Zephyr™ Project

Developer Summit

June 8-10, 2021 ▪ @ZephyrIoT

Using OPC UA with Zephyr

VINCENT LACROIX (SYSTEREL)

Agenda

What is OPC UA ?

Deploying OPC UA on Zephyr with S2OPC

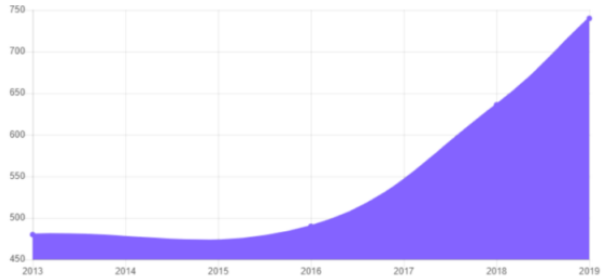
S2OPC continuous integration using Renode

OPC UA: the new industrial interoperability standard

M2M & IIoT communication protocol with SOA capabilities

International standard IEC 62541 since 2008

- Native security / Open & Interoperable / Multi-platform



The OPC Foundation

- OPC UA development & promotion /
  neutral compliance lab / certification authority

Recommended by leading Industrial IoT Consortia

USA

industrial internet CONSORTIUM

Europe

PLATTFORM INDUSTRIE4.0

China

MADE IN CHINA 中国制造 2025

South Korea

Manufacturing Industry Innovation 3.0

Japan

IVI Industrial Value Chain Initiative
Connected ! Manufacturing

A protocol Secure …

Confidentiality: encryption of messages

Integrity and authenticity: signatures

Availability: control messages size

Authentication:

- Application: X509 certificates
- Users: User/password or X509 certificates

Authorization to perform operations based on access rights

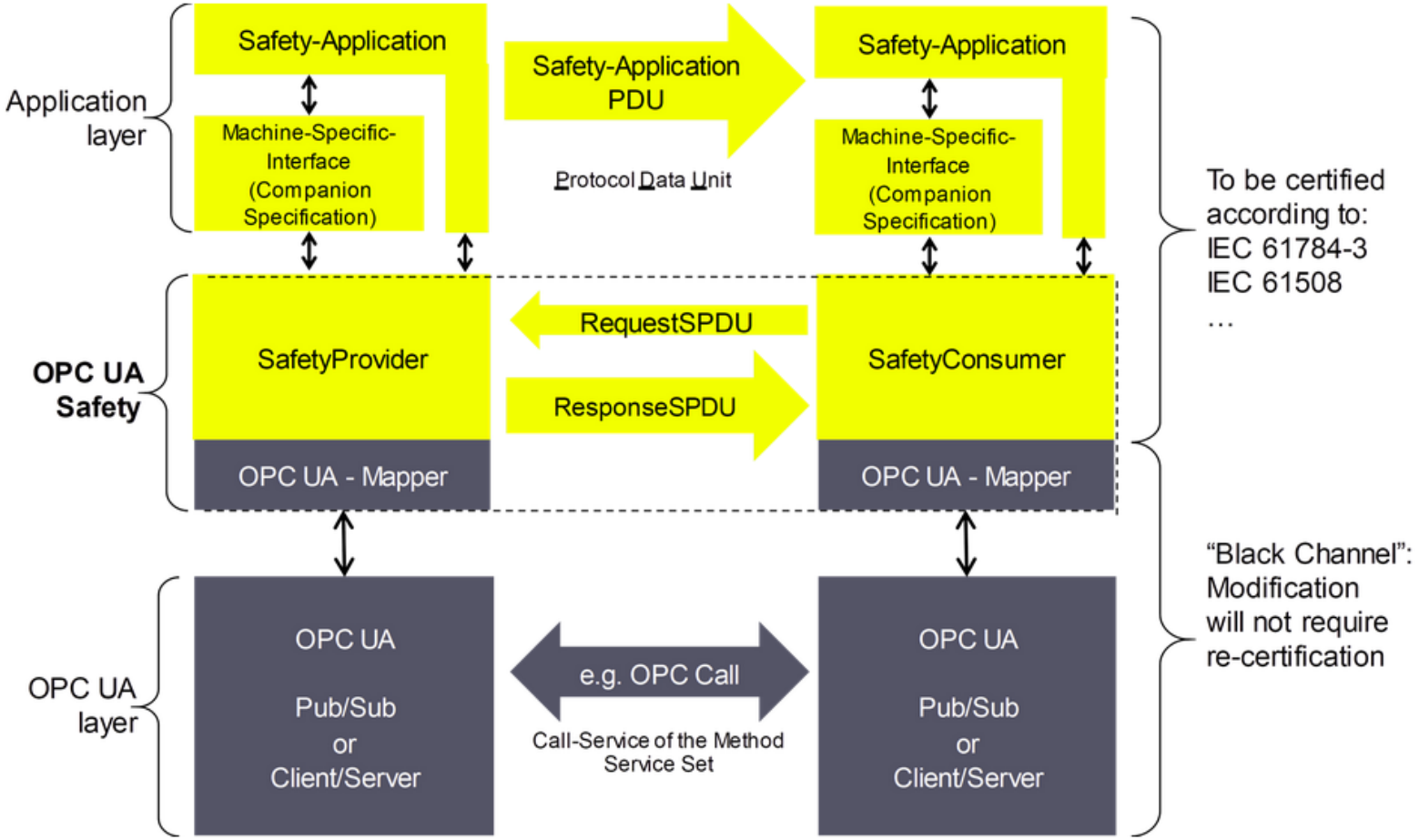
Audit: generation of audit events

In depth analysis from the BSI: no systematic error in specification

.. and Safe

Zephyr™ P
Developer Sum

OPC UA Specification part 15: Safety



Some OPC UA users

Industry:

- devices,
- PLCs,
- SCADA,
- Cloud.

Railway: gateway between trains' equipment and SCADA

Energy,

…

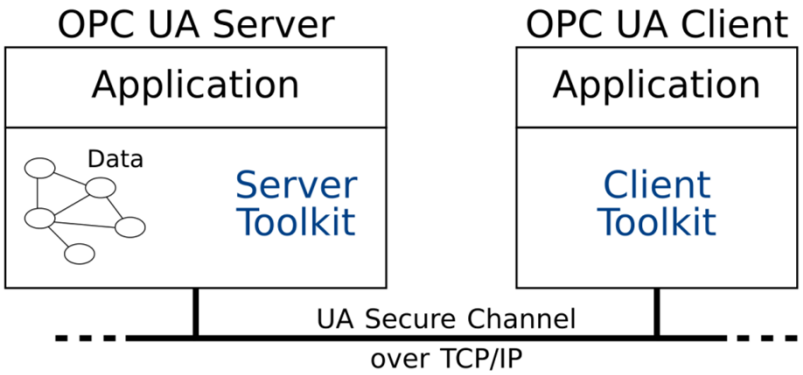
In 2018, 47 millions of automation devices with OPC UA Client technology

(ARC Advisory Group research estimation)

OPC UA Client/Server

Zephyr™ P
Developer Sum

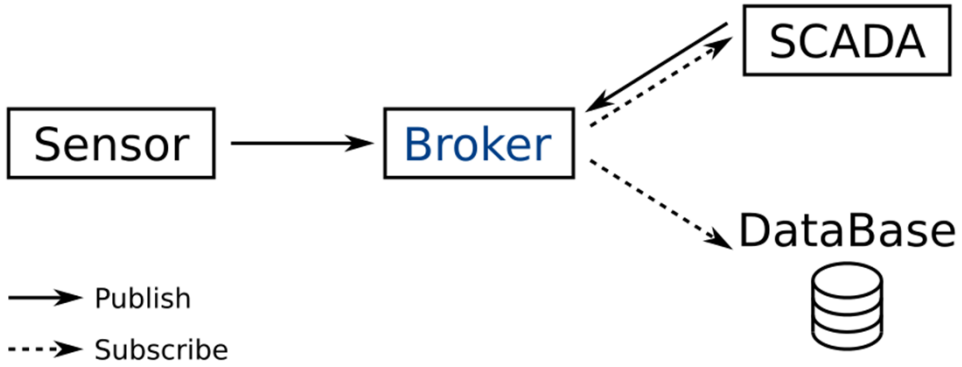
Data exchange protocol



- Server: organizes data
- Client: read/write data, browse/change organization
- subscribe to data changes, remote procedure call
- Toolkit or library or API or Stack and SDK

OPC UA PubSub

- Not OPC UA Services (PubSub ≠ Subscription service)
- Not connected
- One to many
- No Address Space
- Uses OPC UA encodings
- Publisher with cyclic updates
- With or without broker
- Lightweight
- Several transport layers:
  - UDP, Raw Ethernet, AMQP or MQTT



Safe & Secure OPC UA

Designed from scratch for certification

Scalable, High Performance, Real Time

Multiplatform

Open-Source

S2OPC and Zephyr

Same open source license (Apache 2.0)

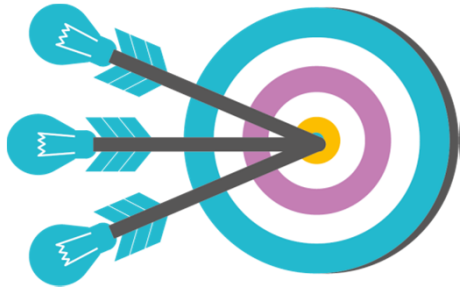
Built with safety and security in mind

Target embedded systems

Open Source

Safety and Security

Embedded Systems



Porting S2OPC on Zephyr makes perfect sense !

Zephyr and S2OPC – Current status

See issue https://github.com/zephyrproject-rtos/zephyr/issues/25046

- OPC UA Server and PubSub
- Mimixrt1064, SAME70, and native POSIX
- Security policies:
  - None, Basic256 and Basic256Sha256 for Client/Server
  - PubSub-Aes256-CTR for PubSub

Found issues:

- #24730: Access to time.h function with POSIX API
- #26435: Seg fault risk with random32_entropy_device

Suggested contributions (Contributing to #2336)

- #26584: Fix generic UDP Multicast emission
- #26585: Add imx1064 UDP multicast support

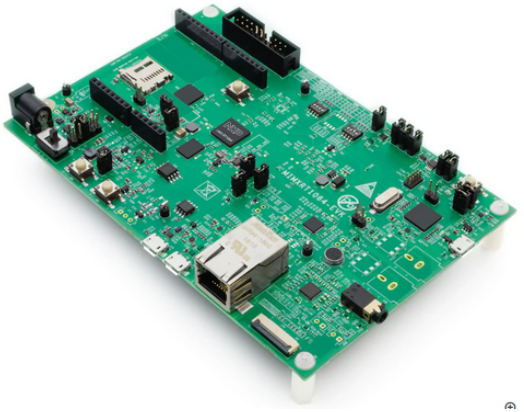
S2OPC on Zephyr – Example with Mimixrt1064

S2OPC team maintains a dedicated Zephyr branch

See instructions on the wiki: Wiki-S2OPC

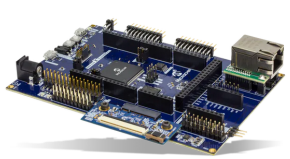
- Add S2OPC as a Zephyr module in west.yml
- Compile using West
- Flash and start application
- OPC UA server exposes its endpoints on opc.tcp://[ip addr]:4841
- OPC UA UDP PubSub messages are cyclically sent

```
*** Booting Zephyr OS build zephyr-v2.3.0  ***

Network initialize called

 Thread safe mbedtls init

Ready to launch application :)
WARNING: log creation failed, no log will be recorded !
ERROR: S2OPC Logs initialization failed!
# Info: Server initialized.
# Info: Endpoint and Toolkit configured.
# Info: Server started.
# Info: PubSub static configuration loaded.
# Info: PubSub configured through Start/Stop Command.
# Info: initialize local SKS with static security data
# Add input with context 0x80043ce0
# RT Subscriber initializer add input 0
# Load security signing key from static buffers... slen = 32
# Load security encrypt key from static buffers... slen = 32
# Load security nonce key from static buffers... slen = 4
# Rt subscriber initialized
# Rt subscriber beat heart thread created
# RT publisher created :)
# RT publisher initializer created :)
# Load security signing key from static buffers... slen = 32
# Load security encrypt key from static buffers... slen = 32
# Load security nonce key from static buffers... slen = 4
# Info Publisher : message created #0
# RT Publisher initializer : Creation of message with publishing valu
# RT Publisher initializer : Creation of rt publisher message handle
# Info Publisher : message created #1
# RT Publisher initializer : Creation of message with publishing valu
# RT Publisher initializer : Creation of rt publisher message handle
# RT Publisher well initialized
# Info: PubSub started through Start/Stop Command.
# RT Subscriber beat heart thread launched
# RT Publisher tick thread: Beat heart thread launched !!!
# Publisher variables monitoring: thread launched !!!
# RT Publisher start callback: Msg id = 0 - Started
# RT Publisher start callback: Msg id = 1 - Started
```



How to perform continuous integration ?

Continuous integration using Renode

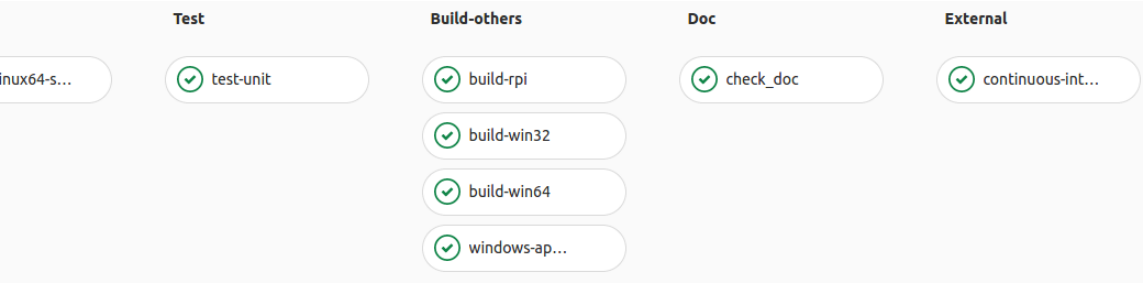


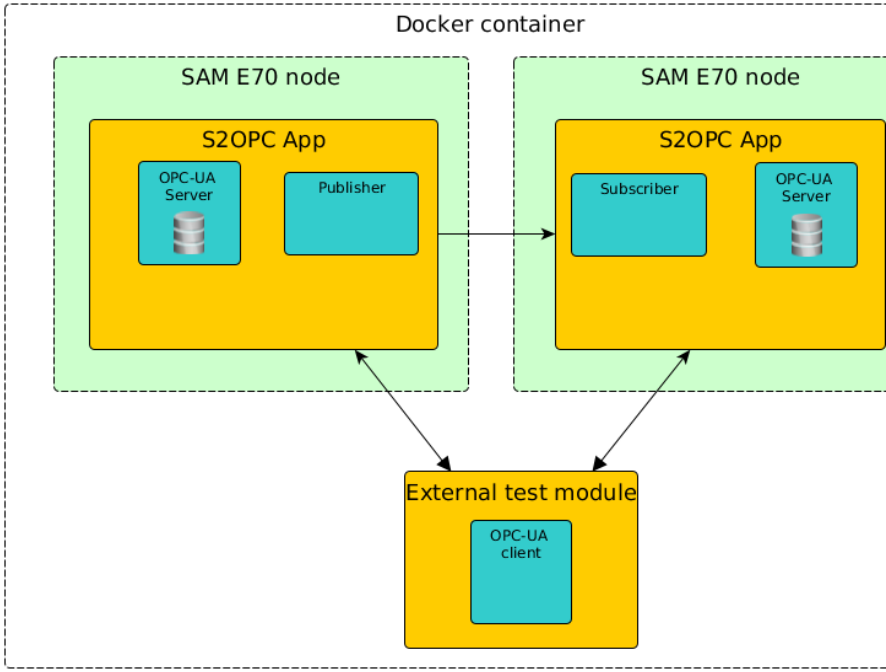
GitLab

Docker image containing Zephyr and Renode

Continuous build and test of typical configurations

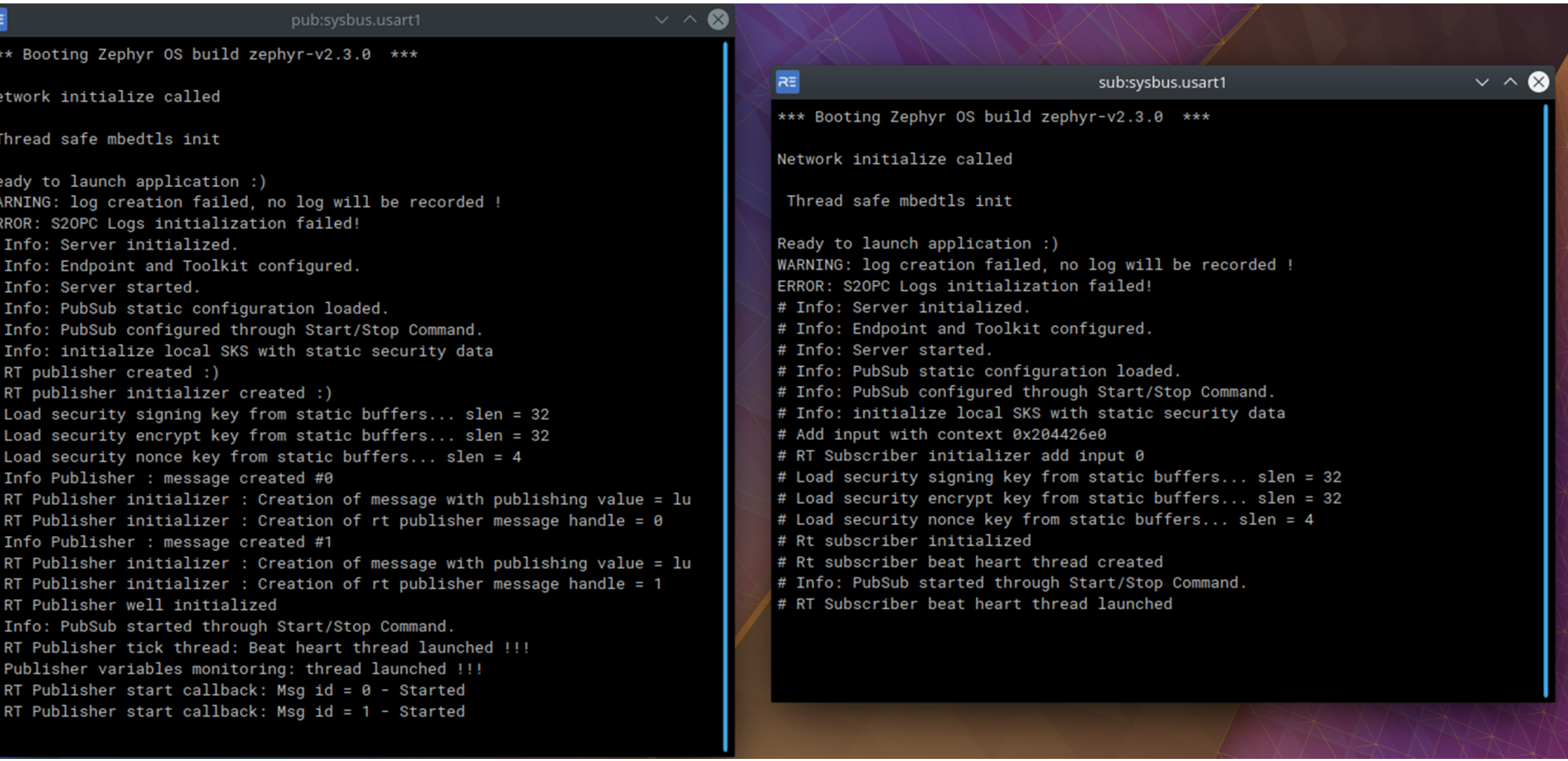
- Example with SAM E70:





Deployment in progress on Gitlab

Continuous integration using Renode



Zephyr and S2OPC – Next step

Support additional boards

OPC UA PubSub on TSN

OPC UA PubSub on MQTT

Zephyr™ Project
Developer Summit

Questions ?


Zephyr™ Project
Developer Summit
June 8-10, 2021 ▪ @ZephyrIoT