KSC-YA-7738
REVISION BASIC

AUTONOMOUS FLIGHT SAFETY SYSTEM ROAD TEST

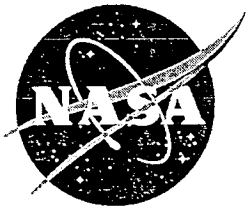
This document does not contain export-controlled technical data.

JUNE 7, 2005

SPACEPORT ENGINEERING AND TECHNOLOGY DIRECTORATE

National Aeronautics and
Space Administration

John F. Kennedy Space Center



KSC-YA-7738
REVISION BASIC

AUTONOMOUS FLIGHT SAFETY SYSTEM ROAD TEST

Prepared by:

James C. Simpson, Ph.D., YA-D7, AST

Concurrence:

Roger D. Zoerner, ASRC-10, Lead Software Engineer

Chris S. Forney, YA-D7, Task Order Manager

Approved by:

Richard A. Nelson, YA-D7, Branch Chief

Temel Erdogan, ASRC-10, Supervisor

JUNE 7, 2005

JOHN F. KENNEDY SPACE CENTER, NASA

1. INTRODUCTION

On February 3, 2005, Kennedy Space Center (KSC) conducted the first Autonomous Flight Safety System (AFSS) test on a moving vehicle—a van driven around the KSC industrial area. A subset of the Phase III design was used consisting of a single computer, GPS receiver, and GPS antenna. The description and results of this road test are described in this report.

AFSS is a joint KSC and Wallops Flight Facility project that is in its third phase of development. AFSS is an independent subsystem intended for use with Expendable Launch Vehicles that uses tracking data from redundant onboard sensors to autonomously make flight termination decisions using software-based rules implemented on redundant flight processors. The goals of this project are to increase capabilities by allowing launches from locations that do not have or cannot afford extensive ground-based range safety assets, to decrease range costs, and to decrease reaction time for special situations.

A Phase I feasibility study was completed in 2000, and a Phase II proof-of-concept hardware system was successfully tested in 2002 using a GPS simulator and Ashtech G12 receivers. A Phase III effort began in 2003 on a flight-qualifiable design to include redundant input sensors and processors. The current sensors are Javad JNS 100 GPS receivers, and the flight processors are IBM Power PCs on PC104 boards running the VxWorks real-time operating system. Sensor solutions are checked for validity and compared with predetermined mission rules and boundaries. The redundant processors each output a Monitor, Arm, Terminate, or Ready-to-Launch signal that will ultimately be sent to a Command Switch Logic and Interlock Circuit (CSLIC) that will perform the voting to decide on the final course of action and then interface with the termination system.

The mission rules are configured for each operation by the responsible Range Safety authorities and can be loosely categorized as follows:

- Parameter Threshold Violation – some trajectory value exceeds an allowed limit.
- Physical Boundary Violation – present position or instantaneous impact point (IIP) is out of the allowed region.
- Gate Rule – determines if a specified point (e.g., vacuum impact) has crossed a gate formed by a line between two points or is ahead of or behind a gate that may be moving or stationary. This rule also applies to chevron lines designed to protect the region behind the launch site from a vehicle that does not pitch downrange successfully.
- Green-Time Rule – how long the missile can fly safely without receiving valid updated tracking data.

2. TEST CONFIGURATION

Hardware

The test hardware was a subset of the final multiprocessor/sensor design, consisting of a single MIP 405 computer, Javad JNS 100 GPS receiver, and battery pack (see Figure 1). A laptop computer was used to monitor and control the AFSS system. The GPS antenna was a simple commercial-off-the-shelf antenna magnetically mounted on the van's roof.

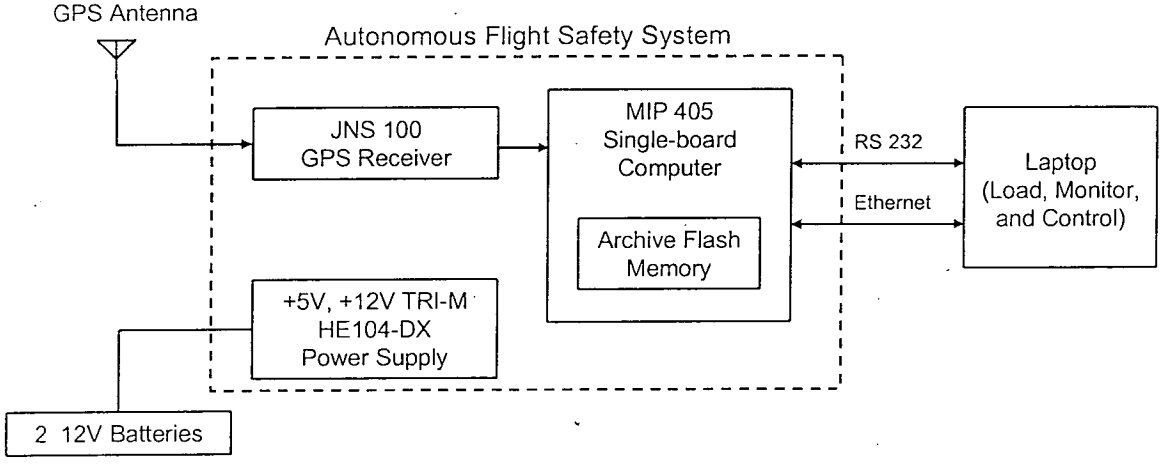


Figure 1. Test Hardware

Test Flight Rules

Because the test was conducted on the ground, present position limits were used instead of IIPs. Tests were made of the first three types of rules: green-time rules were not tested.

- Parameter Threshold Violation – a speed limit was exceeded. This was a contrived but useful test for a generic parameter threshold violation.
- Physical Boundary Violation – the present position went out of the allowed region through a boundary that was not an exit gate.
- Gate Rule – the west end of the test region was defined to be a two-point gate that could be crossed without causing a physical boundary violation.
- Green-Time Rule – not tested.

Several other rules were also tested that do not fit neatly into the four main types described above. These included detecting ignition (i.e., a predetermined acceleration after a ready-to-launch status has been set) and burnout, defined by a predetermined decrease in acceleration. The times to issue Arm and Terminate commands after a rule violation were also tested.

Test Corridor

The test region was a flat rectangular area bounded by easily accessible streets in the KSC industrial area. The latitude and longitude of the vertices were determined from a geographic information system database before the road test and included in the configuration file to define the corridor boundaries.

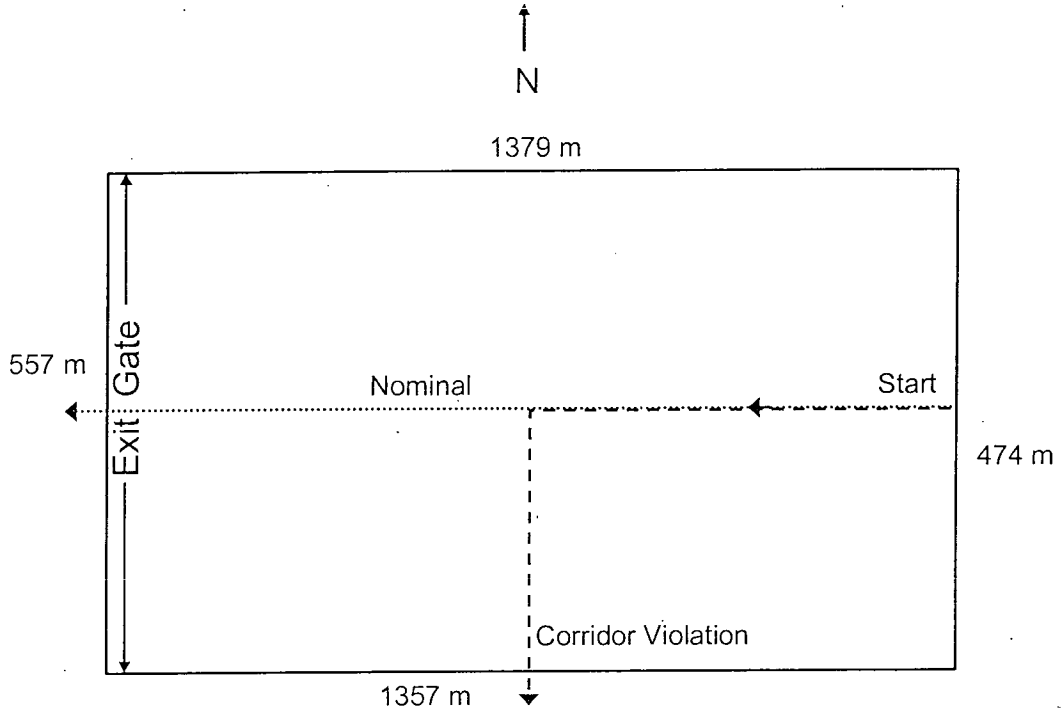


Figure 2. Test Corridor

3. RESULTS

Parameter Threshold Violation

A speed limit of 17.9 m/s (40 mph) was exceeded and resulted in Arm and Terminate actions. This was a contrived but useful test of a generic parameter threshold violation.

Physical Boundary Violation

The present position went out of the allowed region through a boundary that was not an exit gate (see Figure 2) and resulted in Arm and Terminate actions.

Gate Rule

The west end of the test region was defined as a two-point exit gate that was crossed successfully (see Figure 2).

Ignition and Burnout

Ignition and burnout limits of 10 m/s^2 and 8 m/s^2, respectively, were tested successfully, but a road vehicle moving horizontally on the Earth's surface is not the ideal test platform given the current algorithms. The software determines ignition, burnout, and staging events by calculating changes in the applied acceleration, that is, thrust and/or aerodynamic drag, using

$$\vec{a}_{applied} = \vec{a}_{measured} - \vec{a}_{gravity} - \vec{a}_{Coriolis} - \vec{a}_{centrifugal}.$$

The last two terms are necessary because the GPS measurements were made in the rotating, non-inertial Earth Centered Earth Fixed (ECEF) coordinate system. The Coriolis term can be neglected for low speeds and the centrifugal term is small compared to gravity for mid latitudes. This means that gravity becomes an effective normal acceleration of about 9.8 m/s^2.

It was assumed that the van could reasonably accelerate from zero to 25 mph in about 5 s, or about 2.2 m/s^2. Adding this acceleration to that of gravity in quadrature gives the tested ignition limit of 10 m/s^2.

The total rate of change of the speed is calculated in addition to the applied acceleration. If the speed decreases, the magnitude of the applied acceleration is made negative and a burnout is assumed to have occurred. In theory, any burnout limit less than 9.8 m/s^2 would have been appropriate. The empirical limit of 8 m/s^2 used in these tests proved adequate and accounted for numerical and measurement noise.

Time to Arm/Terminate

The times from a violation to issue Arm and Terminate commands of 0.3 s and 0.5 s, respectively, were verified. The actual times can be within ± 0.1 s of these limits because of a combination of certain worst-case assumptions, internal processing delays, and a predetermined tolerance, which was set to ±100 μs for this series of tests (see Figure 3). The broad lines centered at n, n+1, etc., are nominally 0.1 s apart, and the thickness of each line indicates a tolerance of ±100 μs.

Suppose that a violation occurred between times n and n+1. The first data that would indicate a violation is available at n+1, and the software assumes the violation happened at time n. Arm could then happen at either n+3 if the time from n to n+3 was greater than or equal to 0.2999 s, or at n+4 if the time from n to n+3 was less than 0.2999 s, implying a time between the actual violation and Arm of between 0.2 s and 0.4 s. Similarly, Terminate could happen at either n+5 if the time from n to n+5 was greater than or equal to 0.4999 s, or at n+6 if the time from n to n+5 was less than 0.4999 s, implying a time between the actual violation and Terminate of between 0.4 s and 0.6 s.

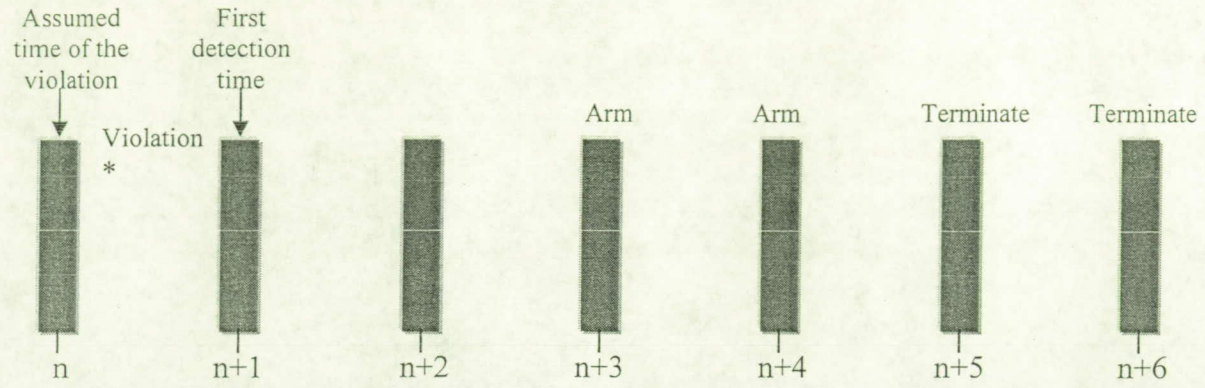


Figure 3. Time to Arm/Terminate Logic

4. CONCLUSION

The system performed as expected. Straightforward rules were violated and handled as expected. Insight was gained into the ignition and staging events limits, and into the logic used for the times to Arm and Terminate after a rule violation. In particular, the timing tolerance of ±100 μs has been changed to ±1 ms, and subsequent simulation tests indicate this is an adequate tolerance to compensate for the single-board computer's multitasking processing delays. The next logical step is to exercise a larger subset of the rules using other vehicles as well as Monte Carlo simulations.