JHU/APL
7393-9331
13 MAR 97

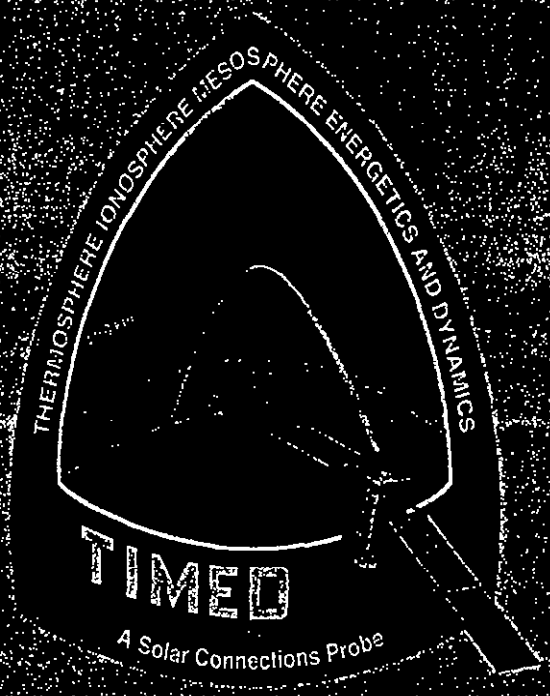
TIMED

Thermosphere - Ionosphere - Mesosphere - Energetics - Dynamics

Software Development Plan

for the

GPS Navigation Subsystem



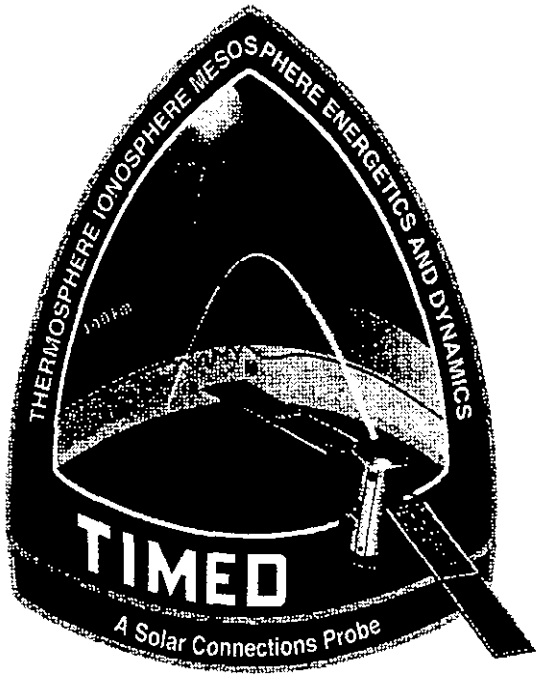
The Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20723

JHU/APL
7393-9331
13 MAR 97

TIMED

Thermosphere - Ionosphere - Mesosphere - Energetics - Dynamics

Software Development Plan
for the
GPS Navigation Subsystem



The Johns Hopkins University
Applied Physics Laboratory
Johns Hopkins Road
Laurel, MD 20723

Software Development Plan
for the
GPS Navigation Subsystem

Prepared by:
Al Chacos
Version 2.1
3/13/97

Approval:

G.E. Cameron
TIMED Program
Mission System Engineer

D. Y. Kusnierkiewicz
TIMED Program
Spacecraft System Engineer

H. K. Utterback
TIMED Program
Software Quality Assurance Manager

L. L. Suther
TIMED Program
Software Lead Engineer

TBD
Flight Software Segment
Software Lead Engineer

3/7/97
W. S. Devereux
GPS Navigation Subsystem
Subsystem Lead Engineer

3-14-97
R. J. Heins
GPS Navigation Subsystem
System Engineer

3/14/97
A. A. Chacos
GPS Navigation Subsystem
Software Lead Engineer

Table of Contents

1.0 INTRODUCTION

1.1 Purpose of the Document

The purpose of this document is to provide an overview of the software development effort for the TIMED spacecraft GPS Navigation Subsystem (GNS). It will describe the products provided by the software, the approach taken to design, implement and test the software, the resources available and the scheduling of those resources to complete the required tasks within the schedule and budget constraints. Although there will likely be changes to the information provided herein, there is no intention to revise and redistribute this document unless major changes are proposed and/or accepted by the program.

1.2 Scope of the Development Effort

The scope of the development effort includes the analysis, requirements definition, design, implementation and testing of all GNS flight software. The GSE (Ground Support Equipment) software required for subsystem integration and test, and pre-launch and post-launch analysis software, while described in this document for completeness, will primarily be procurements or deliverables to the GNS and therefore will not require any GNS provided documentation, build descriptions or software configuration management.

1.3 Overview of the Document

This document provides an overview of the functional requirements of all software related to the GNS, including both flight software and ground software (GSE and analysis). It describes the development, flight and operations computer environments, facilities and resources, and details the line, program and subsystem organizational structures as related to the GNS software development process. This document includes a high-level TIMED spacecraft schedule, high-level GPS subsystem schedule, and GPS subsystem software schedule which define all major milestones and design reviews. In addition, this document describes the personnel allocation, configuration management and documentation provided.

2.0 APPLICABLE DOCUMENTS

1. TIMED GPS Receiver System Requirements, R. J. Heins, SEA-97-xxxx
2. Development Plan for GLT Processing Functions, L. A. Linstrom/R. J. Heins, S2A-95-0118, October 26, 1995
3. TIMED GPS System Mini-CoDR, W. S. Devereux, SEA-96-0060, July 2, 1996
4. TIMED Software Quality Assurance Plan, L. L. Suther and H.K. Utterback, 7363-9101, September 1996
5. Space Department Software Quality Assurance Guidelines, SDO-9989, October 22, 1992
6. Implementing Software Engineering Practices, Fletcher J. Buckley, ISBN 0-471-63386-0
7. TIMED Program GPS Subsystem Software Schedule Rev1, SEA-96-0097, 13 November 1996

3.0 FUNCTIONAL REQUIREMENTS OVERVIEW

3.1 System Overview

The GNS shall provide the TIMED spacecraft with an autonomous navigation capability; namely, position, velocity and time provided on a one second periodic basis. In addition, event flags will be modified to identify the realtime occurrence of defined events, event tables will be generated to identify the predicted times of defined future events and TIMED ephemeris element set tables will be generated for use by the ground system for both antenna pointing and operations planning. Due to the desire for code portability, it should be noted of the intention that all flight software be written in ANSI C (and possibly C++ if available), and that the use of vendor specific language extensions or assembly language shall be permitted only if it can be shown that its use in certain tasks is deemed necessary and/or advantageous. Likewise, utilization of the operating system API's (Application Programming Interfaces) will be isolated into separate modules.

3.2 Realtime Products

The GNS will output position, velocity, time, earth-sun vector and defined event states once a second (valid on the next GNS generated 1PPS output).

3.3 12-Hour Products

Every 12 hours, the GNS will generate and output a table of predicted occurrences of defined events and a table of orbital element sets for ground antenna pointing and operations planning for a defined future interval of time.

3.4 Autonomous Operation

The GNS shall have the capability to operate autonomously, such that, upon power-up, no external commands or data are required for the subsystem to acquire and track GPS satellites, navigate, and continuously generate & output all specified output products.

3.5 1PPS Steering

The GNS shall steer the subsystem generated 1PPS signal such that it is aligned to UTC 1-second epochs.

3.6 On-orbit Reprogramming

The GNS shall implement the capability for receiving command packets & reprogramming all of the subsystem flash memory-based flight software (except the flight boot block software).

3.7 Software Quality Assurance

The GNS shall meet all requirements of the TIMED Software Quality Assurance Plan.

4.0 DESCRIPTION OF MAJOR FLIGHT SOFTWARE TASKS

4.1 Embedded Operating System

The embedded operating system will be Accelerated Technologies, Inc Nucleus Plus which will provide relatively low-cost, reliable functionality such as interrupt handling, interprocess communications, task scheduling and task prioritization.

4.2 Subsystem Executive

This task shall provide all input/output capabilities such as command packet ingestion/parsing and telemetry packet data building/transmission. This software shall also provide the high-level 'supervisory' role for all flight software including subtasks such as power-up and shutdown sequencing, mode selection (such as operational, test or other modes), and system & data integrity monitoring.

4.3 Tracking Loops

This task will provide the low-level interaction with the tracking hardware. This interaction provides the functionality of closing approximately (10) C/A code tracking loops & (10) doppler tracking loops, and extracting GPS message bits. This software may receive tracking aid inputs from the Acquisition Aids Generator software, and shall output a tracking quality factor, tracking modes, an interrupt count and message bits for each of the GPS spacecraft being tracked. This task is an area which is under study in order to evaluate design tradeoffs.

4.4 GPS Message Subframe Builder

This task will input GPS message bits from the Tracking Loop software, build GPS message subframes, perform error checks on the data, and output error-free subframes and subframe timing information.

4.5 GPS Message Subframe Parser

This task will input GPS message subframes from the GPS Message Subframe Builder software, extract the various data fields, and build GPS message data tables.

4.6 Measurement Data Processor

This task will input measurement data from the tracking hardware, input data from the Tracking Loop software & GPS Message Subframe Builder software, and input data from the GPS message tables. The software will process this data and then output a table consisting of time, pseudo-range, and integrated carrier for each of the GPS satellites being tracked.

4.7 Receiver Clock Manager

This task will maintain the GPS receiver clock and a UTC clock.

4.8 1PPS Steering Control Loop

This task will maintain alignment of the GPS steered 1PPS output signal to UTC 1-second epochs to within a specified maximum error.

4.9 Navigation

This task will utilize a Kalman Filter and a 4th order Runge-Kutta orbit propagator with gravity and drag force models (and possibly solar radiation and third body/ earth tides models as well) for estimating position, velocity and time. The software will input the pseudo-range & integrated carrier data, input data from the GPS message tables, and input receiver time, and will output a table of position, velocity, time, and earth-sun vectors. This task is an area which is under study in order to evaluate design tradeoffs.

4.10 Acquisition Aids Generator

This task will determine and build a priority table of the GPS satellites in view. The software will also generate doppler and doppler rate information for all the GPS satellites in view, as well as, those which are about to 'rise' in order to improve the acquisition/tracking process.

4.11 Orbit Estimation/Event Prediction

This task will propagate the current position and velocity out for approximately 4 days every 12 hours. While propagating the orbit, the software will determine if any of a defined set of events will occur and then build a table of these events. This software will also look for the occurrence of any of a defined set of events in realtime in order to set output flags indicating the state of these events.

4.11 Orbit Ephemeris Generator

This task will input the propagated position and velocity data set generated by the Orbit Estimation/Event Prediction software in order to fit the data and generate defined orbital element sets.

4.12 Built-In Test Software

This software shall provide the capability to test the operation of as much of the GPS subsystem hardware and software as possible.

5.0 DESCRIPTION OF GROUND SOFTWARE TASKS

5.1 Algorithm Evaluation Software

This shall be a combination of procured and custom software which will be used in algorithm tradeoff studies to determine the optimal algorithms to be used in the GNS subsystem software. Issues such as which models need to be considered (e.g. gravity, drag, solar radiation, earth tide, third-body interaction - moon & sun) are being investigated utilizing orbital simulation programs (e.g. BG14 Orbital Integrator Program). Other studies which are currently underway or will be performed include determining tracking loop algorithms, designing the event prediction algorithms, and determining the optimal orbital element parameter sets required for predicting position.

5.2 Ground Support Equipment (GSE) Software

This software shall provide the capability for generating and sending command packets to the GNS subsystem and receiving, decoding and displaying of telemetry data packets from the GNS subsystem. The telemetry data shall be displayed in both tabular and graphical formats. This software is in addition to a procured GPS constellation simulator consisting of a hardware unit, (2) computers and associated software, and a Mini-MOC hardware and software system (deliverable to the GNS) which provides a very extensive command & telemetry capability which is intended to supplant the preliminary command & telemetry software described in this section.

5.3 Data Analysis Software

5.3.1 Prelaunch Verification and Validation

This software will provide the capability for completely and thoroughly testing the flight software prior to launch. This will rely heavily on the use of a high-fidelity GPS constellation simulator.

5.3.2 Postlaunch Data Analysis

This software shall process downlinked time-tagged raw pseudo-range & integrated carrier data and message data, compute position, velocity, time, and earth-sun vector and compare these values against downlinked time-tagged position, velocity, time and earth-sun vector values. In brief, we will re-navigate on the ground with the raw data and verify that the same results are observed.

6.0 SOFTWARE ENVIRONMENTS

6.1 Development Environment

6.1.1 Development Computer Hardware

6.1.1.1 Host and S/W Development Computers

(4) Gateway 2000 Inc. Pentium Pro computers with 17" monitors, 32 MByte RAM, 2 GByte hard drives, 12x CD-ROM drives, IOMEGA ZIP drives, and tape drives for system backups.

6.1.1.2 Development Computer Board #1

Integrated Device Technology Inc. R3081 Evaluation Kit. This board will allow for early-on evaluation of the embedded operating system and development tools.

6.1.1.3 Development Computer Board #2

LSI Logic Pocket-Rocket development board. This board is baselined to be used for performing tradeoff studies of the tracking hardware and software prior to committing to a long-term custom integrated circuit fabrication cycle.

6.1.1.4 Development Computer Board #3

Synova, Inc. Turbo-Rocket development board. When this board becomes available, the tracking hardware and software studies will migrate to this computer from the Pocket-Rocket.

6.1.1.5 Development Breadboard

The breadboard is an APL custom design which will be used to test prototypes of the tracking hardware and software, and 'shake-out' the design of the GPS subsystem computer and RF hardware.

6.1.1.6 Engineering Model Board

The engineering model board is an APL design which will meet the 'form, fit & function' of the flight board, but will not necessarily use flight qualified parts.

6.1.1.7 Flight Board

The flight board is an APL design which will be used to perform a final hardware/software acceptance test prior to delivery to the spacecraft.

6.1.2 Development Computer Operating System

Windows NT v4.0 will be the PC operating system. If determined to be desireable, a Windows NT/Windows for Workgroups or Windows 95 dual-boot configuration will be utilized.

6.1.3 Development Computer Software Tools

6.1.3.1 Editor

Premia Corp. CodeWrite v4.0 Professional Editor.

6.1.3.2 Compiler/Linker

BSO/Tasking Inc. CP2002 R3000 Family ANSI C Compiler/ Assembler/Linker Package.

6.1.3.3 Debugger/Monitor

BSO/Tasking Inc. TK2041 CrossView ROM Monitor Debugger - Windows version. This will also provide a boot-loader capability for downloading flight code from the development system to the flight (target) system and will be burned into the processor Console Boot PROM.

6.1.3.4 Software Configuration Management

Mortice Kern Systems Inc. MKS Source Integrity v7.2 will be used to provide overall software configuration management, including version control, build management, project snapshots, and protection from inadvertent loss or overwrite.

6.1.3.5 Miscellaneous Software Tools

For initial software development, Microsoft Visual C++ (ANSI C development only) will be used to take advantage of its very user friendly and capable integrated development environment. MATLAB may be used for various analysis or plotting needs.

6.2 Flight Environment

6.2.1 Flight Computer Hardware

Synova Inc. Mongoose-IV or V MIPS R3000 Rad-Hard Processor with a MIPS R3010 Floating-Point Unit (FPU).

6.2.2 Flight Computer Operating System

Accelerated Technology, Inc. NU930106 Nucleus PLUS Embedded Real-Time Operating System. A predecessor of this operating system is being used on Hubble Space Telescope flight hardware, as well as, heart monitors and X-ray machines.

6.3 Mission Operations Environment

6.3.1 Mission Operations GNS Computer Hardware

This computer will be a PC to support on-orbit checkout and monitoring of the GNS.

6.3.2 Mission Operations GNS Computer Operating System

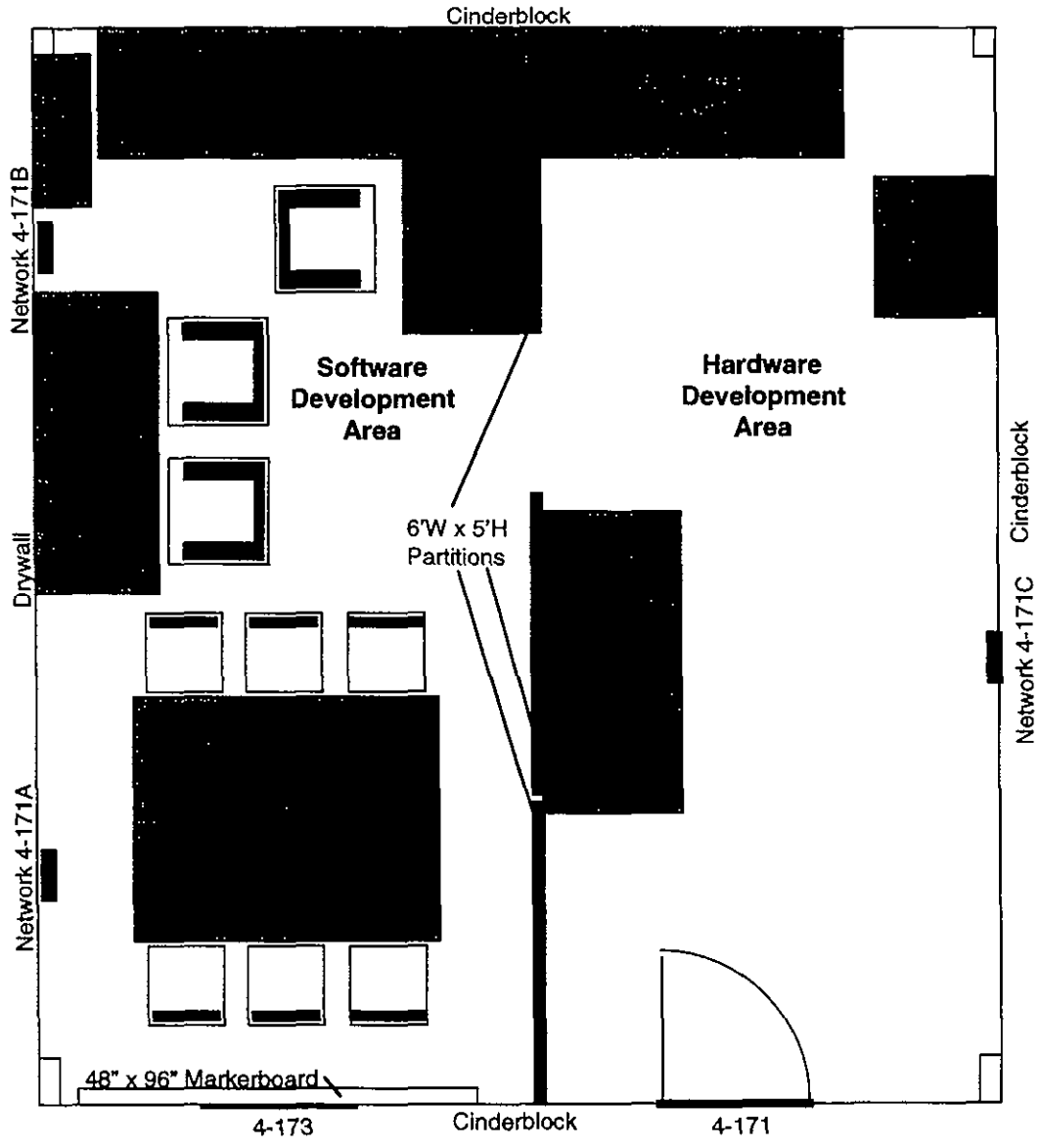
Windows NT v4.0 will be the PC operating system. If determined to be desireable, a Windows NT/Windows for Workgroups or Windows 95 dual-boot configuration will be utilized.

7.0 FACILITIES AND RESOURCES

7.1 Development Space Allocation

Lab space has been allocated for the development of both hardware and software for the GPS subsystem as shown below.

TIMED Program
GPS Navigation Subsystem
Development Room



7.2 Computer Resources

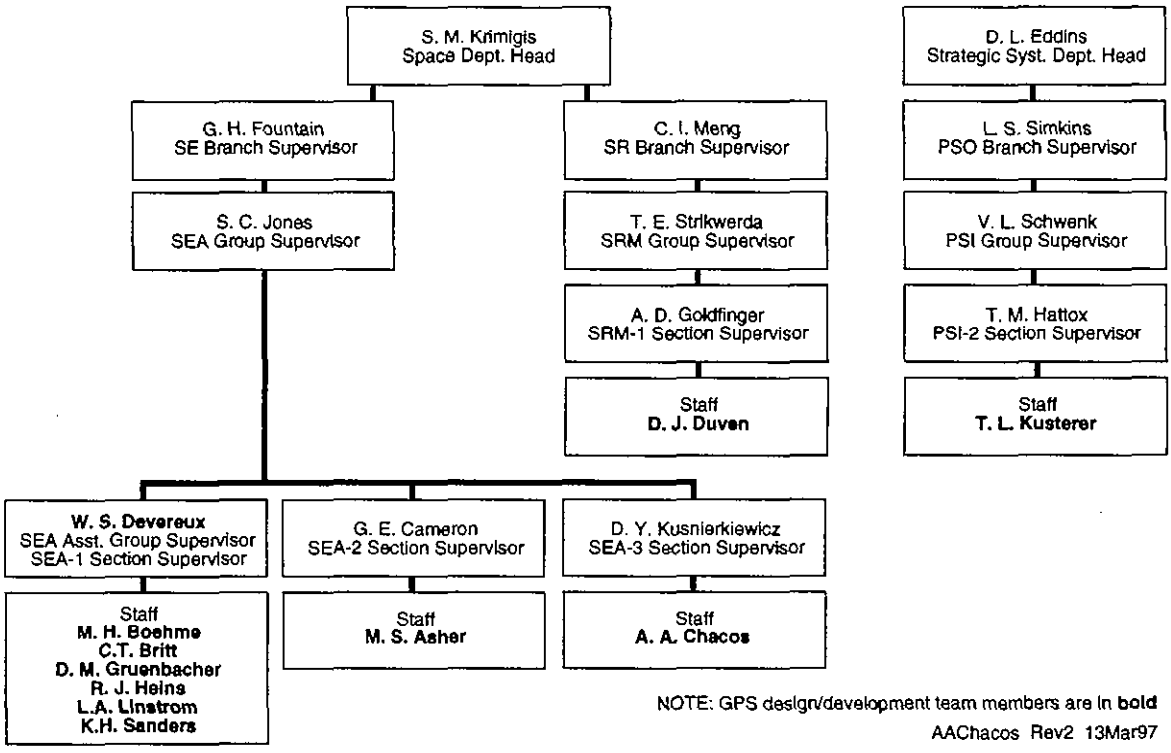
A variety of IBM-compatible computers, Apple Power-PC's, workstations, etc. are available for running simulations and tests.

8.0 ORGANIZATIONAL STRUCTURE

8.1 Line Organization

The following diagram indicates where all GPS subsystem team members are located within the line organization.

TIMED Program
GPS Navigation Subsystem
Line Organization



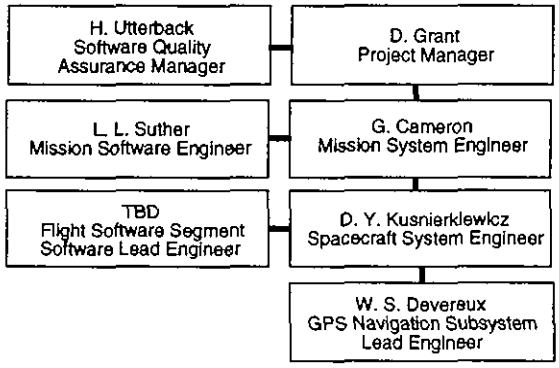
NOTE: GPS design/development team members are in **bold**

AAChacos Rev2 13Mar97

8.2 Program Organization

The following diagram indicates the subset of the program organization as it applies to the GPS subsystem software.

TIMED Program Organization

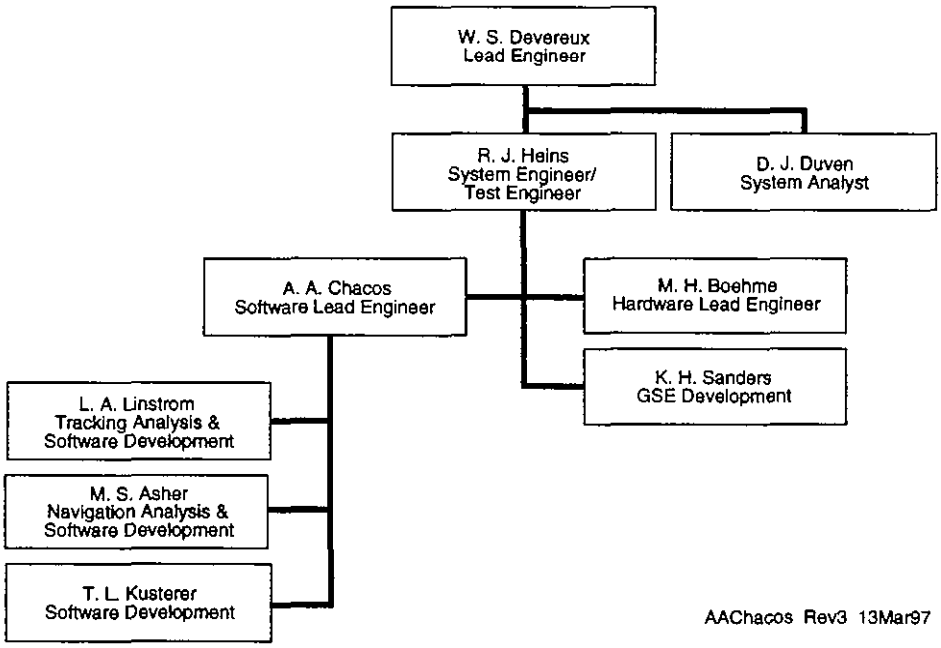


AAChacos Rev2 13Mar97

8.3 GPS Subsystem Organization

The following diagram indicates the subset of the GPS subsystem as it applies to software development.

TIMED Program
GPS Navigation Subsystem
Organization



9.0 MANAGEMENT CONTROL

9.1 Schedules

9.1.1 Program Schedule Major Milestones

GPS Mini-Conceptual Design Review	Jun 1996
Mission Preliminary Design Review	Feb 1997
Mission Critical Design Review	Dec 1997
Spacecraft Integration & Test Start	Oct 1998
Spacecraft Pre-Environmental Review	Jul 1999
Spacecraft Shipment to GSFC	Aug 1999
Spacecraft Pre-Ship Review	Oct 1999
Spacecraft Shipment to VAFB	Oct 1999
Spacecraft Flight Readiness Review	Dec 1999
Launch	Dec 1999

9.1.2 GPS Subsystem Schedule Major Milestones

GPS Subsystem development board available	Feb 1997
GPS Subsystem PDR	Apr 1997
GPS Subsystem breadboard available	Jun 1997
GPS Subsystem flight board #1 complete	Nov 1998
GPS Subsystem system-level test	Nov 1998
GPS Subsystem delivery to IEM chassis	Nov 1998
GPS Subsystem flight board #2 complete	Jan 1999

9.1.3 GPS Subsystem Software Schedule

GPS Software Requirements Review	Apr 1997
GPS Software Preliminary Design Review	Apr 1997
GPS Software Detailed Design Review	Various
GPS Software Integration Readiness Review	Oct 1998

9.1.4 GPS Subsystem Detailed Software Schedule

Refer to section 2.0 Applicable Documents, item 7.

9.2 Personnel Allocation

9.2.1 GNS Subsystem Overall Software Integration

L. A. Linstrom has the overall responsibility for the top-level integration of all GPS subsystem software tasks.

9.2.2 Embedded Operating System

L. A. Linstrom has the responsibility for procuring, configuring and testing the embedded operating system used on the flight processor.

9.2.3 GNS Subsystem Executive Software

L. A. Linstrom has the responsibility for the design, implementation, integration and testing of the 'Executive' software task.

9.2.4 Tracking Loop Software

L. A. Linstrom has the responsibility for the analysis, design, implementation, integration and testing of the 'Tracking Loop' software task.

9.2.5 GPS Message Subframe Builder Software

L. A. Linstrom has the responsibility for the design, implementation, integration and testing of the 'GPS Message Subframe Builder' software task.

9.2.6 GPS Message Subframe Parser Software

L. A. Linstrom has the responsibility for the design, implementation, integration and testing of the 'GPS Message Subframe Parser' software task.

9.2.7 Measurement Data Processor Software

M. S. Asher has the responsibility for the design, implementation, integration and testing of the 'Measurement Data Processor' software task.

9.2.8 Receiver Clock Manager Software

L. A. Linstrom has the responsibility for the design, implementation, integration and testing of the 'Receiver Clock Manager' software task.

9.2.9 1PPS Steering Control Loop Software

L. A. Linstrom has the responsibility for the analysis, design, implementation, integration and testing of the '1PPS Steering Control Loop' software task.

9.2.10 Navigation Software

M. S. Asher has the responsibility for the analysis and design of the 'Navigation' software task (working with D. J. Duven and T. L. Kusterer). T. L. Kusterer has the responsibility for the implementation, integration and testing of the 'Navigation' software task (working with M.S. Asher).

9.2.11 Acquisition Aids Generator Software

M. S. Asher has the responsibility for the analysis and design of the 'Acquisition Aids Generator' software task (working with T. L. Kusterer). T. L. Kusterer has the responsibility for the implementation, integration and testing of the 'Acquisition Aids Generator' software task (working with M.S. Asher).

9.2.12 Orbit Estimation/Event Prediction Software

M. S. Asher has the responsibility for the analysis and design of the 'Orbit Estimation/Event Prediction' software task (working with D. J. Duven and T. L. Kusterer). T. L. Kusterer has the responsibility for the implementation, integration and testing of the 'Orbit Estimation/Event Prediction' software task (working with M.S. Asher).

9.2.13 Orbit Ephemeris Generator Software

M. S. Asher has the responsibility for the analysis and design of the 'Orbit Ephemeris Generator' software task (working with D. J. Duven and T. L. Kusterer). T. L. Kusterer has the responsibility for the implementation, integration and testing of the 'Orbit Ephemeris Generator' software task (working with M.S. Asher).

9.2.14 Built-In Test Software

L. A. Linstrom has the responsibility for the design, implementation, integration and testing of the 'Built-In Test' software task (working with M. S. Asher and T. L. Kusterer). This includes all system integrity and data integrity monitoring software.

9.2.15 Algorithm Evaluation Software

L. A. Linstrom, and M. S. Asher have the joint responsibility for the initial design and implementation of any custom 'Algorithm Evaluation' test software and the procurement of any required software analysis packages. D. J. Duven will be taking on this responsibility around the Preliminary Design Review time. This primarily is utilizing BG14 and designing and implementing MATLAB simulations of tracking and navigation software.

9.2.16 Ground Support Equipment (GSE) Software

TBD has the responsibility for procuring, designing, implementing, integrating and testing software for the GSE system which is required to support the design, development, and pre-launch operation of the GNS subsystem breadboard and flight board designs.

9.2.17 Data Analysis Software

L. A. Linstrom, M. S. Asher, and T. L. Kusterer have the joint responsibility for the design and implementation of any custom 'Data Analysis' software and the procurement of any required software analysis packages required for pre-launch and post-launch testing and performance evaluation of the GNS subsystem.

9.3 Flight Software Builds

9.3.1 Build Overview

It is intended that the following six builds, as a minimum, will be provided during the development process: GNS subsystem executive software build, tracking software build, navigation software build, IEM integration build, spacecraft integration build, and final flight build.

9.3.2 GNS Subsystem Executive Software Build

This build provides the environment and routines which are utilized by all other software executing in the subsystem. All I/O drivers, and command and telemetry data handling will be provided in this build.

9.3.3 Tracking Software Build

This build provides the ability to track multiple GPS satellites simultaneously with or without acquisition aids.

9.3.4 Navigation Software Build

This build provides the ability to estimate position, velocity, time and sun vector on a 1 Hz periodic basis. The input data for testing this build can either be generated by the GNS tracking software, or be from a previously generated file containing formatted measurement data.

9.3.5 IEM Integration Software Build

This build will provide the capability required to verify proper operation within the IEM. This will be provided prior to scheduled IEM integration tests with the GNS subsystem.

9.3.6 Spacecraft Integration Software Build

This build will provide the capability required to verify proper operation within the initial spacecraft environment. This is expected to be all software except Orbit Estimation/Event Prediction and Orbit Ephemeris Generator software. This will be provided prior to scheduled spacecraft integration tests with the GNS subsystem.

9.3.7 Final Flight Software Build

This build will provide a complete and fully tested GNS software system. This will be delivered prior to the spacecraft pre-environmental review.

9.4 Flight Software Configuration Management

9.4.1 Responsibility

Selection and implementation of the GNS flight software configuration management, as well as, coordination with program Software Quality Assurance managers for audits is the responsibility of the GNS lead software engineer.

9.4.2 Software Source Code Management and Archiving

Configuration management of the flight software effort will include work-in-progress snapshots, flight software build management, version control, separation/isolation of code in various stages of development, and protection from inadvertent overwrites or deletions. This will provide the capability to preserve evolving revisions of source code and resource files, enabling the re-creation of past versions of the project for error tracking/debugging purposes, provide for the separation of code being developed, code being tested and completed code, and protection of work in progress from inadvertent loss. It should be noted that due to the small size of the software implementation team (3-4 coders max.), no additional features such as control over shared files or security/access controls will be implemented.

9.4.3 Formal Software Configuration Control

Formal software configuration control will commence after the baseline software delivery (spacecraft integration software build) to the spacecraft.

9.5 Flight Software Documentation

In addition to this software plan, it is intended that the following documentation will be provided for all flight software:

Requirements/Design Matrix
Internal Interface Control Document
External Interface Control Document
Top-Level Design Document
Detailed Design Document
Unit Development Folders

Acceptance Test Plan (as part of the overall subsystem test plan)

Some of these individual documents may be combined to help minimize the effort required for their generation/distribution. Configuration management of the flight software documentation will be controlled by the GNS Lead Software Engineer. Software listings will be the primary documentation for all GSE and analysis software and will be maintained by the developer.

9.6 Verification and Validation (V&V)

9.6.1 Criticality

The program office has designated the GNS subsystem level of criticality as a 'Mission Impact' software system. Although, according to the Space Department Software Quality Assurance Guideline document independent V&V (IV&V) is not required for this level of criticality, the program office has directed that IV&V be performed on the GNS flight software.

9.6.2 Deliverability

The GNS flight software is considered to be a deliverable to the sponsor. In addition, the GNS GSE and associated ground software is also considered to be a deliverable (in-place) to the sponsor for performing on-orbit performance characterization.

9.6.3 Technical Risk

The GNS software is considered to have minimal technical risk. The algorithms which will be implemented (or variants thereof) have been used in industry and at APL for a number of years. The individuals responsible for the design and implementation of those algorithms have a significant background and level of experience in the same and related fields.

9.6.4 Development Risk

The GNS software is considered to have minimal development risk. Studies have been performed and simulations have been run (and more will be run) in order to complete design tradeoff studies. These studies and simulations have shown that the design requirements can be achieved; additional studies and simulations will be performed in order to try to optimize the performance of the GNS subsystem. An associated development risk which is considered to be between a minimal and moderate level is related to CPU utilization issues. The processor type and processor clock speed have not yet been determined, nor will a breadboard be available in the near future for running baseline benchmark tests. This may result in a delayed decision to add a second processor to the GNS in order to meet system requirements, thereby adding hardware and software complexity and related impacts on the GNS development schedule.

9.6.5 Detailed Design Reviews

Detailed design reviews of the flight software are required by the TIMED program Software Quality Assurance Manager (SQAM) and shall be supported by the GNS subsystem software developers. In order

to optimize resource scheduling, a number of reviews of task-level software scope will be organized instead of a single all encompassing review. No detailed design reviews will be held for the ground-based GSE and analysis software.

9.6.6 Code Walkthroughs

Code walkthroughs of the flight software are required by the TIMED program SQAM and shall be supported by the GNS subsystem software developers. In order to optimize resource scheduling, a number of reviews of task-level software scope will be organized instead of a single all encompassing review. The SQAM has directed that the successful completion of a code walkthrough precede unit testing. No code walkthroughs will be held for the ground-based GSE and analysis software.

9.6.7 Unit Testing

Unit testing shall be performed for all flight software developed for the GNS subsystem, but formal documentation and review is not required nor will it be provided. Unit testing informal documentation and reviews shall be required at the subsystem development team level, and details of the testing shall be maintained in the Unit Development Folders.

9.6.8 Testing Responsibilities

The software implementers are responsible for the unit testing of their software and shall keep a record of the testing in their Unit Development Folders. The GNS Software Lead Engineer shall be responsible for the oversight of all software testing, i.e. verifying that all the software has been thoroughly tested and is correctly and efficiently implemented. The GNS System Engineer will be responsible for verifying that the software system has met all requirements and is successfully executing the required tasks.

9.6.9 Software Change Control

9.6.9.1 Software Requirements Change Control

The TIMED program Mission Software Engineer shall be maintain oversight of all software requirement changes and additions.

9.6.9.2 Software Implementation Change Control

After formal software configuration control commences, any changes to delivered software will require processing of a Software Problem/Change Report (SPR). The SPR will specify the change requested and the reasons for why it is required. The SPR will be reviewed by a Configuration Control Board (CCB) selected and convened by the SQAM.

9.6.9.3 Final Changes

At the direction of the TIMED program Mission Engineer, no changes or additions to the flight software will be permitted after the Environmental Readiness Review unless deemed critical by the program office.

9.6.10 Testing Tools

9.6.10.1 Hardware Tools

A procured GPS constellation simulator will provide a key capability for verifying that the software is performing as intended. The simulator will generate GPS signals similar to those which will be received while on-orbit. A logic analyzer will provide a capability for non-intrusive testing which is important for time-critical tasks.

9.6.10.2 Software Tools

Procured software tools will be used to provide added visibility into the software implementation and testing. The tools will help define a minimum number of tests required to verify the software, provide an indication of the degree to which the software is structured, determine the portability of the code, as well as a number of other capabilities.

9.7 Independent Verification and Validation (IV&V)

9.7.1 IV&V Overview

In order to minimize costs associated with the IV&V, the effort will be partitioned into three parts; namely, an algorithm review, a software implementation review, and a 'black-box' subsystem characterization and performance evaluation review. The first two parts will be performed internally within APL and the third part will be performed by an organization specializing in the testing of GPS systems.

9.7.2 Algorithm Review

A detailed review of all algorithms implemented in the GNS subsystem shall be reviewed by an individual or individuals within APL who have a background and a significant level of experience with similar systems. To maintain independence, the selected reviewers shall have had no knowledge of the specific implementation of the GNS design until the review is underway, after which contact between reviewers and developers shall be controlled.

9.7.3 Software Implementation Review

A detailed review of all software in the GNS subsystem shall be reviewed by an individual or individuals within APL who have a background and a significant level of experience with similar systems. To maintain independence, the selected reviewers shall have had no knowledge of the specific implementation of the GNS design until the review is underway, after which contact between reviewers and developers shall be controlled.

9.7.4 GNS Subsystem 'Black-Box' Review

A 'black-box' subsystem characterization and performance evaluation shall be performed by an organization which specializes in the testing of GPS systems. An IEM chassis, hardware engineering model, and a computer will be provided to the outside vendor for a period of approximately 1-2 months for the evaluation. No information shall be provided with respect to the specific algorithms implemented. The tests shall include an evaluation of the interface, functional, and performance capa-

bilities of the receiver. The tests shall evaluate the receiver's navigation solution accuracy, signal acquisition/reacquisition performance, ability to handle interfering signals, performance at nominal and specified maximum space vehicle dynamics, its ability to provide valid navigation outputs, and the evaluation of event prediction and orbital element generation.

SEA-97-0035
25 Mar 97
ICRXPDJB

To: Distribution

From: A. A. Chacos

Subject: Transmittal of TIMED GPS Navigation Subsystem Software Development Plan.

Attachment: TIMED Software Development Plan for the GPS Navigation Subsystem v2.1, 7393-9331, 13 Mar 97.

The attached document is provided for your files.

A. A. Chacos

Distribution:

M. S. Asher	
G. E. Cameron	
A. A. Chacos	
W. S. Devereux	
D. J. Duven	4-106
M. H. Boehme	
C. T. Britt	
D. G. Grant	4-376
D. M. Gruenbacher	
R. J. Heins	
S. F. Hutton	24-E135
S. C. Jones	
D. Y. Kusnierkiewicz	
T. L. Kusterer	8-164
L. A. Linstrom	
K.H. Sanders	
L. L. Suther	24-W161
H. K. Utterback	23-1T
S. P. Williams	4-216
SEA File	
Archives	