PREFERRED
RELIABILITY
PRACTICES

# FAULT PROTECTION

**Practice:**

Fault protection is the use of cooperative design of flight and ground elements (including hardware, software, procedures, etc.) to detect and respond to perceived spacecraft faults. Its purpose is to eliminate single point failures or their effects and to ensure spacecraft system integrity under anomalous conditions.

**Benefits:**

Fault protection design maximizes the probability of spacecraft mission success by avoiding possible single failure points through the use of autonomous, short-term compensation for failed hardware.

**Programs That Certified Usage:**

Voyager, Galileo, Magellan, Cassini

**Center to Contact for Information:**

Jet Propulsion Laboratory (JPL)

**Implementation Method:**

Except during critical event periods, the primary purpose of an autonomous fault protection system is to place the spacecraft in a safe, commandable state which can be maintained for a reasonable period (typically two weeks) following a fault. During critical periods, the primary purpose of the fault protection system is to ensure the completion of the critical event. A simplified block diagram representing the following three general types of fault protection is illustrated in Figure 1:

- Subsystems alone.
- Subsystem to system, and
- System to ground control.

**Fault Protection Allocations**. All on-board, post lift-off, autonomous fault protection is designated as either "subsystem internal" or "system" fault protection. Fault protection engineering elements which have been allocated fault protection responsibility must provide the requirements and design for the associated detections, monitors, responses, and diagnostic data in compliance with project functional requirements. Where science instruments include fault protection in their design, designers must

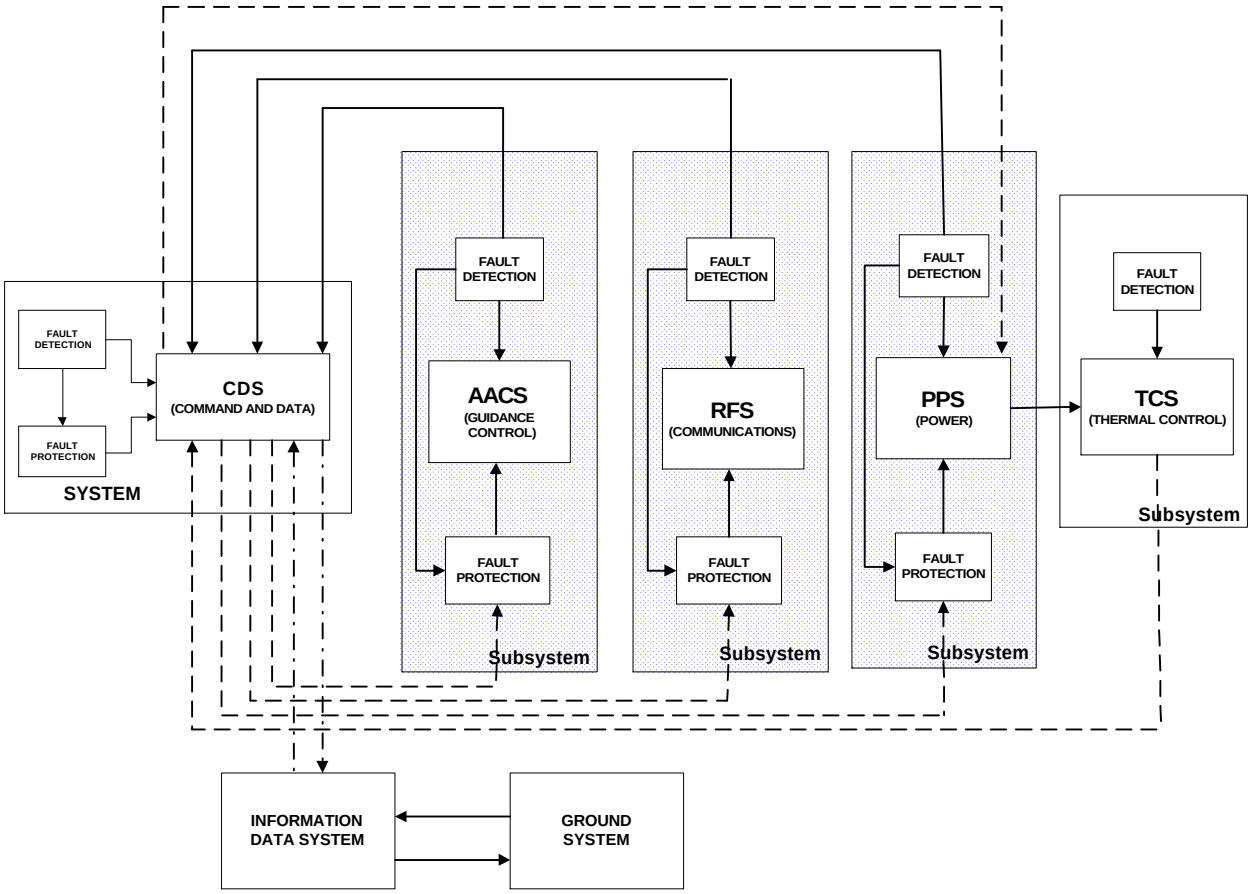


**Figure 1**. Spacecraft Fault Protection Design Block Diagram

still ensure compliance with spacecraft project fault protection requirements if one of the following conditions apply:

- The fault protection internal to the instrument is dependent on non-standard services from another subsystem (or another instrument), or
- Internal failures have an impact external to the instrument (viz, a change in power state, momentum, support to other instruments).

**Spacecraft Safing**. Spacecraft "safing" is a general purpose safe-state response which is initiated by both system and subsystem internal fault protection. The purpose of this response is to provide the following:

- A safe state for the hardware,
- An uplink, and
- A downlink (with some exceptions for specific failure conditions).

To achieve these goals, the normal stored sequence is terminated and non-essential spacecraft loads are powered off.

**Undervoltage Response.** Most spacecraft designs include an undervoltage response, which is designed to protect the spacecraft in the event of a short or a bus overload. The hardware senses when the power drops below an established value for a specified time. If the criteria are met, the power system sheds all non-essential loads from the bus and indicates the undervoltage condition to the Command Subsystem, which will initiate the undervoltage recovery response. Critical spacecraft memories are maintained throughout the undervoltage.

**Functional Implementation Requirements.** Fault protection is typically allocated to the on-board elements of the system in accordance with the following principles:

- *Spacecraft versus ground control.* Autonomous fault protection is included on board the spacecraft only if a response by Mission Operations is not feasible nor practical, or if action is required within two weeks of detecting the failure. Otherwise, ground control is responsible for fault recovery. In both cases, ground control is responsible for failure diagnosis and, if necessary, the configuration of the spacecraft to nominal operations after the fault.
- *Protection against sabotage and operator errors.* To simplify the development of fault protection, autonomous fault protection is not required to protect against sabotage or operator errors, although such protection is not prohibited. There is limited spacecraft protection against these failures (viz, information system data integrity checks and some software checks).
- *Protection against spacecraft hardware and software design errors.* To simplify the development of fault protection, autonomous fault protection is usually not required to protect against spacecraft design errors, although it is not prohibited if practical. The practice of fault protection typically provides some limited spacecraft protection against design errors (e.g., thermal fault responses).

The autonomous fault protection function is responsible for all on-board fault detections and corrections except those routinely required to ensure spacecraft data integrity (viz, EDAC, Reed Solomon encoder, checksums, etc.). Data error detections and corrections may be used, however, for fault protection purposes. The spacecraft information system typically has the primary responsibility for ensuring data integrity.

**Fault Protection Design Requirements**. Management and coordination of fault detection, monitoring, and response, for both system and subsystem internal fault protection, is performed in accordance with the following general rules:

- *Enables/disables for responses.* Where applicable, fault responses should have two enable/disable mechanisms (or the functional equivalent):
  1. an enable/disable by stored sequence, and
  2. an enable/disable by ground control or by fault protection algorithms.
- *Enables/disables for monitor activation of any response.* If a response can be initiated by

more than one monitor, those monitors should include an enable/disable mechanism or the functional equivalent.

- *Enable/disable state specification.* Each enable/disable is specified by a single parameter unique to each fault protection algorithm.
- *Enable/disable strategy - general.* As a goal, fault protection monitors and responses should be designed to be enabled for the entire mission. This reduces the risk of incorrect fault protection states.
- *Enable/disable strategy - critical events.* For critical events, enable/disable strategies may be used to minimize or prevent the effects of an erroneous fault indication.
- *Response initiation.* Fault responses are initiated if and only if spacecraft performance is unacceptable, or there is a significant risk to the mission or to subsystem safety.
- *Parameter modifications.* All fault protection parameters which may reasonably be expected to change as a function of mission mode, type of activity, fault history, or operational experience should be alterable by ground control without requiring flight software modification.
- *Software modifications.* To the extent possible, monitor and response algorithms should be stored in programmable RAM.
- *Configuration compatibility.* On-board fault protection should be designed to respond to a fault while in any possible spacecraft configuration (e.g., fault protection should be able to accommodate all possible combinations).
- *Independence from instruments.* Engineering fault protection should not depend on science instruments or their data.
- *Multiple faults.* At a minimum, fault protection should be designed with the assumption that only one fault occurs at a time, and that a subsequent fault will occur no earlier than the response completion time for the first fault. As a goal, fault protection should be capable of recovering from multiple successive or coincident faults provided that the faults and associated fault algorithms are independent.
- *Propagation of failures.* Autonomous fault protection assumes that spacecraft hardware design ensures that a single failure in a subsystem (including instruments) cannot propagate to its redundant unit or to another subsystem, or prevent switching to its redundant unit. This can be verified by performing a failure modes, effects, and criticality analysis (FMECA) or fault tree analysis (FTA).

**Typical Fault Detection Design Requirements.** Hardware and software detection sources have two criteria:

- *Direct detection.* Detection mechanisms should be as direct as possible (i.e., a direct measurement is preferred over a calculated or derived measurement).
- *Detection coverage.* Detection mechanisms should only be required to detect a failure to the

level at which that failure can be isolated or corrected.

**Design Requirements for Fault Monitors.** Software monitors used by system and subsystem internal fault protection have the following features:

- *Monitor thresholds.* Where possible, thresholds should use reasonableness checks, detection filtering (to exclude certain faults from a previously established fault database), or redundant detections.
- *Threshold modifications.* Monitor threshold values should be alterable by ground or sequence command, or by fault protection responses as appropriate. As a goal, monitors are best designed to detect and disregard failed sensors.
- *Redundant detection.* For detections where an inadvertent trip would result in a severe response (viz, downlink loss, irreversible hardware swaps, large use of expendables, critical sequence cancellation), and where a sensor anomaly could cause an inadvertent trip, independent physically or functionally redundant detections are employed such that simultaneous detections are necessary for response initiation.
- *Fault response tolerance.* Monitors are designed to be tolerant of off-nominal conditions following a reconfiguration resulting from a fault protection response (e.g., thresholds might be relaxed as part of a response).

**Design Requirements for Fault Responses.** System and subsystem internal fault response concerns include:

- *Fault response primary responsibility.* Following an anomaly, fault responses should ensure spacecraft commandability and the maintenance of a safe state for at least two weeks. This requirement is superseded only by a requirement to complete a critical event.
- *Fault protection priorities.* Fault responses are designed with the following priorities:
  1. Protect critical spacecraft functionality,
  2. Protect spacecraft performance and consumables,
  3. Minimize disruptions to normal sequence operations, and
  4. Simplify ground recovery response, including providing for downlink telemetry.
- *Multiple levels of response.* Where possible, response design includes multiple levels of response, with the response actions executed in order of increasing severity.
- *Real-time ground responses.* Autonomous fault protection is designed so as to not require real-time ground responses for recovery from known faults.
- *False alarm tolerance.* Unintended entry into a fault protection response in the absence of a fault must not present a hazard to the spacecraft or mission. For critical event periods, however, this requirement is relaxed and is considered a goal.
- *Use of redundant (and spare) units.* Redundant or spare units may be used by autonomous

fault protection responses if a satisfactory alternative design is not available.

- *Unpowered redundant units.* The transition of a unit from "off" to "prime" must not require ground commands in order to support spacecraft fault protection and mission critical functions.

- *Component warm-up times.* Fault responses should take into account component warm-up times and similar delay requirements.

**Data Handling Requirements.** The following data handling tasks are performed:

- *Recording engineering data.* The combination of sequences and fault responses should ensure the recording of engineering data prior to, during, and after the execution of any fault response. Some exceptions are made for recorder and command subsystems failures.
- *Storage and preservation of diagnostic data.* Fault protection is designed to include the storage of diagnostic data (see Telemetry and Diagnostic Data on page 7), and ensure that data are not overwritten as the result of a response action. This requirement only applies if the writing of diagnostic data is not affected by the original fault.
- *Protection of critical science and engineering data.* Fault responses must not destroy "critical data" stored on-board the spacecraft. "Critical science and engineering data" must be defined by project policy.

**Requirements Interactions with Stored Sequences.** The following interactions with on-board sequences may be necessary:

- *Response design for critical events.* Fault response is designed to ensure the completion of the critical event as and when required, with spacecraft safety having lower priority, until the critical events are completed. Orbit insertion is an example of a critical event.
- *Response design for non-critical events.* Unless required to execute a critical event, fault responses should stop any on-board sequence(s) only if the sequence(s) compromises the integrity of the fault response, or if the fault response compromises the integrity of the sequence.
- *Reactivation of stored sequences.* After completion of a response which terminates a non-critical stored sequence, fault responses should not autonomously resume the terminated sequence.

**Safing Requirements.** "Safing" is defined as a general purpose fault response which results in the cancellation of non-critical sequences, the possible suspension of critical sequences, and a general reconfiguration of spacecraft components. Safing responses typically include the following general features:

- *Uplink communications.* The safing response provides for a spacecraft state and attitude that ensures uplink commandability in the long term.

- *Downlink communications.* The safing response provides for a spacecraft state and attitude that ensures continuous engineering telemetry with positive link margin.
- *Environmental constraints.* The safing response meets boresight and radiator environmental constraints.
- *Safing priorities.* When uplink, downlink, and hardware safing requirements are in conflict, the following priorities apply:

  1. Provide a safe state for the hardware,
  2. Provide an uplink, and
  3. Provide a downlink.

  Ultimately, uplink must be provided for in the long term.
- *Benign spacecraft configuration.* As a goal, safing responses should place the spacecraft in an operationally benign state. This includes but is not limited to the following: (1) powering off of all non-essential equipment including instruments, and (2) stopping of all non-essential spacecraft processes.

**Telemetry and Diagnostic Data Requirements**. The spacecraft design should ensure that following an anomaly, spacecraft telemetry will be sufficient to perform the preliminary failure identification and analysis required for ground control to perform near-term corrective actions. At a minimum, the real-time telemetry stream includes the following:

1. Measurements to unambiguously identify the current engineering subsystem configuration and operating status,
2. Enable/disable states of fault protection software, and enable/disable states of fault protection dedicated hardware,
3. Active/inactive states for fault protection software responses,
4. Identification that a fault response has been activated,
5. Monitor detection logic output (regardless of whether the monitor is enabled or disabled), and
6. As appropriate, cumulative counters of faults detected.

Telemetry design should ensure reasonable and timely sampling frequencies for these items. As a goal, the real-time telemetry stream should include monitor high water marks. A "high water mark" is defined as a measurement which identifies how close a monitor is to indicating a failure.

Following an anomaly involving a fault protection activity, the spacecraft design should ensure that sufficient information is available to reconstruct the audit trail-- the sequence of fault protection events following the anomaly. Diagnostic data is also needed to analyze the anomaly and to identify the sequential effects of the anomaly on spacecraft performance. The data should be available in non-volatile telemetry or stored on-board for later retrieval by the ground system. At a minimum,

diagnostic data should include unique identification and time-tagging of monitors and responses which have been active. It is also desirable that diagnostic data include the value of the measurement which initiated a response.

**Technical Rationale:**

JPL has incurred numerous instances of presumably redundant systems which failed to successfully transfer to the back-up path when the primary path did not function. A rigorous, systematic search, crosstrap FMECA, or sneak paths analysis could have foretold the failure, and redesign could have averted the problem. Prevention of propagating failures has its greatest value when supplemented by an on-board fault protection system.

The practice of failure propagation prevention is of most value in a repairable system. Non-propagation minimizes the number of units requiring repair. In a non-fault protected spacecraft, this capability is limited to the preflight phases of either subsystem or system testing. In a fault protected spacecraft, it can be extended to the flight phase as well. The key to this investigation is a complete diagram of the involved interface circuits which penetrates each unit to a circuit depth sufficient to prove that no possible failures in one unit can propagate to become irreversible hardware failures in a second unit. Another input is a complete list of part or assembly failure modes for hypothesis.

Successful transfer (or equivalently independence of the primary and back-up functions) is a necessity for either repairable or non-repairable systems and requires the same complete interface diagram and complete list of failure modes. Such a list must include items such as:

- Part failure modes (opens, shorts, stiction, etc.),
- Single event effects (latch-up, transfer, etc.), and
- EMI (latch-up, transfer, overvoltage).

  These last two items are critical since they can affect both sides of a redundant pair.

**Impact of Nonpractice:**

A decision to forego fault protection will increase the risk of unrecoverable single point failures.

**Related Practices**

Fault Tolerant Design, Designing for Dormant Reliability, Active Redundancy

**References:**

1. JPL D625-505; Vol. 8, Fault Protection System Design and Operations.
2. 699-CAS-3-330; Fault Protection Requirements, Cassini Project.
3. 699-CAS-3-331; System Fault Protection Algorithms, Cassini Project.