SYSGO
EMBEDDING INNOVATIONS

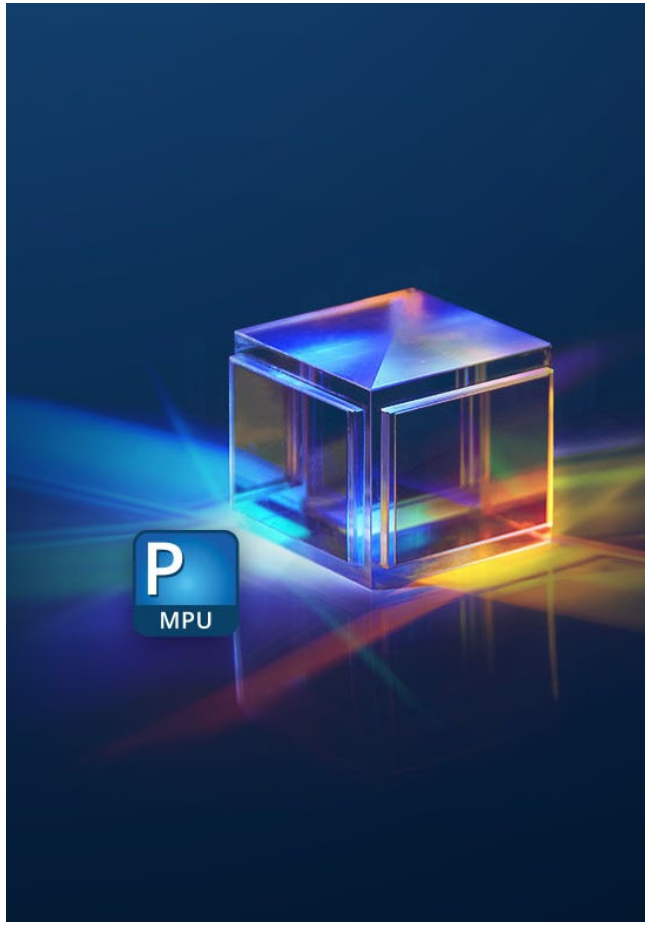
CYBERSECURITY BY DESIGN FOR MIXED CRITICALITY EMBEDDED SYSTEMS

Damir Bartakovic, Sven Nordhoff, Michael Werner, [1] Hans-Jürgen Herpel,[2] Patricia Lopez Cueva,[3] Juan Maria Carranza, [4]
(1) SYSGO GmbH, (2) Airbus Defence and Space (ADS), (3) Thales Alenia Space (TAS), (4) European Space Agency (ESA)
2023

PRESENTER

Damir Bartakovic

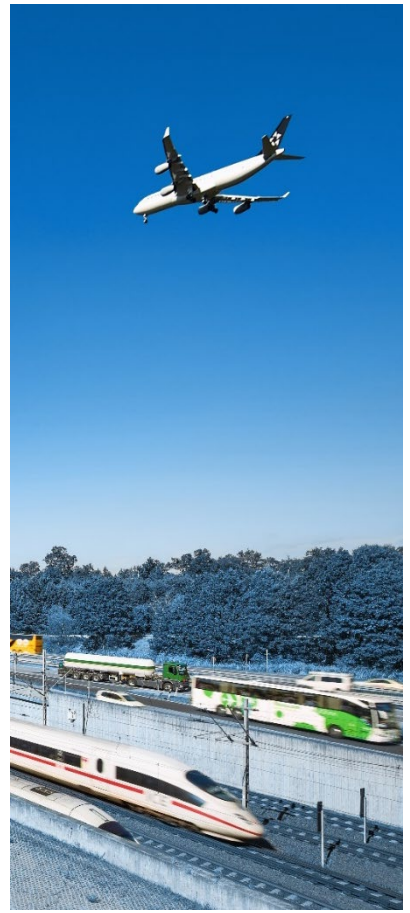
- Since 2018 at SYSGO GmbH
- Since 2021 Project Manger for PikeOS for MPU [1]



[1] https://www.sysgo.com/pikeos-for-mpu

AGENDA

- Background
  Security in Space, Study Overview, NG-ULTRA [1, 2, 3]
- Usage Scenarios, Use Case and Security Objectives
- Security Analysis
- Security requirements
- Validation and Verification
- Certification Support Package



SECURITY (AND SAFETY) OF SPACE SYSTEMS

- Safety – well established in space and aviation
  - Aviation and space safety is the result of measures taken in the design, construction, maintenance and operation of a spacecraft or aircraft to prevent accidents, failures or **unintentional** behavior.
- Security – has been historically low priority for the space and aviation industry but getting more attention nowadays
  - Aviation and space security is highly focused on the avoidance of any **intentional** outside attack or detection of vulnerabilities which in a consequence will lead to accidents, failures or unintentional behavior.

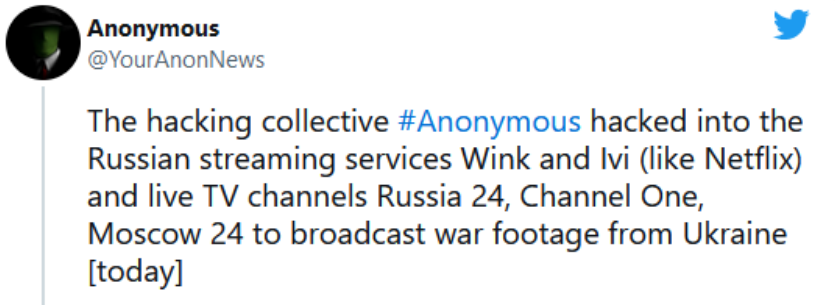
[2]

[3]

Viasat Attack [1]

Suspected target:
Ukraine's defense communication

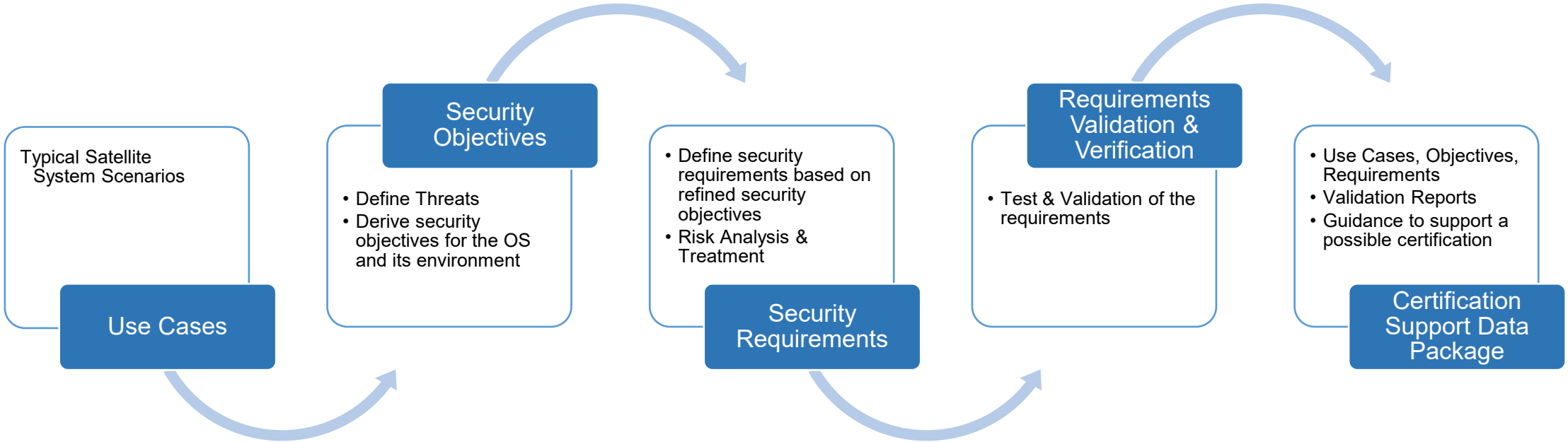
Collateral Damage
17.000 Users affected
~ 6000 wind turbines offline



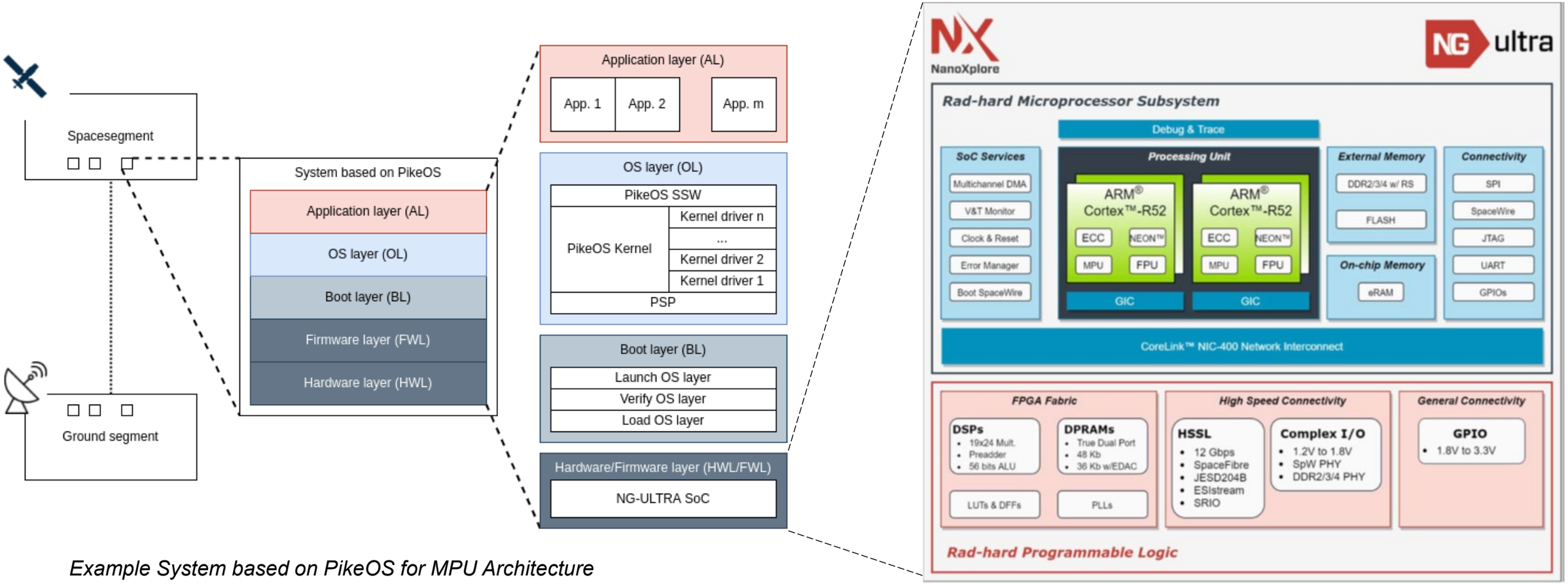
[1] https://www.wired.com/story/viasat-internet-hack-ukraine-russia/
[2] https://hack.cysat.eu/
[3] https://hackasat.com/

ESA STUDY CYBERSECURITY BY DESIGN FOR MIXED CRITICALITY EMBEDDED SYSTEMS

- In a joint cooperation with ESA, Airbus, TAS and SYSGO the result of this study was the proof that the RTOS PikeOS for MPU offers properties and features that allow implementing secure applications with different security sensitivity levels for the spacecraft microprocessor platform NG-ULTRA.



NG-ULTRA & PIKEOS FOR MPU



Example System based on PikeOS for MPU Architecture

[1] https://dahlia-h2020.eu/ (**D**eep sub-micron microprocessor for sp**A**ce rad-Hard app**LI**cation **A**sic)

[2] https://eurospace.org/dasia-conference-aspx/ - Programme : NG-Ultra: a system-on-chip suiting the upcoming space missions (TAS, May 17th 2022)

[3] https://www.sysgo.com/pikeos-for-mpu

USAGE SCENARIOS & USE CASES

In cooperation with ADS and TAS the typical satellite mission usage scenarios e.g.:

- Earth Observation
- Satellite Navigation
- Satellite Telecommunications
- Deep Space

and use cases deemed relevant for security were analyzed and mapped e.g.:

- Protection of the OS layer from applications, and applications from other applications.
- Protection of the communication to and from external systems.
- Secure Boot / Secure Update.

SECURITY OBJECTIVES

The initial security objectives were derived based on the use cases. e.g.:

- Hardware
- Exclusive access to resources
- Privileged executables

Criticality categorization of security consequences has been discussed e.g.:

Name	Level	Type of consequence		
		Dependability	Safety	Security[8]
Catastrophic	1	• Failure propagation (Only for lower than system level analysis)	• Loss of life, life-threatening or permanently disabling injury or occupational illness • Loss of system • Loss of an interfacing manned flight system • Loss of launch site facilities • Severe detrimental environmental effects	
Critical	2	• Loss of mission	• Temporarily disabling but not life-threatening injury, or temporary occupational illness • Major damage to an interfacing flight system	• Major mission data compromission • Major intellectual property compromission

Severity Categories derived from ECSS safety classification (augmented with security aspects)

SECURITY ANALYSIS

The security analysis in this study relies on a risk assessment framework similar to that introduced in [ED-203A]/[DO-356A]. This Framework defines steps to be taken in order to assess and manage the risk e.g.:

1. Identification of potential risks
2. Identification of a scenario for the risk to be exploited
3. Treatment of security risks
4. Identify countermeasures to prevent the risk from being exploited.
5. Categorize potential residual risk effects

The initial security objectives were refined and assumptions, threats added for the analysis.

SECURITY RISK TREATMENT

- The security risks are mapped to use cases identified including an adequate risk classification.
- For each risk the risk treatment means are specified.
- Security risk analysis focuses mainly on the RTOS and handles
  - Attacks on the hypervisor
  - Communication and interaction of threads in the RTOS
  - RTOS Configuration
  - RTOS Data Changes
  - Error and Fault Handling of the RTOS
  - Resource Consumption of the RTOS

SECURITY REQUIREMENTS

- The next step was to systematically map the existing security requirements specified in the security target to the risk treatments. This was done to enable the mapping of the result of the use cases, risk analysis and treatment to the security requirements actually retained in the RTOS specifications.

[ST] Requirement	[D2_D3] Risk Treatment
00106-ST-1233	SR_0092
00106-ST-1233	SR_0104
00106-ST-1233	SR_0142
00106-ST-1233	SR_0170
00106-ST-1233	SR_0176
00106-ST-1233	SR_0188

A **Security Target (ST)** is a document that specifies the security properties, objectives, and requirements of the product or system (the Target of Evaluation = TOE) undergoing the security evaluation.

VALIDATION AND VERIFICATION

- In order to test and validate the security requirements (00106-8000-ST) a mapping to the existing requirements of the RTOS components has been established in a traceability matrix (TRACE).
- This TRACE document has been combined with the test results (TR) to summarize in one single document the security tracing and achievement during testing.

		Valid Test-Result		
Module	**Baseline**	**#REQ**	**#PASS**	**%PASS**
00106-8000-ST	**42.2**	114	114	100,00%
00106-8022-TRACE	**42.1**	501	501	100,00%
00106-2000-KERN-IF	**42.4**	160	160	100,00%
00106-2060-KDEV-IF	**42.2**	66	66	100,00%
00106-3000-PSSW-IF	**42.3**	150	150	100,00%
00106-2500-PGEN-IF	**42.3**	20	20	100,00%
00106-5000-CONF-IF	**42.2**	23	23	100,00%
00106-0236-CCONV-TAR	**40.12**	131	131	100,00%
00106-0237-CCONV-TAR-VMIT	**42.1**	121	121	100,00%
00106-0238-CCONV-TAR-ROM	**42.1**	68	68	100,00%
Summary		**1354**	**1354**	**100,00%**

CERTIFICATION SUPPORT PACKAGE

The package shows which documents and artifacts must be produced, and what evidence must be collected along the lifecycle of the project based on proven standards [ECSS], [ED-203A], [DO-356A] and especially for security [CC].

Following key aspects have been identified.

- Common Criteria or a similar approach as used in this study can be the security standard.
- Certifying the RTOS and SBRTOS (**S**ystem **b**ased on the RTOS) would preferably be done separately.
- Lower-level separation security objectives should be provided and validated by the RTOS.
- Certifying SBRTOS would rely heavily on certifying RTOS.
- SBRTOS certification shall specify, high-level system security objectives including analysis of security risks and security risk treatments.

DOCKER IMAGE TO RE-EXECUTE VALIDATION

- A full-functional test-environment was provided including the
  - SYSGO Test Framework (TFW)
  - The target as Arm FVP R52 (Simulation)
  - PikeOS for MPU
- Single tests or the complete test run can be reproduced
- An introduction with a live demo how

```
nor@nor-VirtualBox:~$ docker run --net=host -it tfwc:customer-version /work/we
------------------------------------------------------------
Files

                    EMBEDDING INNOVATION
------------------------------------------------------------
Welcome
For usage you need to communicate the SYSGO license server.
export SYSGO_LICENSE_PATH=<Enter Your Server>

The FVP simulator does require a license file at '/work/fvplicense'.
To override this with:
export ARMLMD_LICENSE_FILE=<Enter Your Server>
[ChRoot tfwc-deb9.13+25-r2] root@nor-VirtualBox:/#
```

Docker

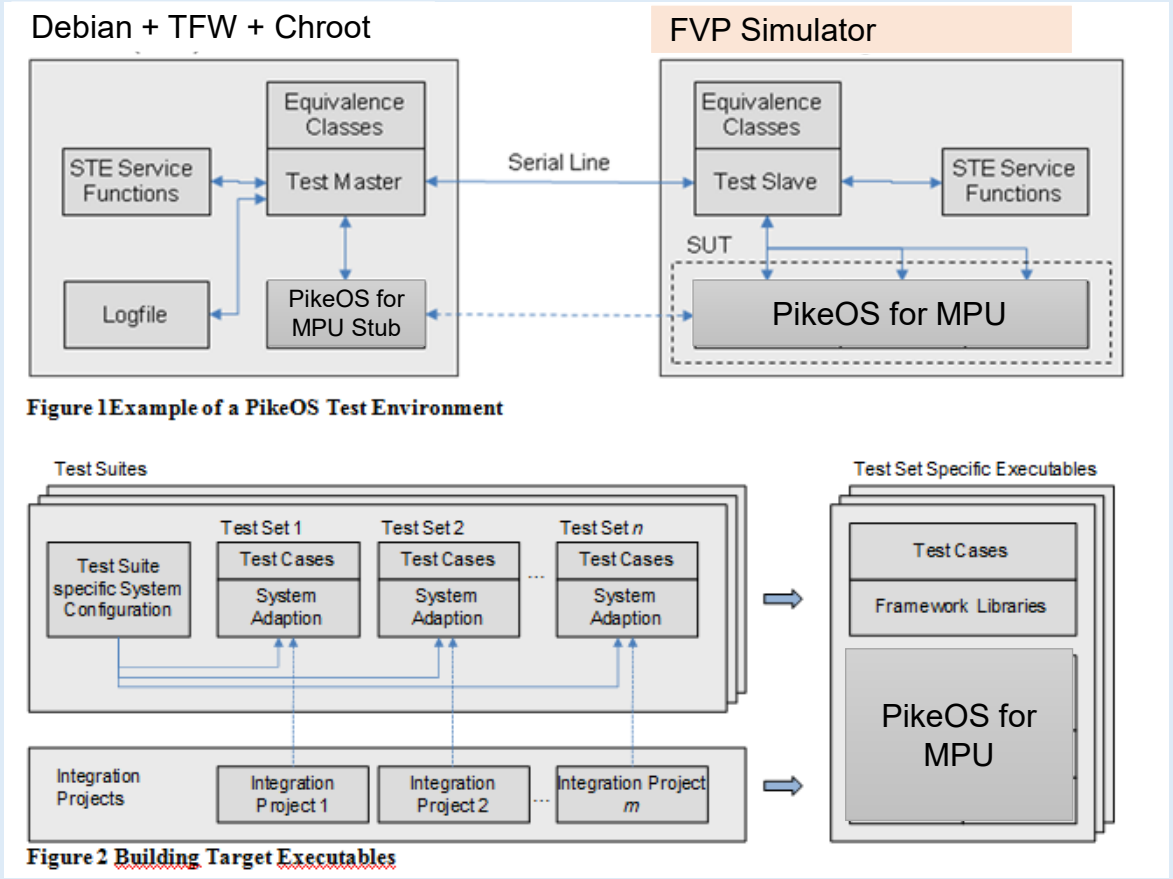


Figure 1Example of a PikeOS Test Environment

Figure 2 Building Target Executables

CONCLUSION

- Security for space has become important in most recent times
- A RTOS with a prequalified set of documentation for security is the basis to provide all security "features" to be used for security for the SBRTOS.
- Security certification on system level need further investigation depending on the requirements for the SBRTOS.
- The combination of the NG-Ultra SoC & PikeOS for MPU have the best prerequisites to fulfil the needed space requirements concerning security while maintaining safety and dependability

THANK YOU FOR YOUR ATTENTION

SYSGO GmbH

Am Pfaffenstein 8
55270 Klein-Winternheim
Germany

Phone: +49 6136 99480
E-Mail: info@sysgo.com

Sales Contact
sales@sysgo.com

Subscribe, Like and Follow:

www.sysgo.com/newsletter

www.sysgo.com/twitter

www.sysgo.com/linkedin

www.sysgo.com/youtube

www.sysgo.com