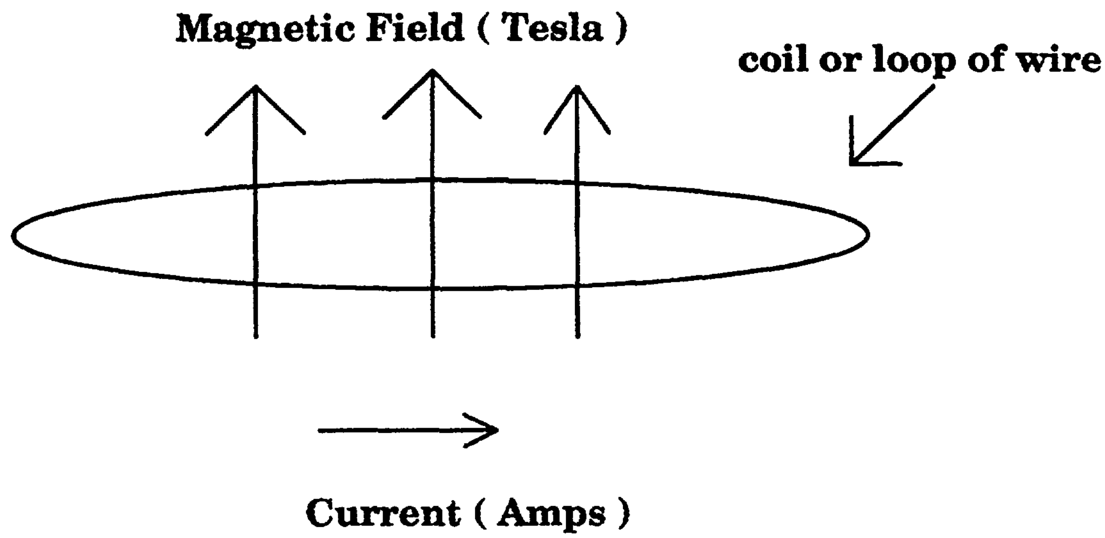
Figure (1) An example of a harness

This summer I helped out with testing of the Spartan 206 onboard systems. For the first two weeks of my internship, I spent most of my time in the Building 7 cleanroom. There we put the Spartan Spacecraft in the cleanroom and did test on its onboard systems dealing with the experiments and maintenance controls. We used the Ground System Equipment (GSE) to test if each experiment and housekeeping system was running properly. The GSE is a console which gives commands to activate the different systems of a spacecraft. I was able to enter the cleanroom five (5) times to connect the cables from the GSE to the spacecraft.

After the two weeks in the cleanroom we moved the spacecraft to a big metal chamber where we did Mass Properties of the Spartan Spacecraft. The mass properties test is a test to see how heavy the spacecraft is. The test also tells the engineers and builders if they gained our lost mass during the course of its construction. When we did the mass properties test for the Spartan 206 Spacecraft we found that we were lighter than expected. The engineers discovered that they did not have the flight hardware integrated in the spacecraft. Also, the engineers discovered that the weights of different parts on the spacecraft came out to be heavier or lighter and predicted.

After the mass properties test, we took the spacecraft to the magnetic calibration facility. There we magnetically calibrated the torque rods on the spacecraft. In the event the spacecraft should go out of control and the Attitude Control System (ACS) is nonfunctional, the spacecraft will use it's torque rods to magnetically align the spacecraft to the Earth's magnetic field.

The facility and the magnetic calibrator was designed to not interact with the Earth's magnetic field. The inside the building is made of wood because concrete blocks have metal in them and produces a magnetic field. Also the magnetic calibrator is raised above the ground to avoid the Earth's magnetic field. The spacecraft is placed in the middle of the magnetic calibration chamber and a magnetic field is produced in the middle of the chamber. In figure below, show the physics behind the production of this magnetic field. When a current is passed through a coil or loop of wire, the inside area of the coil produces a magnetic field.



The mass properties and magnetic calibration testing lasted four weeks.

3.1 My Project for the Spartan 206 Spacecraft

For the last three weeks of my internship, I was responsible for building a switch box which will simulate the Release Engage Mechanism (REM) and the Remote Manipulating System (RMS) switches onboard the Spartan 206 Spacecraft. When the spacecraft is turned off, the REM switch is open or in the BERTHED position. To activate the spacecraft's onboard system you must open the RMS switch, RIGID position, close the REM switch, DEBERTHED position, and then close the RMS switch, DERIGID position. To deactivate the spacecraft you open the REM switch, BERTHED position.

4.0 Acknowledgment

I would like to take this opportunity to thank the following people for their support and encouragement and made my ten weeks at Goddard an enjoyable and exciting experience :

Mr. Bob Stone

Ms. Cindi Lewis

Mr. Tom Gostomski

Mr. Don Carson

Mr. Dan Kreiger

Dr. Joan Langdon

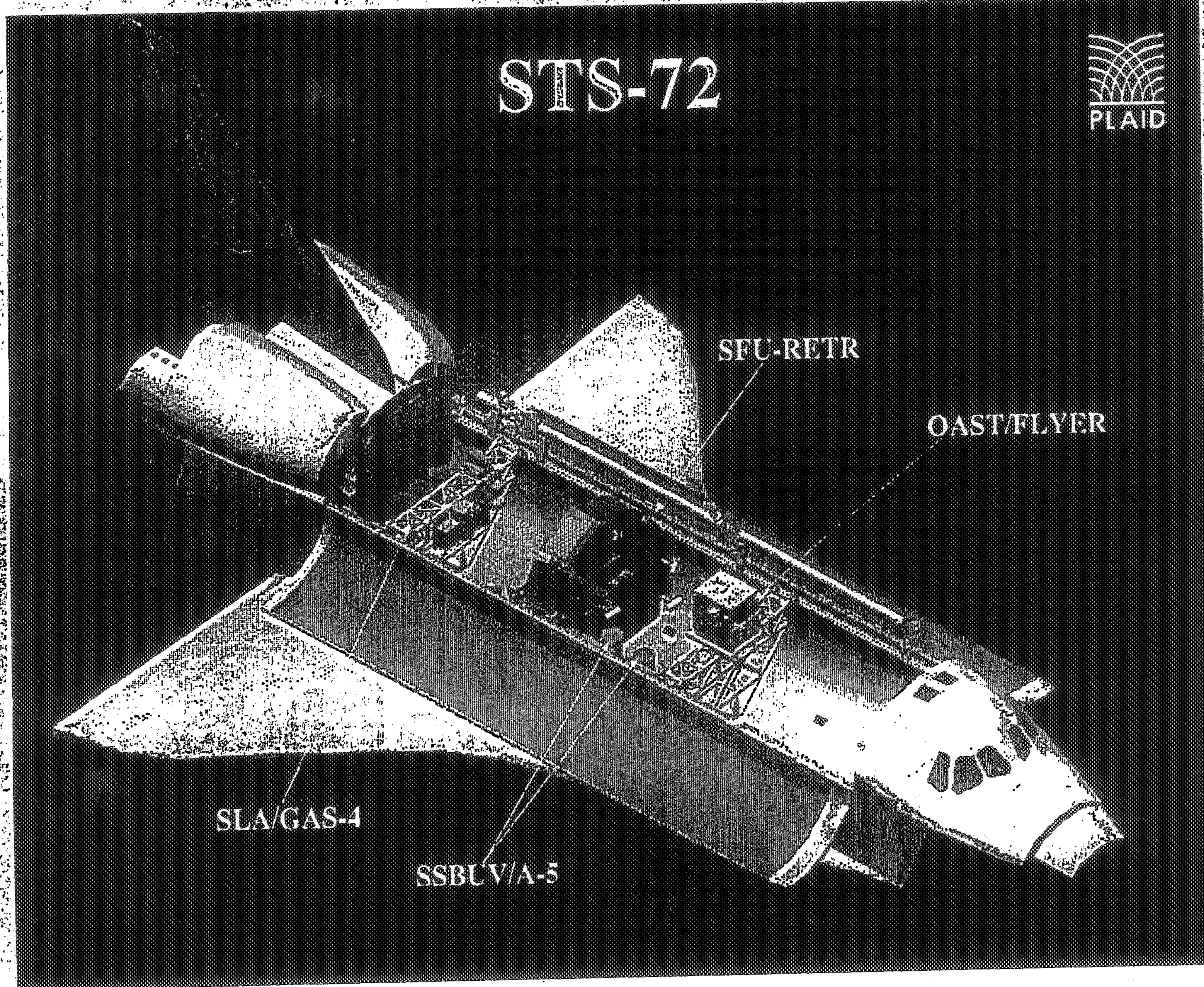
Ms. Mary Lampe

Mr. Nnaemeka Nwosu

Mr. Aaron Rogers

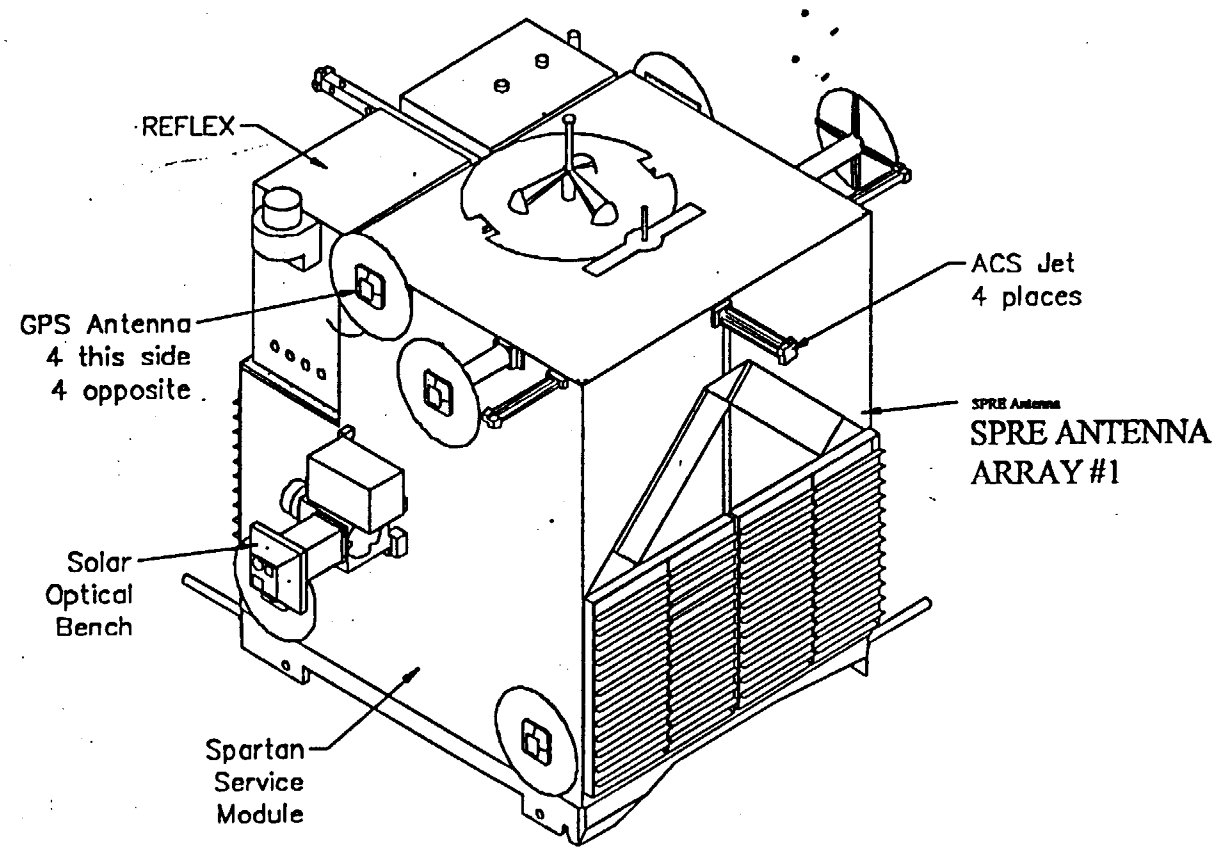
Mr. Vondell Coleman

Mr. Ifeanyi Ezeh



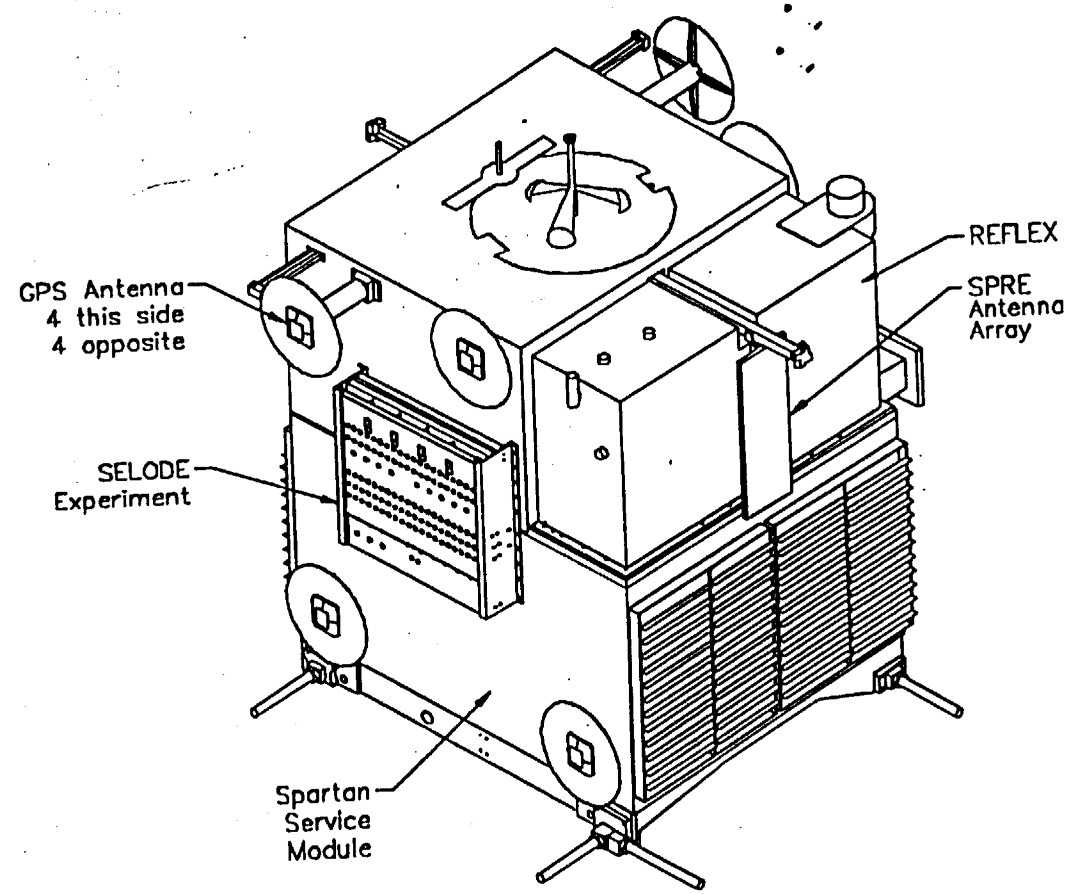
D1

SPRE Antenna Array #1



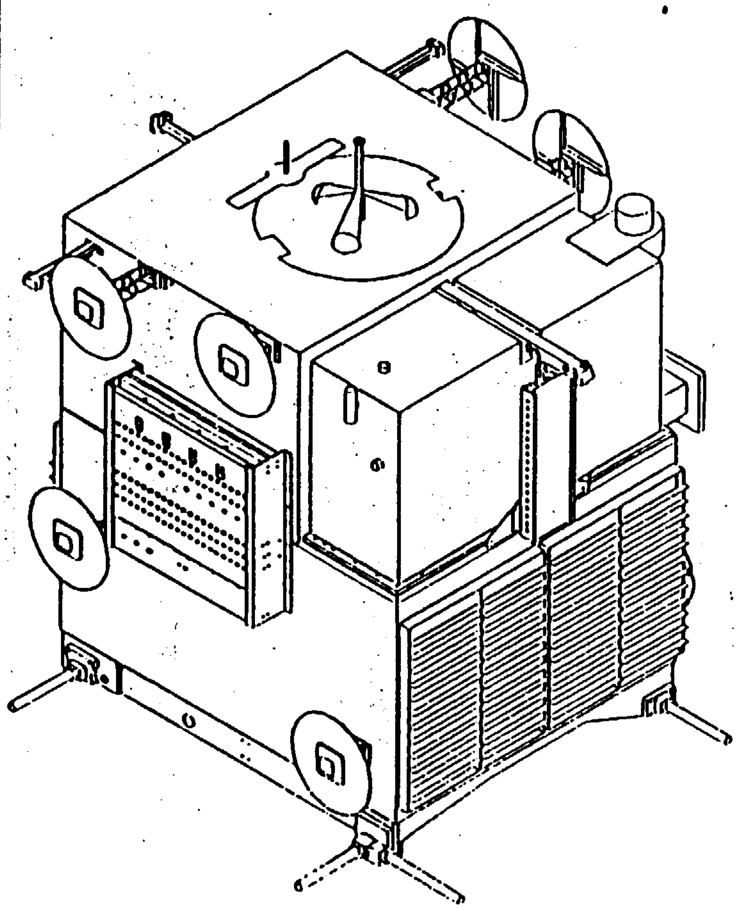
D2

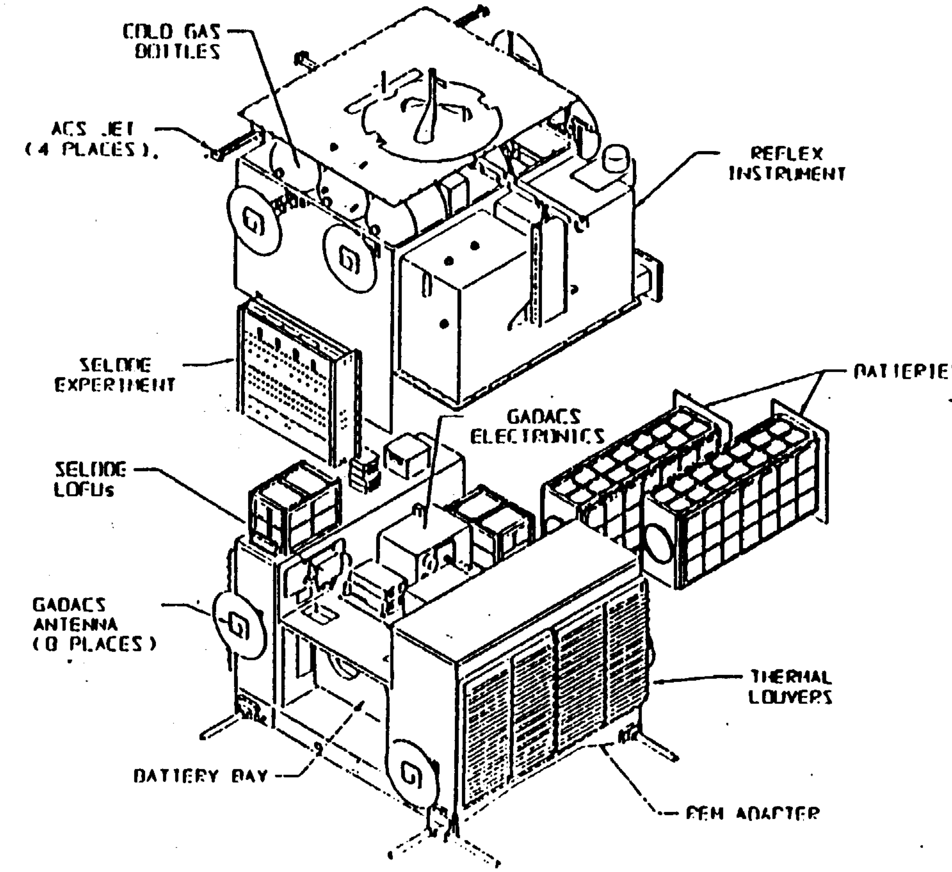
SPRE Antenna Array #2



D3

OAST-Flyer Spacecraft





11127.051895

D4

My Summer Contribution To Goddard Space Flight Center

by
Demetrius Shaffer
SIECA-UG Intern

August 1, 1995

Computer networking plays an important role in today's rapidly growing technological advances. it is critical in the transferring, accessing, and receiving of information via different computers across the world. The Goddard Space Flight Center(GSFC) Center Network Environment(CNE) is the center wide computer network. The CNE interconnects systems ranging from PCs and Macintoshes to workstations to supercomputers. Because of the massive number of connected computers , plenty of information is stored in numerous different databases to keep track of each computer, which is the only source for troubleshooting any problems that go on in the network. My job this summer was to update the data stored in the CNE database(CNEDB), the official GSFC database of all computer in the network. Another similar project I had for the summer was to cross-check the data between three of the databases used by the department. These clean-up jobs were essential to maintain the network more efficiently.

Another project I had for the summer dealt with the CNE web documents, or pages, listed on the Internet. The CNE already had different sites that could be viewed on the Internet. My job was to create another site on the Internet which would give a listing of the top features of the CNE and link any user to those documents. This project consisted of learning the language used to write web documents, hypertext markup language, better known as HTML. In my paper, I will discuss more in detail of the function of the CNE, each of my projects at GSFC, and explain how each of my projects related to the role of the CNE.

The CN what? The CNE who? Just another acronym in the scroll of acronyms at GSFC(another handy acronym) was what I first thought of the department known as the CNE when I walked through the doors of building 28. Could it be the Creatures of Neptune on Earth? Hope not. What it is, I soon came to find out, was a department that Goddard could not survive without--and I do mean that literally. Fortunately for me, I got a chance to participate in the action of the CNE. I had three projects assigned to me there for the summer. Before, I explain them, let me answer the above question--what is the CNE?

The Center Network Environment, Code 933, is the center-funded, center-wide computer network that is open to civil servants and contractors. Access is also provided to off-center networks. Visualize the human body for a second. There is a backbone and then ribs connected at different parts of the backbone. Together, they make one big structure that holds the body together. That's just what the CNE is. It is comprised of inter-building backbones that connect to almost every building at Goddard. Attached to these backbones are different ribs, and on each of these ribs are the systems(nodes) that are registered in the network. Together, all of the nodes make up the GSFC CNE.

OK, so now there's a network. What's next? What is it that can be accomplished with a network? Well, access to any member of the network can be done through another member of the network. If someone needed to access another machine or ,maybe, there are files located somewhere else that need to be viewed, it can be done through networking. Any organization that connects to the CNE use it for NASA-related (but not mission-critical) research, development, administrative, engineering, or support work. Other services that are offered to registered nodes are: electronic messaging (or Email), information archives such as file transfer protocol(FTP) and World Wide Web access, electronic bulletin boards, local access from sites in the metro area, and wide area Internet providers.

All types of systems are interconnected in the CNE on and off site. There are Macintoshes, PCs, workstations, and supercomputers all connected in the network.

Currently, over 12,500 nodes are registered in approximately 50 buildings on and off site. Other NASA centers, other Federal agencies, universities, and commercial enterprises around the world are connected in the GSFC CNE. The CNE Project (the group that operates the CNE) hopes to implement more widespread desktop connectivity in the future.

Two of my summer projects involved working with several databases that the CNE has. Before discussing these two projects, I would like to introduce these databases to give a better picture of what the projects entailed. There were three different databases that I worked with: the Sybase database, the Ghost database, and the Nameserver database. Sybase is the official GSFC database of all the nodes in the network. It is also called the CNE database, or simply, CNEDB. Ghost is the database that was used before Sybase. It is used mainly as a backup reference to check any changes in the CNEDB. Finally the Nameserver is the official name database for all the IP addresses for each node in the network. An IP (Internet Protocol) address is the specific spot on a rib of a backbone that a node is assigned to.

For my first project, cross-checking, I utilized all three of the above mentioned databases. Cross-checking is checking data about a certain node in different databases and making sure that the data all coincide with each other. That is what I did with Sybase, Ghost, and Nameserver. Given a list of node names and their respective IP addresses, I checked a node's data in all three databases to see if they all matched each other. If all the information checked to be the same, then, fine, there was no problem. However, as was the case with almost all of the nodes, there was a problem--ranging from simple to complex. A simple problem would be a node moving to a different location or a node changing its name to something else. Problems such as these I could manually update myself. Other problems weren't as simple, though. Such a problem was a node that was listed in the Nameserver, but was not listed in Sybase and/or Ghost. In such a case, I would have to contact the Rib Manager for that node, who is the person responsible for all the nodes on a particular rib, and ask that person what happened to that node. This, my first project assigned project, was completed in approximately two weeks.

My second project that involved database work was the Sybase clean-up. In this project I had to update the person info data that was in this database. I checked for Email addresses that were no longer in use, incorrectly spelled Email addresses, incorrectly spelled names, duplicated names, and other obvious errors in the data. Once I found these types of errors, I had to correct them using the data that is in the X500 directory. X500 is a current "yellow pages" of Goddard. Any person registered in X500 has their vital information listed in the directory--name, code, building, room number, Email address, and phone number. Therefore, when I saw an error in the data, I checked for that person's listing in X500. If he/she was listed, I corrected the data in the database. If he/she was not listed, there was not much that could be done about it. Most people, though, is registered in X500. This was the last project that was assigned to me and I worked on it until my last day at Goddard. I completed the first half of the nodes listed(there were 5000 total) which was well beyond the expected goal of the project. The goal, when it was assigned to me, was to just "get it started a little".

My final project involved working with the Internet. The Internet, or World Wide Web("Web" for short), is a giant world wide network of information. Any one across the entire planet can access **tons** of information provided by people around the world. A person can go from a listing of Prince concert dates to the weather in Chicago to the recent issue of VIBE magazine. "Everything is on the Internet" may not be a completely true statement, but, in due time, that statement will definitely hold true My project was to created a Web pages for the CNE. The CNE already has listed pages on the Internet, but I was to design and implement a new page that listed the top 10 features of the CNE. I had to learn the language, though. That language is HTML--Hypertext Markup Language.

Hypertext Markup Language is the language used to create documents to be viewed on the Web. It is a collection of styles, indicated by markup tags, that define the various components of a World Wide Web document. Markup tags tell the Web browser how to display the text. A sample of a basic HTML program would be:

```
<title>This is a sample program</title>
<h2>that shows some basic features</h2>
<h3>of an HTML document</h3>
```

All of the words enclosed in <....> are the markup tags that tells the browser how the text is to be displayed. Of course, the program I designed was a bit more complex than this sample, however, the main basic features were still there. After I learned the language, though, creating the page took very little time. The completed Web page can be seen through the CNE homepage on the Web.

My projects played a significant role in the CNE, especially the two database projects. Because of the massive number of nodes registered in the network, a lot of information must be kept about each individual node--such as who is responsible for that node, where it is located in the network, what type of machine it is, and so on. All of this information is stored in different databases, Sybase, Ghost, and Nameserver being three of the major databases used in the CNE. All of this information must be kept up to date. If anything were to ever happen to one of the nodes is the network, network troubleshooting would be virtually impossible without the proper, correct, up-to -date information. My database projects served critical in keeping this aspect of the CNE functioning in a decent manner. They helped to speed up pin pointing any troubleshooting problems in the network. That not only puts a smile on the faces of people who encounter problems, but, also, on the faces of the CNE project. As for my

Web page project, that as well helps the CNE. It gives yet another opportunity for people world-wide to find out more about the CNE.

In conclusion, my summer here at Goddard provided the opportunity for me to have an impact, even as slight as it may be, to the manner by which things are handled in the CNE. My three projects, cross-checking, cleaning-up Sybase, and creating a Web document all had a goal behind them. Each of these goals were achieved here at the CNE. Although the CNE project are "behind the scenes" characters and is easily overlooked, the CNE project is, and will continue to be, an extremely critical part of Goddard.

SPACE PLASMA DETECTOR PROGRAM

Name : Danielle M. Whipp
Program : SIECA
Mentor : Dr. Mona Kessel
Code : 632
Location : Bldg. 26, Room G1
Date : August 2, 1995

SPACE PLASMA DETECTOR PROGRAM

This summer I was assigned to work for Mona Kessel in the Space Physics Data Facility(SPDF), code 632. The Space Physics Data Facility's primary intent is to lead in the definition, development, operation and promotion of collaborative efforts in the collection and utilization of space physics data and models. The SPDF develops and operates a range of programs which serve the data needs of NASA and international space physics science communities. Some of the most interesting programs that SPDF is involved in are the International Solar-Terrestrial Physics (ISTP) program and the Inter-Agency Consultative Group (IACG). My project for the summer involved taking a FORTRAN space plasma simulation program, which originally contained one plasma detector, and add in Hawkeye LEPEDEA (Low Energy Proton Electron Differential Electrostatic Analyzer) to the simulation. My project objectives were a FORTRAN refresher, to understand the existing program, enhance the code, add Hawkeye LEPEDEA, and then run data tests.

My first objective was to refresh my FORTRAN skills. The reason for this was that a period of time had passed since I last used FORTRAN; therefore before doing anything I had to get re familiarized with the syntax. The way I went about this was by pulling out my old FORTRAN book, by going over the program, and by asking Mona a few questions.

After refreshing my skills, I moved onto my next objective, I began to tackle the code. The first step was to see what the old code did. The original program was a computer simulation of measurements by a particle detector. The program accepted plasma distribution input parameters such as density, velocity, and temperature. It also accepted detector characteristic input parameters such as area and geometric factor. After all of the parameters were inputted it then started it's calculations. It calculated simulated observations and then it analyzed the data. After it finished calculating, it then compared the observed (calculated) parameters to the input parameters. The reason for this comparison was to determine the detector's sensitivity. To determine the detector's sensitivity, you would just see how close the observed and input parameters are to each other.

Next, Mona stated that she would like to update and shorten the old code. This now started me working on my third objective, to enhance the code. The two main ways that I updated was to change the common blocks into structures and records, and also add in error checking (see Appendix A). Part of shortening the code was to use the record names in the subroutine calls instead of the common blocks, another way was by deleting choices from the main menu (see Appendix B). I was able to delete choices from the main menu because the original code allowed the user to input values for program options, data binning, and observation parameters; however, Mona decided that these variables could have set values. This allowed me to put these set variables into subroutines, and then take them out of the main menu. I also shortened the code by deleting unnecessary code; however the main part was taking large parts of code and turning them into separate subroutines. This allowed me to exchange the code with a simple subroutine call.

After enhancing the code I then started working on the implementation of LEPEDEA, my forth objective. The first area that needed to be covered was background research. The background research I did was to read an old AMPTE detector article, a LEPEDEA detector article, and an IDL program which included calculations for the LEPEDEA detector. The reason for reading the AMPTE article was to get a better understanding of how plasma detectors work. The purpose of reading the LEPEDEA article was to find out how it worked in contrast to the AMPTE detector. While reading, I found out that LEPEDEA is a twenty year old detector which is archived at the National Space Science Data Center (NSSDC, Code 633). The NSSDC is a multidiscipline archive, presently supporting astrophysics, solar and space plasma physics, lunar and planetary, and Earth science data. The NSSDC acquires data from spaceflight projects, data systems, and individual principal investigators. I also discovered while reading that LEPEDEA is designed for measurements of the energy spectrums and angular distributions of proton and electron intensities. Some of the major objectives for the plasma instrument are: measurements of the angular distributions and energy spectrums of magnetosheath plasmas within the polar cusps, provide detailed plasma observations, and search for the outward flow of ionsopheric ions along field lines threading the earth's polar caps. After the background research was complete, Mona and I started to create new LEPEDEA subroutines for the program. The four new subroutines are: LEPEDEA_bin, LEPEDEA_detect, LEPEDEA_observ and LEPEDEA_extrct. The main job of LEPEDEA_bin is to assign the variables minimum energy, maximum energy, theta angles, and phi angles values. The purpose of LEPEDEA_detect is to assign the geometric factor its values. LEPEDEA_observ's main duty is to take the plasma distributions (temperature, velocity, density) and determine the number of counts. Finally, LEPEDEA_extrct's purpose is to determine the bulk parameter (temperature, velocity, density) values.

After completing the implementation of LEPEDEA, I began my fifth objective which was to run data tests. I started by inputting values for AMPTE ftr into the program. The next step was to take the results from the program and compare them with the inputted data. The last step was to take the bulk parameter's (temperature, velocity, density) data and develop a percent error chart (see Appendix C). Unfortunately, I was unable to run data tests for the LEPEDEA detector because there are still a few bugs in the program. Mona stated to me that some time in the near future she will fix LEPEDEA's errors.

Finally, I would like to summarize my summer here at NASA. First, I would like to comment on what the program now enhances. It now enhances AMPTE ftr and LEPEDEA. In addition to my project, I was asked to create a home page for the world wide web, the URL is: http://www.gsfc.nasa.gov/students/Whipp/danielle.html. In conclusion, I believe this experience has given me a broader outlook on the computer science profession. Before receiving this internship with SIECA, I had only been exposed to computer science in school. I never really knew there was such a big difference between school and the work force. I now know the difference and am a better person for it. NASA has also shown me what is important in the computer science field and has helped me to decide what classes I need to concentrate on in my last year of school. I would like to give special thanks to the SIECA program, my mentor Dr. Mona Kessel, and NASA, GSFC.

APPENDIX A

Updating

Example 1:

```
COMMON/DET/idetect,AREA,FOV,EBAND,EFF(64),H,GTHETA(180)


structure/DETECTOR_structure/
    integer    idetect
    real*4     AREA,FOV,EBAND,EFF(64),H,GTHETA(180)
end  structure
record/DETECTOR_structure/DETECTOR
```

Example 2:

```
10  write(6,*) 'SPECIFY DETECTOR      1 - AMPTE'
    write(6,*) '                      2 - LEPEDEA'
    read (5,*) DETECTOR.idetect
    DO WHILE ((DETECTOR.idetect.NE.1).AND.(DETECTOR.idetect.NE.2))
         write(6,*) 'Selection of detector was incorrect, try again.'
         GO TO 10
    ENDDO
```

APPENDIX B

Shortening

Example 1:

```
        CALL AMPTE_bin(EMIN,EMAX,LEVELS,ITYPE,THST,THEND,
1THDIFF,PHIST,PHIEND,PHIDIF,mode)
```

```
        CALL  AMPTE_bin(DATBIN,mode)
```

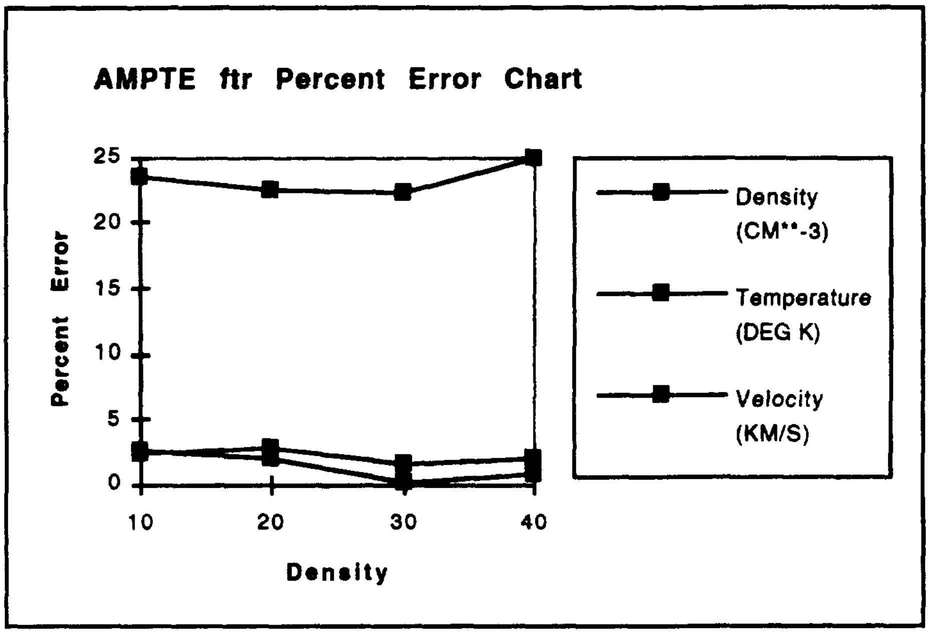
Example 2:

```
write(6,*) 'ANY DATA CHANGES REQUIRED?'
write(6,*) 'SPECIFY CHANGES          1 - PROG OPTIONS'
write(6,*) '                         2 - SOURCE DESCRIPTION'
write(6,*) '                         3 - DETECTOR CHOICES'
write(6,*) '                         4 - DATA BINNING'
write(6,*) '                         5 - OBSERVATION PARAMETERS'
write(6,*) '                         6 - BACKGROUND LEVEL'
write(6,*) '                         7 - RUN PROGRAM'
read (5,*) ITEST
```

```
write(6,*) 'ANY DATA CHANGES REQUIRED?'
write(6,*) 'SPECIFY CHANGES          1 - SOURCE DESCRIPTION'
write(6,*) '                         2 - DETECTOR CHOICES'
write(6,*) '                         3 - BACKGROUND LEVEL'
write(6,*) '                         4 - RUN PROGRAM'
read (5,*) ITEST
```

APPENDIX C

AMPTE ftr (magnetosphere)



NATIONAL ARERONAUTICS AND SPACE ADMINISTRATION
SUMMER INSTITUTE IN ENGINEERING
AND COMPUTER APPLICATIONS
GODDARD SPACE FLIGHT CENTER

HUBBLE SPACE TELESCOPE (HST)

VEHICLE ELECTRICAL SYSTEM TEST (VEST)

CODE #442

ARTURO YANEZ NAVARRETE
SIECA-UG STUDENT

August 2, 1995

TABLE OF CONTENT

INTRODUCTION

The Hubble Space Telescope (HST) Vehicle Electrical System Test (VEST) Facility provides test support and maintenance of the HST during its long term mission. The VEST and VOCC facilities located in building #29 at Goddard Space Flight Center (GSFC) are under contractor of Jackson and Tull Aerospace Division, Code 442. Here, the VEST Operators and Test Engineers are preparing these facilities for the Second Servicing Mission to begin on September 1995.

Through the Summer of 1995 I have been envolved on the preparatives for the next mission. Technical experince as well as engineer operation were adquired through a daily basis. I understand how the VEST and VOCC facilities fuctioned in reference to the HST on-orbit and the responsabilities handled in the Clean Room. Assistance to the VEST operators Technicians to set-up test equipment for the Flight Spare DF-224 and the Rate Gyro Assembly (RGA's) Test, working shoulder to shoulder with the Science Instrument Team in identifing floor space for the new science equipment and the upgrading of the UPS system were some projects involved during this summer.

OVERVIEW OF THE VEST FACILITY

The Hubble Space Telescope (HST) Vehicle Electrical System Test (VEST) Facility provides integration and test support for the on-orbit upgrades and maintenance of the HST during its long term scientific mission. The VEST Facility is located in Building 29 at Goddard Space Fight Center (GSFC). The VEST Facility is made up of a combination of VEST hardware and software subsystems that are divided into three major areas: VEST Structure, VEST Operations Control Center and Science Data Support System.

VEST Structure:

The VEST Structure is located in the High Bay Clean Room and is a high-fidelity electrical harness that is a replica of the HST flight harness. It contains an integrated set of the HST "black boxes" of flight spares or simulators that controls differents areas of the Hubble Space Telescope.

VEST Operations Control Center:

The VEST Operations Control Center (VOCC) regulates the daily operational activities of the VEST system. It is located at building 29 room 100. The VOCC primary function is to command and control the VEST test articles. It sends commands to the VEST Structure and monitors the telemetry data output. The VOCC records all the commands sent and telemetry received form the VEST Structure in wide band tapes, history tapes, line printer and disks.

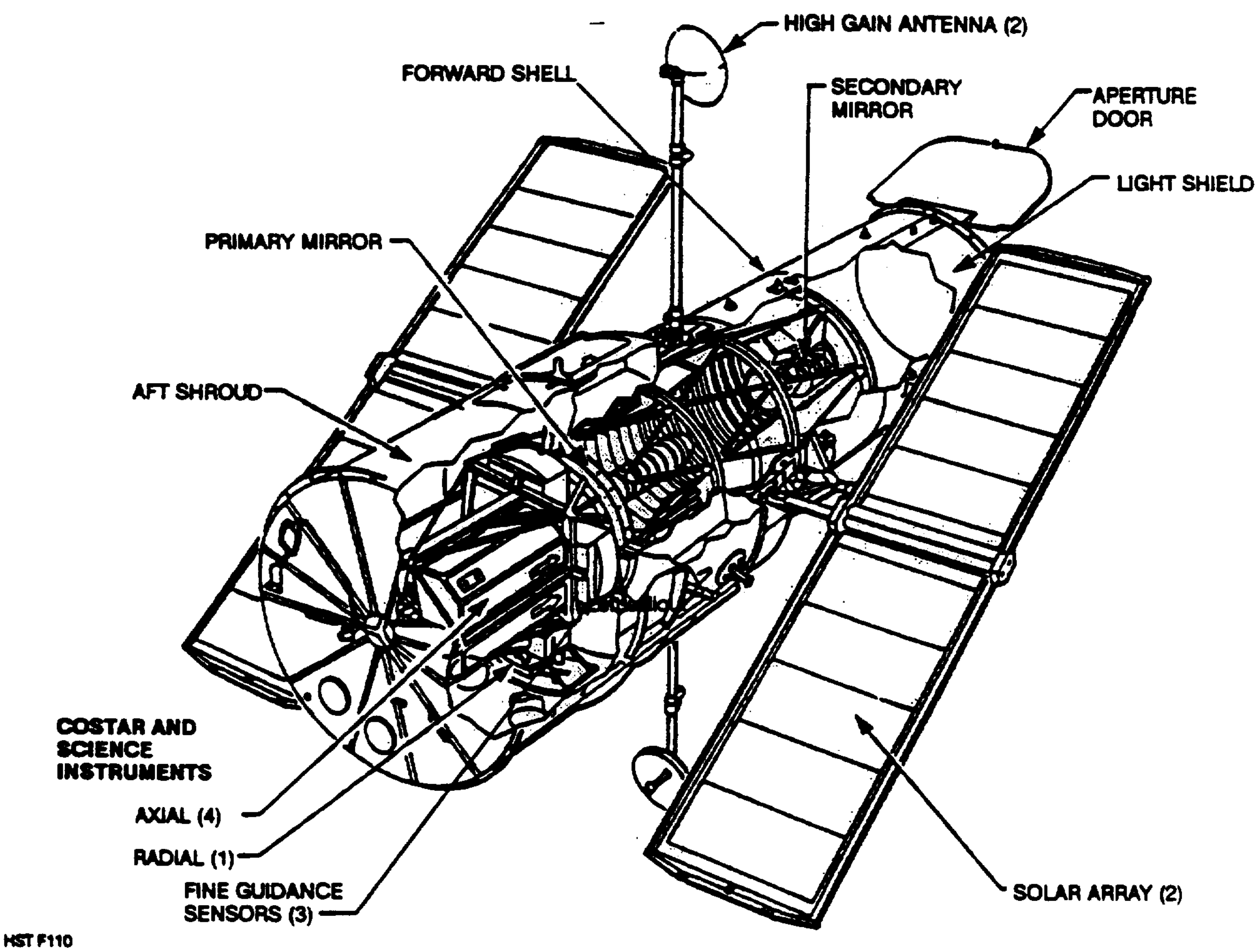


Fig. 1-3 Overall Hubble Space Telescope configuration

VEST Facilities Objectives:

The VEST Facility is the major plataform used to test system software and HST Orbital Replacement Units in an electrical environment. The VEST Facility objectives are:

- To permit ease of mechanical assembly, fault isolation and repair of the HST/VEST components
- Support building or modifying HST/VEST hardware
- Test and verify interfaces and functions of HST flight equipment intended for on orbit installations
- To test and verify HST on-board/ground support software
- Test and evaluate the compatibility of new or revised HST flight software without disturbance of ongoing orbital operations.
- To troubleshoot HST on-orbit anomalies and problems and assist in fault diagnostics
- To stablish safe methods of operating and testing HST/VEST hardware and/or software by developing safe operating procedures, test procedures and commands.

Second Servicing Mission:

The VEST operators and Integration & Test Engineers are preparing the facilities for the Second Service Mission programmed to begin on October 1995. They are going to test scientific instruments as the STSI and NICMOS, the Flight Rate Gyro Assembly, the Solid State Tape Recorder and the Telemetry and Command (TTAC).

The RGA's control the three gyroscopes the Hubble has on board. This gyros control and stabilize the structure on-orbit. RGA's monitors the revolutions and the temperatures of the gyros for continous optimization.

The Solid States Tape Recorders are going to be replaced in the on-orbit HST and they are going to be tested in the VEST facilities before being installed.

The TTAC support the science instruments, NICMOS and STIS, sending commands in the form of telemetry. The new TTAC is able to run 32K of telemetry output as well as the actual system of 4K.

The on-orbit service mission is scheduled to be on February 1997.

CONTRIBUTION TO THE PROJECT

I have been involved in diferent fields at the VEST facility. Technical, design and engineer operations background were adquired during the summer. The responsabilities and safety of the test as well as the rules and regulations to be handled in the Clean Room were adquired.

Floor Plan Room 100

Due to the second servicing mission new equipment is needed in the facilities for the testing and/or to be tested. It was necessary to update and verify the actual floor plan of Room 100. Displaying equipment not shown and verifyng the each measurements of the equipment. The floor plan of room 100 was presented on June 12, 1995 (See Apendix B).

Identifying floor space for the new equipment was the basic in the project. I worked together with the Science Instrument Team to identify floor space for the new equipment arriving: TTAC, STIS console and workstation, and NICMOS console and its workstation. The specifications of each equipment as how long were the cables, where were connected and the UPS system were considered for the design as well as the office environment was observed. Five floor plans showed different configurations of the equipment in room 100. The Floor Plan 4 was accepted and it is in progress (See Apendix B).

Uninterruptable Power Source (UPS) System Overview

The VEST facility provides a 60Hz power backup in the form of Uninterruptable Power Source(UPS). The UPS system is used primarily to prevent electrical damage to VEST Structure. The UPS system allows time to power off the computers when an electrical emergency occurs.

The UPS system is a most to operates the VOCC in a safe mode. Every software equipment need to be connected to the UPS in order to prevent damages to the structure. I updated and verify the equipment connected to each UPS circuit. Some anomalities were discovered and corrected. I presented the actual specifications of the UPS system (See Apendix B). The UPS circuit outlet were traced and found its location in the Floor Plan.

Flight Spare DF-224 Installation

The Flight Spare DF-224 is the main computer of the HST and it is going to be installed in the next servicing mission. The DF-224 were installed on the VEST Structure on June 1995. I assisted the VEST Operators Technicians to set-up the test equipment for the installation of DF-224. I assited in the Electrical Interface Continuity Isolation test (EICIT), Interface Verification Test (IVT) and System Functional Test (SFT). The Flight Spare DF-224 were successfully installed on June 7, 1995.

Rate Gyro Assembly (RGA's) Test

The Rate Gyro Assembly controls and monitors the three gyroscopes the Hubble has on board. It monitors the temperature, revolutions and thrust of the gyros. The RGA's is going to be installed in the next mission. The engineers re-design the RGA's and the VEST Facility was conducting the test. I assisted the VEST operastors technicians to set-up the test equipment in the cleanroom. Also I has been involved in the preparation of cables and breakout boxes needed for this test. I assisted Mr. M. Brunofski , enginner in charge of this test, in finding the commands and the specifications of each command arguments in the data base. I presented him a written report (See Apendix B). During the test I verified the measurements taken on the oscilloscope and Stripchart Recorder. The RGA's test was completed successfully after diversous anomalies.

CONCLUSIONS

Through the understanding how the VEST Facility related to the Hubble Space Telescope on-orbit and the procedures to be handle on the Clean Room developed the basis of the summer project. I had been envolved through different areas in the Facility: engineer operation, engineering design and technical support. Experience were adquired using the oscilloscope, stripchart recorder as well as other test equipment. I began to understand some concepts thoght in college courses as well as I began to develop troubleshooting thinking and analysis.

I will really like to thanks:

Dr. Joan Langdon - Director SIECA Program

Mr. Dan Krieger - Program Coordinator

Mr. Chuck Manns - Mentor; VEST Operations Manager (J&T)

Ms. Cynthia Ivy - VEST Operations Engineer (J&T)

Mr. Eric Barksdale - VEST Deputy Program Manager (J&T)

who helped me through all the summer.

LOGBOOK

Date	*Task Assigned*
May28-June2	Meet the VEST HST Staff at building 29. Trainning on the engineer operation of the VOCC and VEST. ReadingTechnical Materials on VOCC equipment.
June 1	Get in to the PreFab Laboratory where the technicians construct and test the cables to be installed. Test some cables.
June 2	Trainning on the rules and regulations to be handle in the Clean Room. Meeting of the SIECA program in the afternoon.
June 5	Tour around the Goddard Space Flight Center facilities. Keep reading material on the engineer operation of the VOCC.
June 6	Work on the removing the old DF224 and installing the new one. Check and test procedures on the DF224 before connected to the VEST. EICIT test.
June 7	Re-test the DF224. IVT test and SFT test on the DF-224. Assigned to update and check the actual floor plan of room 100. We are preparing the facilities for the next service misssion that will begin in next September.
June 8	Meassurements on the equipment size and working on the floor plan.
June 9	Work on the floor plan. Disk corrupted.
June 12	Floor Plan 1 done.

June 13	Specifications on the new equipment (NICMOS, STIS, SUN worstations) to be bringed for the next mission. Assigned to design a new floor plan to acomodate the old equipment with the new equipment.
June 14	Floor Plan 2 done.
June 15	Asigned to work with Mr. Scott Clough in the Battery Simmulator. Little training on the purpose of the simulator and how it will works.

June 19	Scott is in vacations during the week. Work on the design of alternatives for the new floor plan.
June 20	Present three possible alternatives as Floor Plan3A, 3B, 3C.
June 21	Search in the DATABASE for some PSTOL commands and presented a report to Mr. Mike Brunofski explaining briefly what the meaning of the commands and the arguments.
June 22-23	Present Floor Plan 4 and check minor details in the design and specifications of the equipments.

June 26-30	Asigned to work on the UPS system. Detail report on what equipment to a certain circuit breaker and where they are located in the floor plan. Add the UPS system to the Floor Plan 1 and 4.

July 3	Minor details on the floor plan were fixed. It is the final plan to be presented to the evaluation comitee.

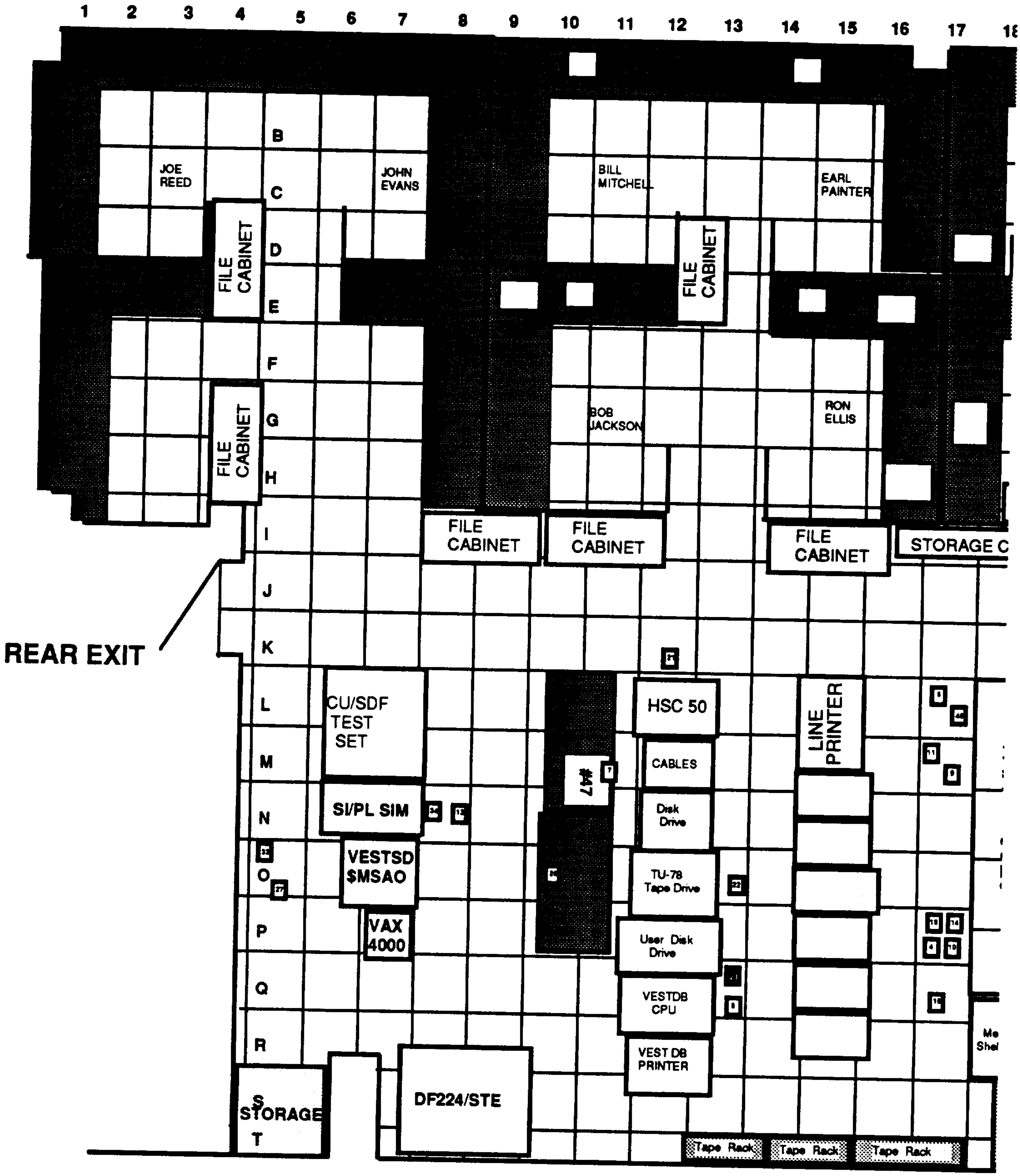
July 5	Batery Simulator proposal is not yet aproved. Begin the rough draft and the abstract for the final presentation.
July 6	Begin to prepare the final report. Give some reading materials to be prepared for next week to test the Rate Gyro Assembly(RGA).
July 7	Technical support in the clean room.

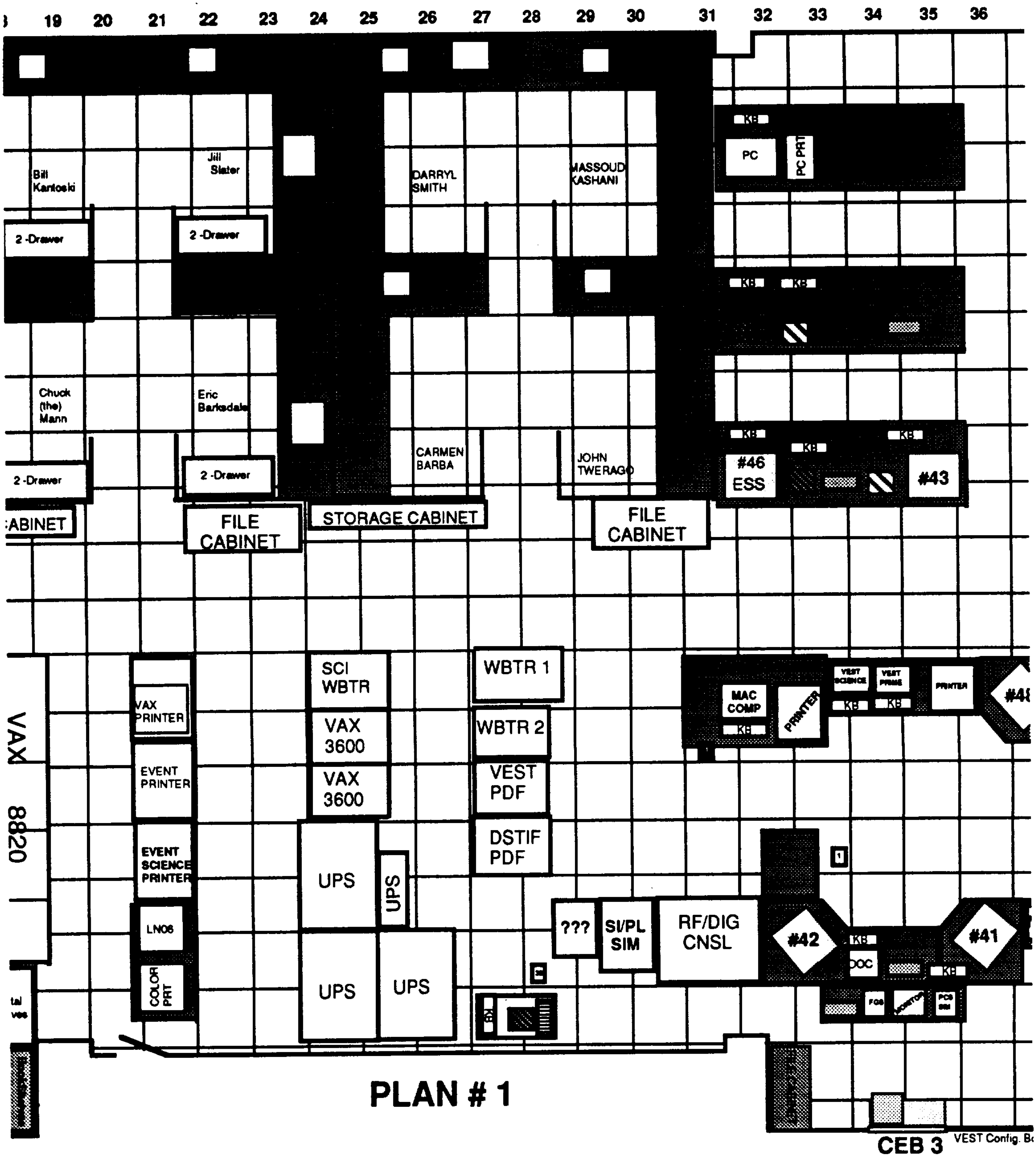
July 10	Meeting during all the afternoon of the SIECA program at Building 8, Conference Room.
July 11	Easy Day. Meeting of the SIECA program with the Director of Goddard Space Flight Center.
July 12	Technical support to Mr. Scott Clough in testing the amount of current can held the 24 gauge-pins in a connector for the solar arrays.
July 13	Organized tech equipment. Miliohm Test on cables at the pre-fab laboratory.
July 14	Milihom test and VJ test on cables. Preparing the equipment to work on RGA's next weeks. Abstract done and handle to Mr. Dan Krieger

---

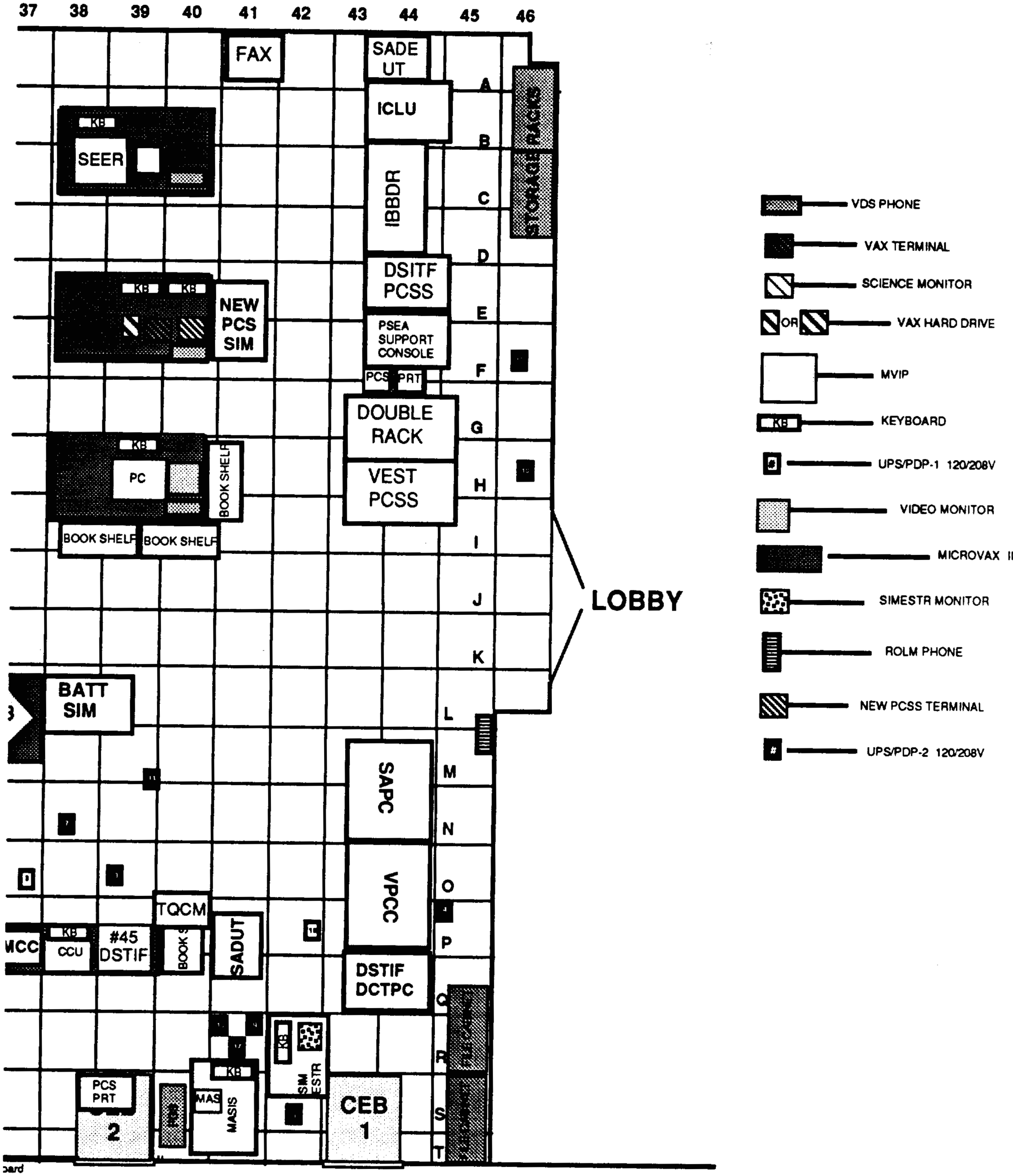
July 17	Setting the equipment in the clean room for the RGA test.
July 18	Veryfying the setting and pre-testing procedures.
July 19 - 21	RGA test is conducted. Technical support.

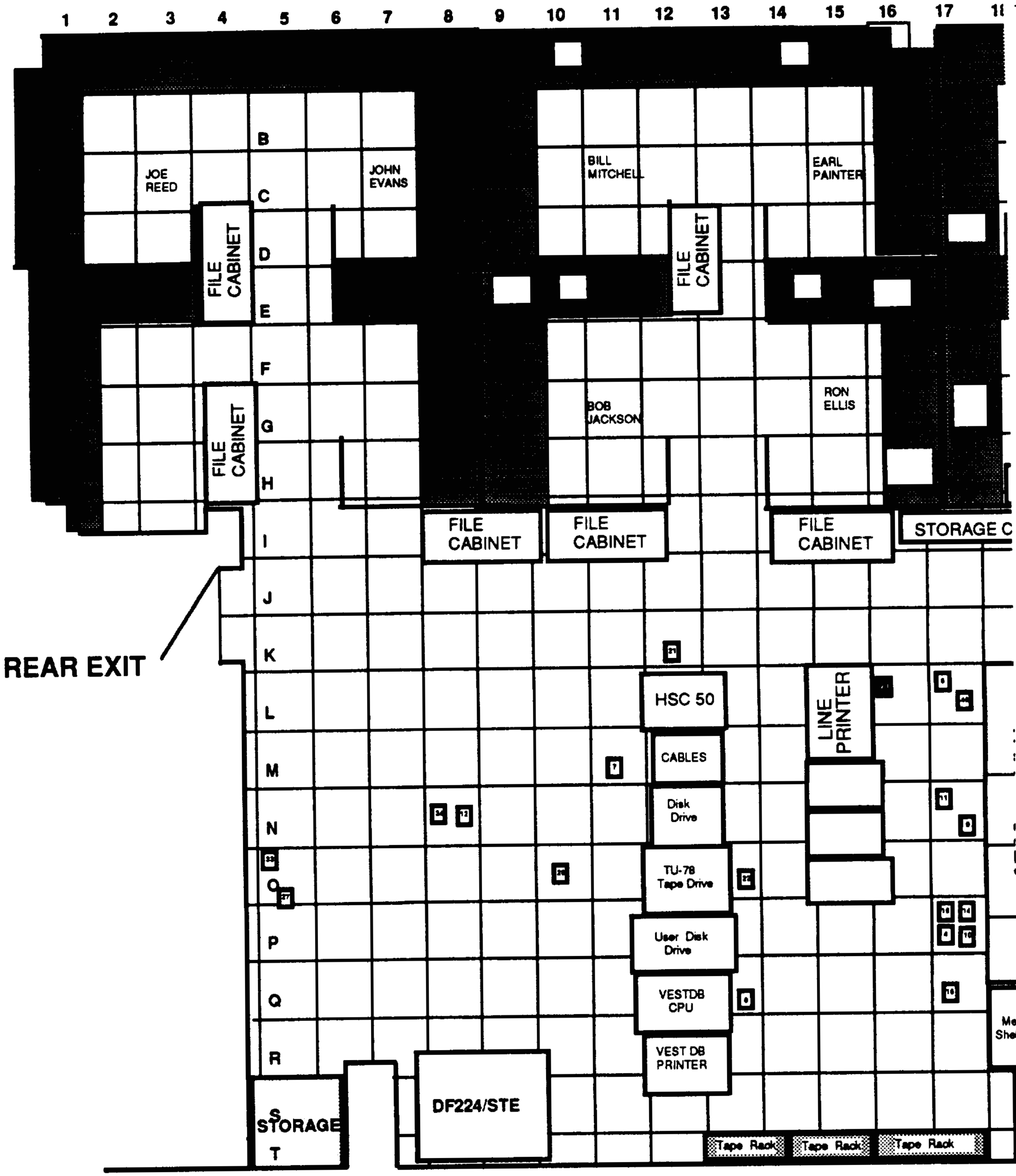
July 24	RGA test keep going. Work on the oral presentation. Give operational support.
July 25-28	Work on the oral presentation and writting the paper. Give operational support.
July 31	Oral presentation at 10:45 in Building 8, Room 121.
August 2	The Award Ceremony at 3:00pm at the Recreation Center.
July 31-Aug4	Proposed operational support. New equipment arriving and sheduled to assist in the installation of the equipment.
August 4	Last Day of the Summer Internship. Pay day. Go back home next day. Thanks to Dr. Joan Langdon and Mr. Dan Krieger for the oportunity and hope to see you again next year.


1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
JOE REED
JOHN EVANS
BILL MITCHELL
EARL PAINTER
BOB JACKSON
RON ELLIS
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
STORAGE C
REAR EXIT
CU/SDF TEST SET
SI/PL SIM
VESTSD $MSAO
VAX 4000
#47
HSC 50
CABLES
Disk Drive
TU-78 Tape Drive
User Disk Drive
VESTDB CPU
VEST DB PRINTER
LINE PRINTER
STORAGE
DF224/STE
Tape Rack
Tape Rack
Tape Rack

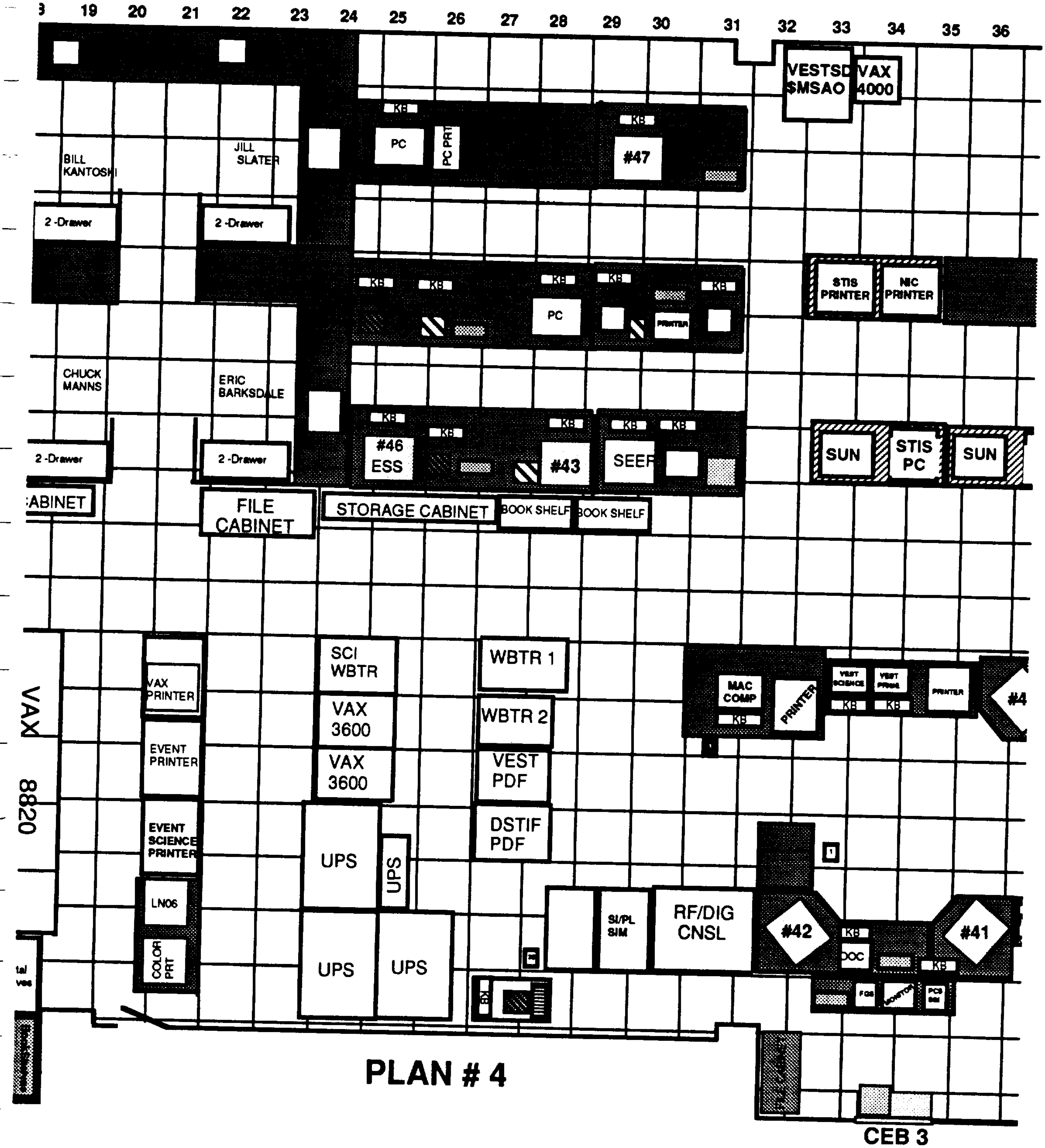

19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
Bill Kantoski
Jill Slater
DARRYL SMITH
MASSOUD KASHANI
KB
PC
PC PRT
2 -Drawer
2 -Drawer
KB
KB
Chuck (the) Mann
Eric Barksdale
CARMEN BARBA
JOHN TWERAGO
KB
KB
KB
KB
#46 ESS
#43
2 -Drawer
2 -Drawer
ABINET
FILE CABINET
STORAGE CABINET
FILE CABINET
VAX 8820
VAX PRINTER
EVENT PRINTER
EVENT SCIENCE PRINTER
LN06
COLOR PRT
SCI WBTR
VAX 3600
VAX 3600
UPS
UPS
UPS
UPS
WBTR 1
WBTR 2
VEST PDF
DSTIF PDF
MAC COMP
KB
PRINTER
VEST SCIENCE
VEST PRIME
KB
KB
PRINTER
#4
???
SI/PL SIM
RF/DIG CNSL
#42
#41
KB
DOC
KB
FOS
PCS SIM
KB
CEB 3
VEST Config. B

PLAN # 1

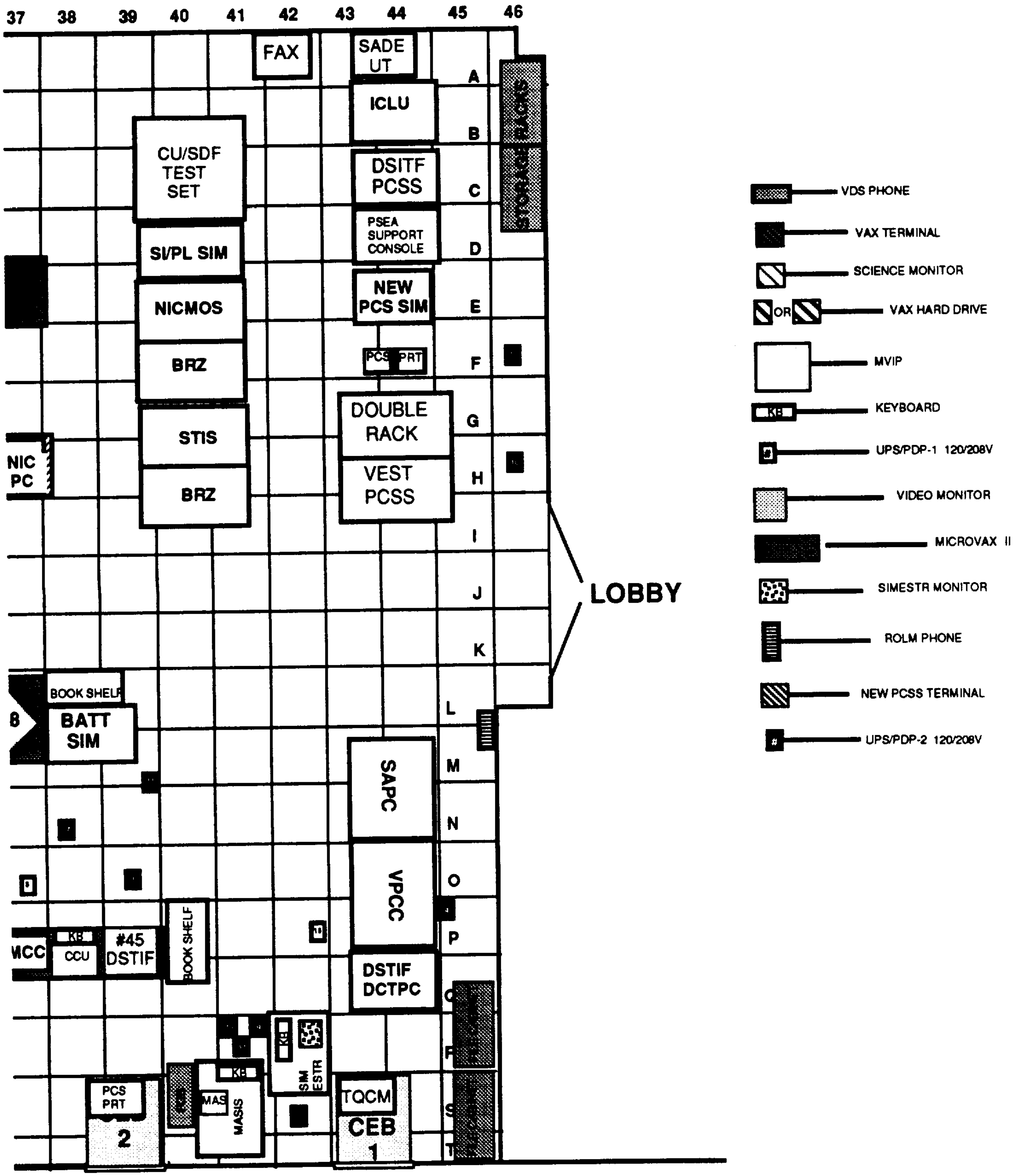

37
38
39
40
41
42
43
44
45
46
FAX
SADE UT
ICLU
IBBDR
STORAGE RACKS
KB
SEER
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
DSITF PCSS
PSEA SUPPORT CONSOLE
PCS
PRT
KB
KB
NEW PCS SIM
DOUBLE RACK
VEST PCSS
KB
PC
BOOK SHELF
BOOK SHELF
BOOK SHELF
LOBBY
BATT SIM
SAPC
VPCC
TQCM
KB
CCU
#45 DSTIF
BOOK S
SADUT
MCC
DSTIF DCTPC
KB
SIM ESTR
KB
MAS
MASIS
PCS PRT
2
CEB 1
VDS PHONE
VAX TERMINAL
SCIENCE MONITOR
OR
VAX HARD DRIVE
MVIP
KB
KEYBOARD
UPS/PDP-1 120/208V
VIDEO MONITOR
MICROVAX II
SIMESTR MONITOR
ROLM PHONE
NEW PCSS TERMINAL
UPS/PDP-2 120/208V



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
JOE REED
JOHN EVANS
BILL MITCHELL
EARL PAINTER
BOB JACKSON
RON ELLIS
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
FILE CABINET
STORAGE C
REAR EXIT
HSC 50
CABLES
Disk Drive
TU-78 Tape Drive
User Disk Drive
VESTDB CPU
VEST DB PRINTER
LINE PRINTER
DF224/STE
STORAGE
Tape Rack
Tape Rack
Tape Rack


19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
BILL KANTOSKI
JILL SLATER
PC
PC PRT
KB
#47
VESTSD $MSAO
VAX 4000
2-Drawer
2-Drawer
PC
PRINTER
STIS PRINTER
NIC PRINTER
CHUCK MANNS
ERIC BARKSDALE
#46 ESS
#43
SEEF
SUN
STIS PC
SUN
2-Drawer
2-Drawer
ABINET
FILE CABINET
STORAGE CABINET
BOOK SHELF
BOOK SHELF
VAX 8820
VAX PRINTER
EVENT PRINTER
EVENT SCIENCE PRINTER
LN06
COLOR PRT
SCI WBTR
VAX 3600
VAX 3600
UPS
UPS
UPS
UPS
WBTR 1
WBTR 2
VEST PDF
DSTIF PDF
MAC COMP
PRINTER
VEST SCIENCE
VEST PRIME
PRINTER
#4
SI/PL SIM
RF/DIG CNSL
#42
#41
DOC
FGS
MONITOR
PCS SIM
CEB 3

PLAN # 4


37 38 39 40 41 42 43 44 45 46
A B C D E F G H I J K L M N O P Q R S T
FAX
SADE UT
ICLU
CU/SDF TEST SET
DSITF PCSS
PSEA SUPPORT CONSOLE
SI/PL SIM
NICMOS
NEW PCS SIM
BRZ
PCS PRT
DOUBLE RACK
STIS
BRZ
VEST PCSS
NIC PC
STORAGE RACKS
LOBBY
BOOK SHELF
BATT SIM
SAPC
VPCC
MCC
KB
CCU
#45 DSTIF
BOOK SHELF
DSTIF DCTPC
SIM ESTR
MASIS
MAS
TQCM
CEB 1
PCS PRT
2
VDS PHONE
VAX TERMINAL
SCIENCE MONITOR
OR
VAX HARD DRIVE
MVIP
KEYBOARD
UPS/PDP-1 120/208V
VIDEO MONITOR
MICROVAX II
SIMESTR MONITOR
ROLM PHONE
NEW PCSS TERMINAL
UPS/PDP-2 120/208V

APENDIX B

UPS/PDP-1 120/208 Volts

(REVISION 06/28/95)

CIR #	*HARDWARE*	*GRID*	*SPECIFICATIONS*
C-1	MICROVIP 42	O-33	20A/125V (DD)
C-3	MICROVIP 41/ MCU	O-37	20A/125V (DD)
C-5	LINE PRINTER/NETWORK COMP	L-17	20A/125V (DD)
C-7	MICORVIP47/HSC50 CONSOLE	M-11	20A/125V (DD)
C-9	PDU1 C-42	N-17	30A/125V/L5-30
C-11	PDU1 C-38	N-17	30A/125V/L5-30
C-13,15,17	VPCC	P-42	3o 60A
C-19,21,23	HSC50	K-12	3o 30A/L21-30
C-25,27,29	OPEN	O-05	3o 30A/L21-30
C-31,33,35	OPEN	O - 5	3o 30A/L21-30
C-37,39,41	NEXT TO SI/PL SIMULATOR	Q-28	3o 30A/L21-30
C-2,4,6	Vax 8820/Unibus/BI Bus	P-17	3o 20A/L21-30
C-8	VESTDB CPU	Q-13	30A/125V/L5-30
C-10	Vax 8820/Unibus/BI Bus	P-17	30A/125V/L5-30
C-12	VESTSD$MSAO	N- 8	30A/125V/L5-30
C-14	TAC1::DU1	P-17	30A/125V/L5-30
C-16	DON$MUAO:	Q-17	30A/125V/L5-30
C-18	Open	P-17	30A/125V/L5-30
C-20,22,24	SYSTEM INDUSTRY DISK DRV	O-13	3o 30A/L21-30
C-26,28,30	HSC000$MUA1 and DON$MUCO	O-10	3o 30A/L21-30
C-32,34,36	OPEN	N- 8	3o 30A/L21-30
C-38,40,42	Vax 8820/Unibus/BI Bus	L-17	3o 60A

UPS/PDP-1 120/208 Volts

(REVISION 06/28/95)

CIR # HARDWARE	GRID	CIR # HARDWARE	GRID
1 MICROVIP 42	0-33	2\\	P-17
3 MICROVIP 41/ MCU	O-37	4 -Vax 8820/Unibus/BI Bus	P-17
5 LINE PRINTER/NETWORK	L-17	6//	P-17
7 MICORVIP47/HSC50 CONSOLE	M-11	8 VESTDB CPU	Q-13
9 PDU1 C-42	N-17	10 Vax 8820/Unibus/BI Bus	P-17
11 PDU1 C-38	N-17	12 VESTSD$MSAO:	N-8
13\\	P-42	14 TAC1::DU1:	P-17
15 -VPCC	P-42	16 DON$MUAO:	Q-17
17//	P-42	18 TAC/LTU CPU	P-17
19 \\	K-12	20\\	O-13
21 -HSC5O	K-12	22 -SYSTEM INDUSTRY DISK DRV	O-13
23 //	K-12	24//	O-13
25\\	O-5	26\\	O-10
27 - OPEN	O-5	28 -HSC000$MUA1 & DON$MUCO	O-10
29//	O-5	30//	O-10
31\\	0-5	32\\	N-8
33 - OPEN	0-5	34 -OPEN	N-8
35//	0-5	36//	N-8
37\\	Q-28	38\\	L-17
39 -NEXT SI/PL SIMULAT	Q-28	40 - Vax 8820/Unibus/BI Bus	L-17
41//	Q-28	42//	L-17

UPS/PDP-2 120/208 Volts

(REVISION 06/28/95)N

CIR #	*HARDWARE*	*GRID*	*SPECIFICATIONS*	
C-1	MACINTOSH IVY; PDF	M-31	SINGLE DUPLEX	
C-3	THINWIRE REPEATER(UNDER FLOOR)	0-39	SINGLE DUPLEX	
C-5	PLUG TO EXTENSION BOX=CLEAR	S-42	SINGLE DUPLEX	
C-7	MICROVIP 48	N-38	SINGLE DUPLEX	
C-9	CLEAR	R-41	SINGLE DUPLEX	
C-11	CLEAR	M-39	SINGLE DUPLEX	
C-13	NO CABLE CONNECTED TO BREAKER	-----	----------------	
C-15	OPEN	R-41	??	
C-17	OPEN	R-41	??	
C-19	ICLU (Directly Connected)	NONE	----------------	
C-21	ICLU (Directly Connected)	NONE	----------------	
C-23	TTAC(new)	L-16	??	
C-2,4,6	DCTP	P-45	3o	??A/L21-30
C-8,10,12	PCSS	H-46	3o	??A/L21-30
C-14,16,18	OPEN	F-46	3o	??A/L21-30
C-20,22,24	VBS	-----	3o	60A/5-WIRE

UPS/PDP-2 120/208 Volts

(REVISION 06/28/95)

CIR #	HARDWARE	GRID	CIR #	HARDWARE	GRID
1	MACINTOSH IVY/PDF	M-31	2\\		P-45
3	REPEATER(under floor)	0-39	4	-DCTP	P-45
5	PLUG EXT BOX=CLEAR	S-42	6//		P-45
7	MICROVIP 48	N-38	8\\		H-46
9	OPEN	R-41	10	-PCSS	H-46
11	OPEN	M-39	12//		H-46
13	NO CABLE CONNECTED TO BREAKER		14\\		F-46
15	OPEN	R-41	16	-OPEN	F-46
17	OPEN	R-41	18//		F-46
19	ICLU(directly connected)	NONE	20\\		- - -
21	ICLU(directly connected)	NONE	22	-VBS	- - -
23	TTAC(new)(open)	Q-13	24//		- - -
25			26		
27			28		
29			30		
31			32		
33			34		
35			36		
37			38		
39			40		
41			42		

P-442-0428
Revision A
March 22,1995

APPENDIX-C

PSTOL LIST

BADECHO(2) *Bad Echo Inhibit Control (1=inhibit, 2=no inhibit)*

BOD(1) *Bright Object Detector (1=enable, 0=disable)*

CSSVOTER(1,1) *Enable/Disable Voted Outputs of the CSSs monitor signal to BOD logic. ($1=CSS 1, $2=CSS 2; 1=enable, 0=disable)*

DIUICMD(0,0,0,1) *Set DIUI Command Registers ($1=DIU 1, $2=DIU 2, $3=DIU 4, $4=DIU5; 0=Side A, 1=Side B)*

DIUIDAT(0,0,0,1) *Sets DIUI Data Registers ($1=DIU 1, $2=DIU 2, $3=DIU 4, $4=DIU5; 0=Side A, 1=Side B)*

DIUPWR(5,1,0) \ *Control Power to select DIU ($1=DIU#:1,2,4,5; $2=DIU*
DIUPWR(5,2,1) / *Side:1=A,2=B; $3=Power: 0/1=Off/On)*

DMUMODE(1) \ *Selects DMU Mode (1=Normal, 2=Bypass, 3=Diagnostic)*
DMUMODE(2) /

DMUMODSW(A,A,A,A,A,A,B) *Configure DMU Module ($1=CIF, $2=FHSTI, $3DIUI, $4=COM $5=TFC, $6=TRI, $7= RGA)*

GYROPWR(11,0) \
GYROPWR(11,1)---*Turn on/off a single Gyro($1=Gyro#, $2=on/off)*
GYROPWR(12,0) /

KADSABLE(0) \ *Sends commands to PSEA every 30s.Disable KA monitoring.*
KADSABLE(1) / *(1=on, 0=off)*

MSSCONF(0,1) *Loads MSS status into PSEA Config Mem($1=MSS !,$2=MSS 2)*

PSEAPWR(1) *Turns on/off PSEA*

RGABIAS(0,0,0) *Loads RGA Rate Biases for ea axis in PSEA Conf Mem(V1,V2,V3)*

PSTOL LIST (Continued)

RGACHDIS (0,1,1,1,1,1) \ *Disable Gyro Control Heaters(RGA 11,12,23,34,35,36;*
RGACHDIS (0,1,1,1,1,1) / *1=not disable, 0=disable)*

RGACONF(1,1,1,1,0,0) *Loads RGA on no-op status into the PSEA Conf Mem (RGA 11,12,23,34,35,36; 0=No Operation, 1=On)*

RGAMODE(0) \ *Controls RGA mode selection via dismond FLAG in Flight*
RGAMODE(2) / *Software (0= Autonomous Mode Switching, 2=command to high mode and disable mode switching)*

RGANUL(0,0,0,0,1,1) *RGA Null Status into the PSEA (RGA 11,12,23,24,35,36; 1=no disable, 0= disable)*

RGASHDIS(0,0,1,1,1,1) *Disable Gyro Survival Heaters(RGA 11,12,23,24,35,36; 1=no disable, 0= disable)*

RGATPCTR(0,0,1,1,1) \ *Selects Temp Controllers for Gyros(RGA 11,12,23,24,35,36;*
RGATPCTR(1,1,1,1,1) / *1=Torque Generatorthermistor, 0= Heat Blanket Thermo)*

RGHTRSEN(1,1,0,0,0,0) *Enables/disable the RGA control and survival heaters*

SADECONF(1,1,1,1,0,1) *Load SADE conf in PSEA Conf Mem($1=Torque Failure Flag Test, $2= Profile Test, $3=Sade Switch Over,$4= Final atempt overridding test, $5= SADE 1 Status, $6=SADE 2)*

SELMCE(0) \ *Select MCE Side in the PSEA (0= A Side, 1= B Side)*
SELMCE(0) /

SMC(0,0,1) \ *Enables/disble SMC for use and selects it to be used when PSEA Safe*
SMC(1,1,0)---*Mode is initiated. ($1=SMC A, $2=SMC B; 0/1=Enable/Disable)*
SMC(1,1,1) / *($3=Use SMC: 1=SMC A, 0=SMC B)*

SMCSWCTL(1,1,0,1) *Loads SMC S/W Control Word into PSEA Conf Mem($1= MTE test, $2= RGA RC, $3= MSS RC; 0/1=Enable/Disable)*

SMHWTST(4) ***NO IN DATA BASE; suppose to be for last service miss***

PSTOL LIST (Continued)

SPCONF(0,0,1,0,1) *($1=Gyro Hold Mode; 0=No hold, 1=Hold)*
($2=Aper Door Close: 0/1=disable/enable)
($3=Sun Point Axis: 0=-V1, 1=+V3)
($4=SP to GG Select: 0/1=disable/enable)
($5=RWA Reas Check: 0/1=disable/enable)

TIMERLIM(4,4,1) *Sets limits for timer A, B, C($1=Timer A,$2=Timer B, $3=Timer C; 1,2,3,4)*

/ COMMANDS

/RPCSSCMD,RRGA11TQ=0
/RPCSSCMD,RRGA11TQ=128
/RPCSSCMD,RRGA11TQ=160
/RPCSSCMD,RRGA11TQ=192
/RPCSSCMD,RRGA11TQ=224
/RPCSSCMD,RRGA11TQ=32
/RPCSSCMD,RRGA11TQ=64
/RPCSSCMD,RRGA11TQ=96
/RPCSSCMD,RRGA12TQ=0
/RPCSSCMD,RRGA12TQ=128
/RPCSSCMD,RRGA12TQ=160
/RPCSSCMD,RRGA12TQ=192
/RPCSSCMD,RRGA12TQ=224
/RPCSSCMD,RRGA12TQ=32
/RPCSSCMD,RRGA12TQ=64
/RPCSSCMD,RRGA12TQ=96
/RRGAS3ON
/RRGAS4ON
/RSGYRCMD,RGYROCNF,C1H,C2H,C3H,C4H,C5H,C6H,C1I,C21,C3A, C4A,C5I,C6I
/RSGYRCMD,RGYROCNF,C1H,C2H,C3H,C4H,C5H,C6H,C1I,C2I,C3I, C4I,C5I,C6I

Programming for the GLAS and the 1.2m Telescope

by

Gilbert Castillo

Summer Institute in Engineering and Computer Applications Program

Mentor: Jan F. McGarry
Code: 920.1
LTP-SGAPO

Head: John J. Degnan

National Aeronautics and
Space Administration

Goddard Space Flight Center

Since its creation, NASA has gathered vast information on both earth and space. Satellite Laser Ranging (SLR) has provided data on the earth for over 25 years; the Geoscience Laser Altimeter System (GLAS) project, when completed and placed into orbit around the earth, will help scientists better understand our planet's changing surface. SLR systems are located around the world and also provide information about the earth and its constant state of gradual flux. GLAS is expected to be ready and launched before the year 2000. The purpose of this system is to use a laser beam to map the earth's surface from an orbiting satellite. Its design is currently being investigated by the Experimental Instrumentations Branch (Code 924) of the Laboratory for Terrestrial Physics (LTP) at the NASA Goddard Spaceflight Center (GSFC). My internship involved working in Code 920.1 (the Space Geodesy and Altimetry Projects Office) providing programming support to both the 1.2m SLR and tracking facility and the GLAS project. I worked closely with my mentor, Jan McGarry, Senior Software Analyst in Code 920.1. Although I was placed in Code 920.1, it is important to point out that the work I did on the GLAS project was mainly for Dr. James B. Abshire in Code 924.

SLR data provides scientists with information to determine the earth's gravity field, crustal motion, polar motion and more. At NASA's 1.2m telescope site at the Goddard Space Flight Center, laser ranging data is gathered and stored for experiments in atmospheric modelling, satellite spin determination and relativity to name a few. The data includes information such as the day and time of the laser firing, the vital round-trip time of flight of the laser, the telescope's pointing angles and other system information. In 1983, an international format standard for full rate SLR data was made. This standard was named MERIT (Monitoring of Earth Rotation and Intercomparison of Techniques) and has been the operational format for the global community since its creation. The 1.2m telescope, however, like most other SLR stations around the world, produces and stores its data in another internal format, making the data unaccessible to the global community. My project for the 1.2m telescope was to convert the data from the internal 232 byte format to the 130 byte MERIT II format. Much of the data in the internal format is not needed in the MERIT II format so it is omitted. I rewrote and modified existing FORTRAN modules that converted the data from the internal format to the older Crustal Dynamics Project 88-byte Mailing Tape Format (CDP-MLT). I added another module that performed a statistical analysis on the calibration data in order to eliminate noise and compute a refined system delay value. One of the first users of the 1.2m telescope data in this format will be Dr. Alley of the University of Maryland who will look at relativistic effects in the Russian GLONASS satellite's orbit. GLONASS is the Russian equivalent of our Global Positioning System (GPS) which, with the use of a special receiver, has the ability to give one's location on the earth's surface to within a few meters of accuracy.

The GLAS is essentially composed of a laser and a receiver (see Fig 1.1, taken from Abshire, et. al). After the laser fires a beam to the earth's surface, the reflected light is gathered and processed by the receiver. A simplistic scenario for our discussion will be sufficient to demonstrate how the system will work. The GLAS satellite, orbiting approximately 705 km above the earth, fires the laser onto the terrain directly below. The laser beam propagation time determines the height of the terrain (See Figure 1.3). The orbital altitude (R) of the satellite is known to an accuracy of approximately 10cm as is the radius of the earth (re). The distance from the satellite to the terrain is the propagation time (Δt) multiplied by the speed of light (c) and divided by two. The height (h) of the terrain is then:

$h = R - (\Delta t * c)/2 - re.$

My project for GLAS was continued from last summer and involved programming a portion of a hardware simulator for the system. The purpose of the GLAS simulator is to investigate and test different hardware configurations that may or may not be used to build the final product to be launched into orbit. The GLAS investigative team is responsible for setting the scientific requirements that drive the engineering requirements. They need reliable hardware that will provide accurate and dependable results. The simulator will be used to determine what hardware will provide their required results. The GLAS science team will also use the simulated returned information to develop and test algorithms to analyze the data thoroughly and accurately.

There is currently a working simulation program that generates data for a two-dimensional environment and performs simulated hardware analysis on the data. This version (ver 3.7) has been completed and released. The GLAS Instrument Team in Code 924 is one of the users of the simulator. There are also other principal investigators at NASA and the University of Texas at Austin who use the simulator. The simulation program consists of three major components: the terrain generator, the simulator, and the statistical analyzer. The terrain generator is used to create different user defined surfaces or interpolate actual terrain data. The user defined surfaces include sloped surfaces and step-like surfaces. Each of these surfaces in turn can be given a fixed, ramped, stepped, or a sinusoidal reflectivity. A high reflectivity value would be indicative of a surface such as ice or water while a low reflectivity value would be indicative of a surfaces such as soil. The simulator is composed of a space to time transformation, a receiver subsystem, and a waveform digitizer. The space to time transformation is what keeps track of the time it takes for each burst of the beam to travel to and from the earth. The receiver subsystem is responsible for gathering the data from the returned beam, filtering it, and yielding a received waveform. The waveform digitizer is then used to sample the received waveform and to compute an estimated time for the propagation of each pulse, or shot, of the beam. The statistical analyzer produces timing statistics on many shots of the simulator.

Looking at Figure 1.2 (taken from Abshire, et. al), we can see how the simulator operates. First, terrain data is input to the simulator. The terrain can be user defined or real data. Next, the laser is fired at the terrain. In the TERRAIN RESPONSE block, the simulator models the return of the pulse from the terrain. A transformation from the space domain to the time domain occurs at this point. The RECEIVER RESPONSE block then models the acquisition of the pulse by the system at the satellite. Here the signal is filtered through a low-pass filter. The pulse is then discretely sampled and stored in the computer for further analysis in the DIGITIZER block. Lastly, the simulator can perform statistical analysis on the timing of the beam and generate histograms of the results or it can plot the estimated heights against the actual terrain data. Upgrading from the two-dimensional simulator to the three-dimensional one will require modifications to the terrain generator which produces the input for the simulator and to the TERRAIN RESPONSE block where the space-to-time transformation occurs.

My contribution to the GLAS project this summer involved modifying and improving the portion of the three-dimensional terrain generator code which uses actual AOL (Airborne Oceanographic Lidar) data to produce actual surfaces found on the earth. AOL is a remote sensing instrument usually carried onboard a NASA P-3B aircraft located at Wallops Flight Facility, Virginia. The data that we were looking at in particular was from an area over Greenland. The code takes the AOL data and transforms it into a format that the simulator's space-to-time transformation routine can use. This involves extracting and converting the latitude, longitude, and height information found in an AOL file into cartesian coordinates of x, y and z on an evenly "gridded" area with positive dimensions that can easily be changed to reflect different scaling (ie. kilometers, meters, centimeters, etc.). The terrain generator creates an xy grid with the corresponding z or height for each grid point. The value of each z coordinate depends on the surrounding AOL data point values. There are areas which are dense in values while other areas are quite sparse as can be seen in Figure 1.4. This type of transformation to a grid is necessary in order to make the data tractable for the simulator. The algorithm that was developed last summer would determine a z value for each xy grid point by taking the closest three data points, with respect to their x and y values, using the distance formula. Once the three data points were found, the equation of the plane containing the three points was then calculated. With the equation of the plane known, the x and y values of the grid point are substituted into the equation and the ***z*** value, or height, for the grid point determined. This process was done with each grid point, creating a mesh of the terrain. This algorithm, however, was inadequate. In order for the terrain to be regenerated as accurately as possible, the data points had to be ample and evenly distributed, otherwise exaggerated anomalous points would be created. Since the AOL data would be sparse and unevenly distributed, the algorithm had to be improved. Instead of taking the closest three points, we now took the closest three points whose ***xy*** coordinates in a sense "boxed-in", or surrounded, the desired grid point. The three points had to satisfy the conditions such that there was a point to the left of and to the right of the grid point and also a point above and below the grid point as indicated in Figure 1.5. This would reduce the number of anomalies.

Several problems were encountered during the coding of the algorithm. Once we received a postscript file depicting the pattern of the data, we were able to determine if the data was being read and displayed correctly. The printout also showed how the data was collected using a circular rotation making certain areas dense with points and other areas rather scarce. There was also the need to shift and rotate all the data points so that they would lay in the positive ***xy*** grid and be oriented as closely as possible to the ***x***-axis. This is done to make the data more tractable for the space-to-time transformation module of the simulator which will follow the terrain horizontally from left to right. However, the data does not form a straight path and the simulator will require additional information so that the path may be followed appropriately. An image of the reproduced and adjusted terrain can be seen in Figure 1.6. IDL (Interactive Data Language) was use to create the image.

The final version of the terrain generator will contain options for creating user defined ramped or step surfaces. These options are complete. The other option, which has been discussed, is to use AOL data to recreate a surface. At this point, the algorithm recreates the surfaces well, but there is still room for improvement by eliminating any extrapolation of the data.

The algorithm has been tested with actual AOL data sets from June 28, 1993. The data has been plotted using IDL (Interactive Data Language) and compared to the plots received from Wallops Flight Facility. Before the terrain generator can be considered ready, an algorithm to follow the flight path of the AOL data must be coded. This will model the satellites path above the earth. Last of all, it must be integrated and tested with the rest of the modules of the program.

My experience at NASA GSFC has been a very positive one. Like last summer, it was a pleasure to be able to work side by side with professionals in the engineering sciences. I had an enjoyable summer and I'd like to thank Jan McGarry for her help and support. I would also wish to thank everyone under code 920 who welcomed me back on my return internship and express my gratitude to Dr. Joan Langdon of Bowie State University for selecting me to participate once again in the SIECA program.

REFERENCES:

Abshire, James B., McGarry, Jan F., Pacini, Linda K., Blair, J. Bryan, and Elman, Gregory C., "Laser Altimetry Simulator, Version 3.0 User's Guide", NASA Technical Memorandum 104588, January 1994.

Special thanks to Serdar Manizade/NASA/WFF/972/EGG

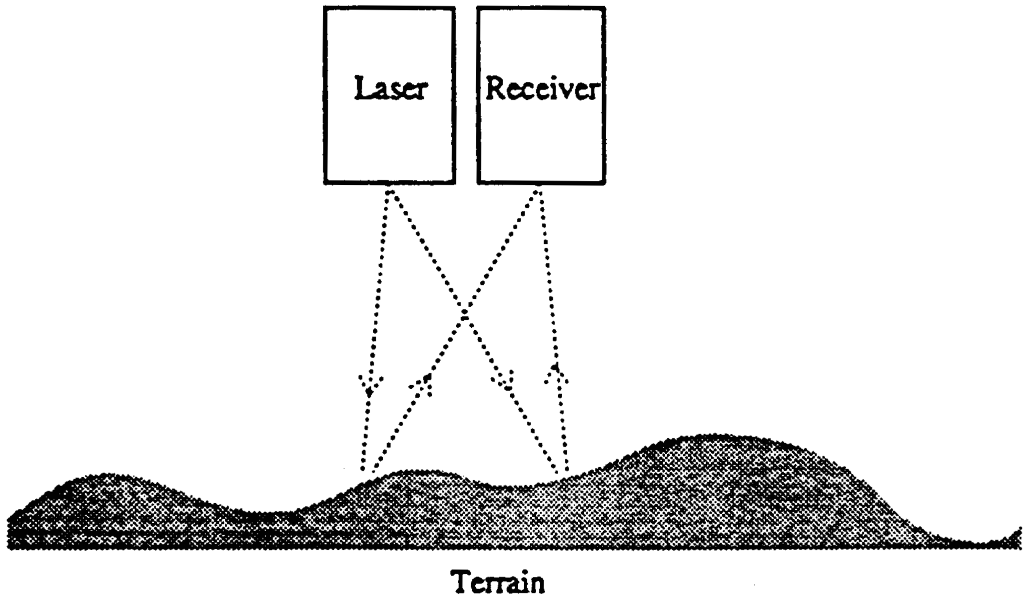


Figure 1.1. Laser Altimeter Measurement Geometry.

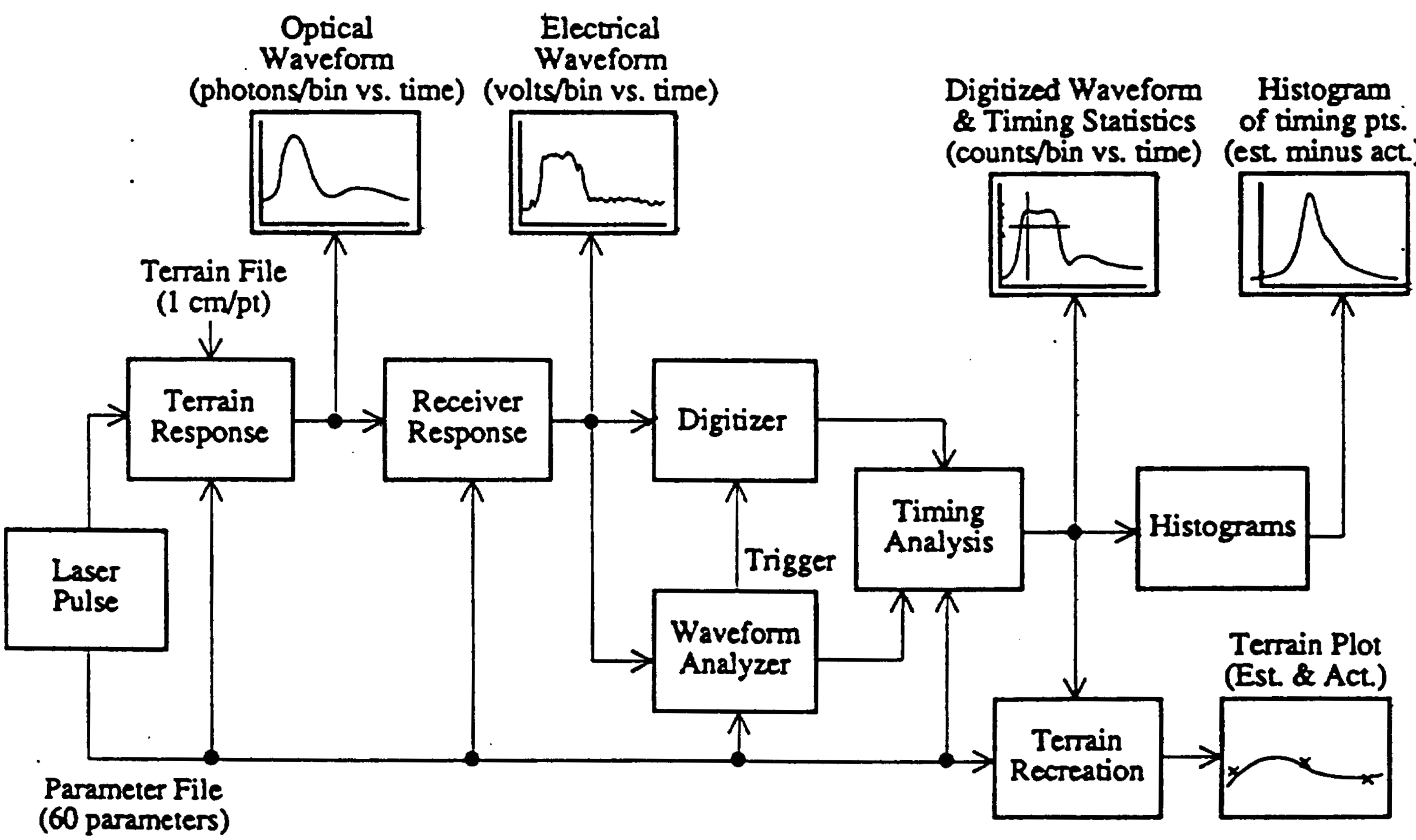
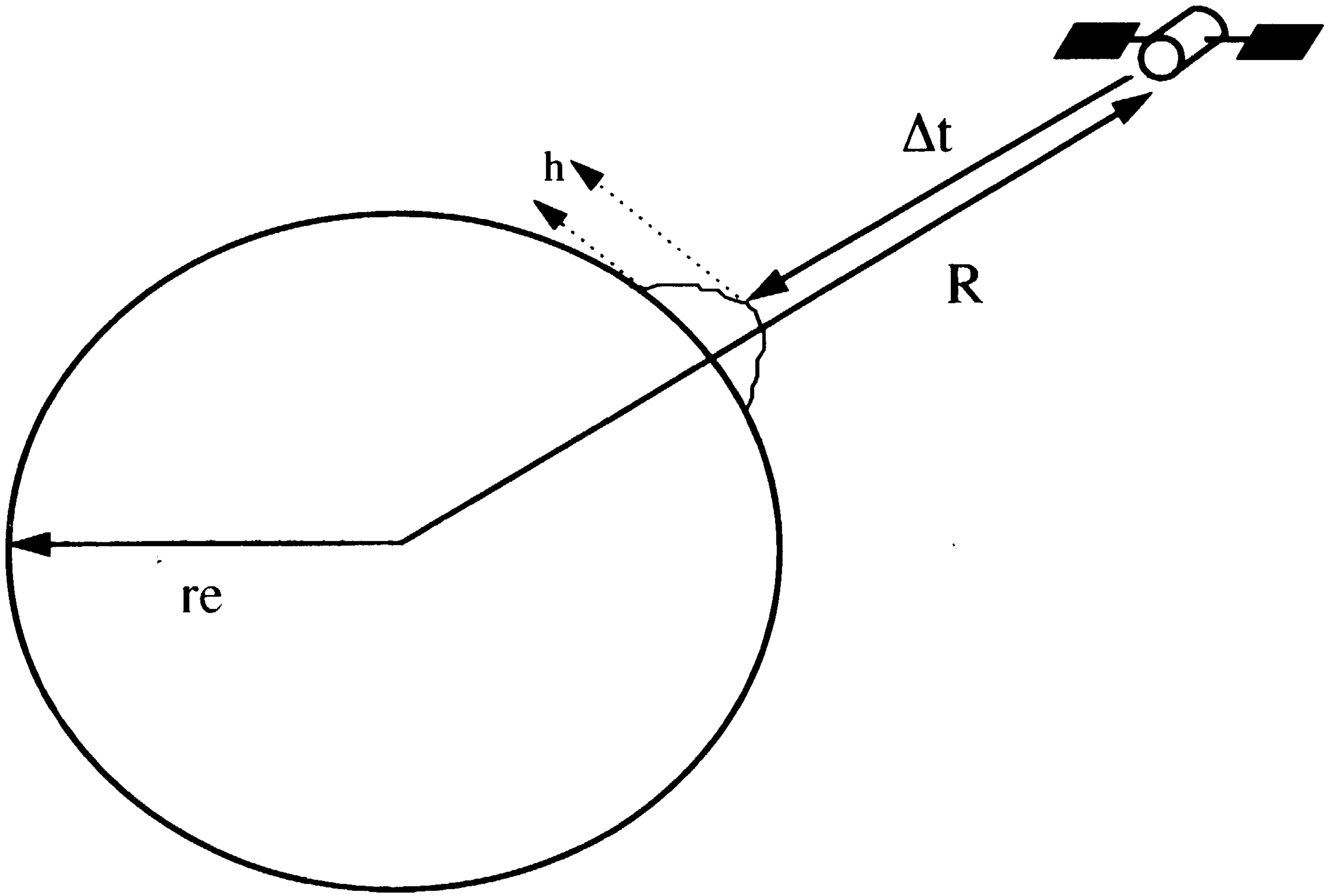
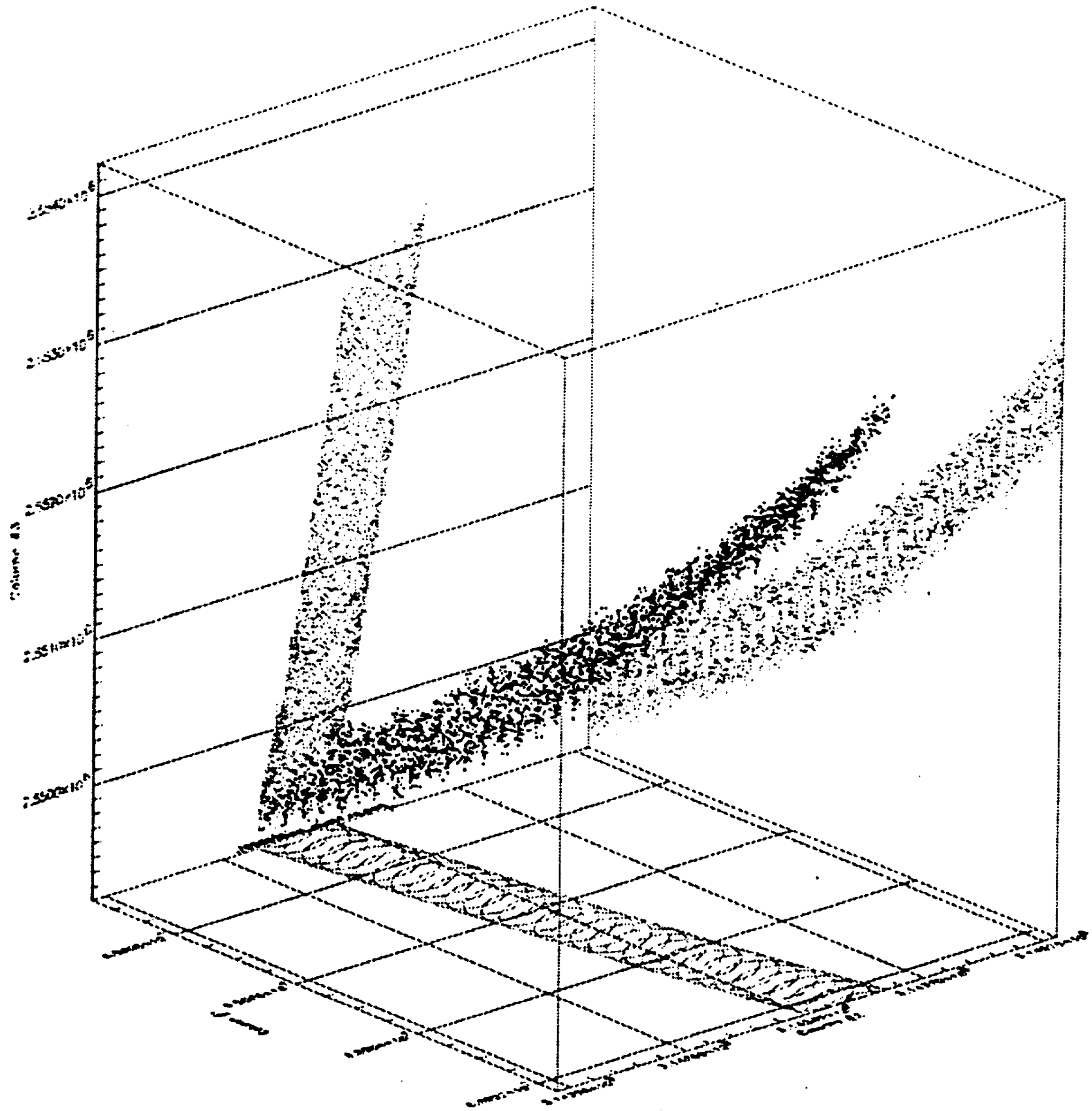


Figure 1.2. Laser Altimetry Simulator Block Diagram.



$$h = R - (\Delta t * c) / 2 - re$$

Figure 1.3



data file is: ./930628aol_clust1.rq.part
directory is: /gen/data34/mcgarry
date : Mon Jul 10 10:47:33 EDT 1995
x axis: data word#2; center= 3.14365e+08; axis span= 21088.0
y axis: data word#1; center= 6.50055e+07; axis span= 7676.00
z axis: data word#3; center= 2.55172e+06; axis span= 5006.00

NASA/GSFC/WFF/EGG/WASC

AIRBORNE OCEANOGRAPHIC LIDAR

Figure 1.4

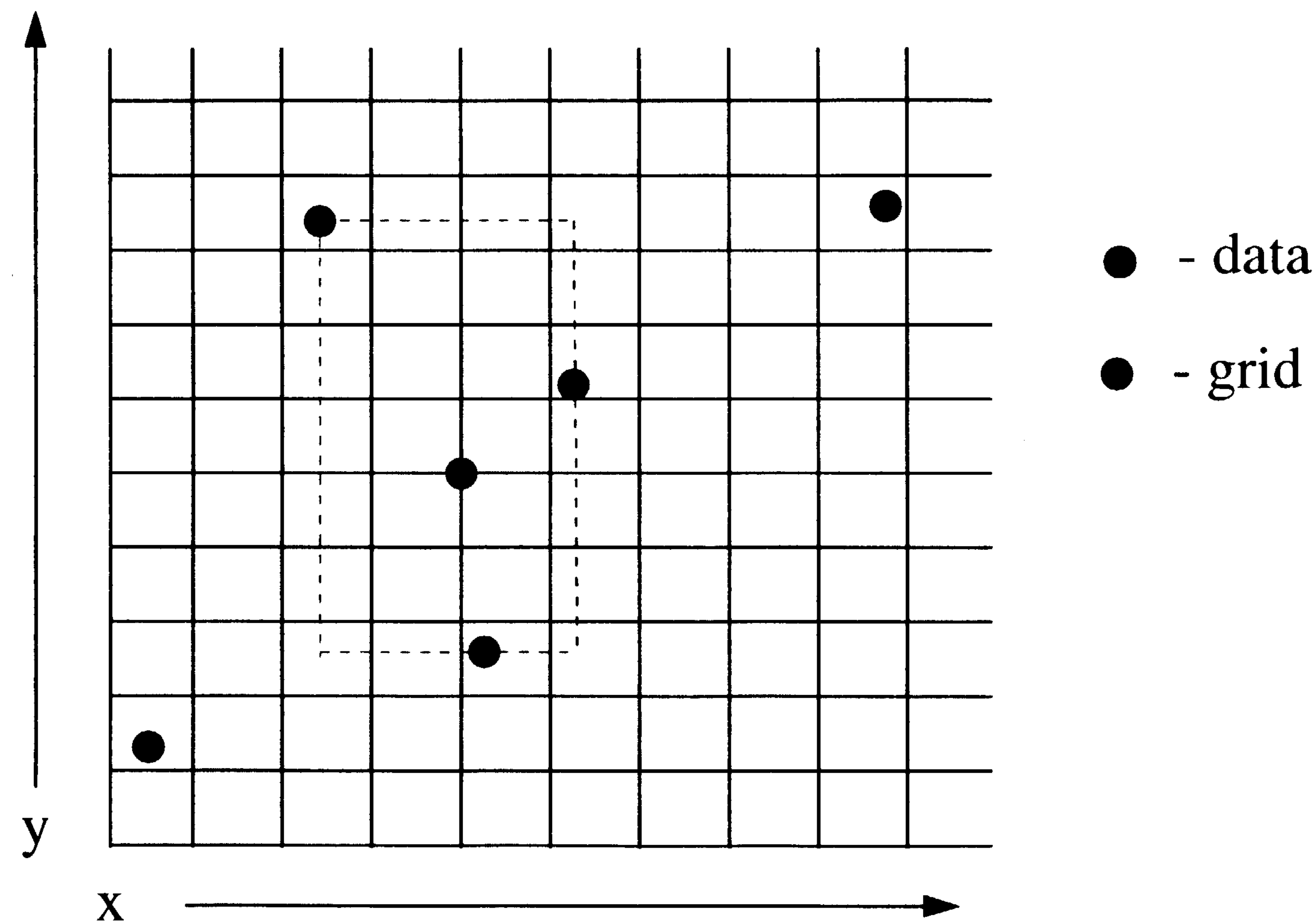
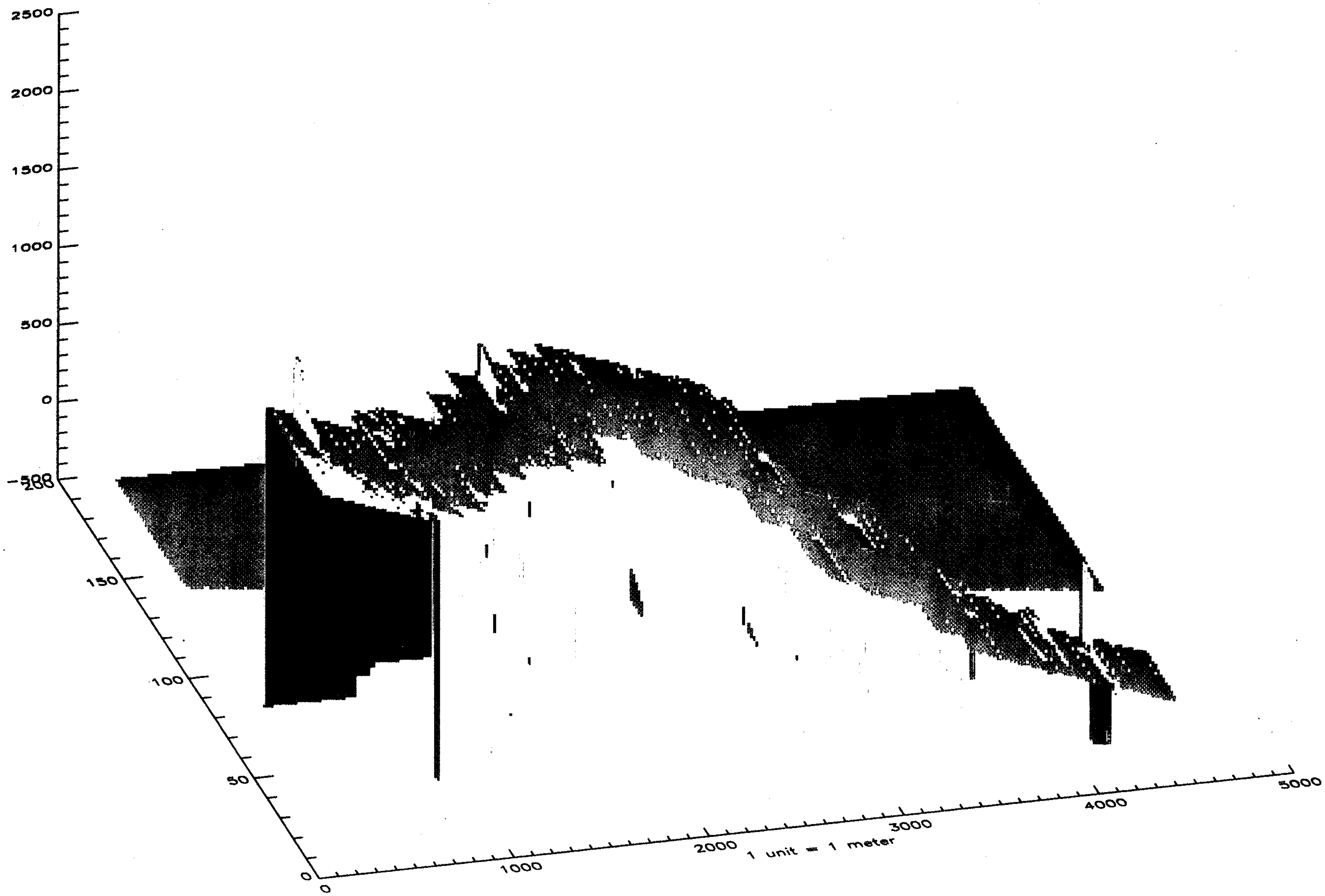


Figure 1.5

FIGURE 1.6


2500
2000
1500
1000
500
0
-500
200
150
100
50
0
0
1000
2000
3000
4000
5000
1 unit = 1 meter

Network Management for the EOS Backbone Network (EBnet)

Shahar Harel

SIECA-Graduate Student

Code 541.3

August 2, 1995

Mentor: Vishal Desai

Table Of Contents

Abstract

The Mission to Planet Earth (MTPE) is a far ranging project established by NASA in order to study the planet as an integrated system of atmosphere, oceans and continents interacting through energy exchange. The Earth Observing System (EOS) includes a constellation of satellites that will collect the pertinent data which scientific investigators will use in their research. Providing and maintaining a reliable network on the ground is an essential component of this mission.

The current design uses a distributed, open systems architecture. The data is sent from a group of satellites, known as the Tracking and Data Relay Satellite System (TDRSS), to White Sands Complex in New Mexico and then transmitted to the EOS Data and Operation Systems (EDOS) at Fairmont, West Virginia for further processing. The ECOM network transmits the data from the West Virginia site to nine different Distributed Active Archive Centers (DAACs) for storage, including one at Goddard Space Flight Center. It is also responsible for transmitting data from White Sands Complex directly to these sites in real time. The EOS Science Network (ESN) provides communications between the different DAACs.

This design is being upgraded and simplified by the EOS Backbone Network (EBnet) which will consolidate both ECOM and ESN into one network. While this change is taking place the network must be continually managed to accommodate operations, simulations, and testing. The Simple Network Management Protocol (SNMP) is ideally suited for this purpose. Hewlett Packard's Network Node Manager implements this protocol and was used in conjuction with Remedy's Action Request System (ARS) Trouble Ticketing software to manage a simulated network.

Role of EBnet Within Mission To Planet Earth

The EOS Backbone Network (EBnet) is an essential part of NASA's Mission To Planet Earth(MTPE). The primary goal of this mission is to study the earth as an integrated system of oceans, continents, and atmosphere which interact though energy exchange. The Mission To Planet Earth also intends to use this knowledge to help form a wise environmental policy. It was determined that more information about the Earth's land, atmosphere, ice, oceans, and biota could be obtained by placing satellites in space then by any other method.

The Mission To Planet Earth will be realized by using the Earth Observing System (EOS). EOS includes both a constellation of satellites in space which will gather scientific data, as well as the ground network which will receive this data and send it to scientific investigators for their research. Figure 1 shows how the data is transferred from EOS satellites down to the ground communication system and eventually to scientific investigators. A discussion of the data's path is provided.

The data is first collected by the EOS satellites. It is then sent to the Tracking and Data Relay Satellite System (TDRSS). It is the responsibility of these satellites to transmit the data to the White Sands Complex in New Mexico for processing. Originally, it was planned that data would then be sent to West Virginia for further processing and synchronization. However, it now seems like all processing will be conducted at the White Sands Complex.

The EOSDIS Internal Network is crucial element in this scheme and its design and implementation are the responsibility of Code 540, NASCOM (NASA Communications). As shown in Figure 1, this internal network provides data access to four main centers. Controlling the EOS satellites is done from the EOS Operations Center (EOC) at Goddard. The System Monitoring & Coordination at Goddard monitors and manages the EOSDIS network. Other governments and companies are also working with NASA on the Mission To Planet Earth. The Japanese, for example, have some instruments on-board the EOS satellites. In order to control their equipment they send requests along the EOSDIS Internal Network to the EOS Operations Center, which will then send these messages up to the EOS satellites. Data is also transferred from these internal networks to nine different DAACs (Distributed Active Archive Centers). These DAACs, geographically distributed throughout the United States, receive the data in its raw form and provide product generation which converts the information into a more useful form. They also archive and distribute the information.

The data is sent from the DAACs to external networks which ultimately send it to end users such as scientific researchers and international data centers. Originally, the "ECOM network" was one aspect of the EOSDIS Internal Network and provided most of the communications and the EOS Science Network was another part and was responsible for communication between the DAACs. However these two science networks have been consolidated into one, the EOS Backbone Network, also known as EBnet.

Two teams are working on the EBnet project. One is involved with the transport aspect of the network. This group is investigating different technologies, protocols, routing algorithms, etc., which can be used to transport the data along the network. The other group is involved with network management. For a full discussion of network management refer to the next section.

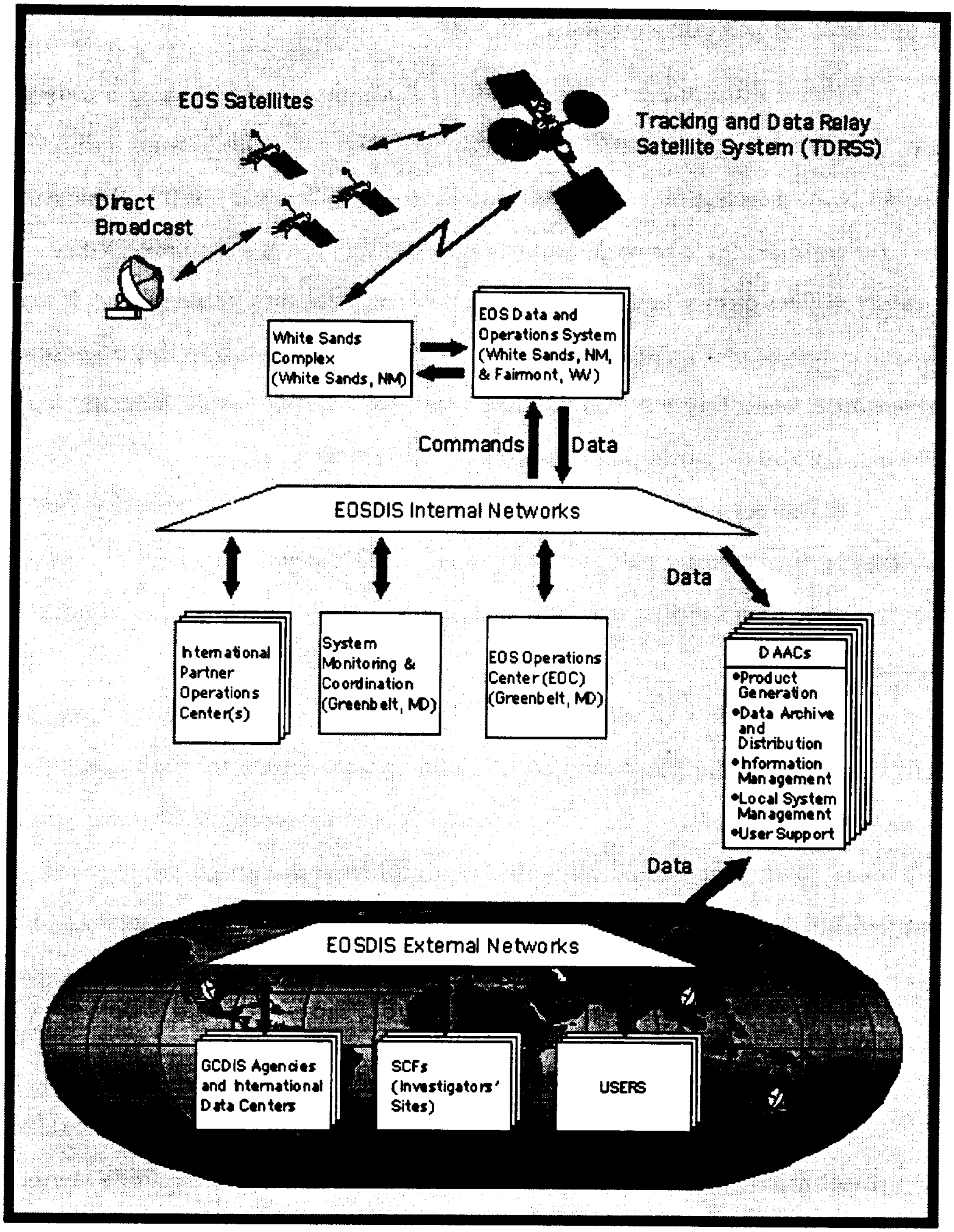


Figure 1 EOS Data and Information System

Network Management and SNMP

Network management is formally defined as the process of controlling a complex data network to maximize its efficiency and productivity. It is the network manager's responsibility to oversee the network, make sure that it is on-line, and that it is operating at peak performance. Since network managers are responsible for an entire network their job typically requires them to isolate problems to the lowest common denominator. If they determine that one node on their network isn't operating as it should then they concentrate on that node, where they may find that the problem lies with one specific terminal. Then they may find that the problem involves a certain interface card.

The network manager also needs to optimize the networks performance. This is achieved by monitoring the traffic on the network. If one branch is overloaded the network manager may, from a remote workstation, change the default routing table of a router to alleviate congestion.

The significance of network management may best be understood from the following everyday example. Many people pay for their groceries at the supermarket with credit cards. If the cashier swipes your card and it doesn't register after a few times people will usually give another credit card. If the transaction goes through the consumer is satisfied, but the company has just lost money. For a Wall Street firm, that transaction could represent millions of dollars. Similarly, NASA has a legitimate interest in managing EBnet. If the network is down valuable data is lost. Moreover, the EOS satellites can not be controlled without an on-line network.

The Simple Network Management Protocol (SNMP) is designed and ideally suited for network management. It allows the network manger to communicate with network devices via two mechanisms - "get" and "set." The "get" command allows the network manager to access certain variables from a network device. These variables could include the number of InOctets (bytes received), InOctetsErrors (number of bytes with errors), etc. The network manager can also set these variables from his or her own terminal.

By using these two commands, the manager is able to control the network. For example, if the manager determines that one workstation has crashed and needs to be rebooted, he can not simply issue a command to reboot that workstation. Instead, the network manager may "get" a variable form the workstation, such as a Time_to_Reboot and then "set" that variable to five seconds.

Personal Involvement

My internship was divided in two parts; a learning phase and a hands-on phase. The learning phase was very extensive and crucial one since I was a new comer to the field of communication networks. In order to familiarize myself with this area I read two textbooks, specifically, *Local and Metropolitan Area Networks* by William Stallings and *TCP/IP Illustrated Volume 1 The Protocols* by W. Richard Stevens. Two self paced courses at the Learning Center were directly related with my work. They are "The TCP/IP Protocol Suite" and "Understanding ATM in Corporate Networks." After completing these course I received certificates from The Learning Center. There were many tutorials and training sessions held by Computer Science Corporation (CSC), a NASA contractor, for their new employees and I was able to attend these. These tutorials covered both areas of communication networks (TCP/IP, ATM, Bridges & Routers, SNMP) as well as EBnet topics (Transport Design, ATM testing) and totaled over thirty hours. I also read from various standards including RFC[1] 1155, 1156, 1157, and 1180. Although I was not able to attend the Interop conference and their workshops, I was given a copy of the workbook associated with their tutorial and written by experts in the field.

Before working with any software I used Hewlett Packard's Network Node Manager, which implements SNMP Version 1, I read their user manual. This application gives the user an unconstrained view of the network to easily monitor and control it. The screen usually consists of different icons which represent different portions of the networks. The color of the icon indicates the status of that component. By double-clicking on a highlighted icon the network manager reaches a new layer of the network and a submap is shown on the screen. The manager repeats this process until the problem with the network is isolated. The network manager must determine the nature of the problem and try to address it. These steps are discussed in the following subsection, "Using

[1] Requests For Comments (RFC) are the written definitions of the protocols and policies of the Internet.

Hewlett Packard's Network Node Manager."

Since I was going to use Remedy's Action Request System (ARS) I read their users guide as well. ARS is known as "Trouble Ticketing Software." It is used to organize reports, or "tickets", that are created at remote locations when a network problem arises there. These reports are filled out by a person working at the remote location and there are different report forms based upon the nature of the problem. The tickets are useful to the network manager because he or she may organize and retrieve them from a database which works with ARS whenever there is a need. Typically, network managers retrieve old tickets to see how similar problems were resolved. Network managers also organize these tickets by separating unresolved problems from those that have been addressed. In our case the Sybase database was used in conjuction with ARS on a SUN workstation.

Using Hewlett Packard's Network Node Manager

Hands-on experience with HP's Network Node Manager supplemented my readings and allowed me to manage a simulated network. The basics of managing a network with this application are discussed in this section. As was previously mentioned the network manager needs to examine the network and find specifically where the problem occurs. Then the manager must determine the nature of the problem. Table I lists the most common network problems that a network manager encounters. There are three basic types of problems and these relate to network connectivity, network performance, and network service. Different procedures exist to deal with each problem.

Once I was to isolate and understand the nature of a problem I used some the Network Node Manager's tools to determine the cause of the problem. These tools included: ping, remote ping, telnet, ARP Cache, Locate Route, Monitor Traffic, and CPU load. Ping and remote ping are standard internet commands to send repeated signals to a

Table I *Diagnosing Network Problems*

Specific Problem	**Probable Causes**	**Nature of Problem**
In the past the user was able to contact particular system (for example, by **ftp**) and now the user can not reach the system	• Connection problem	•Network Connectivity
"Connection Timed Out" Error is received	• System is down • Routing Problem • Low Performance	• Network Connectivity • Network Performance
Remote system could not be found	• System is shut off • System no longer exists • System does not recognize host name • Gateway does not have remote system in routing table	• Network Connectivity
Slow system response	• Traffic overload • Overloaded device	• Network Performance
Loss of data during transmission	• Network overloaded • Device overloaded	• Network Performance
After connecting to system following command is not accepted	• Security Problem	• Network Service
After connecting to system following command is unrecognized	• Particular service is not installed/configured	• Network Service

device and assuming that device is on-line to receive signals, or echoes, back. Ping also lists the average time for all of these signals to come back to the user. Remote ping is the same except that in addition to choosing the target the user chooses which machine to initiate the ping from. Telnet is an internet command which allows the network manager to login remotely to a system. The network manager can also check the ARP (Address Resolution Protocol) Cache of a particular device to check for routing problems. The route taken by a message can be traced with the Locate Route command. Traffic over certain cable can be measured with the Monitor Traffic command. If a particular workstation is running very slowly and a network service problem is suspected the CPU load command is used to check if the particular workstation is overloaded. It was though the use of these tools that I was able to manage a simulated network.

Conclusion

Network management is a vital component of the EOS Backbone Network, which is needed to insure that the network will be running at peak performance. If the network is not managed it may go off line and NASA could temporarily lose control of expensive satellites and lose important data. During my internship, I was able to learn a great deal about the field of communication networks and specifically network management. My readings on this subject were supplemented by invaluable hands on experience in which I was able to use HP Network Node Manager and manage a simulated network.

Acknowledgments

Many people have helped me during the course of my internship. I would like to thank Dr. Joan Langdon for inviting me to participate in the SIECA (Summer Internship in Engineering and Computer Applications) program. I would also like to thank Vishal Desai, my mentor, for spending many hours with me to teach me the principles of communication networks and the EBnet project. I would also like to thank Charles Goldberg, John Steedman, and Brad Torain for their invaluable support throughout the program.

References

Case, Jeffery *Simple Network Management Protocol Workbook* . Interop Co. 1992

Hewlett Packard OpenView Network Node Manager *User's Guide* 1993

Remedy Corporation Action Request System *User's Guide for OSF/MOTIF* 1995

RFC 1155 "Structure and Identification of Management Information for TCP/IP-based Internets"

RFC 1156 "Management Information Base for Network management of TCP/IP based Internets"

RFC 1157 "A Simple Network Management Protocol"

RFC 1180 "A TCP/IP Tutorial"

Stallings, William *Local and Metropolitan Area Networks.* New York: Macmillan Co. 1990

Stevens, W. Richard *TCP/IP Illustrated Volume 1 The Protocols* . Massachusetts: Addison Wesley Publishing Co., 1994.

Image Processing Using Khoros 2.0 Scientific Software Development Environment

By Samone E. Jones
SIECA Graduate Summer Intern
Information Sciences and Technology Branch Code 935

Mentor
Patrick Coronado
Computer Engineer Program Manager
Information Sciences and Technology Branch Code 935

Summer Institute in Engineering and Computer Applications (SIECA) Project 1995

NASA Goddard Space Flight Center, Greenbelt, MD
August 1995

As a summer intern in the Summer Institute in Engineering and Computer Applications (SIECA) Program I had the opportunity of working in the Information Science and Technology Branch (ISTB) Code 935. ISTB's sole purpose is to conduct research that leads to the development of advanced information management and analysis systems that meet NASA's long term needs. Generally ISTB works with other organizations within the Space Data and Computing Division to provide assistance in fulfilling the NASA mission in the area of data and information management systems and techniques for the acquisition, storage, retrieval, manipulation, compression, display and analysis of scientific data and information.

My project assignment correlated with the Advanced Information Systems Program of ISTB. As a part of the Advanced Information Systems Program of ISTB I was assigned the responsibility of determining the full capabilities of Khoros 2.0 data processing capabilities - specifically in the application domain of image processing.

The motivation or objective of this project was to determine if the Khoros Software System could meet the needs of the Advanced Information System Program which include developing:

- Weather System Interfaces and
- Tools to display remotely sensed images and related scientific data

Khoros is a software integration and development environment that focuses on information processing and data exploration. It provides a Scientific Software Development Environment where a rich set of programs may be used for information processing, data exploration, and data visualization. Khoros is designed to shield the user of it's own complexity as well as the complexity of the UNIX and X Window System. Therefore the user is freed from worrying about underlying details.

My responsibilities in the Advanced Information System program of ISTB included determining the capabilities of Khoros 2.0 VIFF data format in order to:

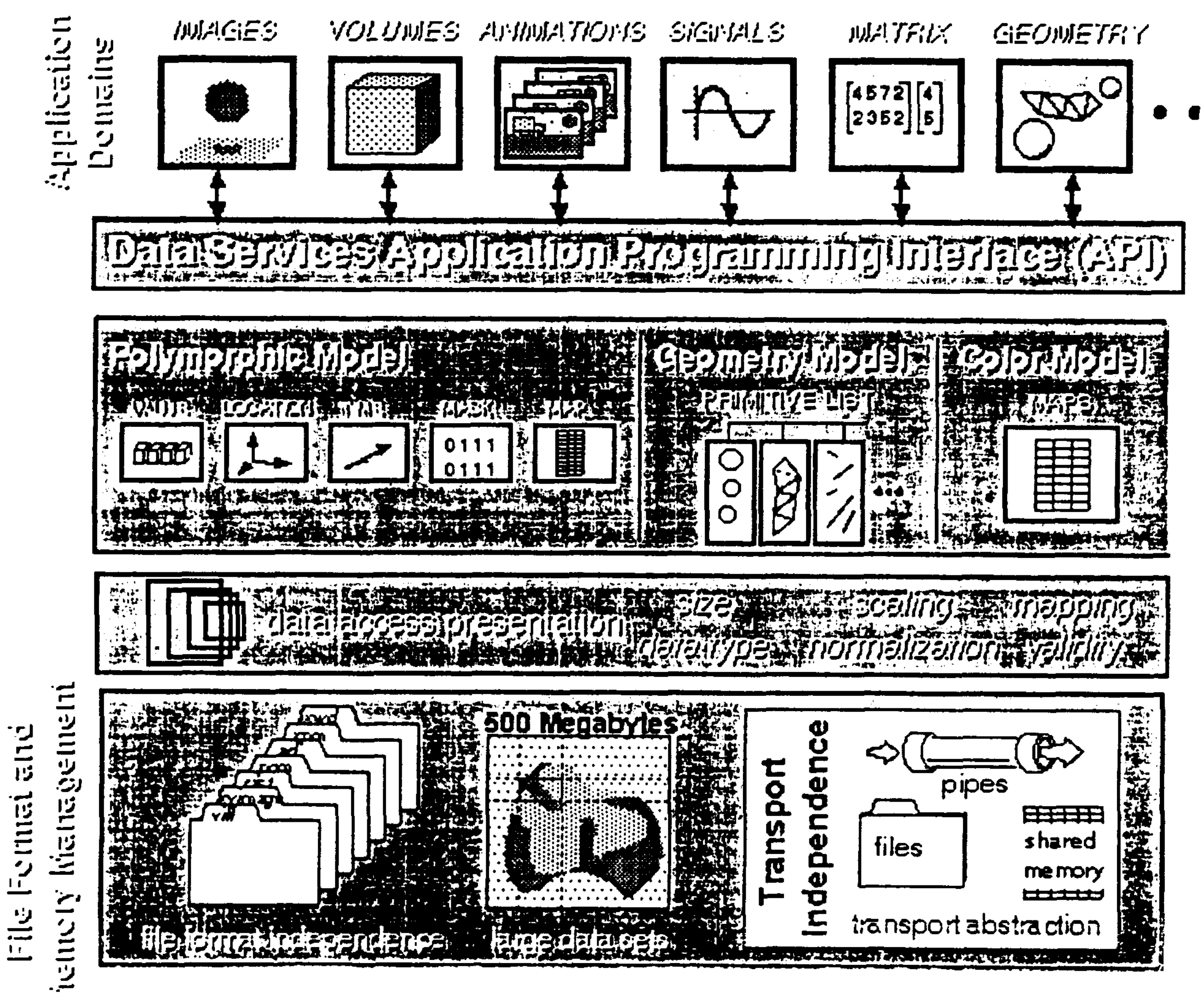
- calibrate science values corresponding to image values
- store navigation information in which earth locations corresponding to
- image pixels
- store information about multiple images in the same file and finally
- determining the xview capabilities of Khoros.

The Khoros software infrastructure consists of three major program services. All data processing and visualization routines in Khoros use the powerful functionality provided by these services.

- Foundation Services - Provide a portable system abstraction
- Data Services -Provide a powerful system for accessing and manipulating data.
- GUI and Visualization Services - Provide capabilities related to graphical display.

The upper level of data services is organized into a series of application-specific services, each with its own data model. Each data model covers the needs of either a specific domain or of a number of similar domains. Even though the data models of each application service are different, the underlying philosophy, design, and API of every service is similar.

There are currently three application data services: polymorphic services, geometry services, and color services. Polymorphic services is designed to cover the majority of application-domains; the polymorphic data model can store anything from signals to images and from animation to volumes. Geometry services is designed to cover the specific needs of the geometry domain; the geometry model provides a range of geometric primitives such as triangles and spheres, in addition to a number of volumetric primitives. Color services is an extension to polymorphic services with very specific functionality relating to colormaps.



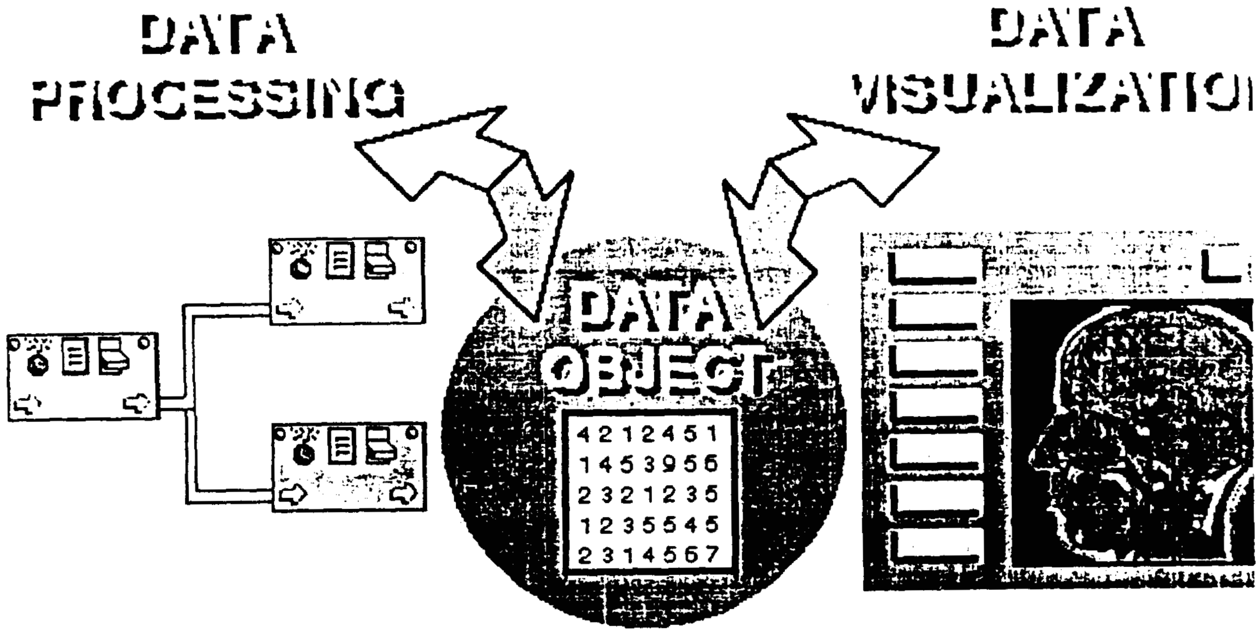


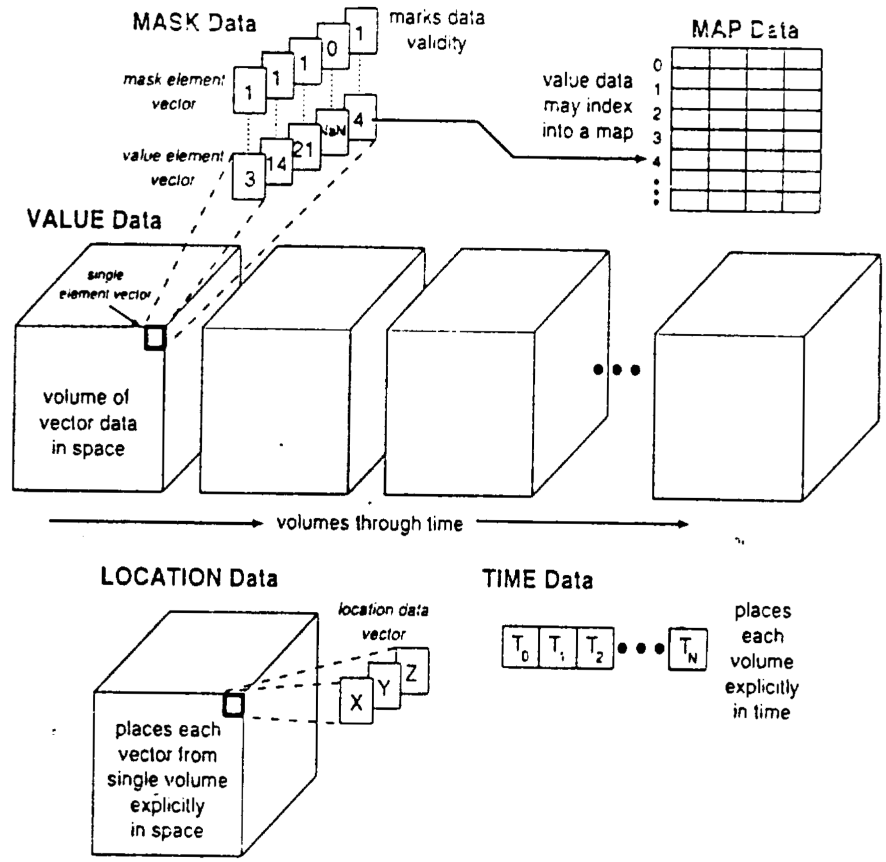
Figure 1: Data services implements a powerful and abstract data object. This data object is used by all Khoros data processing and data visualization programs.

Data Services implements a powerful and abstract data object. This data object is used by all Khoros data processing and data visualization programs.

The data services application programming interface (API) consists of a set of simple library functions which provide the user with access to an abstract data object. This API allows you to store and retrieve data from the data object and to access characteristics of the data without having to worry about complicated data structures or intricate file handling. This API encapsulates extensive functionality which efficiently handles the tedium of data access and presentation. This frees the user to concentrate on the details of implementing a specific algorithm, rather than worrying about how to access the data on which the algorithm is operating.

Many different application domains are able to utilize data services. Each domain performs all data access through the data services API. Data is interpreted according to the data model dictated by the domain. Data services has a series of data models available; each model is designed to meet the need of a single domain or family of domains. The most powerful of these is the polymorphic data model which provides consistent interpretation of data across many diverse domains. A geometry data model and a color data model are also available.

The Polymorphic Data Model



The polymorphic data model implemented by this service is designed to encompass many application domains. The polymorphic data model is based on the premise that data sets are usually acquired from or generated to model real-world phenomena. The polymorphic model thus consists of data which exists in three-dimensional space and one-dimensional time. You can picture the model most easily as a time-series of volumes in space. This time-series of volumes is represented by five different data segments. Each segment of data has a specific meaning dictating how it should be interpreted. Specifically, these five segments are value, location, time, mask, and map. All of these segments are optional; a data object may contain any combination of them and still conform to the polymorphic model.

The value segment is the primary data segment, consisting of data element vectors organized implicitly into a time-series of volumes. The value data may be given explicit positioning in space and time with the location and time segments. The remaining two segments are provided for convenience. The mask segment is used to mark the validity of each point of value data. The map segment is provided as an extension to the value data; the value data can be used to index into the map data. Figure 6 provides an overview of the polymorphic model. This model can be used to represent data for application domains as diverse as image processing. Therefore the polymorphic data model is ideal in meeting the needs for this project.

Out of the three application models, only the polymorphic data model has the file format support needed to accomplish the project goals. The Khoros 2.0 VIFF format is the only supported format which is capable of generally supporting all data segments and attributes.

The Application Programming Interface allows the programmer to manipulate an abstract data object which is used by all Khoros data processing and data visualization programs. Access to the object is done through a set of application specific function calls. These set of functions calls are divided into tow groups - primitives and attributes.

Primitives are used to access data within the data object. Data is stored into the object and retrieved from the object using put_data and get_data function calls. The primitive specified with each of these calls is what determines the amount of data being accessed as well as where in he overall data set that data is located. (What is Khoros ?WWW Page)

Attributes are used to access meta-data within the data object. Meta-data is a term used loosely to cover characteristics of the data such as size and data type and auxiliary information such as the date or a comment. Additionally, meta-data refers to presentation information such as scaling factor or normalization range. Attributes are assigned to and retrieved from an object using set_attribute and get_attribute function calls. Functions also exist for comparing attributes of two objects, for copying attributes from one object to another, and for printing attributes from an object. (What is Khoros ?WWW Page)

The primitives and attributes vary according to the data model that is being used. Each data model has its own set of primitives and attributes. The data format the polymorphic data model uses to process images and perform visualization programming is the VIFF data format. The primitive and attribute Khoros functions are used to manipulate, access, and process data using a data object.

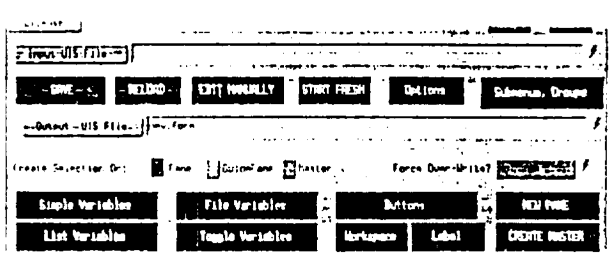
As an example of Khoros Code that consists of primitives and examples used to manipulate a image processing data object please see Appendix A of this paper.

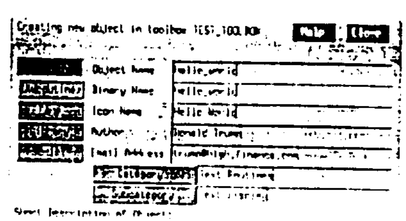
The heart of the Image Processing with Khoros 2.0 project was determining which of the data segments of the polymorphic model needed to be used in displaying a scientific image which correlated to scientific values such as temperature or earth location (longitude/latitude). The data segments of the polymorphic data model needed included:

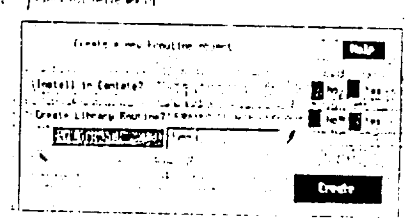
- the Value Data Segment
- the Mask Data Segment
- the Location Data Segment
- and the Time Data Segment.

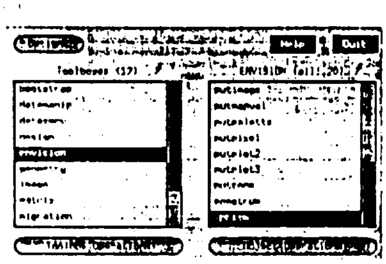
The image data is stored as the value segment data, the mask data segment simply identifies erroneous data and the location and time data segments give the value data explicit location in time.

The crafstman application of the Khoros Software system is used to manipulate files and data formats through the use of Toolbox Operations. A Toolbox is a collection of programs and libraries that are managed as a single entity to develop applications Once a Toolbox is created to manipulate the data object, the Khoros software tool Composer is used to edit the software object created by the Craftsman Toolbox. Finally, the programmer has the option of developing a Graphical User Interface for the software object (GUI) so that it can act as a stand-alone application for the specific task the programmer programmed it to perform.









In conclusion, the Khoros 2.0 VIFF Data Format has been found to have the capability to suit many application domains including many areas in image processing. These areas include storing and displaying scientific values that correlate to image values, storing and displaying overlay information, and storing information about multiple images.

Not only was the Khoros VIFF data format effective in meeting the needs of the Advanced Information Systems Program project of ISTB, but it was also found to be semi-user friendly in that it hid a lot of unnecessary details the user has no need to be aware of in visualization programming.

As an extension of this project, it would be beneficial for ISTB to develop GUISE interfaces to calibrate information for scientific values that correlate to image values and image pixels, support navigation information, hold overlay information for multiple depth images, and hold information about multiple images in the same file.

```
 * Khoros: $Id$
 */

#if !defined(__lint) && !defined(__CODECENTER__)
static char rcsid[] = "Khoros: $Id$";
#endif

 * $Log$
 */

/*
 * Copyright (C) 1993, 1994, 1995, Khoral Research, Inc., ("KRI").
 * All rights reserved.  See $BOOTSTRAP/repos/license/License or run klicense.
 */

/* >>>>>>>>>>>>>>>>>>>>>>>>>>>>> <<<<<<<<<<<<<<<<<<<<<<<<<<
   >>>>
   >>>>         Main program for prob_neural_net
   >>>>
   >>>>  Private:
   >>>>         main
   >>>>
   >>>>   Static:
   >>>>   Public:
   >>>>
   >>>>>>>>>>>>>>>>>>>>>>>>>>>>> <<<<<<<<<<<<<<<<<<<<<<<<<< */

#include "prob_neural_net.h"

clui_info_struct *clui_info = NULL;

/*-----------------------------------------------------------

  Routine Name: main() - Do Non Parametric Bayes Classification

       Purpose: main program for prob_neural_net

         Input:
               char *clui_info->i1_file; {input file}
               int   clui_info->i1_flag; {TRUE if -i1 specified}

               char *clui_info->i2_file; {input file}
               int   clui_info->i2_flag; {TRUE if -i2 specified}

               char *clui_info->i3_file; {input file}
```

```
|               int    clui_info->i3_flag; {TRUE if -i3 specified}
|
|        Output:
|       Returns:
|
|    Written By:
|          Date: Aug 16, 1995
| Modifications:
|
-------------------------------------------------------------*/

void main(
   int  argc,
   char **argv,
   char **envp)
{

        kform  *pane;         /* form tree representing *.pane file */
/* -main_variable_list */
        kobject bilp_file = NULL;
        kobject training_file = NULL;
        kobject sigma_file = NULL;
/* -main_variable_list_end */

        khoros_initialize(argc, argv, envp, "CLASSIFIERS");
        kexit_handler(prob_neural_net_free_args, NULL);

/* -main_get_args_call */
        pane = kgen_initialize(PANEPATH, KGEN_KROUTINE, "CLASSIFIERS", "prob_neu
                prob_neural_net_usage_additions);

        if (!(kclui_check_args()))
            kexit(KEXIT_FAILURE);
        prob_neural_net_get_args(pane);
        kvf_destroy_form(pane);
/* -main_get_args_call_end */

/* -main_before_lib_call */

/* Open the input files and do some error checking */

        if ((bilp_file = kpds_open_input_object(clui_info->i1_file))
                == KOBJECT_INVALID)
        {
          kerror("prob_neural_net","main","Cannot open BILP file");
          kexit(KEXIT_FAILURE);
        }

        if ((training_file = kpds_open_input_object(clui_info->i2_file))
                == KOBJECT_INVALID)
        {
          kerror("prob_neural_net","main","Cannot open training file");
          kexit(KEXIT_FAILURE);
        }

        if ((sigma_file = kpds_open_input_object(clui_info->i3_file))
                == KOBJECT_INVALID)
        {
          kerror("prob_neural_net","main","Cannot open sigma file");
          kexit(KEXIT_FAILURE);
```

```
        }

/* Do the PNN */

        do_pnn(bilp_file,training_file,sigma_file);

/* -main_before_lib_call_end */

/* -main_library_call */
/* -main_library_call_end */

/* -main_after_lib_call */
/* -main_after_lib_call_end */


        kexit(KEXIT_SUCCESS);
}


/*-----------------------------------------------------------
|
|  Routine Name: prob_neural_net_usage_additions
|
|       Purpose: Prints usage additions in prob_neural_net_usage routine
|
|         Input: None
|
|        Output: None
|    Written By: ghostwriter -oname prob_neural_net
|          Date: Aug 16, 1995
| Modifications:
|
------------------------------------------------------------*/
void prob_neural_net_usage_additions(void)
{
        kfprintf(kstderr, "\tDo Non Parametric Bayes Classification\n");

/* -usage_additions */
/* -usage_additions_end */

}
/*-----------------------------------------------------------
|
|  Routine Name: prob_neural_net_free_args
|
|       Purpose: Frees CLUI struct allocated in prob_neural_net_get_args()
|
|         Input: None
|
|        Output: None
|    Written By: ghostwriter -oname prob_neural_net
|          Date: Aug 16, 1995
| Modifications:
|
------------------------------------------------------------*/
/* ARGSUSED */
void
prob_neural_net_free_args(
    int   status,
```

```
    kaddr client_data)
{

        /* do the wild and free thing */
        if (clui_info != NULL)
                {
        kfree(clui_info->i1_file);
        kfree(clui_info->i2_file);
        kfree(clui_info->i3_file);
                kfree(clui_info);
                }

/* -free_handler_additions */
/* -free_handler_additions_end */
}

/TopOfPage 10.75 72 mul def
/LeftMargin .375 72 mul def
%%
90 rotate
18 -850 translate
%%
/Times-Italic findfont
100 scalefont
setfont
LeftMargin TopOfPage 100 sub moveto
(SJONES) show
LeftMargin currentpoint exch pop moveto
currentpoint 85 sub moveto
%%
/Times-Roman findfont
18 scalefont
setfont
(User: ) show
LeftMargin currentpoint exch pop moveto
currentpoint 22 sub moveto
(Request id: postscript-148    Printer: postscript) show
LeftMargin currentpoint exch pop moveto
currentpoint 22 sub moveto
(Options: ) show
LeftMargin currentpoint exch pop moveto
currentpoint 22 sub moveto
(Date: Fri Aug 18 08:56:30 EDT 1995) show
LeftMargin currentpoint exch pop moveto
currentpoint 125 sub moveto
%%
/Times-Italic findfont
75 scalefont
setfont
() show
%%
/Courier findfont
10 scalefont
setfont
%%
showpage
From chettri@fireeater-f.gsfc.nasa.gov Thu Aug 17 16:20 EDT 1995
Received: from fireeater.gsfc.nasa.gov by danville.gsfc.nasa.gov with SMTP
        (1.38.193.4/16.2) id AA03811; Thu, 17 Aug 1995 16:20:34 -0400
Return-Path: <chettri@fireeater-f.gsfc.nasa.gov>
```

```
Received: by fireeater-f.gsfc.nasa.gov; id AA01704; Thu, 17 Aug 1995 16:29:47 -0
Date: Thu, 17 Aug 1995 16:29:47 -0400
From: Samir Chettri <chettri@fireeater-f.gsfc.nasa.gov>
Message-Id: <9508172029.AA01704@fireeater-f.gsfc.nasa.gov>
T   sjones@danville.gsfc.nasa.gov
Subject: pnn.c
Status: RO

/*
 * Khoros: $Id$
 */

#if !defined(__lint) && !defined(__CODECENTER__)
static char rcsid[] = "Khoros: $Id$";
#endif

/*
 * $Log$
 */

/*
 * Copyright (C) 1993, 1994, 1995, Khoral Research, Inc., ("KRI").
 * All rights reserved.  See $BOOTSTRAP/repos/license/License or run klicense.
 */


/* >>>>>>>>>>>>>>>>>>>>>>>>>>>>> <<<<<<<<<<<<<<<<<<<<<<<<<<
   >>>>
   >>>>                File Title
   >>>>
   >>>>  Static:
   >>>>                 _static_routines()
   >>>>  Public:
   >>>>                 public_routines()
   >>>>
   >>>>>>>>>>>>>>>>>>>>>>>>>>>>> <<<<<<<<<<<<<<<<<<<<<<<<<< */

#include "prob_neural_net.h"


/*-----------------------------------------------------------------
|
|  Routine Name: function_name - short description of function
|
|       Purpose: This should be a complete description that anyone
|                could understand;  it should have acceptible grammar
|                and korrect spelling.
|
|         Input: kobject bil_file - The object containing the image in
|                                  BILP (Band InterLeaved by Pixel)
|                                  format. For doing classification,
|                                  this is the best format for fastest
|                                  access to the data for classification.
|                                  Data is stored by value using time
|                                  stamps.
|
|                kobject trai_file - The object containing the training
|                                  data. Data format is BILP with a trailing
|                                  integer value that contains the class to
|                                  which the pixel belongs.
```

```
                 kobject sigma_file - The object containing the sigma values.
                                  The sigma values control the spread of
                                  the Gaussian that is used in the PNN.
                                  The tail of this object contains 3 values
                                  the the following order.

                                    1) NROWS  Number of rows in bil_file
                                    2) NCOLS  Number of cols in bil_file
                                    3) NBAND  Number of bands in bil_file

                                  The reason for this peculiar choice is
                                  because we are storing the data in the
                                  bil_file along the time axis of the viff
                                  format (see Programming Services Vol II
                                  Chapter 1, page 1-10.) and therefore
                                  there is no way to obtain the number of
                                  rows, columns and bands in the image.

         Output: argument4 - explanation
                 argument5 - explanation
        Returns: TRUE (1) on success, FALSE (0) otherwise

     Written By: Chump E. Coyote.
           Date:
  Modifications:

-------------------------------------------------------------*/
do_pnn(kobject bil_file, kobject trai_file, kobject sigma_file)
{

   klist        *objlist=NULL;

/*
   Width, height, depth, time, elements for bil, training and sigma files
*/

   int          w_bil,w_trai,w_sigma,
                h_bil,h_trai,h_sigma,
                d_bil,d_trai,d_sigma,
                t_bil,t_trai,t_sigma,
                e_bil,e_trai,e_sigma;

   int          size;               /* = w_bil*h_bil*d_bil*t_bil*e_bil */
   int          sel_wid;            /* selected width to get data from */
   int          i,k;                /* width, height indices */
   int          vec_siz,num_vec;    /* Not sure what these are */

   unsigned short                   /* Bufs. for image, trai and sigma */
                *bil_buf,tmp,
                *trai_buf,
                *sigma_buf;

   unsigned int nband,nrows,ncols,nclass; /* Number of bands, rows, cols and
                                             classes */

   int          itype;              /* Data input type */

   char         *lib = "kdatamanip";
   char         temp[KLENGTH];
```

```
/* Check if value data exists */

   .f (!kpds_query_value(bil_file))
   {
     kfprintf(kstderr,"No value data in bil_file to operate on.\n");
     return(FALSE);
   }

   if (!kpds_query_value(trai_file))
   {
     kfprintf(kstderr,"No value data in trai_file to operate on.\n");
     return(FALSE);
   }

   if (!kpds_query_value(sigma_file))
   {
     kfprintf(kstderr,"No value data in sigma_file to operate on.\n");
     return(FALSE);
   }

/* Reference object. Supposedly used to avoid "side effects" *

   bil_file = kpds_reference_object(bil_file));
   objlist = klist_add(objlist,bil_file,"KOBJECT");

   trai_file = kpds_reference_object(trai_file));
   objlist = klist_add(objlist,trai_file,"KOBJECT");

   sigma_file = kpds_reference_object(sigma_file));
   objlist = klist_add(objlist,sigma_file,"KOBJECT");

/* How large is the data? */

   kpds_get_attribute(bil_file,KPDS_VALUE_SIZE,&w_bil,&h_bil,&d_bil,&t_bil,&e_bi
   kfprintf(kstderr, "BIL FILE: %d %d %d %d %d\n",w_bil,h_bil,d_bil,t_bil,e_bil)

   kpds_get_attribute(trai_file,KPDS_VALUE_SIZE,&w_trai,&h_trai,&d_trai,&t_trai,
   kfprintf(kstderr, "TRAI FILE: %d %d %d %d %d\n",w_trai,h_trai,d_trai,t_trai,e

   kpds_get_attribute(sigma_file,KPDS_VALUE_SIZE,&w_sigma,&h_sigma,&d_sigma,&t_s
   kfprintf(kstderr, "SIGMA FILE: %d %d %d %d %d\n",w_sigma,h_sigma,d_sigma,t_si

/* Get data type */

   kpds_get_attribute(bil_file,KPDS_VALUE_DATA_TYPE,&itype);
   kfprintf(kstderr, "%d = DATA TYPE\n",itype);
   kfprintf(kstderr, "%d %d %d %d %d %d %d %d %d\n",KBIT,KBYTE,KUBYTE,KSHORT,KUS

   kpds_get_attribute(trai_file,KPDS_VALUE_DATA_TYPE,&itype);
   kfprintf(kstderr, "%d = DATA TYPE\n",itype);
   kfprintf(kstderr, "%d %d %d %d %d %d %d %d %d\n",KBIT,KBYTE,KUBYTE,KSHORT,KUS

   kpds_get_attribute(sigma_file,KPDS_VALUE_DATA_TYPE,&itype);
   kfprintf(kstderr, "%d = DATA TYPE\n",itype);
   kfprintf(kstderr, "%d %d %d %d %d %d %d %d %d\n",KBIT,KBYTE,KUBYTE,KSHORT,KUS

/* Get the data */
```

```
/*=================== bil file ==================*/

   size = w_bil*h_bil*d_bil*t_bil*e_bil;
   bil_buf = (unsigned short *)kmalloc(size*sizeof(unsigned short));

   kpds_set_attribute(bil_file,KPDS_VALUE_REGION_SIZE,
                            w_bil,h_bil,d_bil,t_bil,e_bil);

   kpds_get_data(bil_file,KPDS_VALUE_REGION,(kaddr)bil_buf);

/*=================== trai file ==================*/

   size = w_trai*h_trai*d_trai*t_trai*e_trai;
   trai_buf = (unsigned short *)kmalloc(size*sizeof(unsigned short));

   kpds_set_attribute(trai_file,KPDS_VALUE_REGION_SIZE,
                            w_trai,h_trai,d_trai,t_trai,e_trai);

   kpds_get_data(trai_file,KPDS_VALUE_REGION,(kaddr)trai_buf);

/*=================== sigma file ==================*/

   size = w_sigma*h_sigma*d_sigma*t_sigma*e_sigma;
   sigma_buf = (unsigned short *)kmalloc(size*sizeof(unsigned short));

   kpds_set_attribute(sigma_file,KPDS_VALUE_REGION_SIZE,
                            w_sigma,h_sigma,d_sigma,t_sigma,e_sigma);

   kpds_get_data(sigma_file,KPDS_VALUE_REGION,(kaddr)sigma_buf);

/
   Last 3 values in sigma_buf[] are (in backward order),
     o number of image bands
     o number of image rows
     o number of image cols
   Do some checking to see if these values are ok. If not, print
   error message.
*/

   nband  = sigma_buf[size-1];
   nrows  = sigma_buf[size-2];
   ncols  = sigma_buf[size-3];
   nclass = size - 3;

   size = nband*nrows*ncols;

   if (size != w_bil*h_bil*d_bil*t_bil*e_bil)
   {
     kfprintf(kstderr, "Image size doesn't match. Check BIL or sigma file\n");
     kexit(KEXIT_FAILURE);
   }

 /* Do Processing */

   for (i=0; i<w_sigma*h_sigma*d_sigma*t_sigma*e_sigma; i++)
   {
     tmp = sigma_buf[i];
     kfprintf(kstderr, "%d\n",tmp);
   }
```

The Effect Of The Diurnal Cycle On Precipitation Rates In The Toga Coare IOP

Prem Lall
SIECA Program
Dr. C-H Sui

During my last internship at Goddard, I created a series of computer programs that analyzed radar data to compute parameters for various meteorological experiments. Since that time, my programs have been used to evaluate data collected in the Toga Coare Intensive Operation Period (see fig 1). The resulting information was then used to calculate precipitation rates in certain regions of the Pacific Ocean.

This year, my project involved using these figures to gather information about the causes of rain activity in tropical climates. Specifically, it was my objective to determine the relationship between rainfall rates and the Diurnal Cycle. This cycle is an ongoing process in nature that is driven by subtle changes atmospheric pressure and temperature. However, the influence that this cycle has on the rain activity can be difficult to isolate, because there are many factors that can affect the weather.

The initial stages of my project involved analyzing radar data gathered during a series of periodic scans. Figure 2 contains an example of rainfall activity that was recorded during a particular scan. This "rainmap" is a pictorial representation of weather activity that occurred at a specific time in the Toga Coare experiment. Since our data was collected aboard two ships, this image was actually created from two separate, overlapping scans. Images like this are important to study because they can often reveal features about a storm that are difficult to detect. The different degrees of shading on this map represent areas where rain was falling at various rates. For example, the darker regions indicate that the rainfall is very intense. Therefore, upon close examination one can easily see that it is raining at different rates in different areas of our map. However, since rainfall rates are constantly changing, it is often difficult to know how hard it is raining in a particular place at a particular point in time.(especially in tropical climates, where the weather is at times, very erratic). In other words, because the rainfall is so sporadic, it is very hard to know what will happen from moment to moment. Therefore, it is often more useful to use statistical approximations in our calculations instead of the actual raw data (You might not know exactly how hard its raining in a certain place, but from the available data, you can get a reasonably good estimate). Using these values sometimes allows us to see features of a storm that are hidden or obscured. In addition they often provide more accurate results when we are trying to discover what is occurring over a long period of time.

Therefore, by using certain statistical equations we were able to determine the probability that it was raining at a certain rate, during a certain time period. Once these values were calculated, we were able to computed an individual probability distribution for each scan. Figure 3 contains a graph whose values were derived from raw data depicted in our rainmap. The horizontal axis of this graph contains rain rate values, while the vertical one is comprised of probability values. Notice that a maximum value occurs when it is raining at about 0.05 mm/min (this is quite logical because the shading that represents this number tends to dominate our rainmap). But just as before, this distribution only provides information about one particular scan. In other words, it only tells us about a single moment in time.

In order to see what was going on over a long period of time, we needed to combine the information obtained from many different distributions. Once this was accomplished, we were able to generate what we call a "time series" (see figure 4). This can be valuable source of information because it provides us with a graphical representation of the weather activity over time. Since we had to take time into consideration, we now had to deal with three parameters instead of two. The scale at the bottom of our time series is used to define the various probabilities. Even at this point of our examination, we were already able to detect evidence of a reoccurring cycle taking place. Our next task was to try to uncover the mechanism responsible for producing this pattern (empirical observation alone is not enough to determine the cause of this effect).

Once the data was organized in this way, the relationship between rainfall and the Diurnal Cycle could be examined more effectively. Since this cycle is affected by changes in temperature and pressure, it's behavior is closely linked to the period of the sun (as the sun rises and sets, heat is transferred to the ocean and atmosphere). This type of heat exchange is a crucial component of the Diurnal Cycle. Therefore in order to determine the effect that this process has on the weather, I first needed to see if there was a relationship between rain rates and temperature variations in the surrounding environment. Thus, it was important to examine any process that involved any sort of heat transfer. In order to gain a better understanding of this connection, we wanted to focus on 2 periods that have very different characteristics. (These periods are known as Disturbed and Undisturbed periods respectively).In this way, we hoped to learn how the Diurnal Cycle contributes to weather activity under different circumstances. Also, we suspected that the Diurnal mechanisms that operated during these periods were quite different.

Using additional data gathered during the Toga Coare experiment, we were able to construct graphs of the sea-surface temperature (see fig. 5). These graphs were most important because they served a dual purpose; they provided us with information about heat transfer to the ocean, and allowed us to easily differentiate between our two periods.

By examining changes in sea-surface temperature (abbreviated by SST) readings, I hoped to establish a correlation between variations in the Diurnal Cycle and periods of heavy rainfall. The portion of our graph on the right corresponds to an Undisturbed period. This period is characterized by very sparse cloud cover; thus, the temperature of the ocean gets quite warm, because solar radiation is able to penetrate freely. Therefore when the sun approaches it's zenith, the SST tends to be quite high because a great deal of heat is absorbed by the ocean (this usually occurs around noontime). As the sun goes down however, the temperature sharply declines. These type of temperature fluctuations occur daily during periods like this. (notice that as we approach the middle of the day, the amplitude of the curve gets pretty high). However, things are quite different during a Disturbed period . During this period, the cloud cover is extremely dense, making it very hard for the solar radiation to penetrate. Therefore, unlike the previous case, the SST does not vary much; In fact, it steadily declines as time goes on (see the middle of figure 5). Both of these periods could be easily identified by examining this graph.

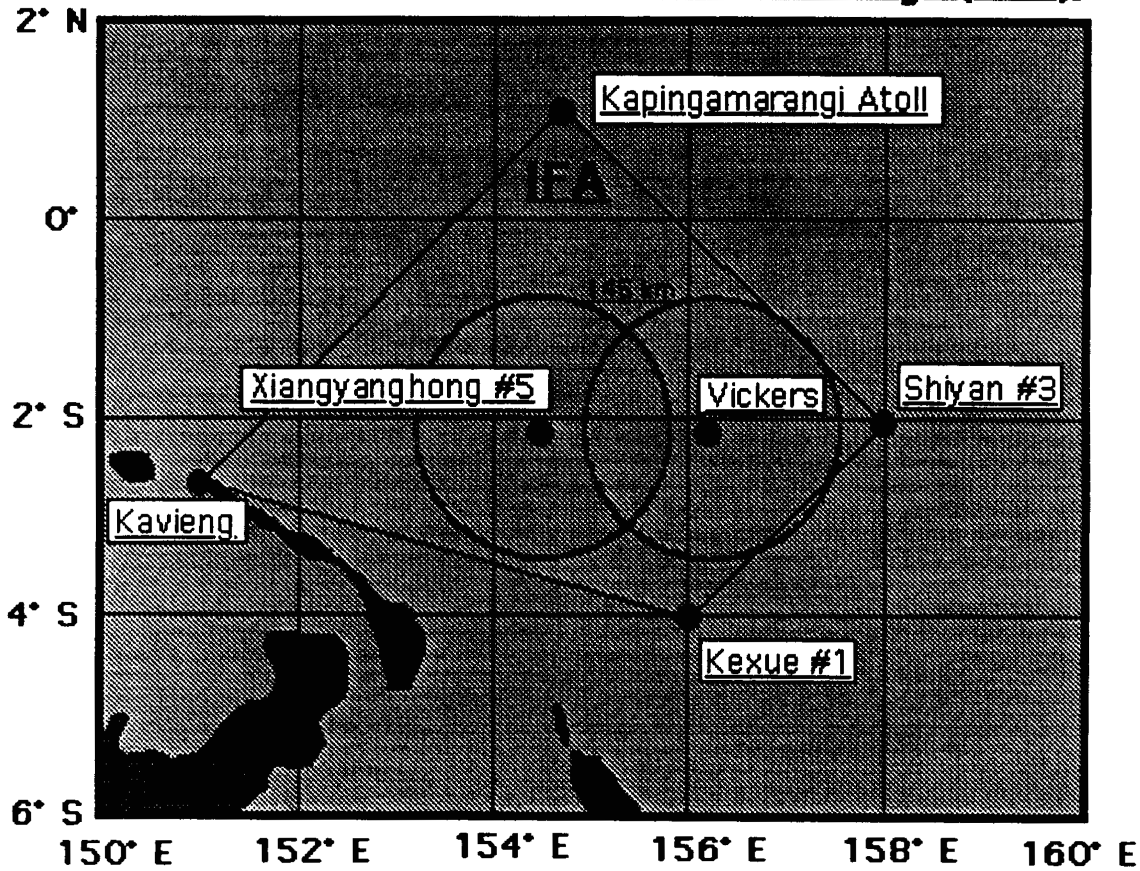
Figure 6 contains two graphs that were composed from information gathered during an Undisturbed Period. The upper portion is a time series comprised of rainfall data, while the lower section contains corresponding SST readings. A cursory observation of both graphs reveals that there is evidence of some sort of rhythmic cycle taking place. Because the heat penetrates so freely, the ocean temperature is greatly influenced by the sun. One can see however, that there is a corresponding rise in the rain activity approximately 3-4 hour after the SST reaches a peak, (the darker areas on the time series correspond to higher probability values). This is due to the fact that, as the SST declines, heat is transferred from the ocean to the atmosphere . This exchange causes convective air currents to develop in the atmosphere that intensify the storm activity. This process of heat transfer between the ocean and the atmosphere is a major component of the Diurnal Cycle during this period. The fact that this entire process reoccurs on a regular basis (almost daily) seems to reinforces our hypothesis that the Diurnal Cycle is playing a part in the weather activity we are observing.

However as stated before, weather conditions are quite different during a Disturbed Period. Rain is falling continuously during this period, because the clouds are so thick. As a result, the ocean is not able to gain much heat. In fact, a gradual decline in temperature can be observed because there is little incoming radiation to heat up the water (see bottom of fig. 7). Therefore we can not rely on SST measurements to establish a connection with the Diurnal Cycle. (because there is little variation in the SST, there is no reoccurring pattern to refer to). However, although the sun did not warm up the sea during this period, it did seem to have some effect nevertheless. By examining our time series we were able to determine that the storm activity tended to subside somewhat at certain times of the day (see top of fig. 7). This phenomenon usually occurred around 12 o'clock each day. Although this effect was not as pronounced as the pattern we observed during the Undisturbed Period, it existence seem to suggest that some sort cycle was taking place. The fact that this pattern could be linked to the solar cycle suggested that it was being influenced by some sort of heat transfer process. After examining the available evidence we were able to conclude that when the sun reached it's high point, it's heat caused the clouds in the upper atmosphere to dissipate. When this happened, heat was transferred to the atmosphere, causing the storm activity to subside.

During both periods, we were able to see a definite correlation between the observed weather activity and the Diurnal Cycle. And, although the mechanisms that influenced the rainfall were somewhat different, in each of these cases, heat transfer played an essential part . For instance, during the Undisturbed Period were able to conclude that there was a strong correlation between variations in the SST and the rainfall. In the Disturbed Period however, the effect of the cycle manifested itself by causing the clouds in the upper atmosphere to disperse.

In addition to these observations, we were also able to detect a reoccurring pattern of nocturnal storm activity. This was not entirely unexpected because the convective currents that drive the storm tended to be quite high during the evening. Also since the temperature is much lower at this time of day (due to the absence of the sun), the atmosphere is not able to retain moisture as effectively. The existence of this pattern reinforces our assumption that the Diurnal Cycle is playing a part here. It is also worth mentioning that the effect of the Diurnal Cycle is often difficult to detect. This is due to the fact that there are so many other factors that can affect the weather. Although it's connection to rainfall was quite evident in the cases presented here, it's influence is sometimes obscured by other phenomenon.

TOGA COARE Intensive Flux Array (IFA)



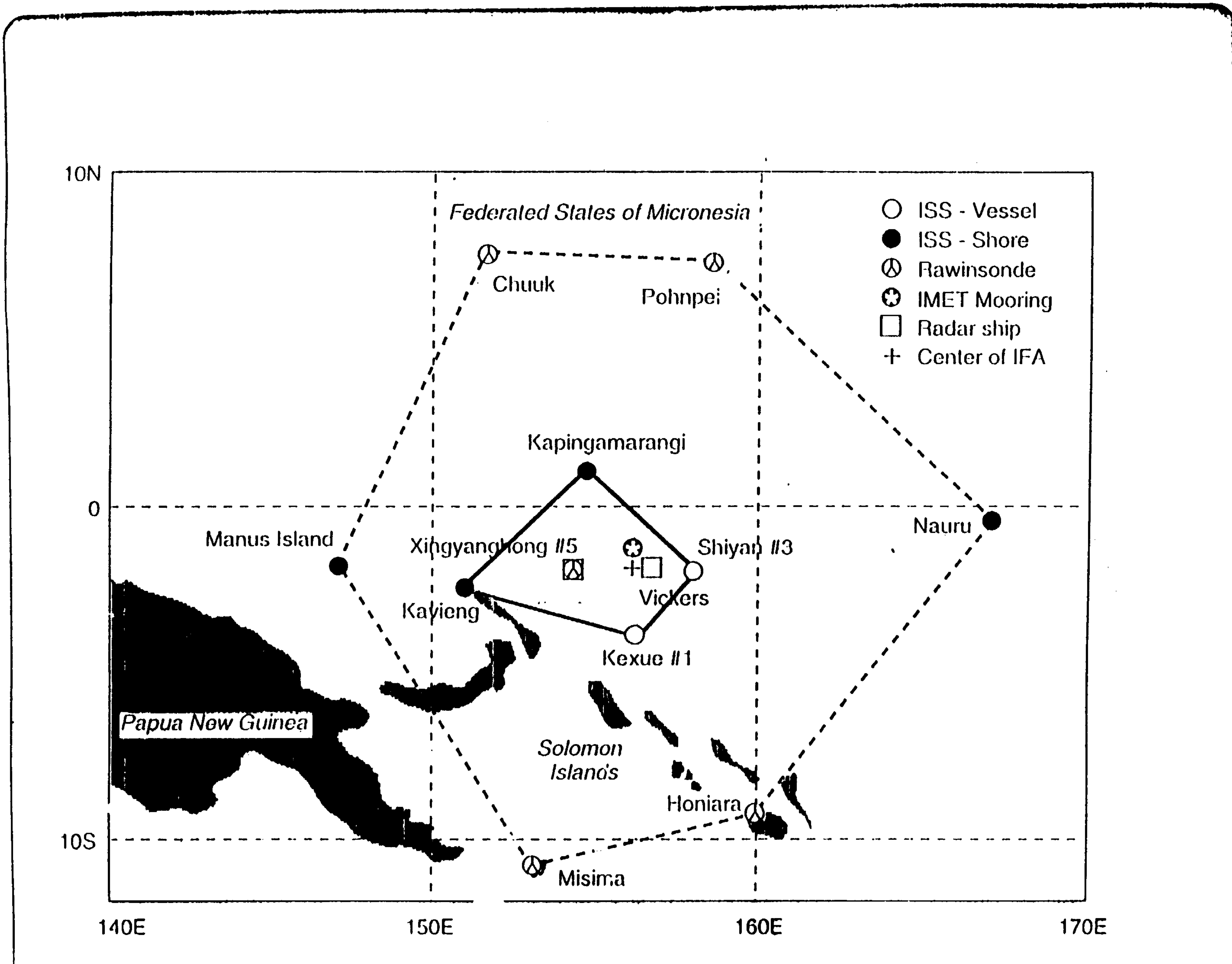
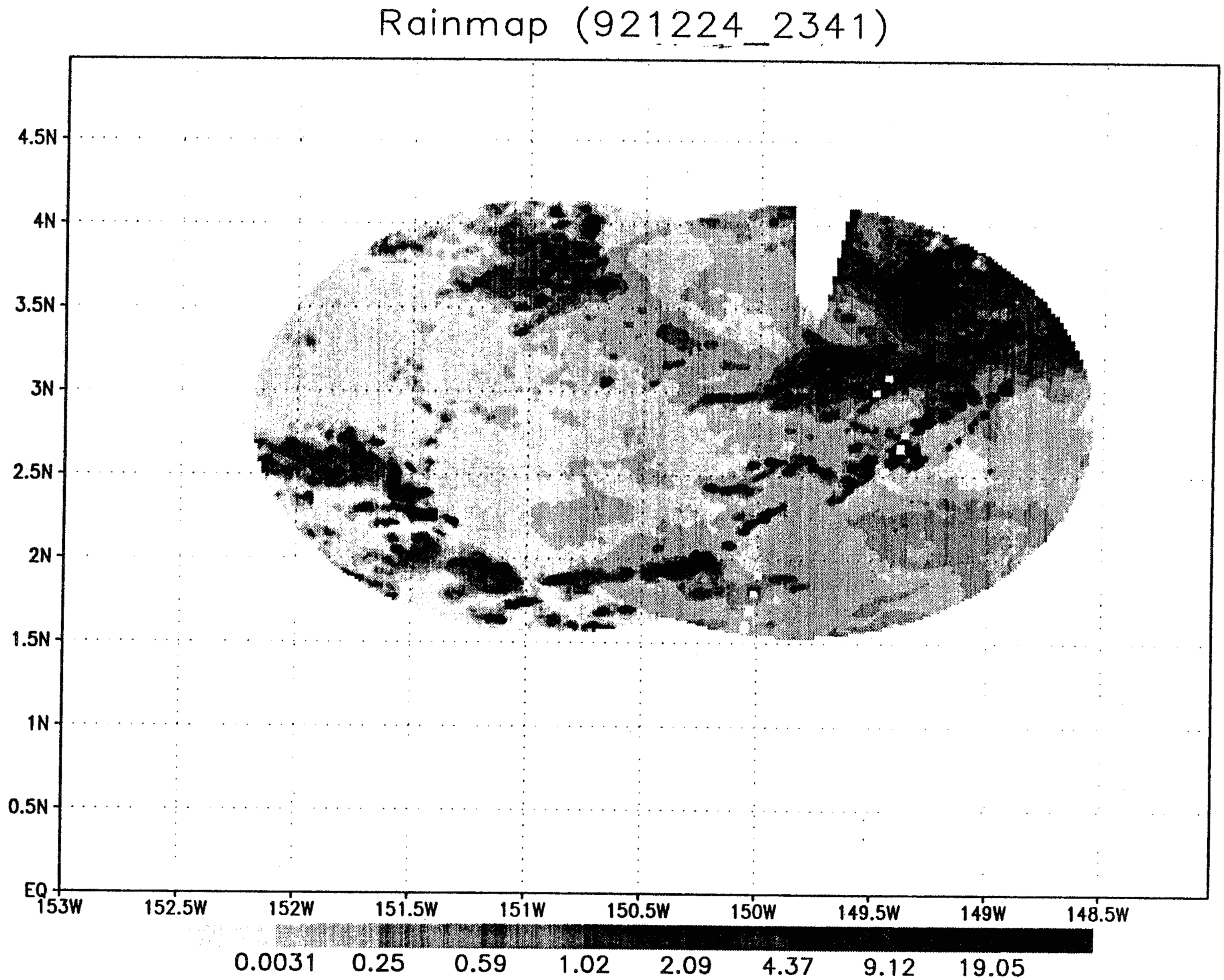

10N
0
10S
140E
150E
160E
170E
Federated States of Micronesia
ISS - Vessel
ISS - Shore
Rawinsonde
IMET Mooring
Radar ship
Center of IFA
Chuuk
Pohnpei
Kapingamarangi
Manus Island
Xingyanghong #5
Shiyan #3
Nauru
Kavieng
Vickers
Kexue #1
Papua New Guinea
Solomon Islands
Honiara
Misima

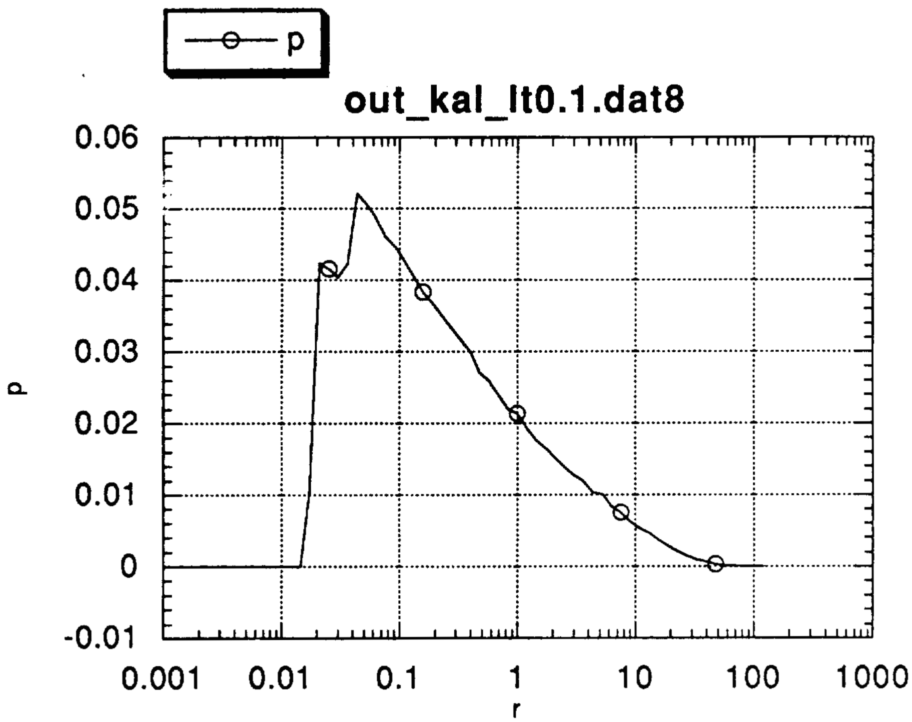
FIGURE 2



Rainmap (921224_2341)
4.5N
4N
3.5N
3N
2.5N
2N
1.5N
1N
0.5N
EQ
153W
152.5W
152W
151.5W
151W
150.5W
150W
149.5W
149W
148.5W
0.0031
0.25
0.59
1.02
2.09
4.37
9.12
19.05

FIGURE 3

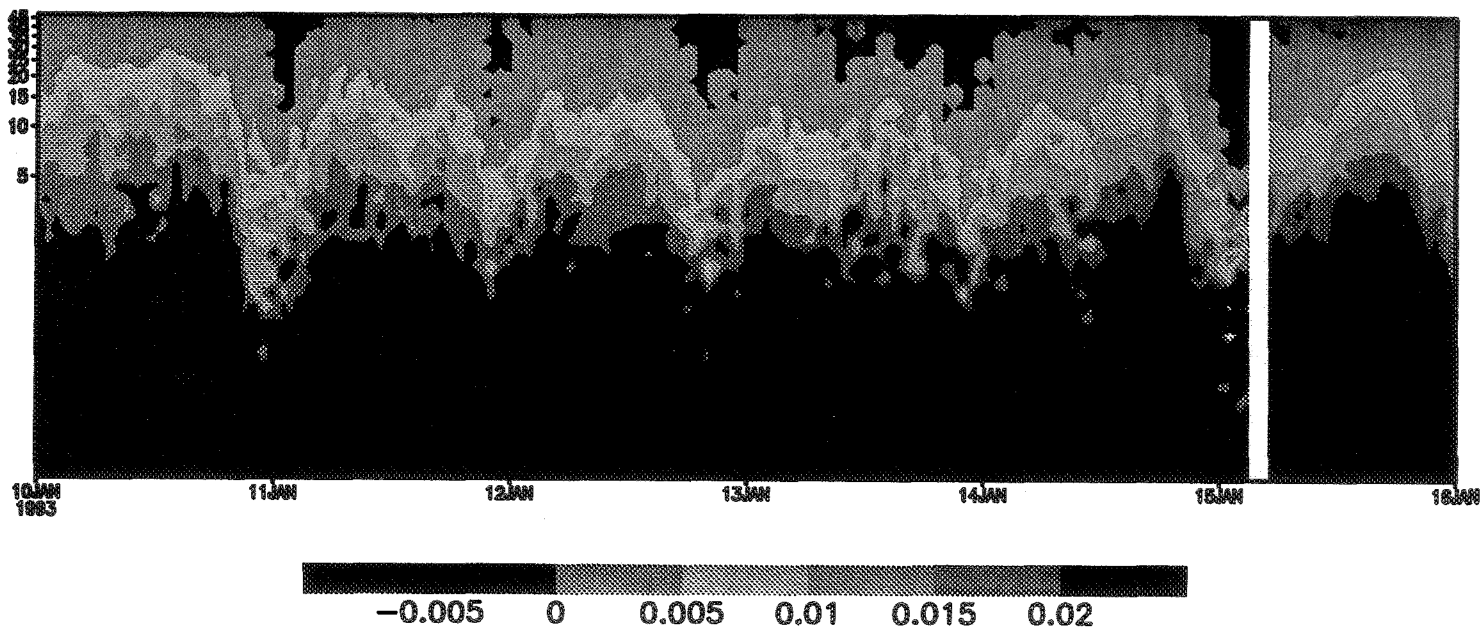
< 0.1



p
out_kal_lt0.1.dat8
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
p
0.001
0.01
0.1
1
10
100
1000
r

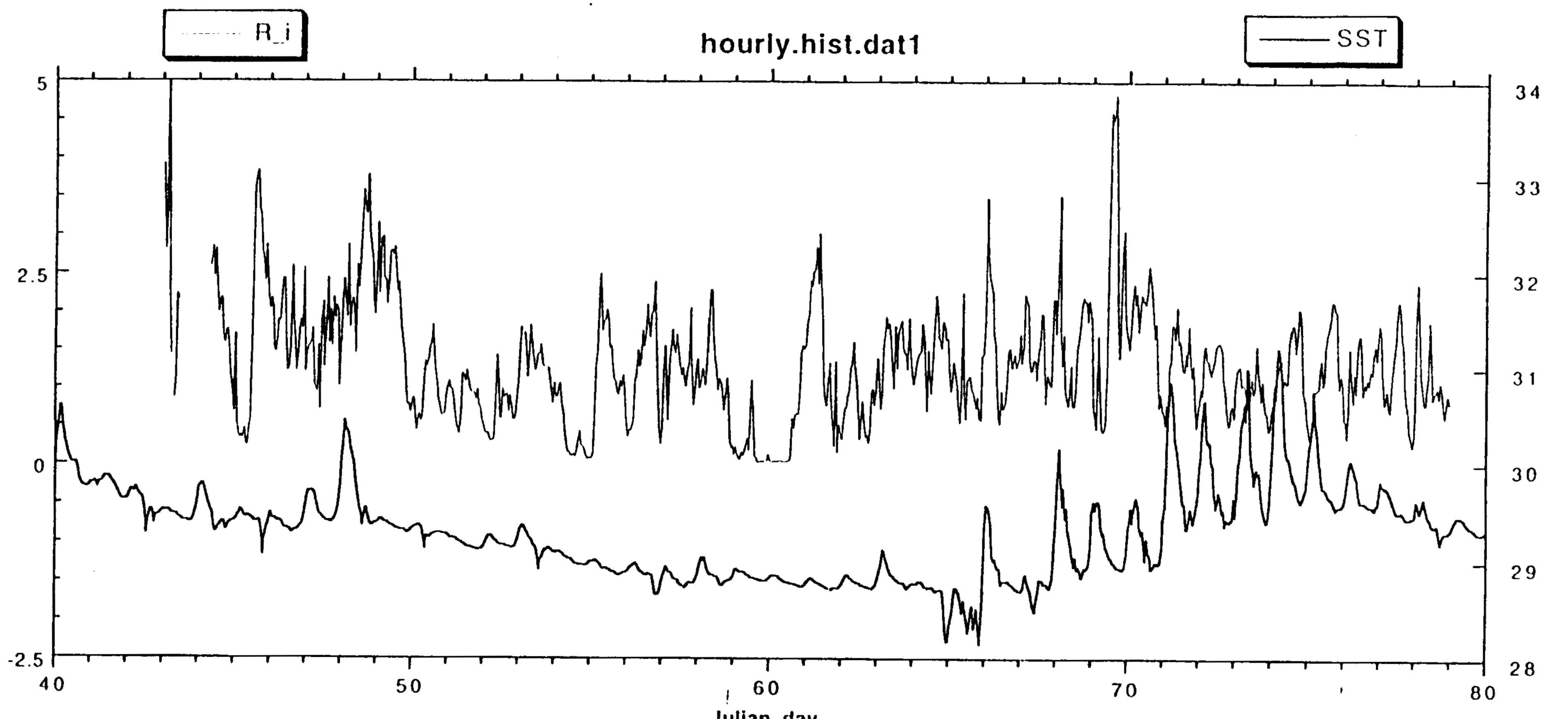
FIGURE 4

(hourly pr within IFA)



: COLA/IGES

FIGURE 5



DISTURBED PERIOD

UNDISTURBED PERIOD

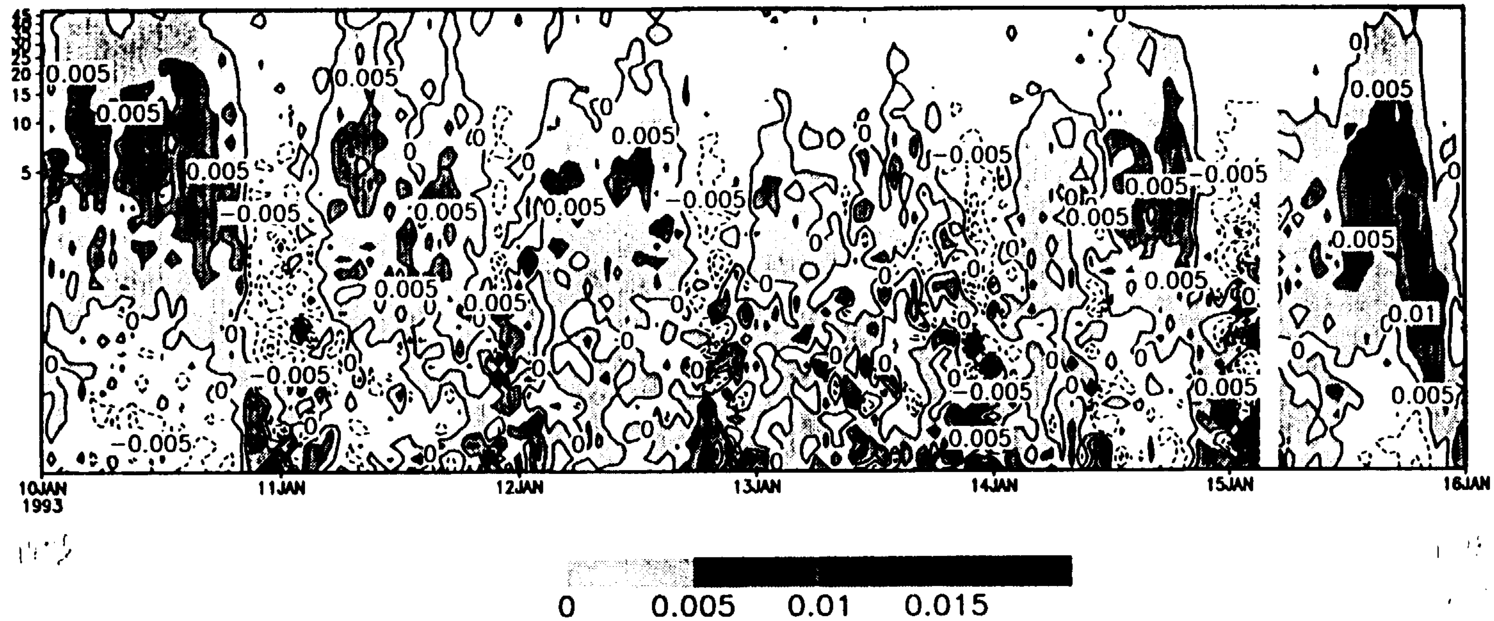


FIGURE 6a

GrADS: COLA/IGES

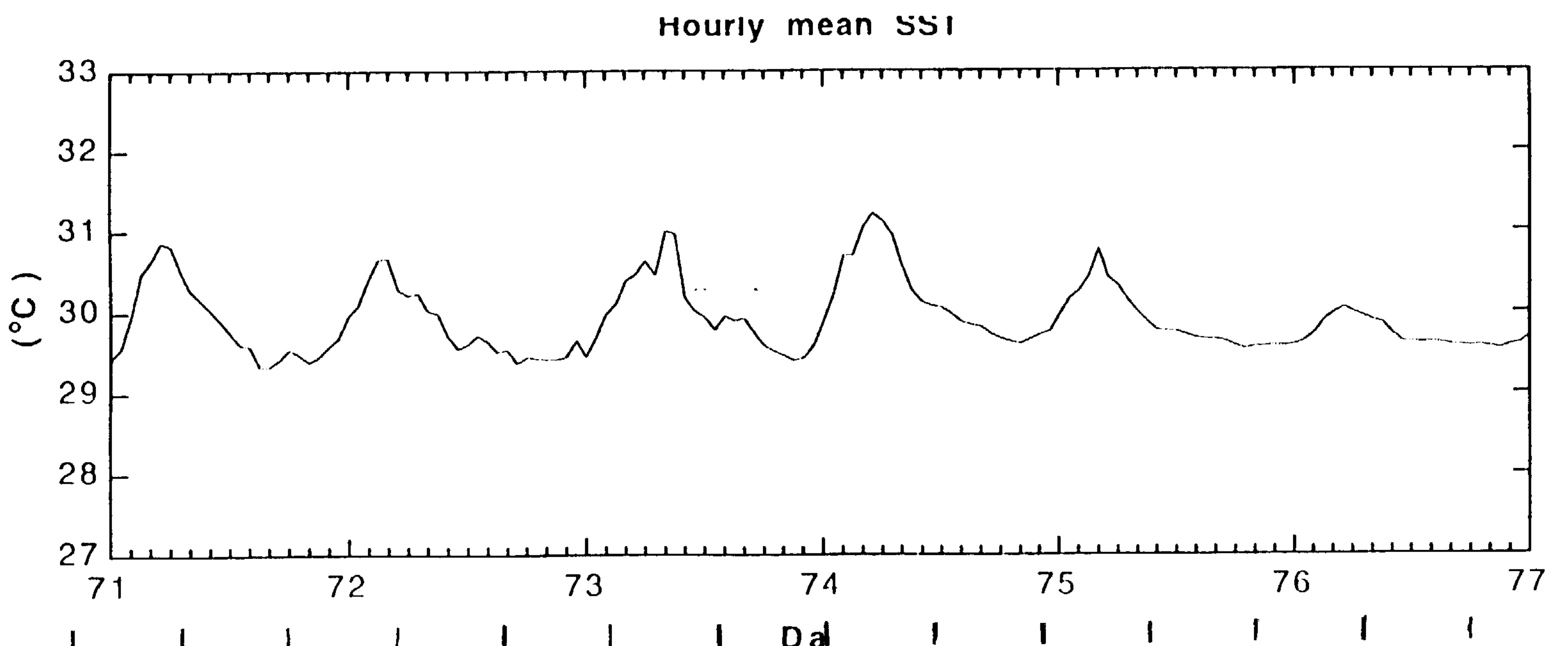


FIGURE 6b

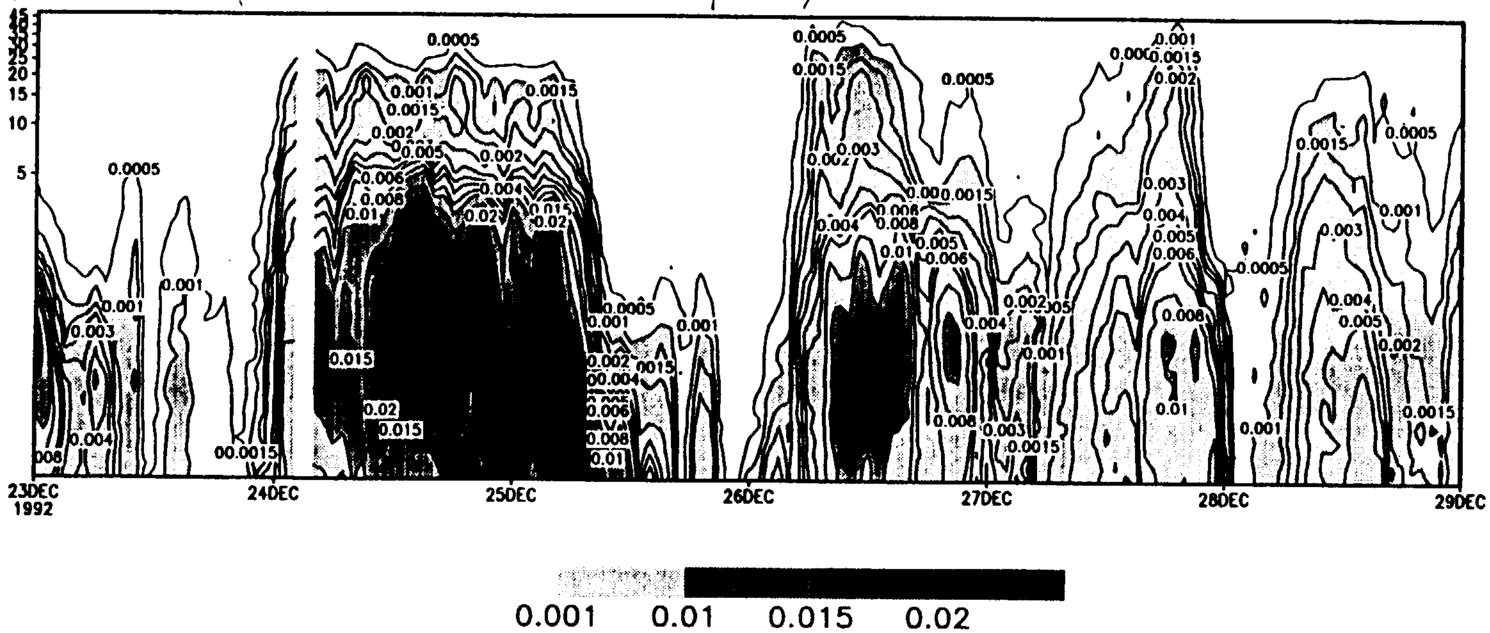


FIGURE 7a

GrADS: COLA/IGES

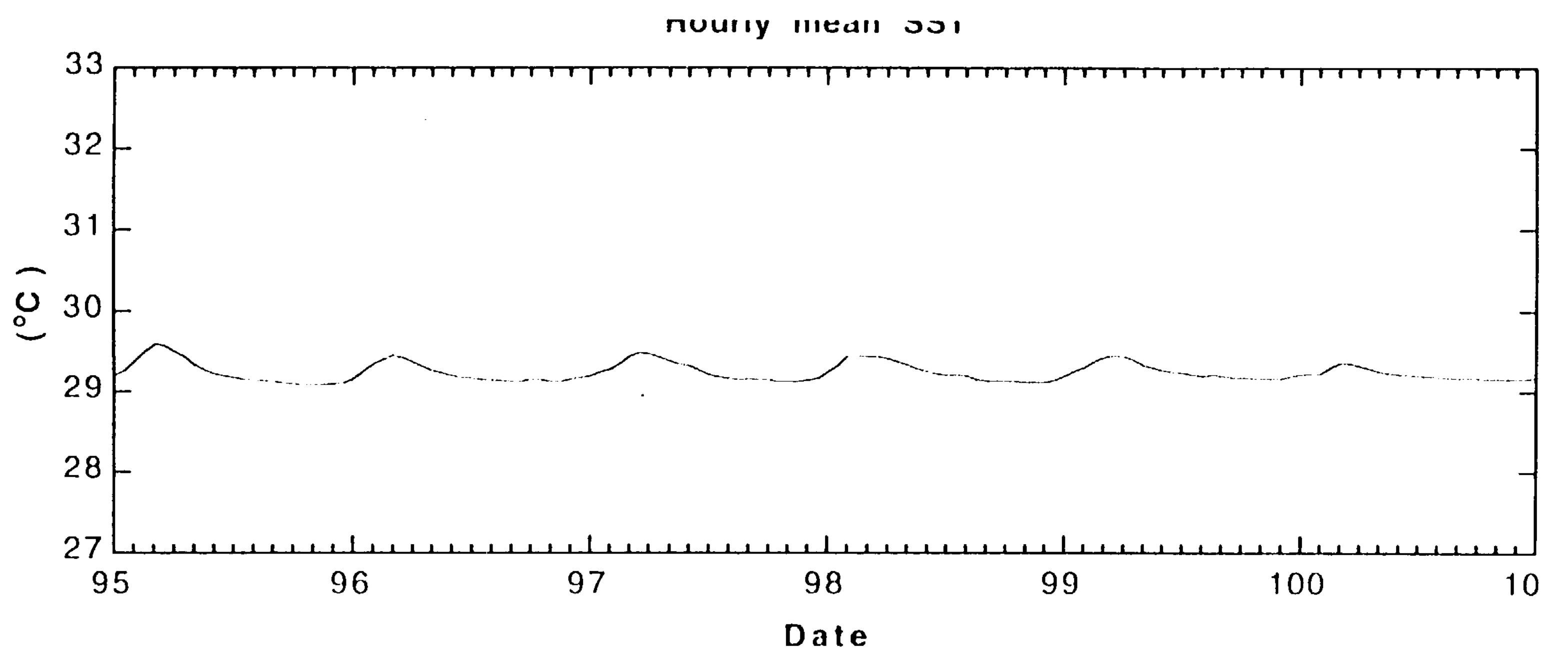


FIGURE 7b

BOREAS Project

C Programming Style and Maintenance

Dorothy Langendorf

SIECA Program

Biospheric Sciences Branch

Code 923

Hughes STX

Mentor: Jeff Newcomer

2 August 1995

PREFACE

The actual report meets the size limitations set forth by Dr. Langdon; however, the number of pages submitted far exceeds this requirement. I have included two copies of one of the programs on which I worked: before and after versions. They are meant for brief perusal only and not for in-depth reading.

ACKNOWLEDGMENTS

In accomplishing my project for the summer, I received a tremendous amount of support from my mentor, Jeff Newcomer. Dave Landis, Amy Ruck, and Fred Irani were always helpful when I went to them for help in trying to understand the various things I had to learn in order to accomplish my assignment.

OVERVIEW OF MAIN PROJECT

Hughes STX, a contractor with Goddard Space Flight Center (Code 923, Biospheric Sciences), has been tasked with supporting the Boreal Ecosystem-Atmosphere Study (BOREAS) and associated information system efforts. BOREAS is a joint United States-Canada interdisciplinary scientific study designed to improve understanding of the interactions between the boreal forest biome and the atmosphere in order to clarify their roles in global change.

The two study areas are located in the boreal forest in central Canada. The Northern Study Area (NSA) is located near Thompson, Manitoba; and the Southern Study Area (SSA) is located near Prince Albert, Saskatchewan. The two areas are approximately 500 kilometers apart. Each study site covers an area large enough to allow the acquisition of useful airborne flux measurements and satellite observations but small enough to allow reasonable coverage with surface instruments.

The field data collection activities began in 1993 and will continue through 1996. Eighty-five science teams divided into six disciplinary groups are participating in BOREAS. The six groups are Airborne Fluxes and Meteorology, Tower Fluxes, Terrestrial Ecology, Trace Gas Biogeochemistry, Hydrology, and Remote Sensing Science.

DATA STORAGE AND MAINTENANCE

The Hughes STX personnel are dedicated to storing and maintaining the extremely large data sets collected by the field scientists, and aircraft and satellite instruments. Data is organized into tables in the Oracle Database, Version 7. The long-range goal is to produce a fully documented data set that will be informative and useful to researchers.

COMPUTER PROGRAMMING AS PROJECT SUPPORT

Several computer programs have been written to aid in the processing of information in the database. Programs perform such tasks as reformatting data for image processing and statistical analysis of information. Because of the tremendously large size of the database, the computer programs have been highly effective in saving time in the project. Revision of the programs is constant in order to clarify the computer code, make the code easier to use, and make the code more efficient operationally. Up-to-date, reliable documentation is essential.

MY ASSIGNMENT AS PART OF BOREAS

My assignment for the internship period was to work with two programs which had been successfully run on on a different project (FIFE), which was also on an earlier version of Oracle. Oracle was upgraded from version 6 to version 7 in June of this year. I was to ensure both programs would run on BOREAS and on Oracle Version 7; and, once accomplished, clean up the programs and comment them for future users. One program dealt with quality assurance of the data; the other extracted data into logical packets of files for distribution to researchers.

<u>OBSTACLES TO OVERCOME</u>

In addition to the C language in which the programs were written, there were several aspects which needed to be considered:

1. The Oracle database was stored on two separate computers: BORIS for permanent data entry and TASHA for testing of various aspects of the data. (Learning to maneuver around the various directories was a challenge in itself.)

2. Oracle used ProC (the C language version of SQL Plus) to query the database. SQL Plus is a language to embed code within C programs. It has its own syntax and structure.

3. Embedded SQL commands within a C program is the means by which the programmer can have his program communicate with the database. The C program file has the extension .pc. The ProC precompiler converts the embedded commands into regular C code and generates the extension .c file, which can then be compiled as a regular C program.

4. Commands for precompilation, compilation, and linkage are not the normal commands used to do this. This

information was not neatly laid out anywhere; it was somewhat of a search to find the necessary commands. Precompilation command is ''PROC1 filename,'' and this must be done on TASHA. The .c file must be transferred to BORIS where it can be compiled and linked. Compilation command is ''CC/NOOP filename.'' The program must link with the ProC library; the command is ''LNPROC filename.exe filename.'' The executable file must be transferred to TASHA for running. This was a very time-consuming activity. This information is now part of the documentation in the program so that the next user will not meet the same difficulty.

5. Each program required user input. Format for the input was not documented anywhere within the program; therefore, it was necessary to do a little searching to arrive at the proper form of input so that the programs would run.

REARRANGING THE PROGRAMS

After overcoming all the obstacles to successful execution of the programs, there was the task of deciphering the programs and putting them into a more understandable form. Since the programs were written by someone who was very familiar with the database and the output expected of the programs, I had a lot to understand about several areas in order to understand the workings of the programs. The author was no longer employed by Hughes STX; therefore, there was no one to whom questions could be directed.

Both programs were written in a style that was difficult to read and decode. I called upon all the instruction from various instructors as to how to write a readable, understandable, and well-defined program.

PROGRAM 1 - ''AUTOQA''

I began with the program that did quality assurance. ''Autoqa'' performs quality checks of the data looking for constant readings (four or more readings that do not change), extreme values (those values more than four standard deviations away from the mean), and spikes (those values that have jumped significantly from the mean delta).

My first step was to put some ''white space'' into the program. There was very little indentation, which I remedied quickly. This allowed me to more readily see what statements were connected with a given condition. Also, the main function was about nine pages long and did too many things. I removed most of the main function and put it into several smaller functions which were called by the main ''driver.''

Once the program was easier to read, I began the process of trying to understand exactly what it did. I started with the embedded SQL commands in order to get a true feeling for what was happening between the C program and the Oracle database. The following are examples of embedded SQL commands which will be converted to C code by the precompiler:

```
EXEC SQL BEGIN DECLARE SECTION;
EXEC SQL DECLARE C CURSOR FOR S;
EXEC SQL CONNECT :uid IDENTIFIED BY :pwd;
EXEC SQL COMMIT WORK RELEASE;
```

I commented all the EXEC SQL statements so that any reader of the C program would be able to understand what was happening, even if he had no knowledge of ProC.

A feature of this program which I had not encountered before was a WINDOW pointer variable declaration. The Systems Analyst helped me locate the curses.h file which contains the commands for creating a window on the text display screen of the VAX computer. This window identified the program that was running and remained on display during the entire execution. It prompted for password and user identification and displayed a confirmation when connected to the database. There were also informational messages which were displayed to tell what was happening at a given time during execution. (NOTE: The window on the screen is not to be confused with anything like Windows on DOS. It was just C code which allowed screen information to be displayed a bit more conveniently.)

Variable names in the program were not named as well as they could have been. Since they were taken directly from a sample program in the ProC handbook, some of the variables were declared but never used in the program. I deleted extraneous variables, renamed some of the poorer-named variables, and commented other variables. Some one-letter variable names were uncontrollable--they were generated by ProC embedded commands. The best I could do was type the list of one-letter variables as an explanatory paragraph at the beginning of the program.

Some variable names were indicative of items in the previous data managing project (FIFE). I changed these names to be more in line with BOREAS nomenclature. Additionally, some Oracle column names were different for BOREAS, and part of the program was to do a string compare of these column names. I had to synchronize these.

I forced a pagefeed at the beginning of each function. Also, each function was begun with a block of asterisks with a simple explanation of what the function accomplished.

Since this program was confusing initially due to lack of documentation, I included an entire page of comments to explain such things as the format of input and how to compile, link, and run the program.

PROGRAM 2 - ''DOWNLOAD''

The second program, ''Download,'' extracts information from the database and organizes it into logical packets of files which will be put into some type of finalized product (possibly compact disk format) and distributed to researchers.

This program was a similar undertaking to the first, but much easier since I had the basic understanding of the database, ProC, and the programming style of the program author. The same steps were taken on this program. Since this was the shorter of the two programs, I have included ''before'' and 'âfter'' programs of ''Download'', for which I renamed my version ''NewD.''

OVERVIEW OF THE PROJECT

This was an extremely informative experience at Hughes STX. The amount of knowledge that I gained is tremendous, and being able to work ''hands-on'' had a greatly reinforcing effect.

Things I learned this summer:

1. Oracle Version 7
2. SQL Plus
3. ProC to Embed SQL Commands
4. Windows declaration in C
5. VAX VMS Commands and Directories
6. File Transfer Protocol (ftp)
7. World Wide Web
8. Hypertext Markup Language (HTML)
9. ...and Dilbert

SUMMARY

The programs with which I worked this summer were used on Hughes STX's previous project, FIFE. Ensuring that these programs would also work on BOREAS was a step toward finalization of the project. Increased readability and documentation of the programs will allow future users of the programs to better understand and utilize them.

PROGRAMS AVAILABLE

FOR REVIEW

UPON REQUEST

SATÈ: System Administration Task Environment

Tony Miller

Abstract

Maintaining large numbers of UNIX workstations requires much attention to detail. With complex installations from multiple manufacturers, the stock administration tools which ship with the operating systems are insufficient for handling even simple tasks. The common duties of adding, deleting, and manipulating user information become a matter of hand editing configuration files on several different machines.

My project involves the creation of an administration tool which will provide a single repository for information concerning all users and machines operated by the Laboratory for Terrestrial Physics, Code 920.2. The user interface, the part that is seen and used for interaction, utilizes the World Wide Web. Through the use of HTML files and PERL scripts, any networked machine which can execute the latest version of the Netscape WWW browser can modify system files. This means that even a PC or Mac on the network can aid in administrative duties. Preserving the security of such a system is vital to preventing unauthorized individuals from gaining access. SATÈ utilizes security features of the Web server run by the LTPCF to ensure that only people with proper privileges can manipulate the system's database. Current functionality includes the ability to add, delete, and modify users. To allow for rapid modification of many users, Netscape tables are employed to present an entire database on one screen, with an option to modify any field value of any number of users at one time. Log files are kept to record an changes made to the database, thus providing a record of all interactions with the tool and the ability to restore previous information.

The tool currently works as a self contained system, but will provide a platform for expansion. Future plans include keeping track of all the equipment operated by the facility as well as related information such as the purchasing records and service contract data.

Introduction

System administration is as much art as science. Keeping up with the latest software and hardware requires frequent studying. Sophisticated installations demand constant attention to ensure that all of the machines and software work properly. Various tools, both commercial and shareware, exist to make the job easier. My project this summer was to implement such a tool to help the UNIX system administrators in the Laboratory for Terrestrial Physics Computing Facility (Code 920.2). The tool utilized the existing World Wide Web system run by the LTPCF and the web browser Netscape. By using the WWW, the administration software is available to any machine in the facility which is on the network

Background

The World Wide Web is a part of the Internet which allows for the transmission of text and binary information. Documents are created using HTML, the HyperText Markup Language. Special programs called Web viewers or browsers utilize commands embedded in the HTML to display a formatted document which may contain text, graphics, or selectable references to other documents or files. The references, called links, serve to create an endless Links between documents can be created which allow someone to go from one document to another anywhere on the net. The development of the World Wide Web, spurred by the creation of a graphical HTML viewer called Mosaic, has grown by leaps and bounds. The most popular Web browser is Netscape, which supports extensions to the original HTML 1.0 and 2.0 specifications not otherwise available. Besides reading documents and exploring various sites, the possibility for interaction exists on the Web. HTML currently supports forms, which present an interface that allows for the bidirectional transfer of information, rather than strictly from the server to the person browsing. With forms support, search engines have been created which allow an individual to request information on any subject, have a program search across the Internet for sites which match the search criteria, and return the information in some easily readable format. On the server side of this information transfer, programs must run which take the request and parse it into something usable, scan the database, and create new HTML on the fly which will present a new page to the user. These programs are called CGI scripts, where CGI stands for Common Gateway Interface. Although any language which allows for the creation of stand-alone executable software can be used to create these programs, the environment of choice is PERL. Perl stands for Practical Report Extraction and Report Language and was developed by Larry Walls. It combines the functionality of shell scripts, the C programming language, and the UNIX utilities sed, awk, and grep into one powerful interpreted language. Because of its flexibility and excellent debugging facilities, Perl is used not only as a major source of cgi scripts, but as a tool to automate repetitive and complicated system administration duties, a perfect match for this project.

Design

The existing universe of the World Wide Web provided much inspiration for the look and feel of the interface. In particular, the software package SATAN (Security Administrator Tool for Analyzing Networks) utilizes an interface which provided an excellent prototype for this project. SATAN is a tool for system administrators to test for security holes in their installations. By combining the free flowing nature of hypertext and the interactive capability of forms, a commercial quality package can be produced with much less effort than when employing a conventional programming language and user interface generation library. The customers and end users for this project are quite familiar with UNIX workstations. They each have several years of system administration experience at large installations. Although basic tenets of good graphical interface design should be followed in any project, many of the more restrictive guidelines do not apply when designing a tool for experienced users. Nevertheless, a consistent look and feel and logical navigation method prevent errors and reduce necessary interaction time. HTML forms work with fewer interface elements than are possible in a full fledged gui environment. The possible "widgets", as they are called, include radio buttons, check boxes, and text boxes. Push buttons serve only to submit queries or to restore forms item default values. Thus, the interaction possibilities are somewhat more limited compared to an X11 or Microsoft Windows application development environment. All user interaction takes place in a page oriented manner, rather than the window and dialog box style available with full fledged windowing systems. Despite these shortcomings, the WWW interface advantages of rapid development and great portability outweigh the disadvantages.

The interface for the tool, while an important project focus, was not the only part which required planning. The back end, which updates the system databases, had to be designed as well. Major goals of SATÈ's library routines include flexibility and extensibility. In addition, the requirements of the system administrators had to be met. A methodology for maintaining databases of user accounts and related information had to be developed. By making the back end independent, SATÈ can be used even without its interface. Thus, even if the WWW link ever dies, as long as a single workstation is up and running, the entire system can still be updated.

Implementation

User interfaces are dynamics entities. They must respond to input in a logical manner, constantly reporting current status and available options. Meeting the requirements of a gui in the WWW browser environment necessitated many small CGI scripts to direct the flow of control. Perl scripts transform the specially formatted stream of information from the browser into discrete variables and values that can be used by the system. Two libraries exist which perform this conversion, cgi-lib.pl written by Steven E. Brenner and cgi_handlers.pl written by James Tappin. Once the requested information is obtained, the CGI script can print an HTML formatted document.to the UNIX file STDOUT (the standard output device). This has the result of returning a new page to the SATÈ user. By creating documents "on the fly," SATÈ can simulate the dialog boxes of a more traditional gui. This method of information retrieval also obviates the need for hard coded HTML files.

The back end library routines create, update, and maintain the database of user information. Since they are likewise written in Perl, they can be accessed from the command line of the workstation serving as the SATÈ host. Thus, batch jobs can be executed which update the user database without utilizing the graphical interface. Most interaction with the SATÈ tools will be with the WWW interface, however. The library is written in standard Perl, even though Perl version 5 with object oriented extensions is available. This was a deliberate decision on my part, because most of the pre-existing code utilizes the functional paradigm. In addition, I have had more experience designing functional systems than object oriented systems. Separate subroutines exist for reading in the data, performing searches, updating with new values, and writing the data back out. This follows the standard input-process-output model of information processing, tried and true in the computer science world. By combining the two elements, interface code and database library, a unified system is created which performs standard administration tasks.

Testing

Testing is always an important part of any software project. Testing must be performed both at the module and system level. Individual functions have to be tested with small stub programs to validate their correctness. Given spurious input, crash resistance must also be checked. Installing a new module into the total system must not break any previous functionality. If it does, those bugs must be identified and eliminated. Because the front and back ends were developed separately, integration testing had to be performed on the entire system to ensure proper functionality. Although Perl provides descriptive error messages, working with Perl CGI scripts can be frustrating. Without a standard text output for error messages, most errors result in an empty page being displayed on the browser. Special testbed programs had to be written to simulate the input to the CGI scripts and determine that the HTML sent to STDOUT is correctly formatted. Exactly following the syntax for HTML isn't a requirement, but simple errors can render a page unreadable.

SATÈ Features

When a user enters the SATÈ environment, a main menu is presented (figure 1). The choices include maintaining the user database, the cluster database, and modifying an individual user's general information and password. The cluster and user database maintenance options bring up a second menu (figure 2) which allows searching the database or simply displaying all the items in the current database. The search utility allows searching for any string in any of the categories for that particular file. Since the database field descriptions are included in the database files, the search categories presented by SATÈ are always up to date. Viewing an entire database brings up a table (figure 3) with all the fields and their associated values. Changing a value is as simple as picking some user(s), choosing a category, and entering a new value. The page is returned with the new values displayed. Working with individual users is as simple as typing in a username and picking the modify option. Alternatively, a user can be deleted or added. Care has been taken that no duplicate users can be added and that usernames which do not exist cannot be modified. Adding a new user is aided by the existence of default values for all of the fields. These defaults come from the various databases and can be altered from within the SATÈ environment. One of the important functions that the administrators need for the LTPCF is the ability to manage clusters. Clusters are groups of workstations that provide a single operating environment for all of the users on those machines. Users need to have their information managed on the cluster level, and this need is addressed in the cluster modification section of the individual user (figure 4). Membership in a cluster can be added or removed at will. The particular information for the user in that cluster can also be modified (figure 5). Finally, user passwords can be changed to a specific value or returned to a default.

A further consideration with the project was security. It is imperative that no unauthorized users can obtain entry into the system and wreak havoc with the user accounts. To satisfy this condition, a password system has been utilized on the server. Without the proper userid and password combination, no one can obtain access to the tool. In addition, no user passwords are displayed on the screen in plain text format, ensuring that someone looking over a system administrator's shoulder won't see other people's passwords.

Future Work

Although SATÈ is a stand alone system at the present time which is capable of performing maintenance duties, further work is warranted. The first item to be added is multi-level administration. With multi-level support, cluster managers can add and delete users from the machines they manage, but not in machines not under their care. In addition, the entire system will be revamped so that further modularization exists. Enhancing the loosely coupled nature of the system will allow for the autodetection of new modules. Thus, the main selections will indicate all functionality without having to create new HTML or modify existing documents. Making SATÈ available for demonstration and viewing by the internet community is also a goal of mine. I will continue to support the software and make improvements even as I go back to school in the fall.

Conclusion

SATÈ has met the goals defined at the beginning of the summer. I am pleased at how far it has come, and excited about the future prospects. With further work, the SATÈ system may be used by administrators across the net. At any rate, I have made the jobs of the administrators in Code 920.2 somewhat easier.

Figure 1

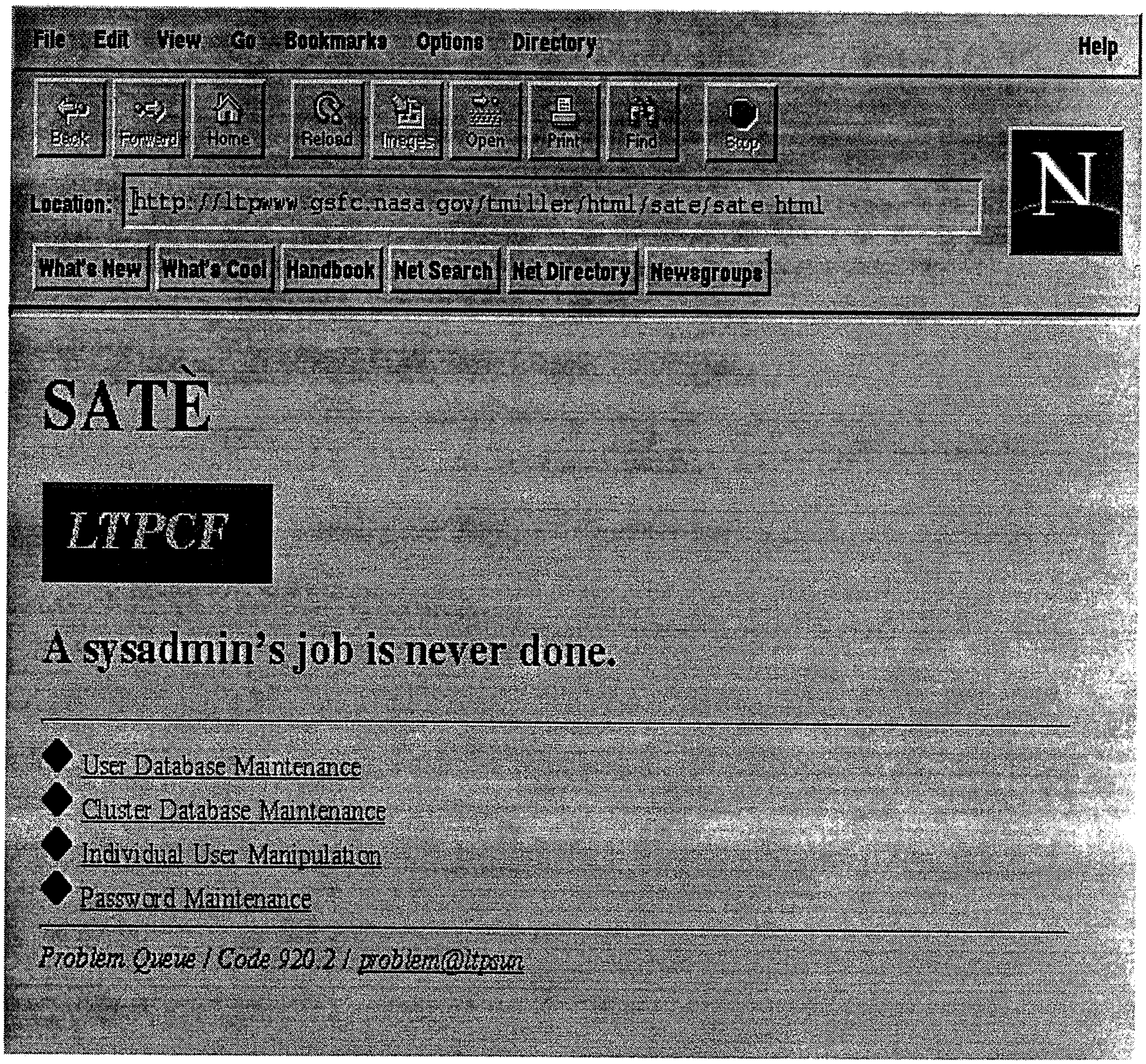

File Edit View Go Bookmarks Options Directory Help
Back Forward Home Reload Images Open Print Find Stop
Location: http://ltpwww.gsfc.nasa.gov/tmiller/html/sate/sate.html
What's New What's Cool Handbook Net Search Net Directory Newsgroups
N
SATÈ
LTPCF
A sysadmin's job is never done.
User Database Maintenance
Cluster Database Maintenance
Individual User Manipulation
Password Maintenance
Problem Queue / Code 920.2 / problem@ltpsun

Figure 2

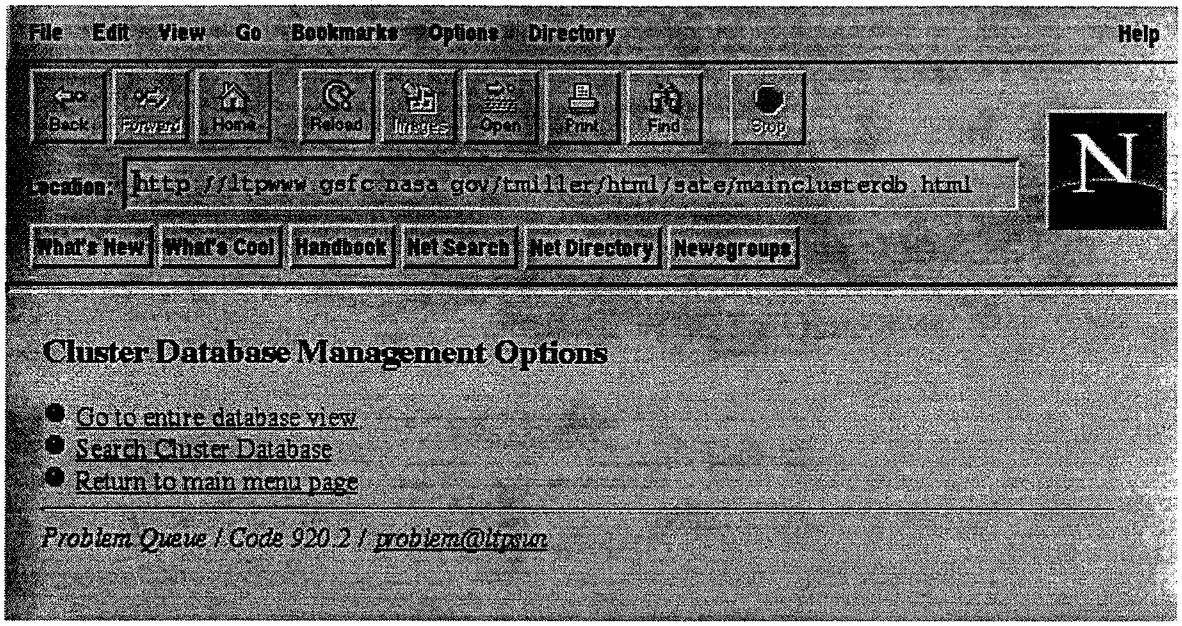

File Edit View Go Bookmarks Options Directory Help
Back Forward Home Reload Images Open Print Find Stop
Location: http://ltpwww.gsfc.nasa.gov/tmiller/html/sate/mainclusterdb.html
What's New What's Cool Handbook Net Search Net Directory Newsgroups
Cluster Database Management Options
Go to entire database view
Search Cluster Database
Return to main menu page
Problem Queue / Code 920.2 / problem@ltpsun

Figure 3

Cluster Database

	cluster	userid	primary_group	home_machine	home_dir	shell
☐	LTPCF	default	20	ltpnfs	/home/default	/usr/local/bin/tcsh
☐	LTPCF	curt	21	curt	/users/curt	/usr/local/bin/tcsh
☐	LTPCF	miw	21	ltpnfs	/home/miw	/bin/csh
☐	LTPCF	tmiller	21	thuja	/home/tmiller	/usr/local/bin/tcsh
☐	LTPCF	dbk	21	burma	/usr/people/dbk	/usr/local/bin/tcsh
☐	FED	default	20	forest	/usr/people/default	/usr/local/bin/tcsh
☐	FED	dbk	10	burma	/usr/people/dbk	/usr/local/bin/tcsh
☐	MCST	default	20	ltpnfs	/usr/people/default	/usr/local/bin/tcsh
☐	MCST	dbk	20	burma	/usr/people/dbk	/bin/csh
☐	MCST	dbk	20	burma	/usr/people/dbk	/bin/csh
☐	GSC	default	20	modisl	/users/default	/bin/csh
☐	GSC	dbk	20	modisl	/users/dbk	/bin/ksh

New value for selected users

Go Clear

Figure 4

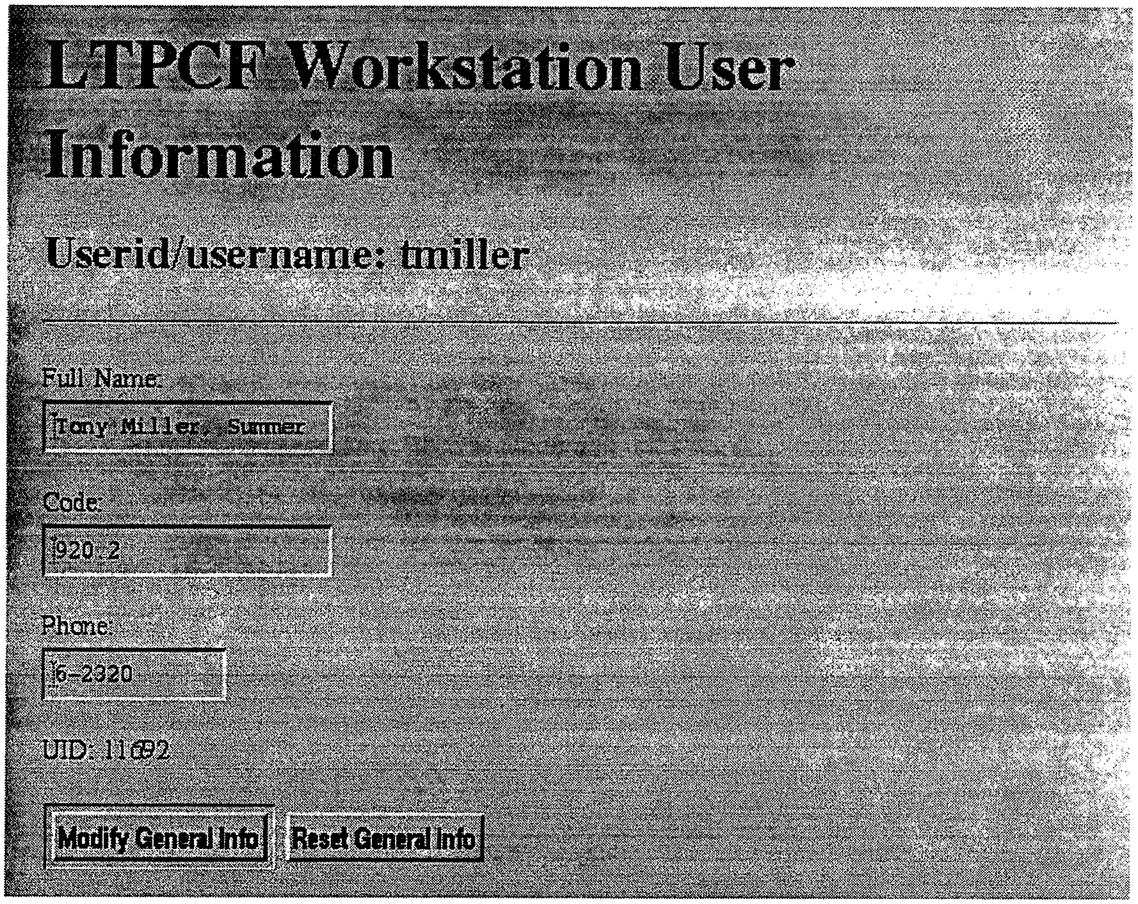

LTPCF Workstation User Information
Userid/username: tmiller
Full Name:
Tony Miller, Summer
Code:
920.2
Phone:
6-2320
UID: 11692
Modify General Info
Reset General Info

Figure 5

Pick a Cluster for Modification

LTPCF

Delete User from Cluster

LTPCF

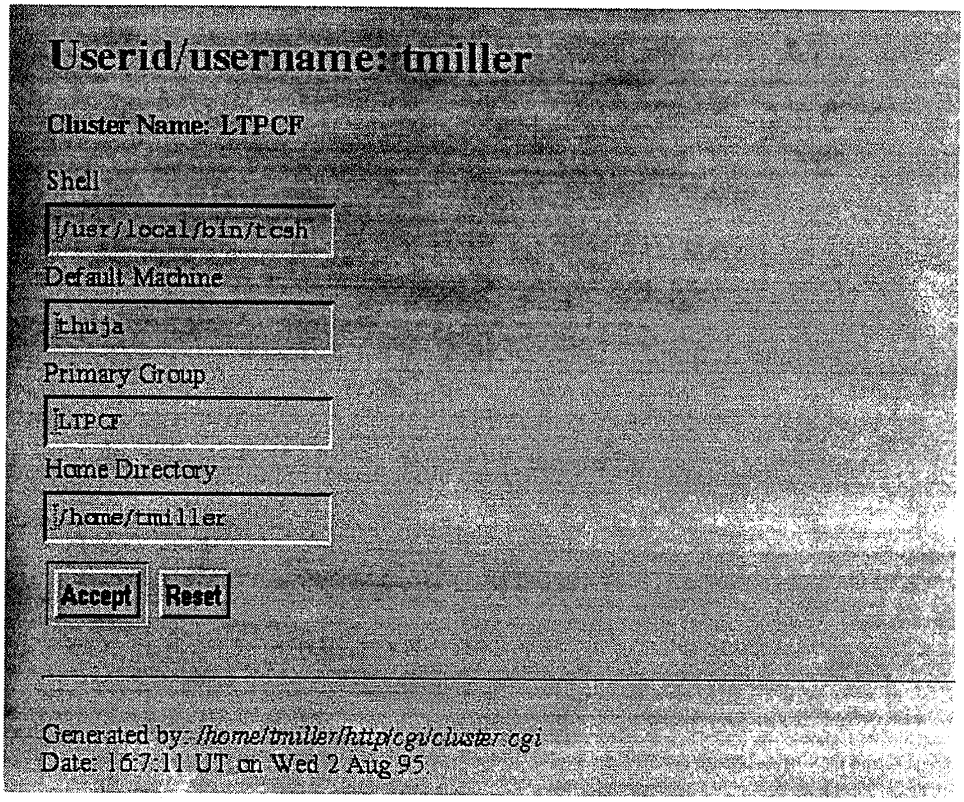
Add User to Cluster

FED
MCST
GSC

- Return to User Data Entry Page
- Return to main menu

Generated by: */home/tmiller/http/cgi/clusterstuff.cgi*
Date: 16:6:58 UT on Wed 2 Aug 95

Figure 6


Userid/username: tmiller
Cluster Name: LTPCF
Shell
/usr/local/bin/tcsh
Default Machine
thuja
Primary Group
LTPCF
Home Directory
/home/tmiller
Accept
Reset
Generated by: /home/tmiller/http/cgi/cluster.cgi
Date: 16:7:11 UT on Wed 2 Aug 95

National Aeronautics and Space Administration
Goodard Space Flight Center
Electromechanical Branch. Code 723.2
Greenbelt, Maryland

Simulation of a Tracking Device Controller for PAMS Experiment

by:
Alberto Rodríguez

mentor:
Javier Lecha

for:
SIECA
(Summer Internship for Engineering and Computer Applications)

August 2, 1995

Abstract

The objective of this project is to develop a mathematical model to simulate the dynamic behavior of the gimbal in the Attitude Measurement System (AMS) of the Passive Aerodynamically Damped Satellite Experiment (PAMS). The purpose of the PAMS shuttle test flight is to demonstrate the feasibility of using aerodynamic stabilization and magnetic damping to passively stabilize small satellites. The data obtained from the PAMS experiment will be used to verify the analytical predictions made by Langley Research Center (LaRC). The PAMS experiment will be flown aboard the shuttle in the spring of 1996.

In order to accomplish this, a satellite test unit (STU) is going to be ejected from the shuttle bay and its trajectory tracked as it decays into the earth's atmosphere. The STU has an array of corner cubes mounted on its face and its attitude is going to be measured by shooting a laser beam at it. The STU reflections are going to be read from the corner cubes using a charged coupled device (CCD) array camera. To center the reflection of the STU into the field of view (FOV) of the CCD array camera, the reflected laser beam is first bounced off a gimbaled mirror and steered to the center of the camera FOV by adjusting the gimbal angles. The gimbal angles are adjusted by means of a feedback system that measures the position of the beam relative to the CCD FOV using a Quadrature Avalanche Photo Diode (QAPD). Using a computer control system the gimbal motors are positioned to null the system's position error.

The mathematical model of the AMS gimbal that I am developing includes the optics and QAPD, gimbal mechanics, control electronics and accounts for errors induced by the shuttle attitude control system. The designed mathematical model is being simulated using an object based programming tool called Simulink, designed by Matlab's Mathworks. The model parameters are being curve fitted and different scenarios are being simulated to find the system's best tracking performance.

Acknowledgments

I am deeply in debt with a group of people without whom this project would not have been a reality. I want to thanks specially Mr. Willie Blanco for all his technical guidance and support during this project. Thanks to Mr. Javier Lecha, my mentor, who believed in me, to Mr. Alfonso Hermida and Mr. Rajeev Sharma who also helped me during this summer project.

I am also in debt with Mr. Dan Krieger who help us to make this an enjoyable experience, thanks Dan. To Mary and Dr. Langdon for believing in me and giving me the opportunity to perform this summer. If forget somebody I apologize, it is not my intention, thank you all.

Table of Contents

Table of Figures

Introduction

In the past decades man power were used to control many processes. Progressively, controls engineering started to automate those processes that were exclusively performed by humans. Therefore, it can be said that control engineering is concerned with the understanding and control of materials and forces of nature for the benefit of mankind.

Today, the control of physical systems with a digital computer is becoming more and more common. Aircraft autopilots, mass transit vehicles, oil refineries, paper making machines, and electromechanical servomechanisms are among the countless existing examples. To create a stand alone system that controls a complex process is not an easy task. For that reason, computer simulations of the processes to be controlled are often done. Simulations try to predict the actual plant or process behavior. If an engineer can accurately predict the process behavior a precious design time can be saved.

Today, the electrical engineering field of control is growing. There are many places were automatic control mechanism are being designed and NASA Goddard Space Flight Center is not an exception. This summer I was assigned to the Electromechanical Branch at NASA Goddard Space Flight Center. My assigned project is called PAMS (Passively Aerodynamically stabilized Magnetically damped Satellite). The objective of my project is to develop a mathematical model to simulate the dynamic behavior of a gimbal in the Attitude Measurement System (AMS) of PAMS experiment. The purpose of the PAMS shuttle test flight is to demonstrate the feasibility of using aerodynamic stabilization and magnetic damping to passively stabilize small satellites. Passive stabilization means the use of no active systems such as electronics, which consumes power, or fuel rockets. The stabilization will be obtained by installing magnetic rods in the satellite fuselage. When the rods are in contact with the earth magnetic field, the satellite is going to align itself with the electromagnetic field achieving the expected stabilization. The data obtained from the PAMS experiment will be used to verify the analytical predictions made by Langley Research Center (LaRC). The PAMS experiment

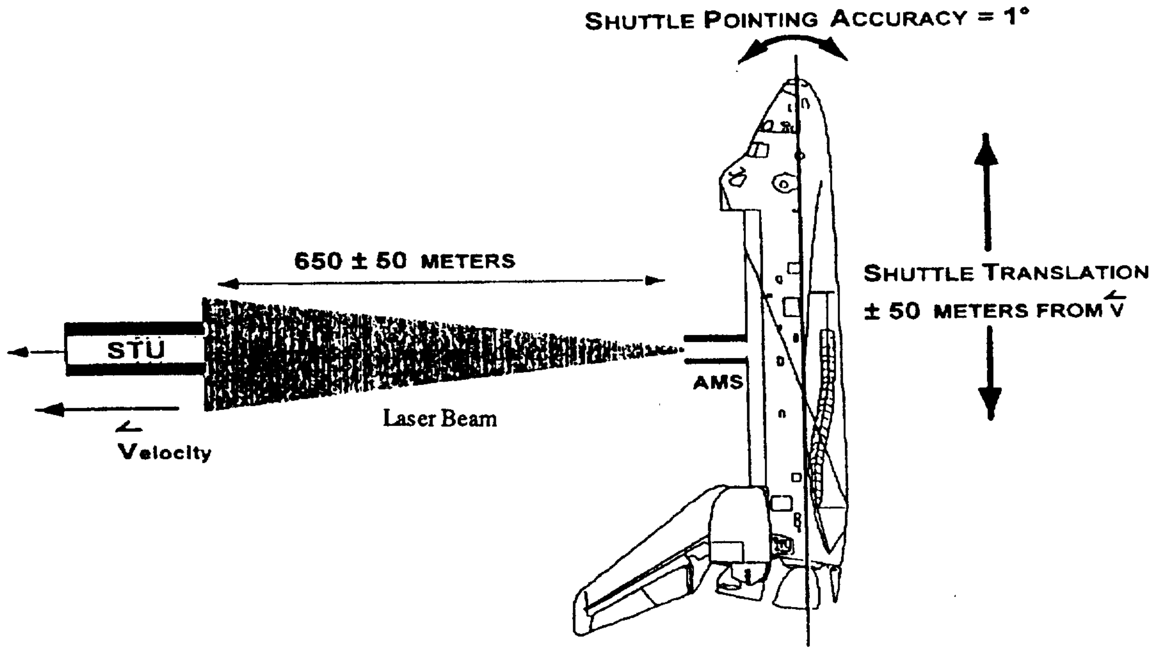


Figure #1. Shuttle Test Flight Configuration

will be flown aboard the shuttle spacecraft in the spring of 1996.

In order to accomplish a good attitude measurement, a satellite test unit (STU) is going to be ejected from a hitchhiker canister from the shuttle cargo bay and its trajectory tracked as it decays into the earth atmosphere. The STU is going to be positioned at 650m ± 50m away from the shuttle spacecraft, see Figure #1. The AMS (Attitude Measurement System), which is also located in a canister in the shuttle's cargo bay, is going to keep track of the STU, see Figure #2. The STU has an array of corner cubes mounted on its face, see Figure #3, and its attitude is going to be measured by shooting a laser beam at it, Figure #1. The STU reflections are going to be read from the corner cubes using a CCD array camera. To center the reflection of the STU into the field of view (FOV) of the CCD array camera, the reflected laser beam is first bounced off a gimbaled mirror and steered to the center of the camera FOV by adjusting the gimbal angles. The gimbal angles are adjusted by means of a feedback system that measures the position of the beam relative to the CCD FOV using a Quadrature Avalanche Photo Diode (QAPD). Using a computer control system the gimbal motors are positioned to null the position error. The mathematical model of the AMS gimbal that I am developing includes the optics and

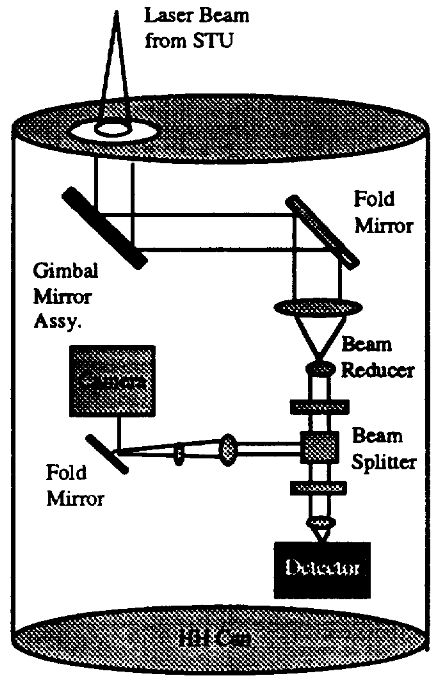


Figure #2. Attitude Measurement System Hitchhiker Configuration

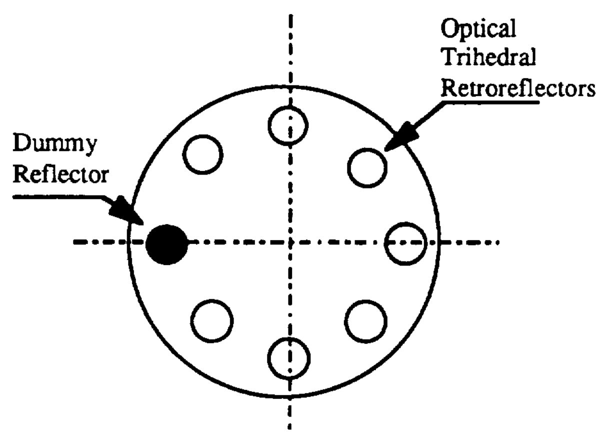


Figure #3. Satellite Corner Cubes

QAPD, gimbal mechanics, control electronics and accounts for errors induced by the shuttle attitude control system. The designed mathematical model is being simulated using an object based programming tool called Simulink, designed by Matlab's Mathworks. The model parameters are being curve fitted and different scenarios are being simulated to find the system's best tracking performance.

Simulink's Simulation

The PAMS Gimbal's Control System Simulation has been performed using a program called Simulink. It is an object based programming language that allows visual programming by dragging and dropping objects into a canvas. Simulink's libraries contain many objects that enable the programmer to create complex systems in a relative short time. The two axis gimbal's simulation contains the necessary details to resembles the actual gimbal's control system. The simulation contains the DC servo motors transfer functions, the effects of the back electromagnetic field, friction, inertia, sampling stages, disturbances, optical stages and many more. To help the non-technical reader to understand the simulation, the following paragraphs are dedicated to explain basic concepts.

It is important to remember that the system to be controlled contains two DC servo motors. A DC servo motor can be characterized using a simple RL (Resistor and Inductor) circuit as shown in Figure #4. In order to implement the motor characterization in Simulink is important to obtain the motor transfer function. Using the circuit drawing of Figure #4, the following equation can be derived,

$$I_m = V_{dc}\left(\frac{1}{R + sL}\right),$$

where I_m is the current flowing at the motor stator, V_{dc} is the voltage applied to the motor, R and L are the motor resistance and inductance respectively, and s is the Laplace transform variable. A DC servo motor also contains mechanical characteristics, the following equation presents the DC motor mechanical characteristics

$$B_{emf} = k_e\,\Omega,$$

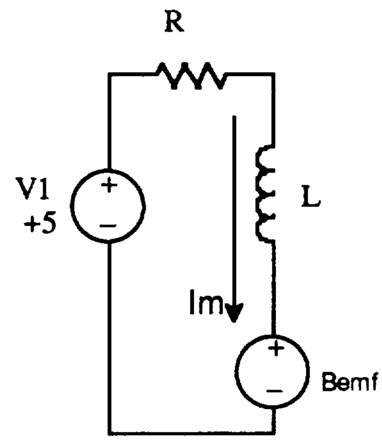


Figure #4. DC Servo Motor Equivalent Circuit

where B_{emf} is defined as the Back EMF (ElectroMagnetic Field) that can be measured at the DC servo motors windings, k_e is the back EMF constant and Ω is the motor speed. As the equation implies the back EMF is proportional to the motor speed. Another important motor relationship is given by the following equation,

$$I_m = \frac{T}{k_T},$$

where, I_m is the current flowing in the motor, T is the motor torque and k_T is the motor torque constant.

The simulation also contains the effects of torque and friction losses. These looses can be described using the following equation,

$$T_N = T_M - B\Omega.$$

The equation above says that the net torque is the developed torque minus the viscuous friction, which is proportional to the motor speed. The effect of the system's inertia is simulated using a simple gain. The Simulink's block diagram for one axis control system is shown in Figure #5.

At this particular time, when the simulation model is already done, the most important task is to fine tune the controllers. In order to do that, the zeros and poles of the system should be identified. The zeros of a system are the root of the numerator of the system's transfer function. On the other hand, the poles are the root of the denominator of

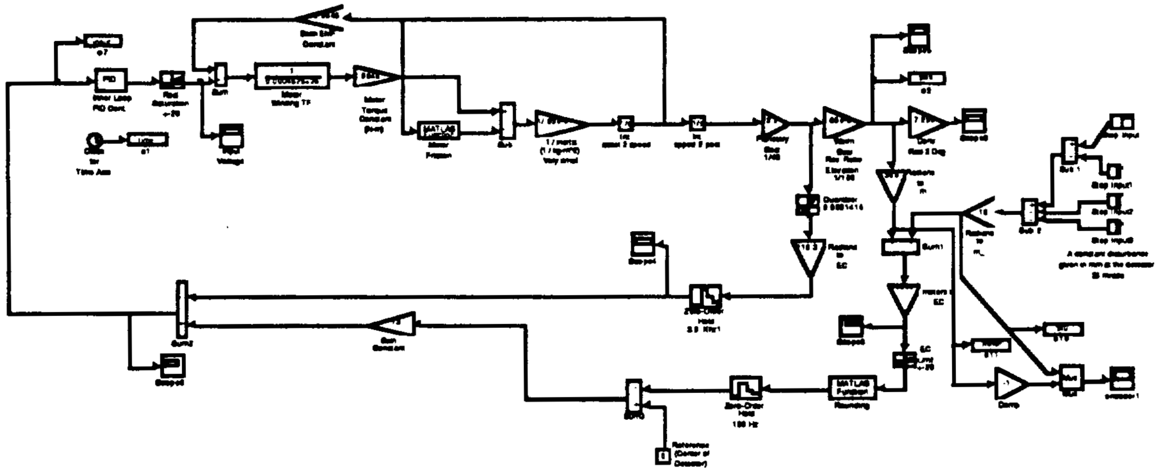


Figure #5. One Axis Gimbal Control System

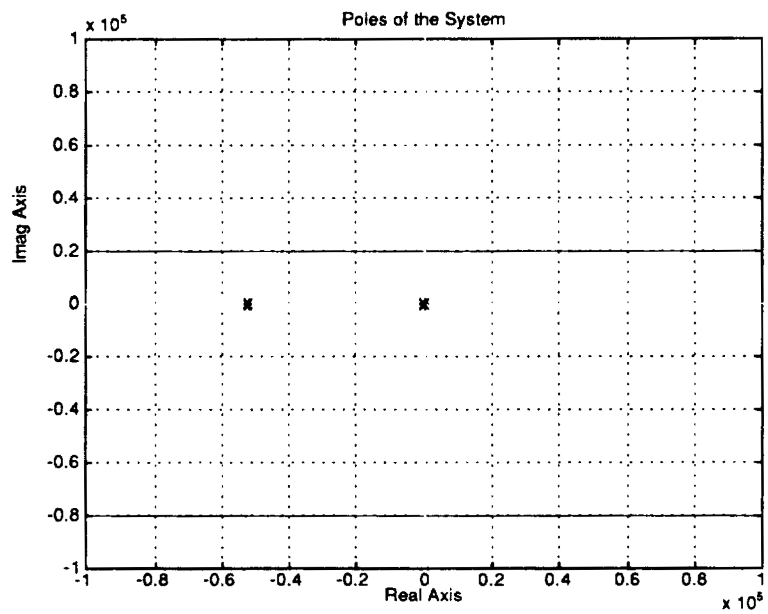


Figure #6. Poles of One Axis Gimbal Control System

the transfer function. After careful study and analysis of the system shown at Figure #5, it can be seen that the system possesses three poles and no zeros, this analysis does not includes the system's controller poles and zeros. The position of the system's poles are shown in Figure #6, this analysis is very useful to design a stable system. The stability issued is resolved making sure that the poles and the zeros of the system always lie at the negative side of the real axis. Also, is important to mention that in Figure #6 the system

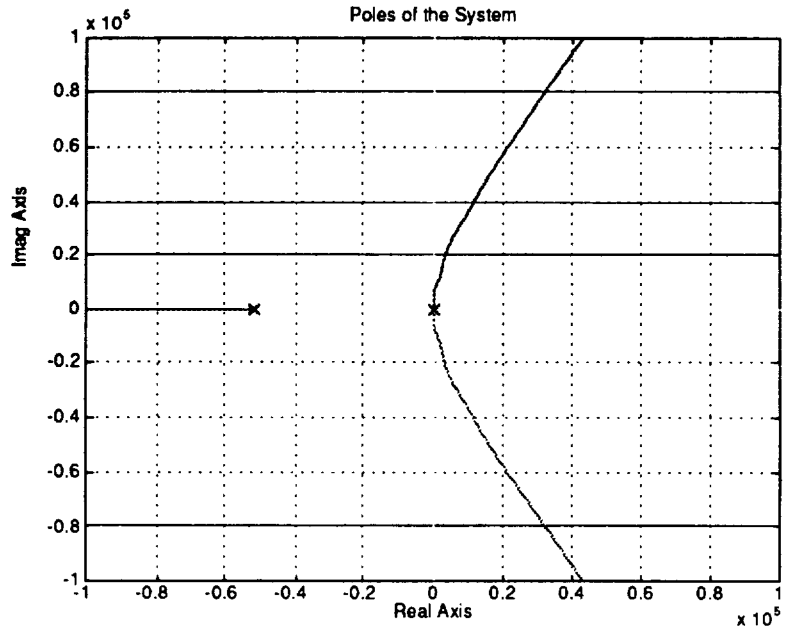


Figure #7. One Axis Control System Root Locus Analysis

has unitary gain. If the gain keeps increasing there is a probability that the system may becomes unstable, this possibility is verified doing a root locus analysis. The analysis basically starts positioning the zeros and poles of the system in a graph and begins to increase the system gain up to infinity. Figure #7 shows the root locus analysis for the one axis system with no controller present. It can be seen that after certain gain, that can be easily determined, the system becomes unstable.

Close to the origin or the center of the plot, Figure #7, there are two poles. Actually one is located at the (0,0) position and the other is located around (-40,0). Those two poles are doing unstable the system. In order to avoid unstability is important to cancel the effect of them. In order to do so, the insertion of zeros in the system is desired. One way to insert poles and zeros is to include a controller. The controller can be P, which stands for proportional, or PI, which stands for proportional integral, or PD, which stands for proportional and derivative and finally the PID, which stands for proportional integral and derivative.

The proportional controller only inserts a gain, the integral controller inserts a pole at the origin and the derivative controller inserts a zero. The central idea is to design the position of the controller's zeros position to assure stability at all times.

Results

By the end of the 10 weeks summer period the gimbal controller system simulation is working correctly. As an example of the systems status Figure #8 shows how it behaves in presence of a disturbance. Proportional controllers were used for both loops. The sharp sawthooth signal is the disturbance at the system, and the smooth curby signal is the system moving with the disturbance to cancel its effect.

In conclusion, during this summer I have acquired many experiences that will change my career. I learned a lot of automatic control systems, zeros and poles positioning design criteria, feedback loop control systems and many other things.

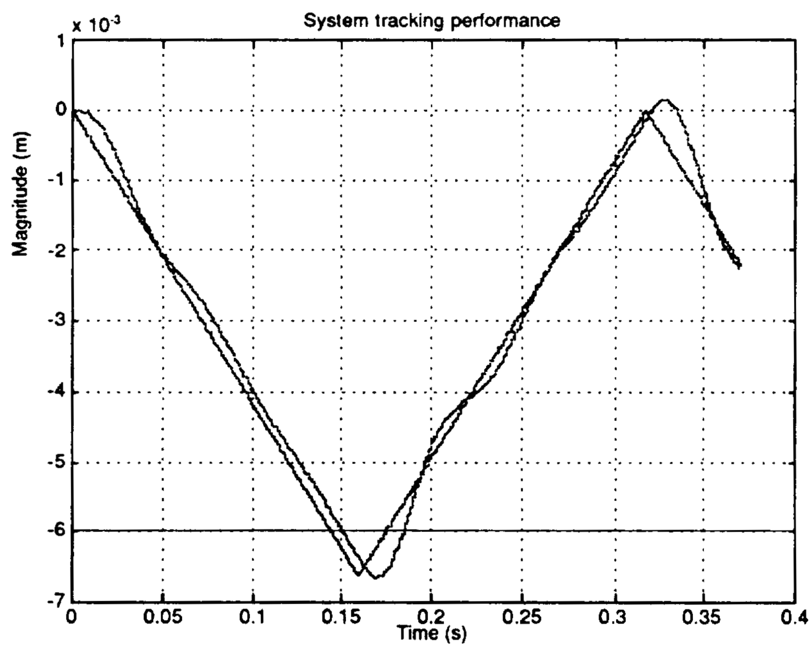


Figure #8. One Axis System Tracking Performance.

References

[1] Richard C. Dorf, "Modern Control Systems," Addison Wesley Publishing Company, 1981.

[2] Gene F. Franklin and J. David Powell, "Digital Control of Dynamic Systems," Addison Wesley Publishing Company, 1980.

[3] The Mathworks Inc., "Simulink, A Program for Simulating Dynamic Systems," 1992.

Summer Internship Project

Kenneth Russell
August 2, 1995
SIECA - Graduate Student

North Carolina Agricultural And Technical State University

CODE 735.3
Mentor: Alan Cudmore

The Flight Software Section, 735.3, was in need of an application that would allow them to have remote access to Ground Support Equipment telemetry via personal computer when designated workstations are not available or are in use. While the thought process behind this project was still going on, it was also realized that this application, if designed, could save travel time from home back to work when problems arose during simulations and other tests.

One of the main ideas of the project was portability. The ability to use any pc at home or work was appealing because it would not require a person to use a specific machine in a certain area. Any personal computer with network access would be available.

In June of 1995, this idea became reality and the Remote Telemetry Monitoring System project began. Once complete it would allow access to any telemetry providing a valid network connection could be established to the Advanced Spacecraft Integration and System Test GSE workstation(s).

In order to understand the totality of the RTMS/95 project, its underlying structure must be examined. Code 735.3 is responsible for designing and developing flight software for certain programs at GSFC. Along with this, analysis and support for the on-board data management systems is performed while coordinating the data processing requirements for payload development.

Flight Software Section is an integral part of the Flight Data Systems Branch which includes four (4) other section: Analysis of Radiation Effects, Data Processing Devices, Command and data handling and advanced packaging. The entire Engineering Directorate (700) culminates a wide program of technical research and development for space applications and science programs.

Code 735.3 develops flight software for missions such as XTE and TRMM. In order to test and verify the flight software, 735.3 must monitor the real time data output from the spacecraft called telemetry. This is monitored through the Advanced Spacecraft Integration & Systems Test, developed by code 733, along with Ground Support Equipment. The Advanced Spacecraft Integration & Systems Test and Ground Support Equipment allows the user to interact with the spacecraft, sending commands and viewing real time telemetry from workstations. Because of the cost and design of the Advanced Spacecraft Integration & Systems Test and Ground Support Equipment, they are located only in development labs and spacecraft integration and test sites. This restriction requires all testing and viewing of telemetry to be done at these specific sites. There is, however, a feature designed in the Advanced Spacecraft Integration & Systems Test and Ground

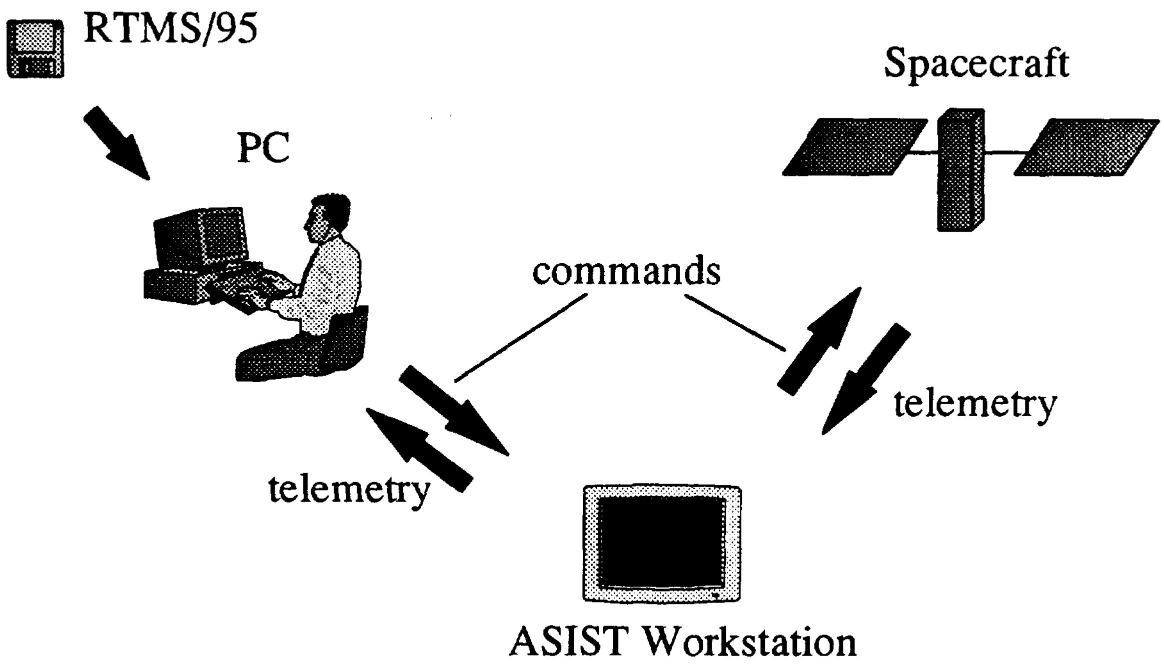
Support Equipment which allows spacecraft telemetry to read over a network. This is where RTMS/95 fits into the picture.

Once the hardware issue was sorted out, the matter of what type software to use was presented. Microsoft's Visual Basic was chosen because it was relatively inexpensive, accessible and it develops effective user interfaces. These interfaces or screens would have to be designed to enable a person with minimal computer experience to easily maneuver though the system.

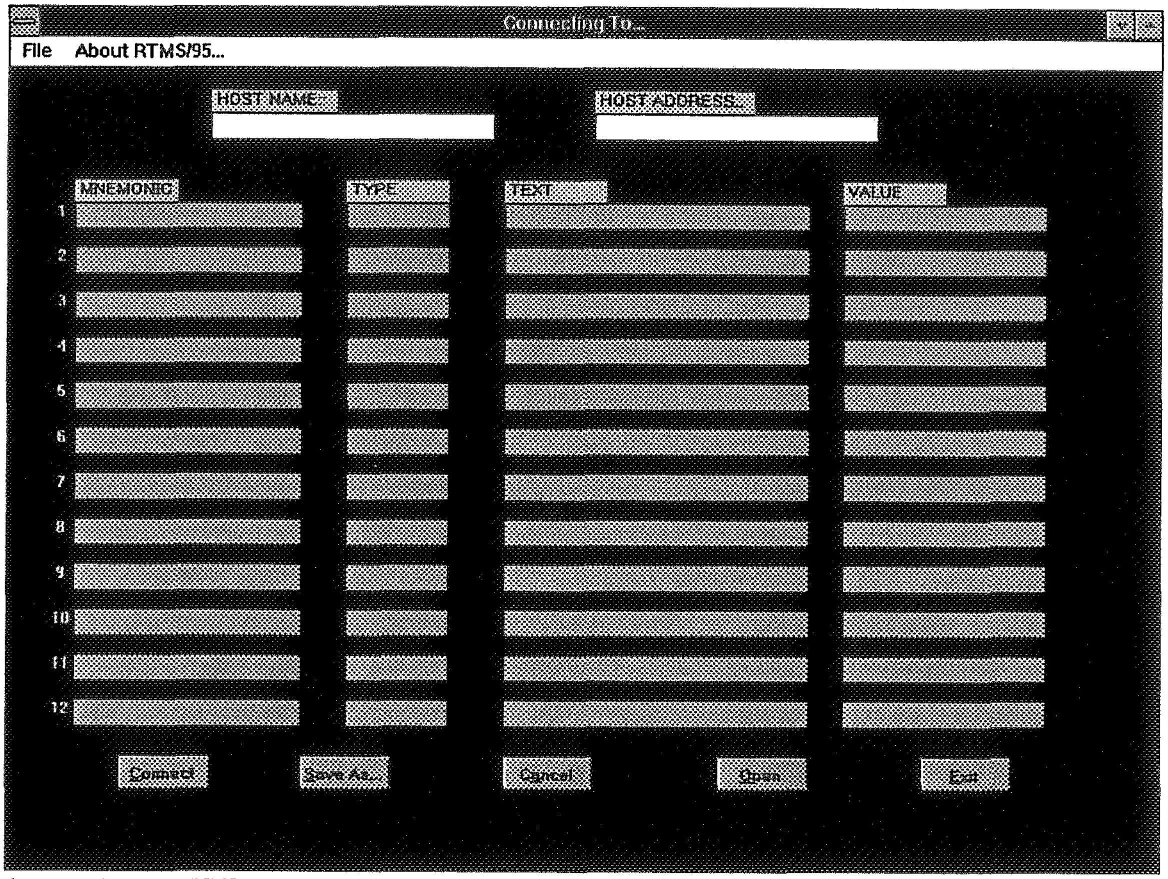
As the testing phase begins on projects, unexpected events are likely to happen and support personnel would be called. The capability of viewing the telemetry was not possible from all locations, so long, tedious phone conversations could ensue. Most of the time taken was used to verbally relay variables and values between the personnel. If the problem could not be resolved this way, the support personnel would more than likely have to travel back on site to try and remedy the situation. (This is not to say that RTMS/95 has eliminated this altogether, some travel back to site will be required to fix certain problems.)

Another situation that RTMS/95 addressed was the projects where funds were not appropriated for workstations and the present workstations were at a maximum level of usage. The RTMS/95 program would allow access to certain data via an internet connection to an Advanced Spacecraft Integration & Systems Test and Ground Support Equipment workstation. A user now could monitor some spacecraft telemetry from their very own desktop. With the availability of Microsoft Visual Basic, this program can be run virtually anywhere.

How RTMS/95 Works



Visual Basic code and ASIST workstation commands are use to send a request for connect to the ASIST workstation. Once a connection is established, the workstation then passes the request for the designated mnemonics to the spacecraft. Mnemonics are the cryptic names for over 10,000 commands available for spacecraft testing and implementation. The respective mnemonic value, or telemetry, is relayed back to the workstation, where its value is displayed on the pc. RTMS/95 ver. 1.0 only holds twelve (12) mnemonics. In order to view more, another session of RTMS/95 would have to be run.



(example RTMS/95 screen)

The interface is done using Ethernet connection with Transmission Control Protocol/Internet Protocol (TCP/IP) as the transport source. A two way socket is then established for communicating via messages. Each message delivered by the client (pc) is confirmed either by an accept or reject on the server end of the connection.[1]

Clients are initially in the READY state. They are not yet connected to any telemetry stream nor are they ready to receive any data. To receive data, a client must sent a CONNECT message. When this message is successfully received, the client enters the CONNECT state. From this position the client can either enter a Block Definition, which transport several telemetry items, return to the READY state or advance to the

RECEIVING state. This is where all requested telemetry resides.[2] The following is sample code used to initiate a connection:

```
'close old connection - if any
IPPort1.Connected = False

If txtHostName <> "" Then
  IPPort1.HostName = txtHostName.Text
  txtHostAddress.Text = IPPort1.HostAddress
  ElseIf IPPort1.HostAddress <> "" Then
     IPPort1.HostAddress = txtHostAddress.Text
     txtHostName.Text = IPPort1.HostName
  Else
     MsgBox "Please specify a host."
     Exit Sub
End If

'ASIST telemetry interface
IPPort1.Port = 4202

'ask for connection
IPPort1.Connected = True

'wait until it is achieved
Do Until IPPort1.Connected: DoEvents: Loop
Me.MousePointer = 11

'send the command to go to connected state
IPPort1.DataToSend = "UUUU0016CONNECT SCTLM -T"
Do Until Signal <> 1: DoEvents: Loop
```

The RTMS/95 project is very easily modified. Other improvements can be added to it as well as the application being used in conjunction with similar products. RTMS/95 leaves in its wake a host of advantages. Some of which are:

- portability - any desktop pc is appropriate for its use
- can be used anywhere internet access is available
- spacecraft tests can be monitored from other buildings or from home
- integration and testing can use it to supplement expensive workstations

With these major advantages and other plans, RTMS/95 will become an integral part of the 700 Directorate and ultimately NASA/GSFC.

[1] NOVELL NetWare v3.11 TCP/IP Transport Supervisor's Guide

[2] ASIST Workstation User's Guide

RTMS/95

Remote Telemetry Monitoring System/95 ver. 1.0

User's Manual

created by: Alan Cudmore (753.3)
developed by: Kenneth A. Russell (SIECA-NCA&TSU program)
August 1995

I. INTRODUCTION

The Flight Software Section, 735.3, was in need of an application that would allow them to have remote access to Ground Support Equipment (GSE) telemetry via personal computer (pc) when designated workstations are not available or are in use. While the thought process behind this project was still going on, it was also realized that this application, if designed, could save travel time from home back to work when problems arose during simulations and other tests.

One of the main ideas of the project was portability. The ability to use any pc at home or work was appealing because it would not require a person to use a specific machine in a certain area. Any pc with network access would be available.

Once the hardware issue was sorted out, the matter of what type software to use was presented. Microsoft's Visual Basic was chosen because it was relatively inexpensive, accessible and it develops effective user interfaces. These interfaces or screens would have to be designed to enable a person with minimal computer experience to easily maneuver though the system.

In June of 1995, this idea became reality and the Remote Telemetry Monitoring System (RTMS/95) project began. Once complete it would allow access to any telemetry providing a valid network connection could be established to the ASIST GSE workstation(s).

II. Terms and acronyms used in this manual:

- **RTMS/95** - Remote Telemetry Monitoring System/95
- **ASIST** - Advanced Spacecraft Integration and Systems Test
- **.TXT** files - Text file default extension
- **Workstation** - Any ground support equipment that is utilizing ASIST commands e.g. lauraa, davem, opus2
- **Visual Basic** - Windows interfacing and development tool

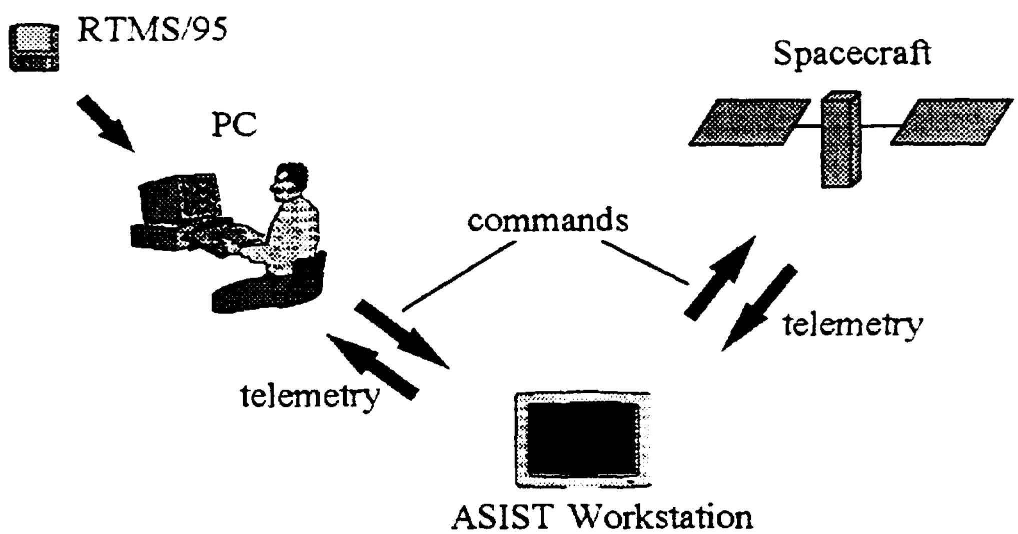
III. To start RTMS/95:

There are two ways to run RTMS/95:

- run the RTMS/95.exe from your pc hard drive or,
- from a diskette: run B:\RTMS/95.exe

Once the application is running, it takes on the attributes of any window based file.

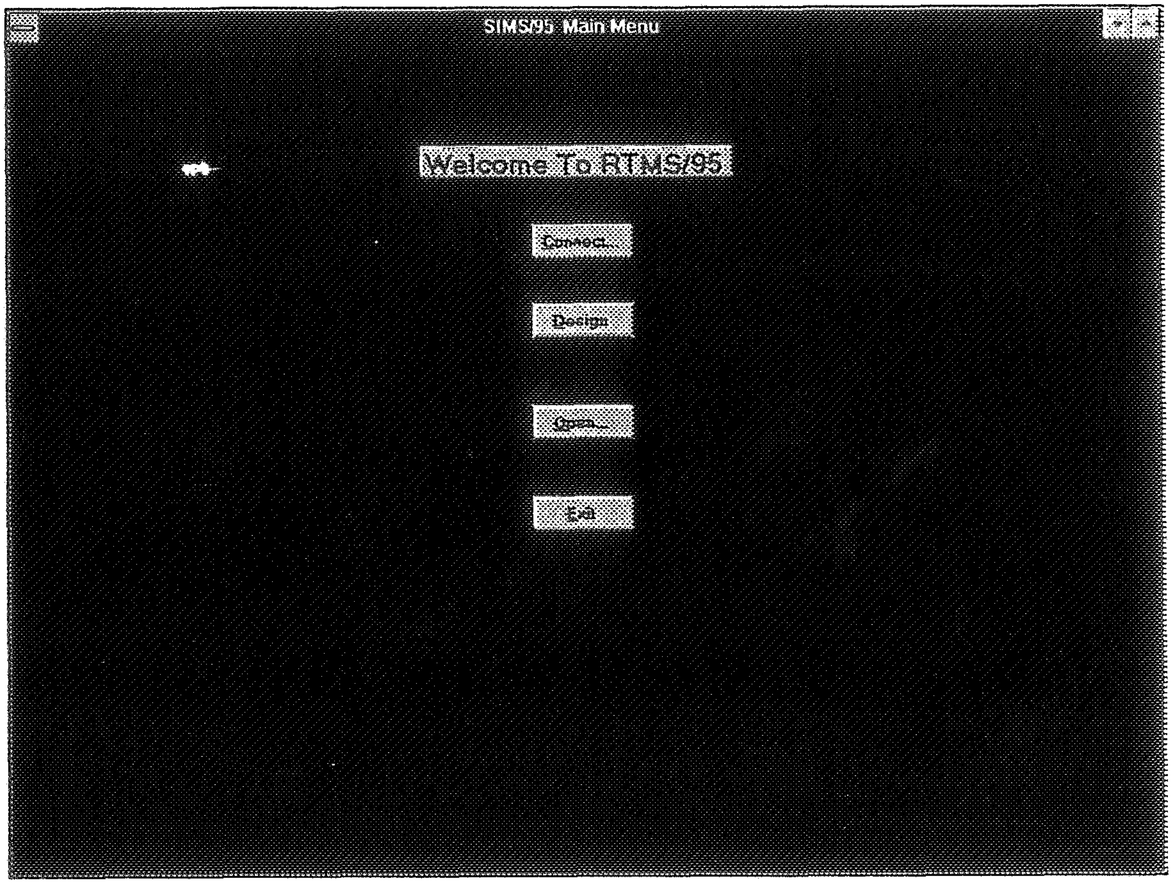
How RTMS/95 Works



NASA/GSFC

Visual Basic code and ASIST workstation commands are use to send a request for connect to the ASIST workstation. Once a connection is established, the workstation then passes the request for the designated mnemonics to the spacecraft. The respective value, or telemetry, is relayed back to the workstation, where its value is displayed on the pc. RTMS/95 ver. 1.0 only holds twelve (12) mnemonics. In order to view more, another session of RTMS/95 would have to be run.

IV. Operating within RTMS/95:



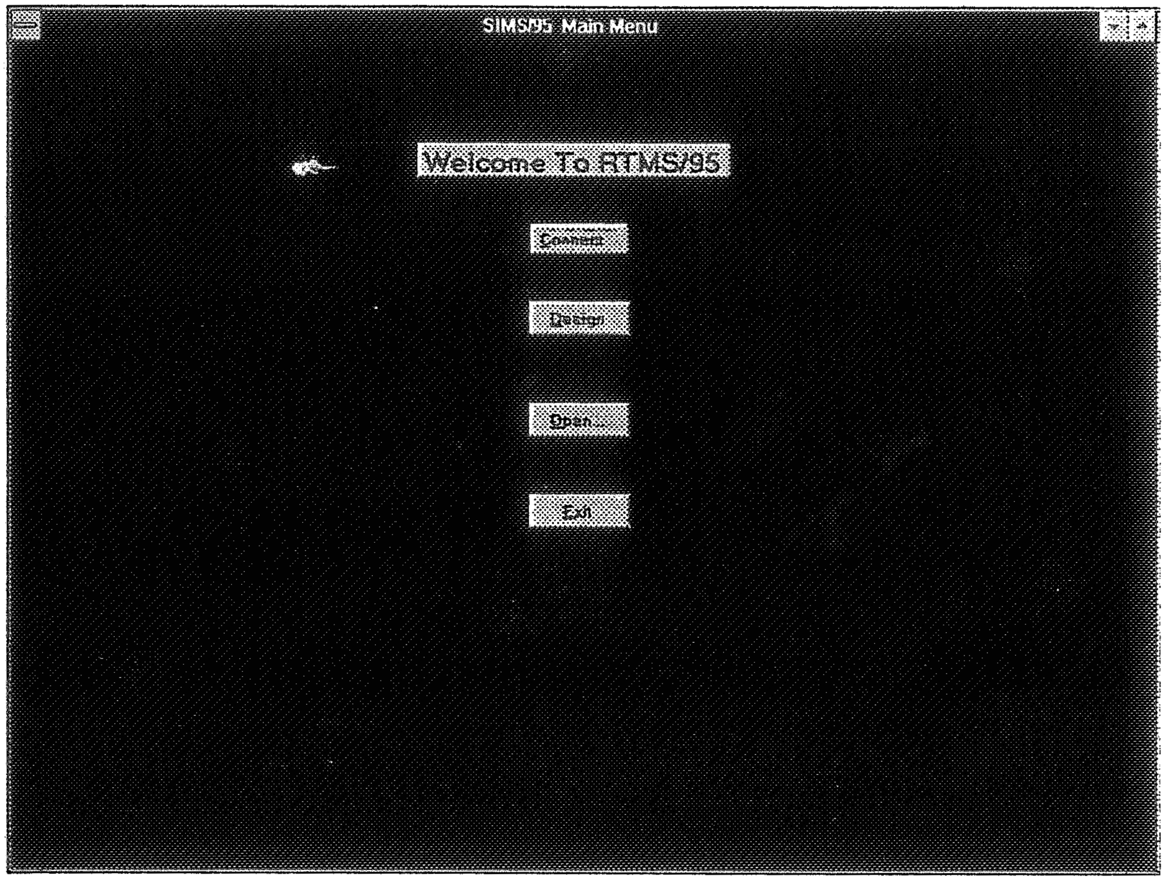
Once the application has been started, the user will be met by a screen similar to the one above. At this point, the user has three (3) options:

1. Connect - if the user has not previously saved a format to a .txt file, they can type in mnemonics, text types and field descriptions on this screen.

2. Design - an option that will be placed in a later version.

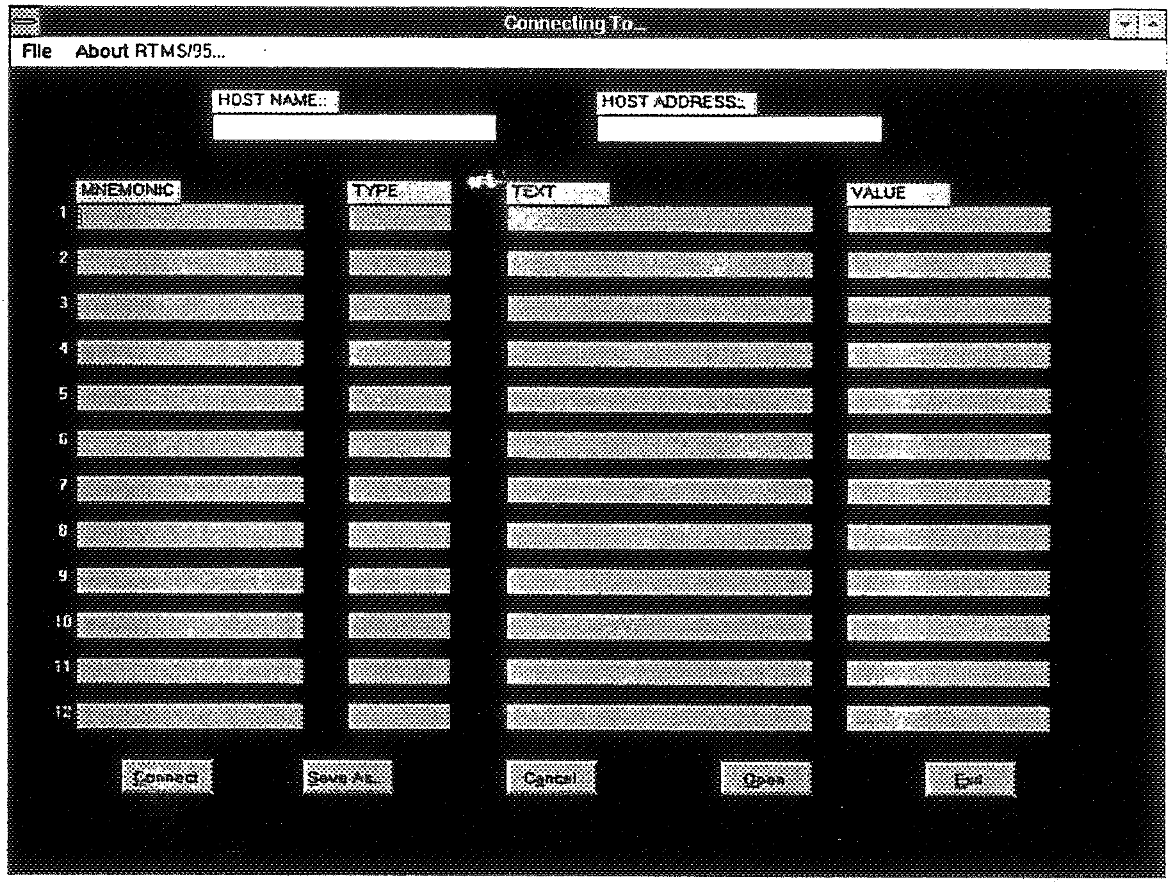
3. Open - if the user has saved a previous .txt file and does not wish to make any changes to if before connecting to an ASIST workstation.

V. Connecting to a Workstation:



1. Select the Connect.... button. This screens primary use is for the development of new .txt files. Other .txt files can be opened from this screen if there is a need.

Connecting to a Workstation: (continued)



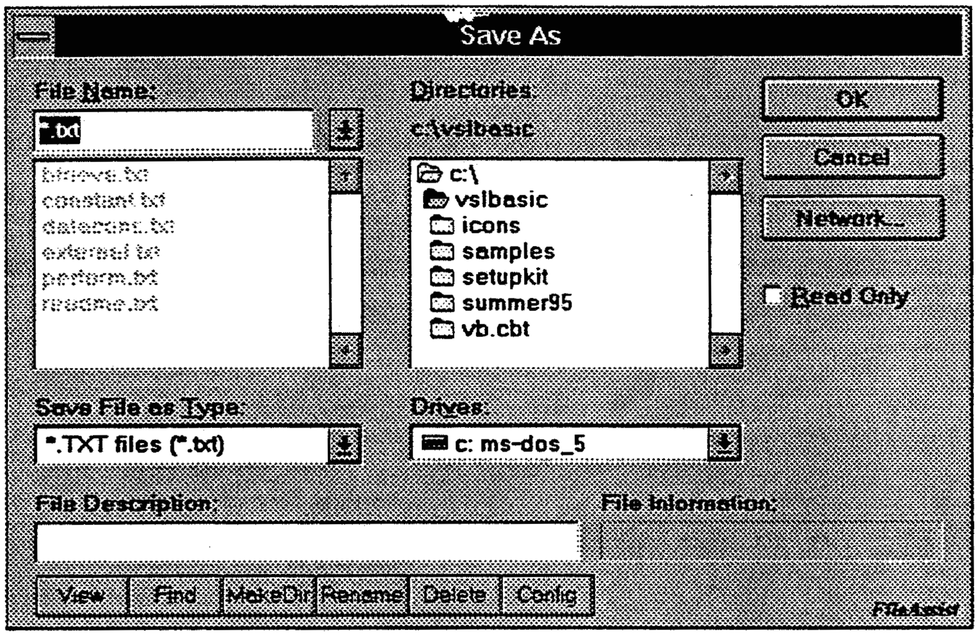
As shown on this screen there are four (4) groups of fields: Mnemonic, Type, Text and Value.

- Mnemonic - needs only cryptic names placed in it, e.g. sciapkpc, scicken. There must be a mnemonic placed in at least the first field for the program to execute.
- Type can only be represented by the following:
  - h - hexadecimal
  - d - decimal
  - s - string
  - f - float
- Text can be any description the user wants to have to represent the mnemonic field.
- Value is respective system data output only from the ASIST workstation.

The user can now place the mnemonic(s), type(s) and text in the appropriate fields. Once done, the Host Name or Host Address is placed in its field and the user can click on Connect to start the program.

VI. Saving to .TXT files:

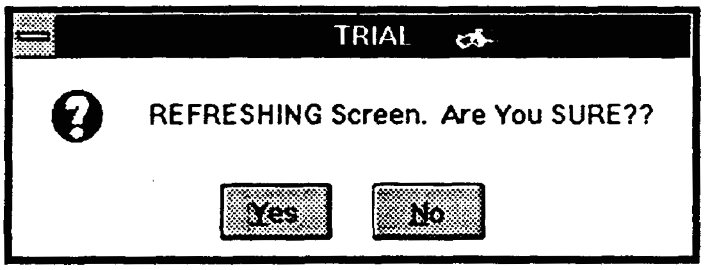
When the user is finished executing the program, there are several options available to them. The first is saving the current work to file. This is done by clicking on the Save As... button which will bring the following screen to view.



It is advised that the user only save to .txt files. For this reason, a default extension will automatically save using .txt. After the user has chosen a name and entered it into the File Name box and selected the correct drive on which to save the file, click the OK button. You have now saved your file to the hard drive, diskette or network.

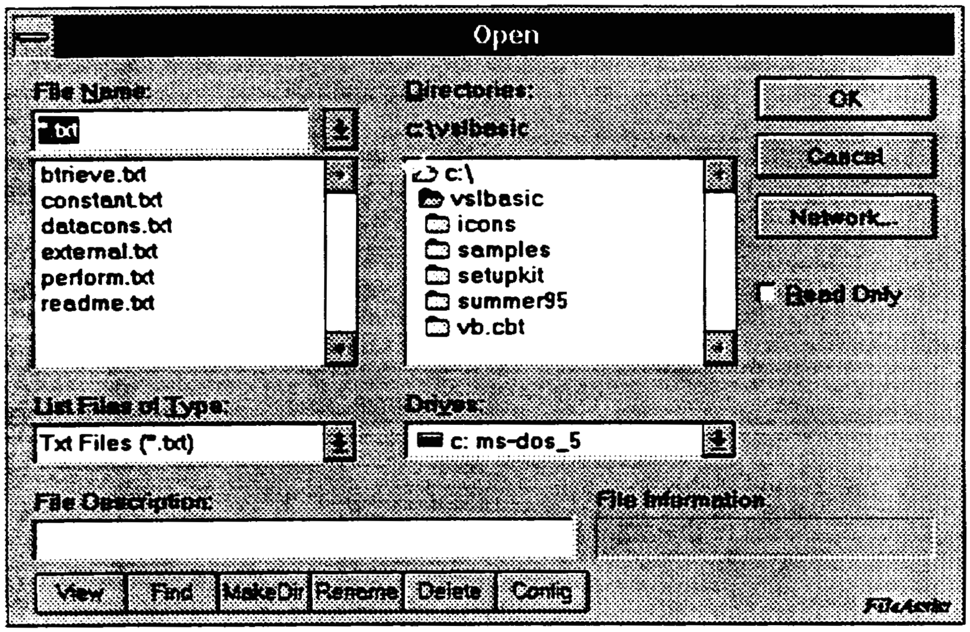
VII. Cancelling the connection:

When the user as finished with a particular file or does not wish to save the current screen information, they can click on Cancel which will refresh the screen.



All existing information will be deleted and the connection to the workstation stopped. At this point the user can open a file or enter new information onto the Connect screen.

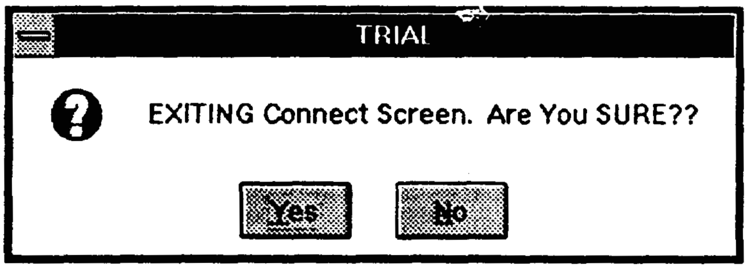
VIII. Opening files:



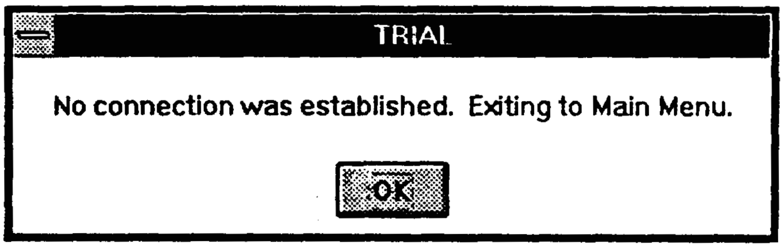
The open dialog box allows the user to access any previously save .txt file either on the hard drive or on diskette. Once the user has chosen which file and which drive to locate the file, the program automatically loads and places the information from the file into the respective fields in the RTMS/95 screen.

IX. Exiting the Program:

When the user wants to exit from Connect Screen, click the Exit button and the message below will appear.



Once done, the program will inform the user that any workstation connections will be dropped if they were made and return the user to the Main Menu.



Click on the Exit button located on the Main Menu to quit execution of the program.

SIECA PROGRAM 1995

THE USE OF SWINGBY AND ADVANCED VIDEO TECHNOLOGY TO MAKE A QUICKTIME MOVIE

SOFTWARE ENGINEERING BRANCH
AUTHOR : RONTRILL SWAIN
MENTOR : JON VALETT
CODE 552.3

TABLE OF CONTENTS

ABSTRACT

Swingby is a software tool used in analyzing and creating missions within the Flight Dynamics Division. It's ability to describe the orbital movements of a spacecraft with the moon, sun, earth and other gravitational bodies has provided analysts with the ability to plan numerous kinds of missions. Specifically, by modeling these orbital movements Swingby can plan a spacecraft's trajectory, illustrate how a spacecraft is transferred from a low earth parking orbit into geostationary orbit and demonstrate how a particular technique can be used to send a spacecraft to the Earth - Sun libration point.

The wide variety of uses for Swingby make it a valuable software program for the mission analysis community. Increasing awareness of Swingby will provide more analysts with these capabilities. In order to provide such increased awareness, Swingby was used to create 3 missions that provided reasonable images to be captured by advanced video application. These missions displayed different orbital movements of the spacecraft with respect to specific goals and variables of particular events.

Thus, this study is designed to increase the awareness of Swingby -- **A Mission Analysis and Design tool** -- by creating and executing sample SWINGBY movies on Code 550 World Wide Web site.

THE USE OF SWINGBY AND ADVANCED VIDEO TECHNOLOGY TO MAKE A QUICKTIME MOVIE

CODE 552.3

INTRODUCTION:

For the past few years, The usage of Swingby has provided analysts with several strong ideas about other areas to explore within the Spacecraft trajectory. By definition, Swingby is a mission analysis and design tool capable of designing missions that include the movements of a spacecraft in orbit with the moon, earth, sun, and other user defined bodies. In addition, by modeling these movements, Swingby can plan a spacecraft's trajectory as well as illustrate how a spacecraft is transferred from a low earth parking orbit into a geostationary orbit, a lunar trajectory, or the interior Earth - Sun libration point. All of these functions are wonderful features of Swingby; but there are a wider variety of uses for Swingby that makes it a valuable software program for the mission analysis community. Therefore, it became a need to setup a project with the intentions of increasing the awareness of Swingby by way of displaying sample Swingby movies on the World Wide Web.

PROJECT DESCRIPTION

The goal of this project was to create sample Swingby missions and make them available on the World Wide Web in order to increase the awareness of Swingby. This goal was accomplished in the following steps.

First, 3 missions were created and executed by Swingby. These missions were **HALO**, **DEMO**, and **WIND** missions. Each of these missions consisted of three basic objects that were provided with color for better interpretation. For example, the Moon is denoted by the large blue circle, the Sun is denoted by the straight yellow line, and the moving green line symbolizes the spacecraft in orbit.

Second, by using Swingby, these missions provided graphical images that were captured by an advanced video application i.e. video camera and placed into a VHS format.

Next, these images were converted to a quicktime movie through the use of an advanced video software program known as Adobe Premiere and later transformed into mpeg form using the Sparkle software application. Finally, these movies were then transported to code 550 WWW site for further observation.

MISSIONS CREATED AND ANALYZED

A brief description and pictorial display of the missions created is provided as the following :

MISSION 1 : HALO
Filename : RUN1.MIS (Regular - Run Mission without a Target).

This mission was primarily set up to take advantage of short filming of short distinct images provided by Swingby. In this mission, the orbital movement of the spacecraft starts near the moon. It then moves from its initial parking orbit close to the moon to a farther location away from the moon.

MISSION 1 (CONTINUED) - HALO
Filename : TARG1.MIS (Regular - Run Mission with a Target).

This portion of the mission takes on the same orbital movements as the previous. However, variables and goals were set. As a result, the completion of the mission is much faster and more efficient. It is also signicant to point out that each event or set of events such as the manuevering, propagating and finite burns within the targeter option is denoted by distinguishing colors to provide a distinction with respect to the movement of the spacecraft.

MISSIONS CREATED AND ANALYZED - (CONTINUED)

MISSION 2 - WIND

Filename : RUN1.MIS (Regular - Run Mission without a Target).

The WIND Mission was primarily set up to record a longer mission sequence. The time it took to complete this mission was tied to the number of events, in this case 45, that were attached to the motional movement of the spacecraft in orbit. This mission illustrates the spacecraft starting in orbit near the moon and manuevering to maintain that orbit until the mission was completed.

MISSIONS CREATED AND ANALYZED - (CONTINUED)

MISSION 3 - DEMO

Filename : RUN1.MIS (Regular - Run Mission without a Target).

This mission shows a spacecraft in a parking orbit about the earth then it propagates to the moon, orbits around the moon for a specific duration and exits back to the earth.

MISSION 3 (CONTINUED)

Filename : TARG1.MIS (Regular - Run Mission with a Target).

This portion of the mission follows the same plan as the Regular - Run Mission without a Target. However, this mission has established goals it must meet before completion.

SUMMARY / RESULTS

After the experimental steps were fully completed, the resulting product was a quicktime movie in MPEG form for each of the missions created and analyzed in this project.

These graphical movies now hold their own URL location on the World Wide Web Server under the address of **http://fdd.gsfc.nasa.gov/Swingby.html.** In addition, brief descriptions of each of the missions are provided that describe the motional movements of the spacecaft with the gravitational bodies involved in the images such as the moon, sun, and earth.

CONCLUSION

All in all, This project was setup to increase the awareness of Swingby and to demonstrate why the tool is a valuable software program to the mission analysis community by making sample Swingby missions available on the World Wide Web.

This project also illustrated that running software programs can be captured by video and converted into a useable movie format on the World Wide Web.

Special thanks ....

SIECA PROGRAM

Mentor : JON VALETT

DAVID MATUSOW

JOHN BRISTOW

DESIGN OF A LUNAR-BASED TELESCOPE

Ebony Alexis Waller

SIECA Progam

Mentors:
Ron Oliversen
Peter Chen

In the future, lunar telescopes will be essential to the progress of astronomical studies. These telescopes will enable scientist to observe objects far more fainter than what the earth-based telescopes can detect. Lunar-based telescopes have a potential for a large collecting area and for optical interferometry. With these applications, scientists can search for planets and extra stars. Lunar-based telescopes can be used for astrophysical research at very high angular resolutions or very faint objects. The telescopes can also be used for the detection of protostars, extrasolar planets and formation and dynamics of galaxies. Scientist have even looked into building an observatory but this is very unlikely to happen because it is expensive and difficult. Despite all of the advantages, there are problems that need to be addressed.

Cost is a very big problem. Building an observatory on the moon would assume a human presence there. This is highly unlikely in the near future. The cost for a manned mission to the moon is approximately $10 billion. Unmanned missions are not as expensive but they are still costly. For a one meter non-steerable (transit) telescope the approximate cost is $500 million.

There is also a lack of heavy launchers that can deliver a sizeable payload (tons) to the moon. Below is a table showing the vehicle, payload capacity and the cost to fly.

VEHICLE	PAYLOAD CAPACITY	COST ($M)
PEGASUS	0	10
PEGASUS XL/STAR	50 kg	15
TAURUS XLS/STAR 37FM	200kg	30
DELTA 7925	350kg	60
ATLAS IIAS/STAR 48B	700kg	120
SHUTTLE/IUS	700kg	?
TITAN IV/CENTAUR	1750kg	300

Since the moon is at a far distance, the launch vehicle will require more fuel. This will also add to the weight of the launch vehicle. Lunar-based telescopes must operate in a close to non-zero gravity enviroment. Therefore, more equipment is needed to stabilize the telescope and give it a low center of gravity. There could also be multiple launches to the moon that could bring up the telescope piece by piece. Astronauts would be required to set up the telescope and since this would not be feasible in the near future, robots are being researhed to do the job.

On the moon, the tracking rate must be 28 times slower than a telescope on the earth. The telescope must be able to move at 0.05 arcseconds or less. The night time temperature on the surface of the moon is 60 to 70 K. Telescope mechanisms must be able to operate in these cold temperatures. Also, mechanical bearings and lubricants wear out with use and would not be sufficient for applications on the moon. Lunar-based telescopes must operate in a cold, dusty vacuum for years or decades. There is also a need for a mount and bearing that can handle 200 kg or more. This is a difficult task.

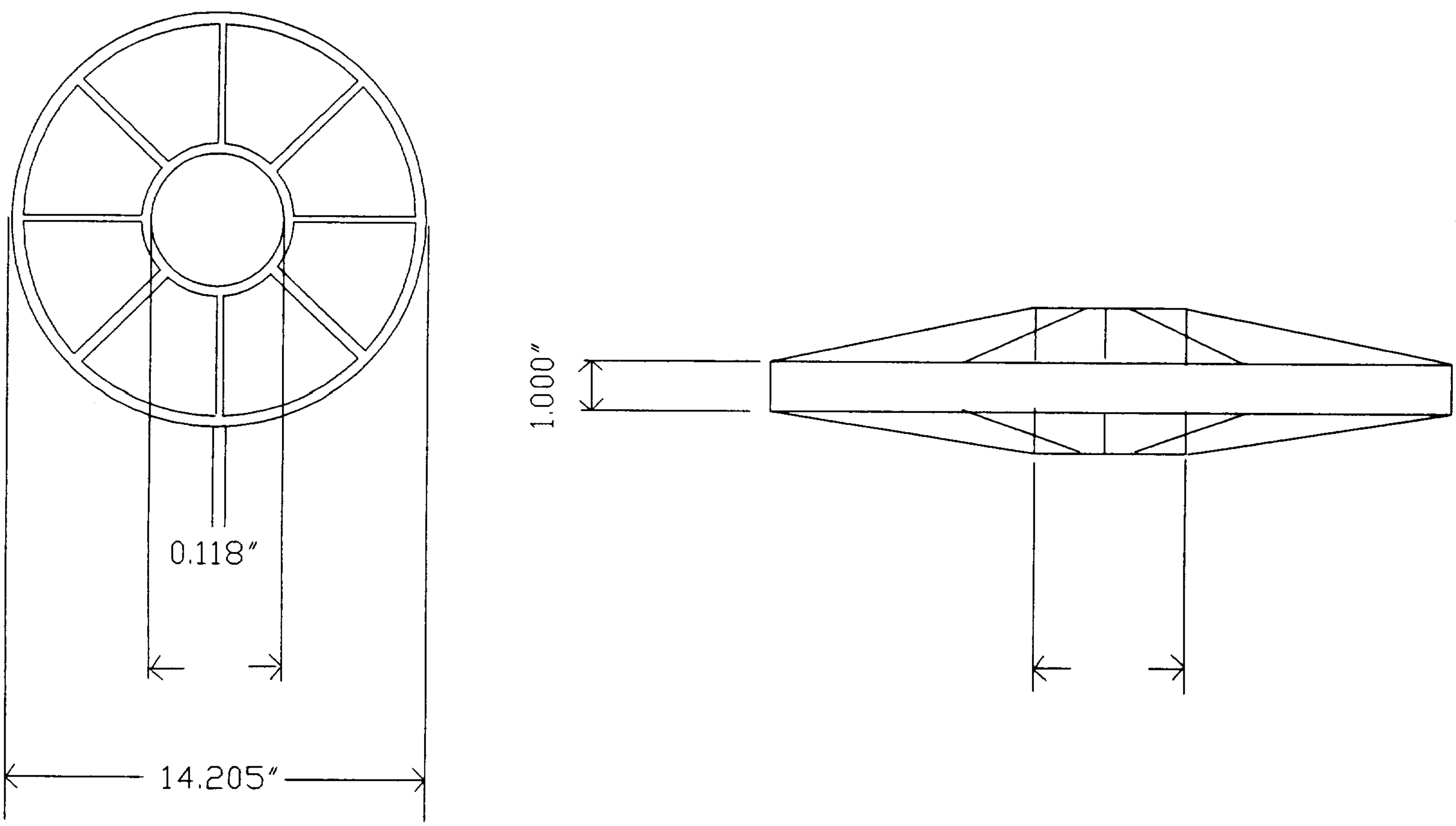
Radiation is also another big problem that needs to be addressed. Electronics that are on board are adversely affected by the high radiation levels that are present on the surface of the moon. The CCD (Charged Coupling Devices) camera used on the Hubble Space Telescope (HST) is very "soft" to radiation. There is also cosmic ray noise. On the HST, 0.1% of 800 by 800 pixels are affected in the shortest exposure time. On the moon, cosmic rays are 10 times higher than that of the HST while it is orbit. Shielding for the CCD cameras is not an option on the lunar surface. To shield the cameras, a sphere of tantalum with approximately a one meter diameter is needed. This shield would have a mass of approximately two metric tons. To transport such a heavy mass it would require the equivalent of four shuttle payloads. Another option is to bury the detector under two to three feet of densely packed lunar soil. This, too, is not an option due to lack of human presence and difficulty digging lunar soil. There is also a concern for solar flare radiation on the CCD cameras.

There is also a concern of dust control. There will be expected dust movement when the spacecraft descents. At this time, the telescope will need to be covered. The only other time that the telescope will be exposed to dust movement is during meteorite impacts. These are rare events that are not expected to occur. A telescope that is made of two to three millimeters of a tough composite material is enough shielding to protect the telescope from micrometeoroid impacts.

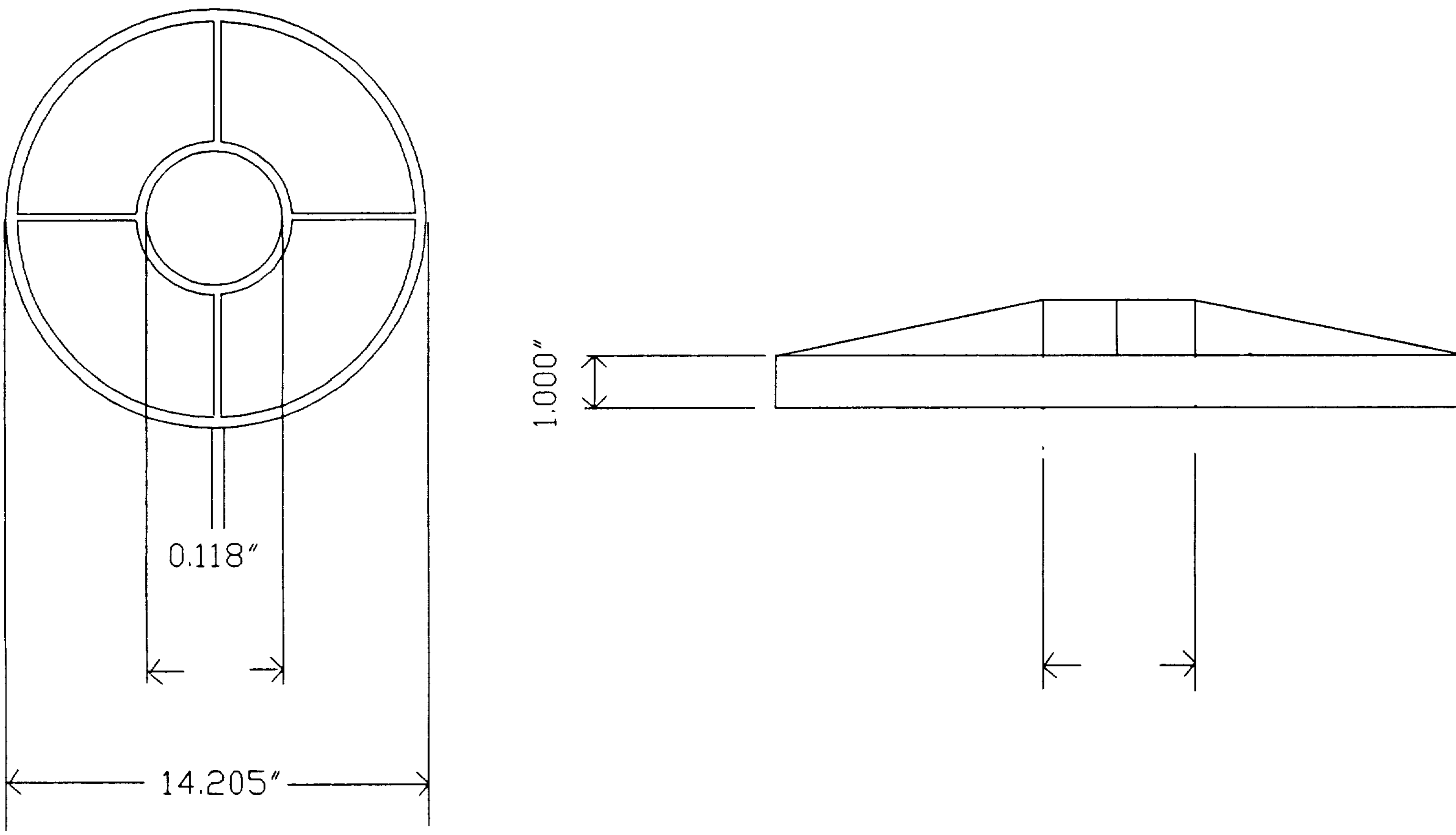
There are some solutions to the problems that exist. The biggest solution to the problem is to make the optics and materials as lightweight as possible. Extremely light weight mirrors can be used by utilizing the replication process. This process is simply just making an impression for a mirror using a mold. Mirrors have been made using graphite-epoxy. High Temperature superconductors can be used for telescope bearings. The absence of contact that occurs with gears can provide long term unattended operation without mechanical wear. The lunar environment is suited well for superconducting material because the temperature of the moon is at superconducting temperature (100 K). There are also radiation tolerant detectors that can be used. These cameras are know as Charged-Injection Devices (CID). CID cameras will allow scientist to monitor bright objects and be able to throw away the interference caused by cosmic ray events.

I was responsible for designing a lunar-based telescope. This telescope needed to meet several criteria. First, it needed to be lightweight. To do so, materials such as graphite-epoxy are used. Also, the replication process is used to create the primary and secondary mirrors used on the telescope. . Second, the telescope must be able to operate on its on once it is on the surface of the moon. It needs to do this because there areno manned missions to the moon or a space station on the moon where human involvement can take place. The telescope must also be self-deploying. It must be able to land on the surface of the moon (after being deployed from a rocket or some type of spacecraft) and set itself up for operation. Finally, it must be inexpensive to build, test and launch.

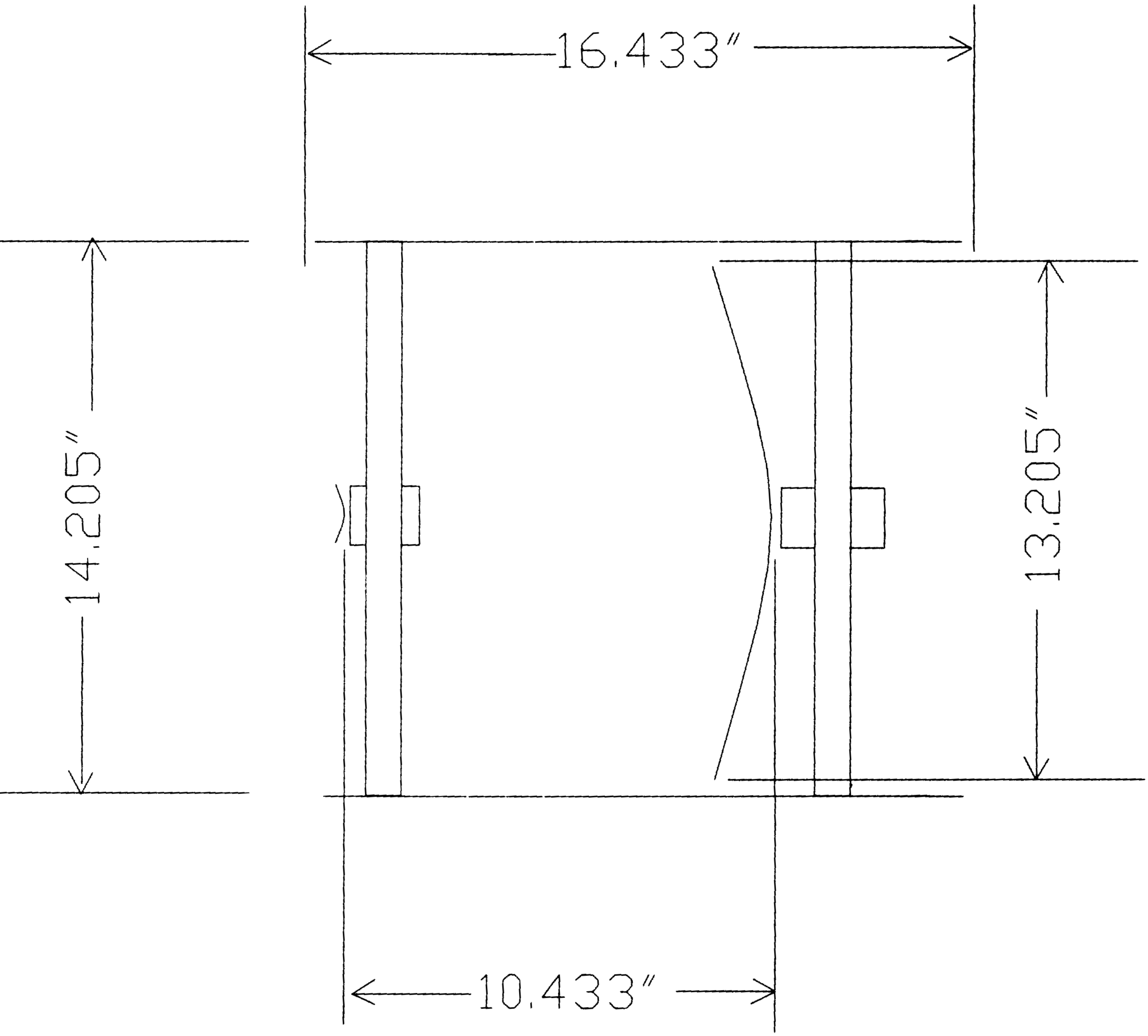
I was able to design a telescope that would satisfy these criteria. I also designed a detailed drawing of the spider structures found inside of the telescope. Spider structures hold the primary and secondary mirrors in place inside the telescope tubing. These structures help to reduce stress on the mirror. I was able to come up with a design for a superconducting bearing that would also be used to control the tracking of the telescope. I was unable to build a model of the telescope because the parts that were needed did not come in during the ten weeks that I was here at Goddard. I was also able to go to a telescope shop and look at possible mounts that I could be used to mount the telescope. I had the opportunity to take apart two telescopes, a 14 inch and a 19 inch telescope, and see the structural design. This also gave me better insight on how to design the telescope. I became famiar with a drawing program, Autosketch. This drawing program is similar to Autocad.



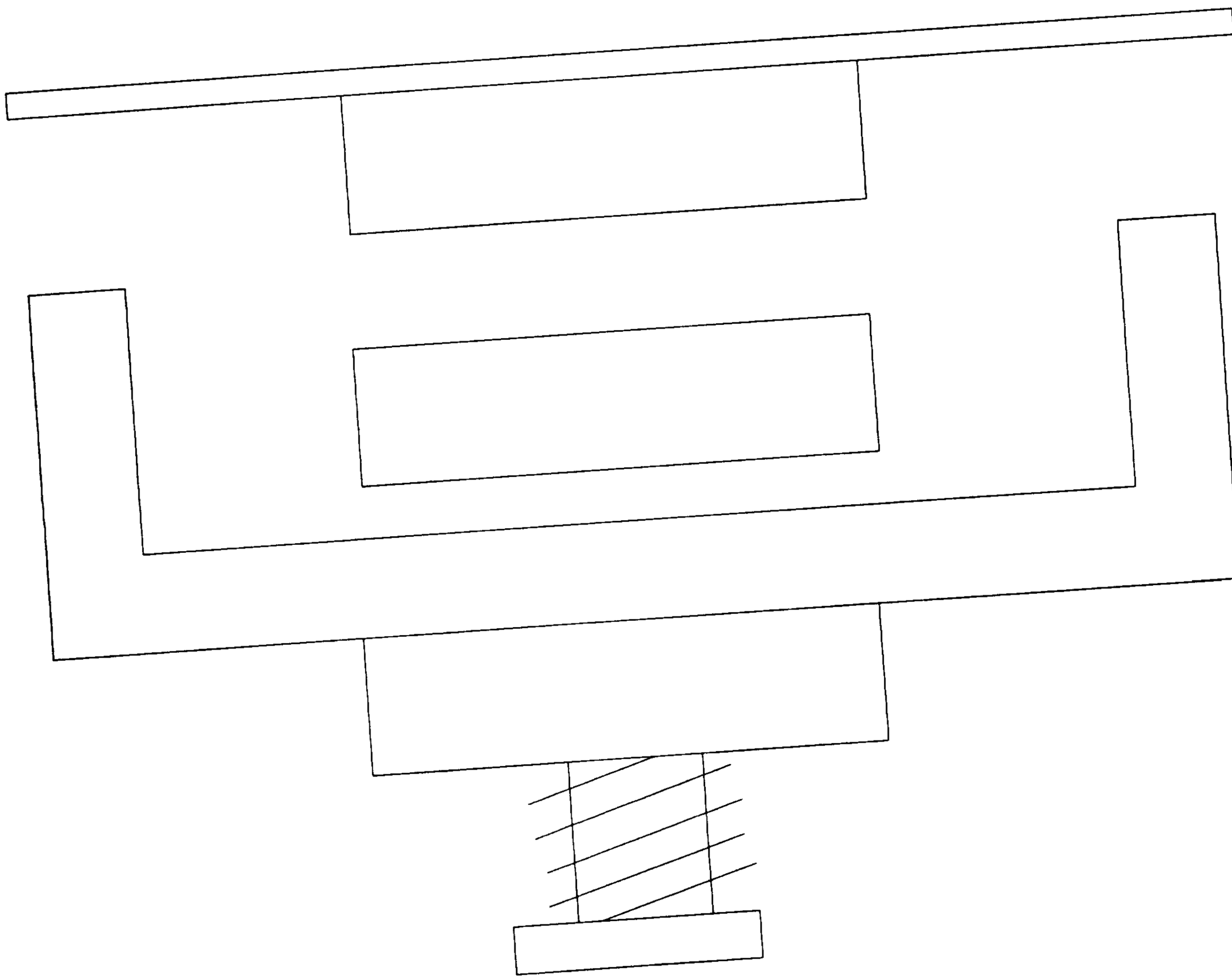
Spider for Primary Mirror



Spider for Secondary Mirror



Telescope (Side View)



Superconducting Bearin

Skylab

X-ray Images

Made Readily Accessible

Edwin Beckford
Dr. David Batchelor (mentor)
SIECA-UG Program
May 30, 1995 - August 4, 1995

Skylab X-ray Images Made Readily Accessible

My name is Edwin Beckford. I am a junior at Norfolk State University, Norfolk, Virginia majoring in Applied Mathematics/Computer Science Applications. I was one of fifteen undergraduate selected to participate in the Summer Institute in Engineering and Computer Applications (SIECA) Intern Program. The SIECA program is designed primarily to provide space-related research experience for minority undergraduate students. It came into existence during the summer of 1970 as the result of a proposal made by Bowie State University to NASA's Goddard Space Flight Center (GSFC) in Greenbelt, Maryland.

I was assigned to the Space Physics Data Facility (Code 632) under the mentor ship of Dr. David Batchelor, a renown astrophysics. The Space Physics Data Facility's (SPDF) main function is the development and operation of a range of programs serving the needs of the NASA and international space physics sciences communities. SPDF supports the Coordinated Data Analysis Workshops and the Satellite Situation Center software development effort. Of greatest interest in this are programs such as the International Solar_Terrestrial Physics program and the Inter-Agency Consultative Group solar-terrestrial research initiative that are central science thrusts within the Space Sciences Directorate. SPDF and its systems are playing key roles in the Space Physics Data System in collaboration with other elements of the Space Science Data Operations Office and the Space Science Directorate laboratories.

Before my arrival to GSFC, I knew that I would be using FORTRAN, C or Pascal to perform a Goddard space related project. I hoped to work with C, my second choice was FORTRAN, for I was already familiar with Pascal and preferred the challenge of something new and different. Since I owned a set of Pascal books I only purchased text for C and FORTRAN, I wanted to be ready of whatever. Never had I heard of

Goddard Space Flight Center (GSFC) nor Bowie State University. Therefore I began reading up on them and studying the maps I received from Goddard and the American Automobile Association. Although this was my first internship I had a pretty good feel for the lay out. I had my maps, text books, and was feeling very comfortable with DOS, Pascal, BASIC, IBM PC, WordPerfect, Quarto Pro, and dBase. I was ready to meet the challenge.

DAY 1

On my first day I was introduced to my terminal. It consisted of an X Window Ver 2, a Bolero client/server, and a UNIX operating system. Next I was informed that my project involved image processing of SKYLAB X-ray exposures that had to be reformatted from obsolete binary image data into modern Graphic Interface Files (GIFs). Then I was to review and give feed back on a couple of C programs, written by my mentor which needed debugging. One was designed to convert Extended Binary-Coded Decimal Interchange Code (EBCDIC) into ASCII and the other floating decimal to a hexadecimal integer array.

Day 2

On my second day, I was exposed to the World Wide Web (WWW), Hyper Text Multimedia Language (HTML), Mosaic and the netscape browsers. I was introduced to the vi and emacs editors, computer security, and told that my second project was to write a Web Home Page, using links, graphics, color backgrounds, images and a self photo. It was becoming apparent that I was in for more then I had anticipated.

Day 3

The next day I found my way to the GSFC Learning Center. I enrolled in UNIX, C, X-Windows, Image Processing, Bourne Shell Script and Internet classes to acquire some knowledge of these relevant subjects. I also began reading two books about Skylab, "A New Sun by John A Eddy and "A House In Space" by Henry Cooper. As for Bolero, my mentor taught me all I needed to know

After 2 Weeks

Within my first two weeks I had successfully completed all my courses and the two books. I also visited Gallery 111, titled "SPACE", at the Smithsonian's National Air and Space Museum. The courses prepared me with the technical background needed; the books and gallery briefed me about Skylab and enhanced my enthusiasm and excitement.

Now I was able to exchange ideas with my mentor. I was able to read his C programs and furnish him with some positive feedback. As a result of these sessions, we established a mutual respect for one another and a common bond. At last I was able to understand the task at hand and was ready to get busy.

Project 1

Skylab, the first long range orbital observatory, took over 200,000 exposures of the sun between May 1973 and February 1974. This was accomplished by using nine telescopes, each uniquely designed to capture images of the sun within pre-designated wavelengths. Two of these telescopes, the S-054 (wavelengths 2 to 60 A) and S-056 (wavelengths 6 to 33 A) provided the X-ray images. My project revolved around the images taken with the S-054. The objective of my project was to make these images readily accessible to the public though gif. files, internet, and CD ROM.

Prior to my arrival, David Batchelor (my mentor) had made it possible to view these images on PC screens. This required a complex sequence of case sensitive UNIX and IDL commands to be manually implemented.

I wrote a UNIX program, using Shell Script, that executed the sequence of UNIX commands upon typing only its file name at the UNIX prompt. Next, I wrote a IDL program, using the IDL buffer, that executed the sequence of IDL commands upon typing only its file name at the IDL prompt. Then I nested the UNIX program as a function of the IDL program, resulting in the display of an image upon typing only one word. This effectively eliminated syntax errors and saved valuable time in my future research.

At this stage, the images still could only be viewed and temporary gamma corrections be made. Therefore, I had to modify the IDL program to compile the current format into a gif file. This was a major undertaking and accomplishment.

At last, from EBCDIC to gif, the Skylab images taken by the S-054 telescope are readily accessible and will soon be on CD ROM.

Project 2

My second project was the presentation. In preparation I completed the following causes at the Learning Center: Technical Presentations, Macintosh Basics, and Macintosh WORD (I only knew IBM PCs and WordPerfect). I also had to learn how to use Adobe Photo Shop and scanners to make color transparencies. And I had to learn Power Point to make slides and transitions.

Project 3

My third project was to assembly a RS-4 Radio Receiver. The RS-4 is a ground-based INSPIRE receiver used for monitoring electromagnetic waves generated from a modulated electron beam on board the MIR Space Station. This is the results of a joint space physics research agreement between INSPIRE and IKI (The Russian Space Agency) known as the MIR-INSPIRE Project.

The RS-4 resistor came in kit form. It consisted of four bags of components. Bag 1 held the resistors. Bag 2 held the capacitors. Bag 3 held the IC, transistors, diode, and coil (transformer). And Bag 4 held the potentiometer, switches, jacks, plug, terminal strip, and miscellaneous hardware.

The assembly instructions began with the following message: "The following assembly instructions should be followed carefully. The RS-4 Receiver Kit is NOT a simple electronic assembly. If you follow the instructions carefully you should be successful in building a receiver that works. If you are not careful, you run the risk of having a problem that is very difficult to locate and fix. Be careful, take your time, and GOOD LUCK!"

It required that I identify the resistors by color bands converting the colors to their numerical equivalent. I had to solder components and wires, assuring that the capacitors were placed with the proper polarity. I had to overcome discrepancies and ambiguous statements. I had to reconstruct circuitry to derive at the required resistance. I made it work! I assembled it the day after obtaining it, just in time to be used at 2:15 am the following morning, August 1, 1995.

At about 1:30 am, August 1, 1995 I met with William Taylor, Director of Space Sciences at Nichols Research Corporation in Washington, DC, President of The INSPIRE Project, Inc. After testing the INSPIRE radio receiver ,which passed, we were ready to use it for reading frequencies emitting from the MIR (a Russian Space Station) that was to pass overhead precisely at 2:15.

This project was a total success. Dr. William Taylor was very impressed with by capabilities. He will also revise the assembly instructions in lieu of the discrepancies I encountered and the suggestions I made.

Again on August 4, 1995 at 12:30 am, I met with Dr. William Taylor to perform more readings and test. The MIR was to pass overhead at 1:07 am and we were all set up. While Dr. Taylor was monitoring and recording the frequencies I was filling in the Log Cover and Log sheets.

Log Cover Sheet:

Ground Station Leader - Dr. William Taylor

Start Time - 8/4/95 05107 UT (0100 CDT)

Describe site location - Hains Point, East Potomac Park, Washington DC

Site latitude - 38 degrees, 54 minutes, North

Site longitude - 77 degrees, 2 minutes, West

Site description - confluence of Potomac and Anacostia Rivers, nearest street lights 150 meters away

Receiver - R channel active

Recorder - DAT

Antenna - large loop, 3 meters vertical

Personnel:

Dr. Bill Taylor, Mrs. Taylor, Ms. Alma Smith, Mr. Tom Smith, Mr. Edwin Beckford

Log Sheet:

Time

Description (Omegas, Whistlers, Tweeks, and Sferics)

Miscellaneous Projects and Accomplishments

1. Contributed some ideas and concepts that will be incorporated into a MIR-INSPIRE type role playing game for K-10 students.
2. Assisted Danny Clark in troubleshooting and debugging the installation of a new color printer on Bolero.
3. On July 11, 1995 I represented Norfolk State University at a Collage Fair.

4. Made and posted a bulletin board depicting Skylab, its telescopes and images of the sun.