Redundant Space Systems

Mission Objective Accomplishment in Case of System Failures

Key element of the Space segment is the capability to pursue the mission objectives under all conditions, even when parts of the satellite or probe are no longer working due to a malfunction of any kind. The ECSS explicitly mandates this in the following requirement:

*"Provision of adequate control functions to configure the on-board systems for the execution of nominal mission operations, failure detection, identification, isolation, diagnosis and recovery, and maintenance operations." **

**Quotation Source: ECSS-E-ST-70-11C*

General requirements: The control functions (telecommands) provided at each level of the system hierarchy shall be capable of achieving the mission objectives under all specified circumstances. This can include the use of redundant equipment to meet the overall system reliability requirements. In terms of FDIR, recovery can be implemented by means of redundant systems. This usually involves an active and a backup subsystem.

A key factor for this is the capability of PikeOS to evaluate the status of the partition via the InstruMon internal monitoring facility which allows to get information of the partition status via OS monitoring.

Leading Prime Contractor • ECSS-compliant • Quality „Made in Germany“

www.sysgo.com

Redundant Space Systems

Mission Objective Accomplishment in Case of System Failures

Redundancy is usually classified by 3 ways:

COLD (passive or standby): Subsystem in which the backup is not operating (or even unpowered) and is activated only when the main system is detected to be malfunctional (e.g. TM Encoder for satellite telemetry).

HOT (active): Both sub-systems are active at the same time, fed with the exact same input data (e.g. satellite telecommand subsystem, which allows to send/receive commands to/ from a remote system like the ground segment).

WARM: Subsystem with active backup system, but runs with reduced functionality (e.g. in a satellite the sub-system that stores a copy of the telemetry-critical data (only), which will allow the download in case the active data storage has a failure.

Failure of Subsystem ALPHA where COLD Backup is available

This first use case considers the failure of ALPHA HW when it is currently in use and a COLD Backup HW (BETA HW) is being available. In this case, the failover takes place when ALPHA HW fails during the computation and no handshake occurs.
The main processing subsystem (MAIN COMP application) may detect the failure upon the missing handshake within a pre-determined timeout, or due the subsystem responsible to monitor ALPHA raising a failure message.
COLD Backup management can be achieved in several ways; for example MAIN COMP application may request the MONITOR to deactivate the driver for ALPHA HW and activate the one for BETA HW. Yet MAIN COMP can still receive the same input from the same channel, but only from BETA and no longer from ALPHA. So, in this case we can exploit the advanced capabilities of PikeOS with custom drivers to achieve redundancy.
Important notice: If the ALPHA system is able to recover, it will not get active automatically. When ALPHA and BETA are implemented via PikeOS partitions, the error in ALPHA can be detected by the instrumentation subsystem and given to a Monitor Application in a service partition, which has privileged access to PikeOS HyperV API and can start/stop other partitions. So in this case the system would stop ALPHA, start BETA and act on the data input to direct the data inflow to BETA (red command lines in Fig. 1).

Failure of Subsystem ALPHA is needed where HOT Backup is available

In this use case we have a HOT Backup GAMMA. The failure detection is the same as in the previous one, however, as the HOT Backup is already running in parallel, we do not need to start the HOT Backup as it is already generating output data.
Also, in this case, the MAIN COMP from now on will only use the GAMMA HW DRV (HOT Backup) data for the needed functionality. What is important to notice is the fact that we have a degraded situation, as now the functionality provided by ALPHA is no longer having a backup.
When ALPHA and GAMMA are implemented by means of PikeOS partitions, the error in ALPHA can be detected by the instrumentation subsystem and given to a Monitor Application in a service partition, which can specify to the output part from which subsystem the valid data will arrive (red command line in Fig. 1).

PikeOS Software Architecture

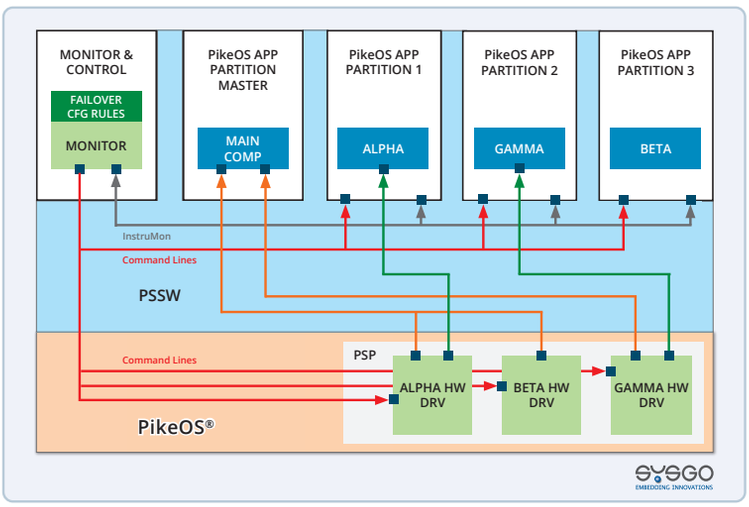

Figure 1

Founded in 1991, SYSGO became a trusted advisor for Embedded Operating Systems and is the European leader in hypervisor-based OS technology offering worldwide product life cycle support. We are well positioned to meet customer needs in all industries and offer tailor-made solutions with highest expectations in Safety & Security.

More information at www.sysgo.com/space

SYSGO Headquarters
Phone +49 6136 9948 500
sales-de@sysgo.com

SYSGO France
Phone +33 1 30 09 12 70
sales-fr@sysgo.com

SYSGO Czech Republic
Phone +420 222 138 111
sales-cz@sysgo.com

www.sysgo.com

SYSGO EMBEDDING INNOVATIONS