The safing routines on all robotic deep-space vehicles are designed to put the vehicle in a power and thermally safe configuration, enabling communication with the mission operators on Earth. Achieving this goal is made a little more difficult on Curiosity because the power requirements for the core avionics and the telecommunication equipment exceed the capability of the single power source, the Multi-Mission Radioisotope Thermoelectric Generator. This drove the system design to create an operational mode, called “sleep mode”, where the vehicle turns off most of the loads in order to charge the two Li-ion batteries. The system must keep the vehicle safe from over-heat and under-heat conditions, battery cell failures, under-voltage conditions, and clock failures, both while the computer is running and while the system is sleeping. The other goal of a safing routine is to communicate. On most spacecraft, this simply involves turning on the receiver and transmitter continuously. For Curiosity, Earth is above the horizon only a part of the day for direct communication to the Earth, and the orbiter overpass opportunities only occur a few times a day. The design must robustly place the Rover in a communicable condition at the correct time. This paper discusses Curiosity’s autonomous safing behavior and describes how the vehicle remains power and thermally safe while sleeping, as well as a description of how the Rover communicates with the orbiters and Earth at specific times.

I. The Curiosity Mission

The Curiosity Rover (see Figure 1) landed in Gale Crater on Mars in August 2012. This ambitious vehicle has many instruments, the capability to drive, a robotic arm, and an articulating remote-sensing mast, all powered by a single 114 W Multi-Mission Radioisotope Thermoelectric Generator (MMRTG) and two 43-Amp-Hour Lithium-ion batteries. Most of the onboard autonomous fault management incorporates a fail-operational strategy. If a problem is detected, the system simply stops doing the offending activity, but allows the rest of the Rover activities to continue. However, a severe failure (such as an unexpected processor reset, a low battery condition, or no commands from Earth after a certain time) requires a coordinated system fault protection response. These responses abort the commanded activities, sometimes swap critical components, and then they autonomously place the Rover in a safe configuration until the mission operators can determine the nature of the problem and then recover back to nominal operations. The focus of this paper is Curiosity’s autonomous behavior design when it goes into safe mode.
II. Safing goals

The autonomous safing routines on all robotic deep-space vehicles are designed to put the vehicle in a power and thermally safe configuration, enabling communication with the mission operators on Earth. Achieving this goal is a little more difficult on Curiosity because of the energy and communication constraints unique to landed missions. The Rover must autonomously communicate with the orbiting assets and Earth at the correct times, but it must also manage the limited energy by sleeping periodically to recharge the batteries.

A. Energy safe

The energy demands of the core avionics and the telecommunication equipment on the Rover exceed the energy production capability of the MMRTG, even if the vehicle is simply communicating directly with the Earth. This deficiency led the designers to create a system that allowed the Rover to integrate and store the generated energy over periods of time when the energy demands of the Rover could be guaranteed to be less than the energy production capability of the MMRTG. This low power vehicle state is called the “sleep” state. In this state, most of the loads on the vehicle are turned off, including the computer that controls the science gathering activities. A few of the instruments may stay on in the sleep state to continue to capture science data day and night. While the loads on the Rover are off, and the Rover is functionally quiescent (asleep), the MMRTG produces sufficient energy to charge the two batteries. When the Rover wakes up from its slumber and resumes its activities, it draws power from both the MMRTG and the replenished batteries.

Software on Curiosity will put her to sleep whenever unless any of the following conditions are true: The vehicle (1) is under master sequence control, (2) is attempting to communicate with Earth or the orbiters, or (3) is running a system fault protection response. Under nominal conditions, the mission operators control the sleep time by executing command sequences each day. If a major fault occurs, the system fault protection aborts these sequences and the vehicle is under autonomous sleep control.

Given the energy constraints and the fact that the computer must be on to communicate with Earth, the safing system was required to have a robust shutdown and wakeup design. This was achieved by utilizing dual Power
Analog Modules (PAMs) that contain the Rover’s switches, mission clocks, alarm clocks, and the wakeup logic in Field Programmable Gate Arrays (FPGAs). Two battery control boards monitor the cells in the batteries to protect the batteries from over-discharging. If the system detects an undervoltage, PAM FPGAs turn off all non-critical loads and force the system to sleep for 16 hours in order to fully recharge the batteries.

B. Thermally stable

Another goal is to keep the Rover thermally stable, even while the vehicle is sleeping. Curiosity incorporated a Heat Rejection System (HRS) that pumps fluid through tubing in the Rover body and around the MMRTG. This system distributes heat to the internal Rover components and rejects the heat from the MMRTG via heat exchangers. A detailed description of the HRS is given in Reference 1. The HRS requires only one of the dual HRS pumps to be on, so the backup pump is nominally powered off to save energy. Bimetallic mechanical thermostats installed near the coldest and warmest parts of the tubing are designed to turn on the backup pump if the primary pump fails. This design keeps the system thermally safe whether the Rover is awake or asleep and does not require onboard software, except to swap the primary pump designation.

C. Communicate with Earth

Curiosity nominally communicates with mission operators in two ways (Figure 2). The first method is the direct path, via an X-band link using either the low gain omnidirectional antenna or the articulated high gain antenna. Direct-from-Earth (DFE) communication requires the Small Deep Space Transponder (SDST) to be powered on to receive command packages from Earth. Direct-to-Earth (DTE) additionally requires the 15W Solid-State Power Amplifier (SSPA) to be powered for the Rover to send telemetry. Since the Earth is only above the Martian horizon for a short part of the day, the mission operators program the daily communications each day. The operators load “windows” that specify the time, the duration, which antenna, the data uplink and downlink rates, and the specific telemetry item types desired for that communications pass. The onboard software coordinates the powering on of the correct SDST and SSPA (if needed), configures them to the correct state, then starts tracking the Earth with the high gain antenna, and finally feeds the telemetry to the transponder for broadcast. The software also receives any command packages sent from Earth. When the window duration is complete, the software gracefully shuts down the components and stows the high gain antenna.

Figure 2: Curiosity's Telecommunication Network
The second method of communicating with Earth uses a satellite relay. The Rover’s Electra-Lite Transponder (ELT) radio links with the Ultra-High Frequency (UHF) radios on the orbiting assets: Mars Reconnaissance Orbiter (MRO) and the Mars Odyssey orbiter. Mission operators schedule these communication windows on both the Rover and the orbiter. When the window starts, the Rover software turns on the prime ELT, configures it to a specified rate, then sets the coax transfer switch to make sure the right radio is connected to the UHF quad-helix antenna, and then finally queues up telemetry. When the orbiter hails the Rover, the radios on both spacecraft sync and data are exchanged. The orbiter may send commands to Curiosity at the same time the orbiter receives telemetry. Hours later, the orbiter relays the telemetry to Earth when it is scheduled to do so.

A major fault could happen at any time and the Rover will autonomously safe. She must eventually autonomously communicate with the mission operators. On most deep space vehicles, this involves turning on both the receiver and transmitter, then transmitting its status continuously. This concept is not possible on Curiosity due to the energy constraints described above. Additionally, Earth is above the horizon only part of the day and the orbiter overpass opportunities occur only a few times a day. The safing design needs to place the Rover in a communicable state at the correct time.

Additionally, the autonomous behavior design needed to be independent of the often-rushed commanding that the mission operators encounter daily. Moreover, the nominally scheduled communication windows are only stored on the primary computer’s memory. The autonomous safing behavior design is required to work on either computer, in the event of an autonomous computer swap. To meet these requirements, the designers implemented a mode that uses System Fault Protection (SFP) communication windows rather than the nominally scheduled communication windows. The window times are configurable by parameters saved in non-volatile memory on both computers.

Several types of SFP windows may be specified. SFP UHF windows are used to communicate with the orbiters. SFP X-band receive-only windows allow the mission operators to send commands directly to the Rover through the low gain antenna, at a very slow bit rate. An X-band “tone” window will allow the Rover to transmit a specific frequency that can be detected by NASA’s Deep Space Network. Note that tone windows are used in nominal operations to tell the mission operators if the daily command load has been received or not. Based on the time and length of the tone, the team can tell if the Rover received the new command load, is still running the old sequence, or if the system has safed.

Before the window start time, the autonomous system wakes up the Rover computer, turns on the appropriate transponder, and configures the telecom path from the telecom interface board to the correct antenna. If the window is a UHF window, the software stages telemetry in the radio and then waits for the orbiter to fly over (which could be hours later). Once the orbiter issues a hail signal, its radio syncs with the radio on the Rover to exchange data. A detailed description of the telecom system is given in Reference 2. Rover software feeds telemetry to the orbiter until the orbiter sets over the horizon.

III. MSL Safing Behavior Description

A. Communication opportunities

If the opportunities to communicate with the orbiters and the Earth are scaled for a Mars day, the daily orbiter over-flights line up within a few hours of each other. A solar day (sol) on Mars is 88,775.24409 seconds, or 39 minutes 35.24409 seconds longer than a day on Earth. Scaled to a 24-hour sol, Figure 3 shows all the opportunities to communicate with the primary orbiters (MRO and Odyssey) over Curiosity’s prime mission, which lasts for one Mars year. Orbiter over-flights occur between 2 and 5 local time each morning and afternoon throughout the prime mission. Over the Martian year, the Earth rises and sets later each day, until the beginning of January 2014 (winter on Mars). Using this information, the mission operators set the parameters to specify when Curiosity should perform each type of communication, should the vehicle experience a major failure.
B. A day in the life

The team optimized the safing communication parameters for landing day (assuming this was the most likely period to experience problems and potentially safe). A typical safing day is shown in Figure 4. The autonomous system sleeps the Rover whenever the vehicle is not communicating (or waiting to communicate). So if Curiosity experiences a fault that puts her into safing, she wakes up just before 2 a.m., turns on the Electra-Lite radio, and queues up telemetry in the radio to transmit as soon as the orbiter initiates the link. When the two vehicles sync up, telemetry is sent to the orbiter and Curiosity receives any commands forwarded from the orbiter. After the orbiter sets, Curiosity goes back to sleep to charge up the batteries.
At 8 a.m., the Rover will wake up again and place herself in X-band receive-only mode. An hour later, Curiosity turns on the transmitter and sends a tone for 50 minutes. Afterwards, the mode is returned to receive-only so the Rover remains awake, which allows the mission operators to send commands. Since the mission operators on Earth know this behavior, they can choose to wait until the “tone” confirmation at 9 a.m. (signaling that the Rover is in a normal safing configuration) before commanding her back to a nominal configuration. Note that the mission operators must take into account the one-way light time delay (between 4 to 24 minutes) to ensure the commands reach the Rover before she naps to recharge up for the afternoon UHF passes.

The autonomous safing behavior gives the mission operators a predictable configuration to diagnose issues and to eventually recover Curiosity back to regular operations. Commanding is possible either directly from Earth through the X-band link or through the UHF forward-link command capability via the orbiters.

The power profile of a safing day is shown in Figure 5. The battery state-of-charge (SOC) increases as the vehicle sleeps, then slowly decreases when the Rover is awake and waiting for the orbiter to hail it between 02:00 and 05:00. The “tone” at 09:00 is the highest power draw, so the parameters are specified to ensure that window lasts only 50 minutes. At the end of the day, the safing behavior has increased the battery SOC by 22%.

![Figure 5: Safing Day Power Profile](image)

C. Safing variation – hail mode

Note that the safing behavior depends on the Rover knowing the correct time. If the time is wrong, the Rover will wake up and turn on the radios at the wrong time, potentially while the orbiters or the Earth are below the horizon. Therefore, Curiosity has another operational mode called “hail mode”. In hail mode, the Electra-Lite radio is turned on and left on while the rest of the Rover sleeps. When the orbiter flies over and “hails” the Rover to
initiate a communication session, Curiosity’s radio will issue a wakeup signal to the PAMs. The Rover will wake up and after the computer initializes, the software sends telemetry to the orbiter and receives commands through the orbiter forward-link. Mission operators will then be able to diagnose the clock problem and, on a subsequent overpass, correct the problem. Hail mode is selected autonomously when the Rover detects a problem with the mission clocks or if it has not received a specific command after a specific time.

IV. In-flight Experience

Curiosity’s first 330 sols on the Martian surface have been relatively fault-free. The fail-operational strategy has worked well, where various problems with actuators and science instruments isolate the problem to that single device, but allow the rest of the normal operations to continue. As of Sol 342, Curiosity has safed twice in surface operations. Sol 200 is described below. Another software flaw was discovered on Sol 218 and the vehicle safed itself. A quick analysis by the anomaly team allowed operators to recover out of safing two days later.

A. Sol 200 Anomaly

On Sol 200, an uncorrectable memory corruption occurred in the non-volatile memory on the primary computer. The data product catalog became corrupted, which uncovered several software defects, with the worse effect that the vehicle did not process the commanded shutdown request. The anomaly team debugged the problem quickly from the telemetry that was downlinked from the nominal MRO pass. Then the team recreated the problem in the testbed, developed recovery commands, and tested them. Less than 24 hours from the initial failure, the anomaly team used the knowledge of the safing behavior to issue commands to reboot the primary computer into an isolated state. This allowed the hardware fault protection to react to the non-responsive computer and turn on the backup computer. When this new computer booted up, the software recognized that there was a processor swap. The fault response autonomously configured the Rover into safing.

This commanding was done at 08:15 local time, when the Earth had risen high enough over the horizon to reach Curiosity. This left enough time before the 09:00 “tone” window for the hardware fault protection to perform the computer swap and for the software to capture the instrument data before turning them off, to capture data from the problem computer before turning it off and to fully safe the vehicle. The Rover slept for roughly 15 minutes, then woke to perform the tone window. From Earth, via the X-band fault tone, the mission operators could verify that the commanding appeared to work as expected. It wasn’t until the later UHF window that the team received the telemetry to confirm that all had worked as planned. Mission operators opted to keep the vehicle in this safe configuration for a few days, until a full assessment of the vehicle health could be performed and a plan for checking out the failed memory could be made.

One surprise discovered on this first safing entry was that the first tone came several minutes later than expected. This is due to the mission clock drift that is nominally adjusted for by mission operators. The mission operators will adjust the safing parameters to deal with the drift, but this action is only needed roughly once a year.

D. Other changes

Mission operators have adjusted the SFP window parameters such that Curiosity will communicate with the Earth later during a Sol. This change was made because the Earth rises later in the sol than it did at landing time (see Figure 3) and to separate the nominal beeps from the off-nominal set. Additionally, the Odyssey operators want to shift the orbit for mission science reasons, so Curiosity’s SFP windows will also need to shift.

V. Conclusion

All deep space robotic missions are required to autonomously keep the vehicle thermally and energy safe in the event of a major system failure. Curiosity’s energy constraints resulted in a design that incorporated a sleep mode to ensure the batteries could be charged by the RTGs. Keeping the vehicle thermally and power safe while the computer was off involved several fault-tolerant systems as well as robust hardware design to ensure the computer could be powered back on to communicate with the orbiters or directly with the Earth. Unique to Mars landers is the need to communicate with orbiters and the Earth while they are above the horizon. Curiosity’s autonomous surface
safing behavior provides the Rover with a safe, predictable, and configurable design that provides visibility to the mission operators into the state of the vehicle and provides commandability to recover back to normal operations.

Acknowledgments

Designing a system to keep the Mars vehicle safe and communicating with Earth was a system-wide endeavor. Kudos and many thanks go to the Avionics team for making a robust wakeup and shutdown system, the Power team for the battery design, the Thermal team for the no-fuss heat rejection system, the Flight Software team for the code to communicate with the orbiters, and the remarkable Telecom team for making the complicated linkage between the Rover and orbiter actually work. The Test team provided a flexible test environment and the Systems team spent many hours proving out the design.

The work described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References
