	<p align="center">SCORPYOS Executive Summary PUBLIC</p>	<p>Proj.: 2021-002-SCORPYOS Ref.: EVODE-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 1/15</p>
---	--	--


Project Name :SCORPYOS


Document Type : RP

Title : D5.2 – Executive Summary

Issue : 1A



	Name & Function	Signature
Approval :	Rodolfo Martins (Project Manager)	

	<p align="center">SCORPYOS Executive Summary PUBLIC</p>	<p>Proj.: 2021-002-SCORPYOS Ref.: EVOLEO-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 2/15</p>
---	--	---

Document Category

A: Document for Approval		R: Document for Review		I: Document for Information	X
--------------------------	--	------------------------	--	-----------------------------	---

Distribution List

Name	Org.	Function	Copies
Kostas Marinis	ESA	Technical Officer	Electronic Copy

Contact List

Name	Org.	Function	Contact
Rodolfo Martins	EVOLEO	Project Manager	Phone: +351 229 424 327 Email: info@evoleo.tech

Document Change Record

Iss./Rev.	Date	Affected Sections	Reason for Change
1	2025-03-12	All	First Issue
1A	2025-03-31	All	Proprietary, Copyright information and references removed

The rest of the page has been left intentionally blank

Table of Contents

1 GENERAL INFORMATION 4

1.1 INTRODUCTION 4

1.2 ACRONYM LIST 4

2 OBJECTIVES AND CONSTRAINTS 5

3 FEEDBACK FROM INDUSTRY 6

3.1 AIRBUS 6

3.2 ASP 6

3.3 REALTRA 6

3.4 SUMMARY 6

4 DESIGN SUMMARY 7

4.1 MICROCONTROLLER SELECTION 7

4.2 BUDGETS AND SPECS 7

4.3 MECHANICAL DESIGN 8

4.4 ELECTRICAL DESIGN 8

4.5 FIRMWARE DESIGN 10

4.5.1 Verification Methods 11

4.6 CONFIGURATION TOOL 11

5 TESTS PERFORMED 14

5.1 SIMPLE USE CASES 14

5.2 SUPERVISE A PROCESSING BOARD AND AN EXTERNAL PSU OR DC/DC 14

6 CONCLUSION 15

Figure List

FIGURE 4-1 – IMPLEMENTED MECHANICAL ASSEMBLY8

FIGURE 4-2 – BOARD AFTER PRODUCTION8

FIGURE 4-3 – SCORPYOS BLOCK DIAGRAM9


FIGURE 4-4 – FIRMWARE COMPONENTS.....10

FIGURE 4-5 – MAIN MENU11

FIGURE 4-6 – UNIT DESIGN MENU12

FIGURE 4-7 – FDIR OPTIONS13

FIGURE 4-8 – MONITOR MENU13

	SCORPYOS Executive Summary PUBLIC	Proj.: 2021-002-SCORPYOS Ref.: EVODE-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 4/15
---	--	--

1 GENERAL INFORMATION

1.1 Introduction


The SCORPYOS project, developed by EVOLEO, has created and presented a breadboard of a reference design for a "Minimalistic Supervisor." This equipment encompasses both hardware (HW) and a suite of software/firmware (SW/FW) tools aimed at facilitating the seamless configuration and parameterization of the operation of inputs, outputs, and internal functions without necessitating specific coding skills.

This document is the Public Executive Summary of the work that has been carried out under ESTEC contract 4000135813/21/NL/GLC/ov "SCORPYOS – Supervisor CONcept foR sPace SystemS" – Minimalistic Supervisor based on ATMegaS128.

1.2 Acronym List

Acronym	Description
COTS	Commercial off the Shelf
DIG	Digital Design
EGSE	Electrical Ground Support Equipment
ELE	Electrical Engineer
EM	Engineering Model
FDIR	Failure Detection Isolation and Recovery
FMEA	Failure Mode & Effect Analysis
FPGA	Field Programmable Gate Array
FW	Firmware
IP	Intellectual Property
LEO	Low Earth Orbit
MIMPS	Mega Instructions per Second
MPSoC	Multi Processor System on Chip
OBC	On-Board Computer
OS	Operative System
P/L	Payload
PF	Platform
PLIU	Payload Interface Unit
RTD	Resistor Temperature Dependant
RM	Reconfiguration Module
RR	Requirement Review
RTOS	Real Time Operating System
S/C	Spacecraft
SEE	Single Event Effect
SEFI	Single Event Functional Interrupt
SEL	Single Event Latchup
SYS	Systems Engineering
TBD	To Be Defined
TC	Telecommand
TID	Total Ionizing Dose
TM	Telemetry

Table 1-1 – Acronym List.

	<p style="text-align: center;">SCORPYOS Executive Summary PUBLIC</p>	<p>Proj.: 2021-002-SCORPYOS Ref.: EVODE-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 5/15</p>
---	---	---

2 Objectives and constraints

The widespread use of COTS in LEO is constrained by their susceptibility to faults and the difficulty in implementing custom fault detection and recovery actions (FDIR) closely to the system in question.

Thus, faults may pass undetected until they propagate outside the COTS system and cause a significant impact on the spacecraft. This delay in recovery actions may contribute to lower in-orbit reliability and availability. The challenge is in supporting designers who wish to protect their systems via recurrent methods and tools which are also cost-effective.

EVOLEO proposes to develop and demonstrate an elegant breadboard of a reference design for a "Minimalistic Supervisor", including HW and a set of SW/FW tools to easily configure/parametrize the operation of the inputs/outputs and internal functions without the need for specific coding.

The SCORPYOS project targets the development of a Supervisor Concept for Space Systems, operating alone or as a supporting unit, focusing on small subsystems, high simplicity of usage and LEO missions.


The ultimate goal of the implementation of the SCORPYOS concept is to significantly increase the availability of the COTS system under supervision via a reference design HW and a set of SW/FW tools that allow a user to easily configure/parametrize the reference design to its use-case.

Therefore, there are three basic key goals to be accomplished within this activity that drive the entire thinking of the project:

1. To design and implement an elegant breadboard (EBB) reference design, HW and SW/FW tools, for a minimalistic supervisor to support COTSbased subsystems capable of fulfilling a large spectrum of the market/use cases needs in terms:
 - a. Technical performance and capabilities (FDIR functions, dependability, flexibility of use)
 - b. Commercial soundness (cost/time to market/recurrence)
2. To validate via testing the applicability of the reference design to, at least, two actual use case scenarios via streamline (re)configuration and parametrization of the reference design to the use-case scenario.
3. To layout in collaboration with industry partners future steps for adoption of the developed technology.

These objectives are to consider the application of the technology development to a:

- Closer to production, flexible and recurrent oriented reference designs for minimalist supervisor for COTS-based subsystems.
- Small to minisatellite classes (+20kg) in LEO for New Space using COTS.
- Small subsystems which require improvements in availability and overall reliability.

	<p style="text-align: center;">SCORPYOS Executive Summary PUBLIC</p>	<p>Proj.: 2021-002-SCORPYOS Ref.: EVODE-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 6/15</p>
---	---	---

3 Feedback from Industry

EVOLEO investigated use cases, including similarities and distinguishing factors, and consulted with partners such as Airbus, ASP and Realtra, to develop a clear scenario of use cases of interest to the industry.

Below is the feedback obtained by the different companies.

3.1 Airbus

Can we use this to supervise Xilinx Versal? And as boot method for KU060?

Consider PMbus for power monitoring

To protect/snoop AI System-on-module (ex: NVIDIA Jetson board)

3.2 ASP

May be relevant to consider SCORPYOS has basis for NewSpace PSDU – more granular

Should there be redundant supervisors?

Consider memory required to store and process all power telemetries

What telemetries and how can we access them? – integration with OBC FDIR concept

3.3 Realtra

Can we consider SCORPYOS for hard/non - LEO missions? Lunar for example, where we can't fly VPU/GPU?

What faults can you detect without changing current VIKI (Ariane VideoKit) design and HK datapool?

Will provide test telemetry (nominal and fault) for post-processing using SCORPYOS algorithms.

3.4 Summary

The SCORPYOS system is designed to be versatile and applicable to various equipment, extending from Commercial Off-The-Shelf (COTS) devices to those operating beyond Low Earth Orbit (LEO).

Its primary objective is to safeguard specific System on Chip (SoC) and AI engines, including Versal, Jetson, and Myriad.

The value of intelligent monitoring will first be demonstrated on standard equipment, establishing its efficacy and reliability.

This foundational success will then pave the way for moving towards more tailored designs, incorporating the advanced capabilities of the SCORPYOS system.

The rest of the page has been left intentionally blank

4 Design Summary

4.1 Microcontroller selection

Options have been identified and trade-off metrics defined.

Devices pre-emptively excluded from analysis: **FPGAs** and **Complex SoC**.

The choice between several candidates was based on costs and delivery times, with **SAMV71** being the choice to start the development process.

4.2 Budgets and Specs

The supervisor unit electronics card provides processing and interface features to supervisor and control an external system. The main features of the unit are:

Processing	ARM-Cortex M7 @ 300MHz with FPU
Memory	2048Kb Flash 384Kb SRAM 256Mb SRAM (option) 256Mb NOR Flash (option) ECC protected
Interfaces	2x High Density flight connectors 4x RS-422 (one of them used for configuration) 2x CAN 7x GPIO 4x Digital inputs (5V Logic Level) 4x Analog inputs (0V~5V) 1x SPI (3x Slave Select)
Power	Vin: 8V~12V (LCL protected with UVP/OVP/OTP) <1.2W
Dimensions	145x139x29.5mm
Mass	390g
Operating temperature	-40°C – 100°C
Reliability	Automotive, QML-Q equivalent and JAN Latch up and high current protections Overtemperature protection Under-voltage & Over-voltage LCL protections Granular current monitoring and power switching
Radiation tolerance	SEL > 40MeV*cm2/mg TID > 20krad

Table 4-1 – Summary of unit specs

4.3 Mechanical Design

The mechanical design follows the proposed concept with the goal of achieving a small and lightweight unit. Flight and ground connectors are placed in adjacent faces of the unit for easier AIT.

Estimated mass:	390g
Dimensions:	145x139x29.5mm

Table 4-2 –Physical dimensions

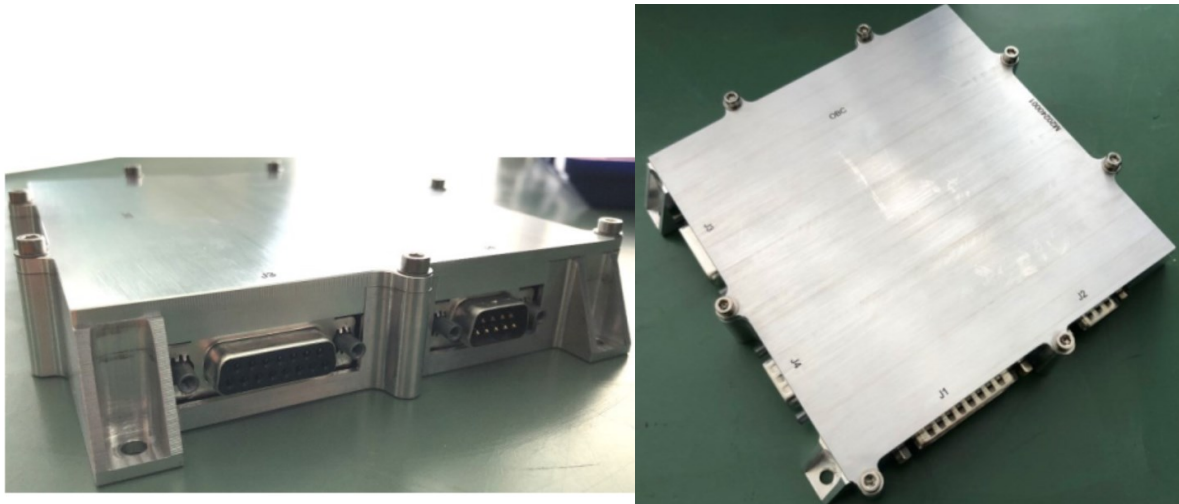


Figure 4-1 – Implemented mechanical assembly

4.4 Electrical Design

The unit provides two sets of interfaces – flight and ground.

Flight interfaces (J1, J2) are meant for interfacing with other S/C units such as power supplies, system under supervisor, main OBC.

Ground interfaces (J3, J4) are dedicated to programming and live monitoring during AIT.

All connectors are rectangular DSUB High Density (HD), except for J4 which is a regular 9 pin DSUB.

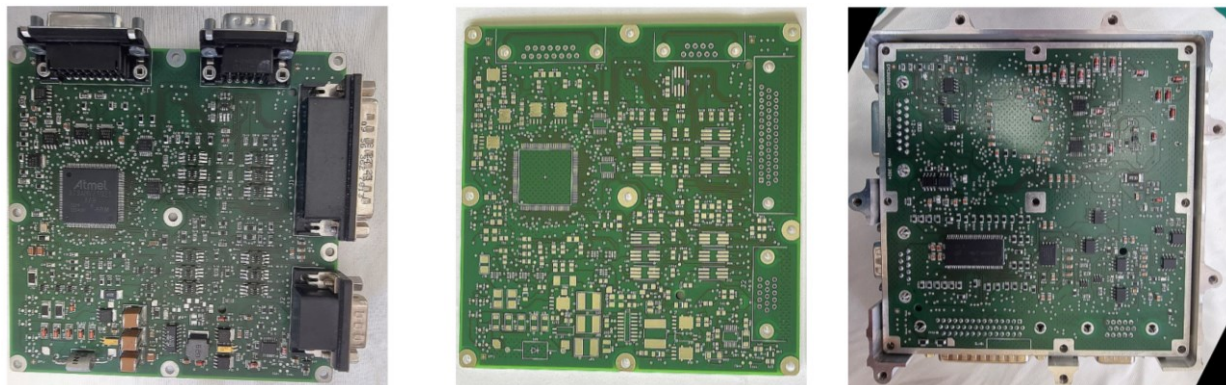


Figure 4-2 – Board After production

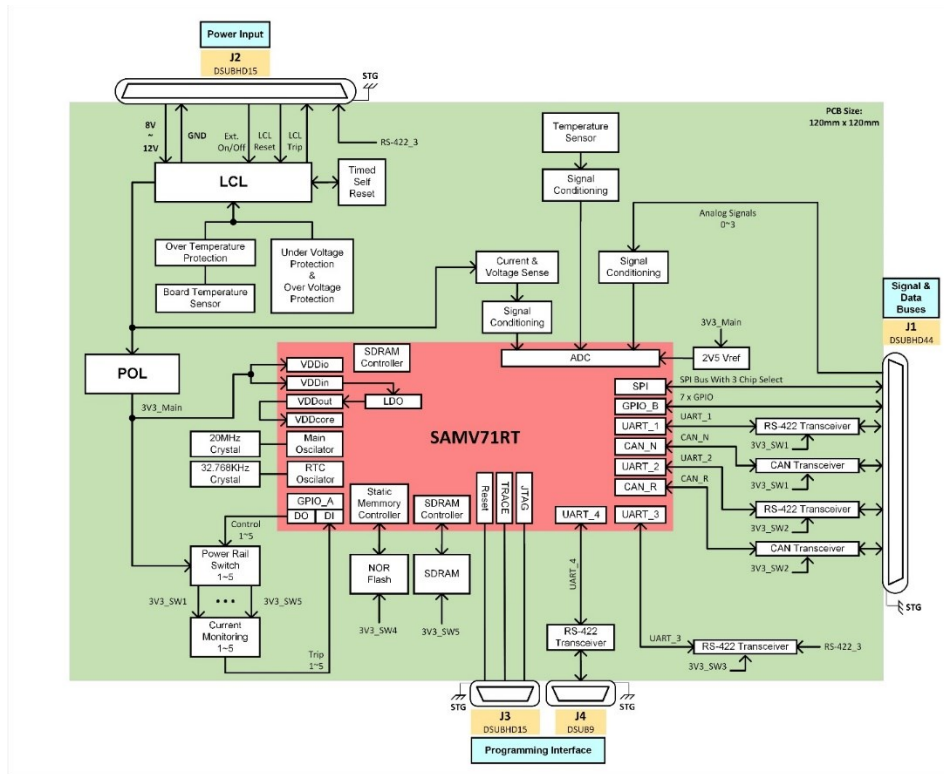


Figure 4-3 – SCORPYOS block diagram

J1 – HD44 – Data and control		
Type		Nº pins
RS-422_1	Standard	4
RS-422_2	Standard	4
CAN_N & CAN_R	Reliable	4
SPI (3x CS)	Standard	6
GPIO (I/O and interrupts)	Standard	7
Digital Inputs	Standard	4
Analog Inputs (comparison and samples)	Standard	4
GND	Standard	8

J2 – HD15 – Power		
Type		Nº pins
PWR_main_in	Reliable	1
PWR_main_rtn	Reliable	1
GND	Reliable	5
LCL reset	Reliable	1
LCL Trip	Reliable	1
RS-422_3	Reliable	4
LCL Control	Reliable	2
-----	-----	-----

Table 4-3 – Flight connectors summary pinout.

J3 – HD15 – Debug	
Type	Nº pins
TRACE _ debug	5
GND	2
JTAG	5
Reset test	2

J4 – DSUB9 - Programming	
Signal	Pin
RS-422_4	4
GND	5
-----	---
-----	---

Table 4-4 – Ground connectors summary pinout.

4.5 Firmware Design

The firmware implements a component-based architecture focused on the decomposition of the design into individual functional or logical components that represent well-defined communication interfaces containing methods, events, and properties.

It provides a higher level of abstraction and divides the problem into sub-problems, each associated with component partitions.

The firmware is structured in several layers, providing a clear separation between the different components. These will be organized into application, handler and driver layers, ensuring modularity and facilitating system maintenance and evolution. The following diagram shows the components that will be developed to meet the use cases mentioned above.

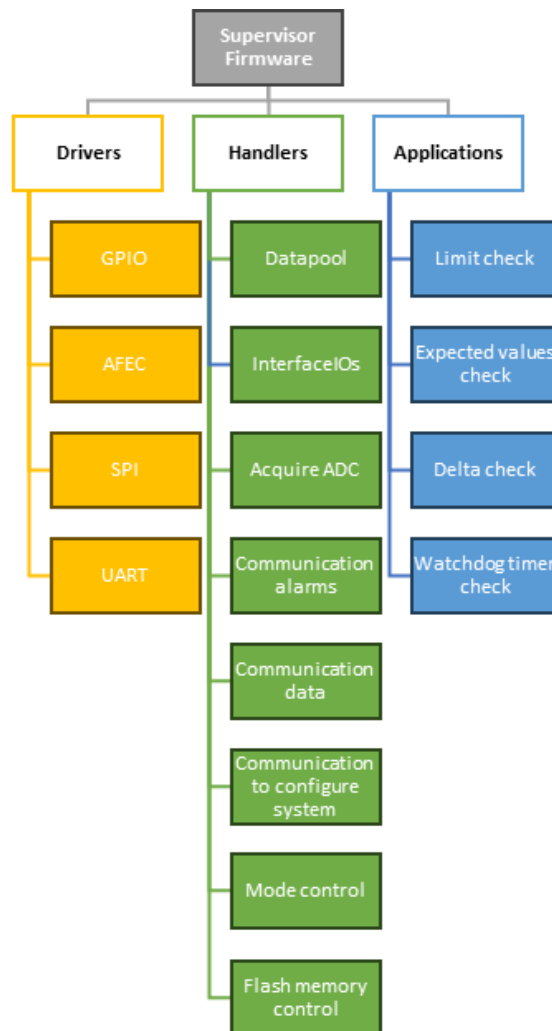


Figure 4-4 – Firmware components

The applications are created to implement the different versions methods, expect the data field check, because this is implemented directly on the communication data handler.

This modular firmware structure facilitates maintenance, improves code readability, and allows the system to be scalable as new requirements arise.

4.5.1 Verification Methods

Limit Check is a fundamental verification method used in supervisory systems to ensure that the values of specific parameters remain within predefined ranges, considered safe or operational. This method is particularly effective in monitoring analog readings such as temperature, current, voltage, reaction wheel RPM, although it can also be applied to certain types of digital data.

Expected Value Check is a sophisticated verification method used in digital supervisory systems, particularly effective in monitoring parameters that contain information encoded in specific bits. This method involves applying a bit mask to a parameter and then comparing the result with a predefined value. Could be used to check component status and operation modes or detect errors in Communication Protocols.

Delta Check is a verification method used to monitor the variation between consecutive readings of a given parameter, ensuring that this variation remains within predefined limits. This method is particularly useful in detecting abrupt or abnormal changes in dynamic systems.

The **Watchdog Timer Check** is a critical supervision method used to monitor the continuous and correct operation of external units or processes. This method is based on the principle that a functional unit must periodically signal its normal operation by resetting a dedicated timer. This should be used to monitoring critical subsystems.

Data Field Check is a versatile verification method used to monitor and analyze specific data fields received through various communication protocols, such as UART (Universal Asynchronous Receiver-Transmitter), SPI (Serial Peripheral Interface), or CAN (Controller Area Network). This method allows the application of verification techniques traditionally used for analog or digital values to data transmitted by serial communication systems.

4.6 Configuration Tool

The configuration software is divided into two subsystems: the configuration and the system monitoring tools.

The configuration tool allows for the setup and customization of system parameters, ensuring that the system operates according to the specific requirements of the application.

The system monitoring tool provides real-time data and diagnostics, enabling users to observe and analyse the system's status.

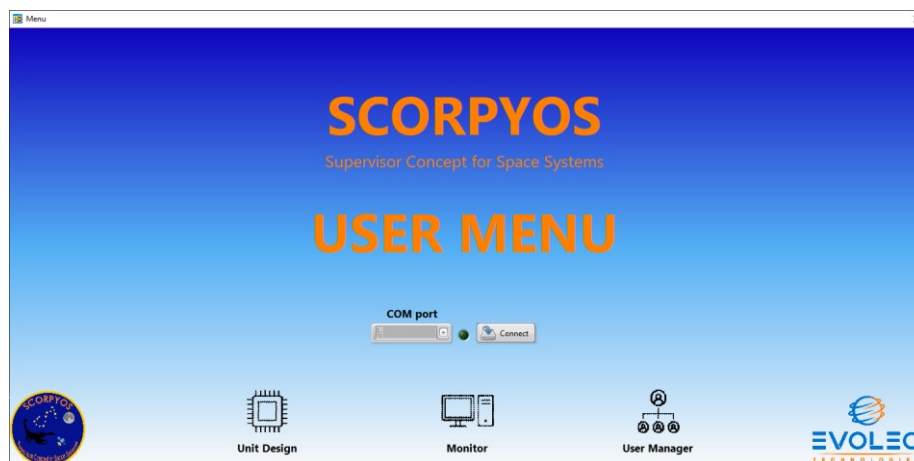


Figure 4-5 – Main Menu

In the system configuration menu, users will have the capability to save various types of configurations referred to as use cases. For each configuration, it will be necessary to first define the hardware settings, followed by the operational mode of the observability system.



Figure 4-6 – Unit Design Menu

To configure the hardware, the following menus are available:

- DIOs (Digital Input/Outputs)
- Analog Inputs
- UART (Universal Asynchronous Receiver-Transmitter)

Each of these systems can be configured as follows:

- Expected Value Check:
- Watchdog Timer Check:
- Delta Check:
- Limit Check:

These comprehensive configuration options ensure that the system can be tailored to meet the specific needs of various use cases, providing flexibility and precision in its operation.

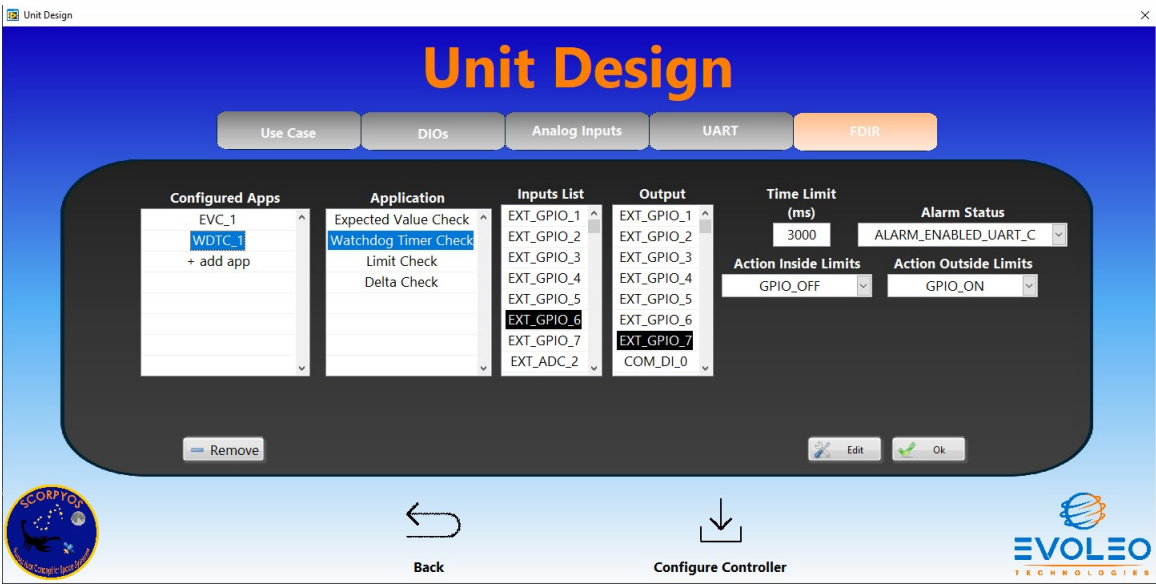


Figure 4-7 – FDIR options

The monitoring menu allows users to send either manual or automatic commands to the system, with the frequency of automatic commands defined by the user.

These comprehensive monitoring capabilities ensure users can efficiently oversee and manage system performance, ensuring optimal operation and quick identification of any issues.

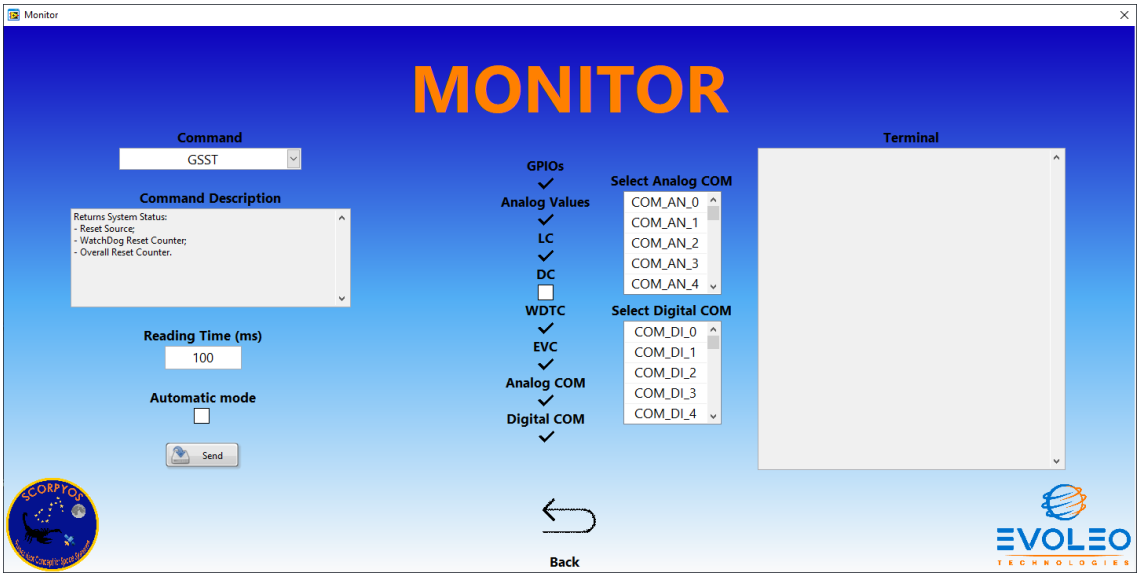



Figure 4-8 – Monitor menu

	<p style="text-align: center;">SCORPYOS Executive Summary PUBLIC</p>	<p>Proj.: 2021-002-SCORPYOS Ref.: EVODE-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 14/15</p>
---	---	---

5 Tests Performed

5.1 Simple use cases

Three fundamental use cases were defined and demonstrate the adaptability and effectiveness of SCORPYOS in different operational scenarios.

These use cases cover the supervision of digital, analog and communication systems, illustrating the ability of SCORPYOS to integrate with diverse architectures and respond to a wide range of monitoring requirements.

- The **first use case** demonstrates the application of SCORPYOS in the monitoring of digital inputs, using verification methods such as Expected Value Check and Watchdog Timer Check. It illustrates the system's ability to interpret logical states and respond to specific patterns or absence of "heartbeat" signals.
- In the **second scenario**, SCORPYOS is configured to monitor the analog signals, using methods such as Limit Check and Delta Check. This case highlights the system's ability to detect deviations from predefined values and significant variations between consecutive readings.
- The **third use case** focuses on SCORPYOS' ability to monitor and analyse data transmitted via communication interfaces, such as UART. It demonstrates the integration of message validation with content analysis, applying methods such as Data Field Check and Limit Check to received data.

5.2 Supervise a Processing board and an external PSU or DC/DC


In this test, a Processing Board (based on the Xilinx US+ MPSoC, also developed by EVOLEO) was used with SCORPYOS for being supervised.

Additionally, there was another independent system, a power supply, used to simulate a DC/DC converter.

SCORPYOS was responsible for detecting any issues within this system, actuating a GPIO to inform the system of any problems encountered.

This use case demonstrates the application of SCORPYOS monitoring the status of another system, and to do that, it will monitor the digital inputs, analog inputs and communication protocols, using verification methods such as Expected Value Check and Watchdog Timer Check, Limit Check and the Data Field Check.

It illustrates the system's ability to interpret logical states and respond to specific patterns or absence of "heartbeat" signals, the system's ability to detect deviations from predefined values and the SCORPYOS' ability to monitor and analyse data transmitted via communication interfaces, such as UART.

	SCORPYOS Executive Summary PUBLIC	Proj.: 2021-002-SCORPYOS Ref.: EVOLEO-SCORPYOS-RP-038 Date: 2025-03-31 Iss./Rev.: 1A Page.: 15/15
---	--	--

6 Conclusion

Following the completion of the tests, it was confirmed that the system is fully functional and capable of analysing external issues, effectively communicating any detected problems to the system. The system accurately describes the nature of the issues through communication channels.

In each of these cases, SCORPYOS demonstrates its flexibility through:

- Adaptive Configuration: Using the UART D interface, the system allows precise adjustments of operational parameters, adapting to the specific requirements of each application.
- Varied Verification Methods: Implementation of different verification techniques, suitable for each type of signal or data monitored.
- Automated Responses: Ability to generate digital outputs and alerts in response to detected anomalous conditions.
- Data Integration: Storage and processing of data in an internal data pool, allowing complex analyses and correlations between different inputs.

The use cases not only demonstrated the technical versatility of SCORPYOS, but also its applicability in critical scenarios where accurate monitoring and rapid response are essential.

The SCORPYOS integrated tests intended to demo the system's main features, regarding:

- System's protection (Over Voltage and Under Voltage Protection)
- System GPIOs (Analog Inputs, Digital Inputs and Outputs)
- System communication ports
- System's Alarms

Looking ahead, there is potential for further enhancement of the system's configurability.

Future work could include the ability to define whether inputs are pull-up or pull-down, implement new observability features, enable logical operations, and interconnect multiple observability systems.

As this is a demonstrator system, future versions can be evolved according to emerging needs. Each module can be seamlessly integrated into the existing system, allowing for easy expansion and continuous improvement.

END OF DOCUMENT