MQTT Interface for Omron PLCs to EPICS

Science and Technology Facilities Council

ISIS Neutron and Muon Source

Mateusz Leputa[1], Ajit Kurup[2]

1. ISIS Neutron and Muon Source, Harwell Campus, Oxfordshire, OX11 0QX, UK
2. Imperial College of Science and Technology, London, SW7 2AZ, UK

Introduction

The ISIS Neutron and Muon Source accelerator control system is in the process of migrating from Vsystem software running on the OpenVMS operating system to an EPICS/Linux system (**TUPDP108**). New Omron NX and NJ Programmable Logic Controllers (PLC) have been integrated solely into EPICS. These devices were initially connected using a Python-PVA Server based on a library that implements the Common Industrial Protocol (CIP) communications protocol, a library which is no longer supported meaning an alternative communication method had to be found. Leveraging existing developer experience with Python and MQTT, as well as availability of high level MQTT libraries from manufacturers, a development of an MQTT to EPICS interface was undertaken.

MQTT

MQTT is a lightweight communications protocol which allows devices of limited capabilities, such embedded devices, to exchange data on a wider scale. The protocol follows a publisher-subscriber model where a client will send (publish) data to a broker which will then notify other clients (subscribers) listening for specific data often sent in a JSON format. MQTT is already in use across the accelerator control systems stack, and it is supported by the target PLC's manufacturer OMRON.

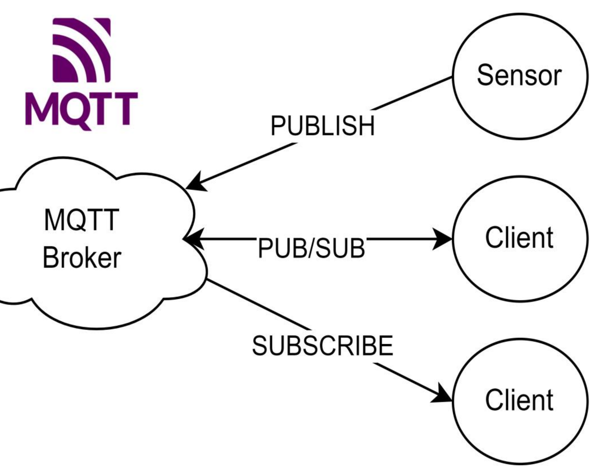

Figure 1: MQTT subscriber/publisher scheme.

Architecture and Logic

Fig. 2 shows a simplified overview of the solution system. The system is implemented in three distinct components:

- The PLC is the source of truth for the PV structure and data
- The MQTT broker which manages the publishers and subscribers
- The p4p based IOC that converts the data sent to PLC to MQTT to an EPICS PV variable.

AN existing Eclipse Mosquitto MQTT broker is deployed as a docker service on ISIS accelerator controls' high-availability infrastructure. The development effort was focused on the PLC and IOC.

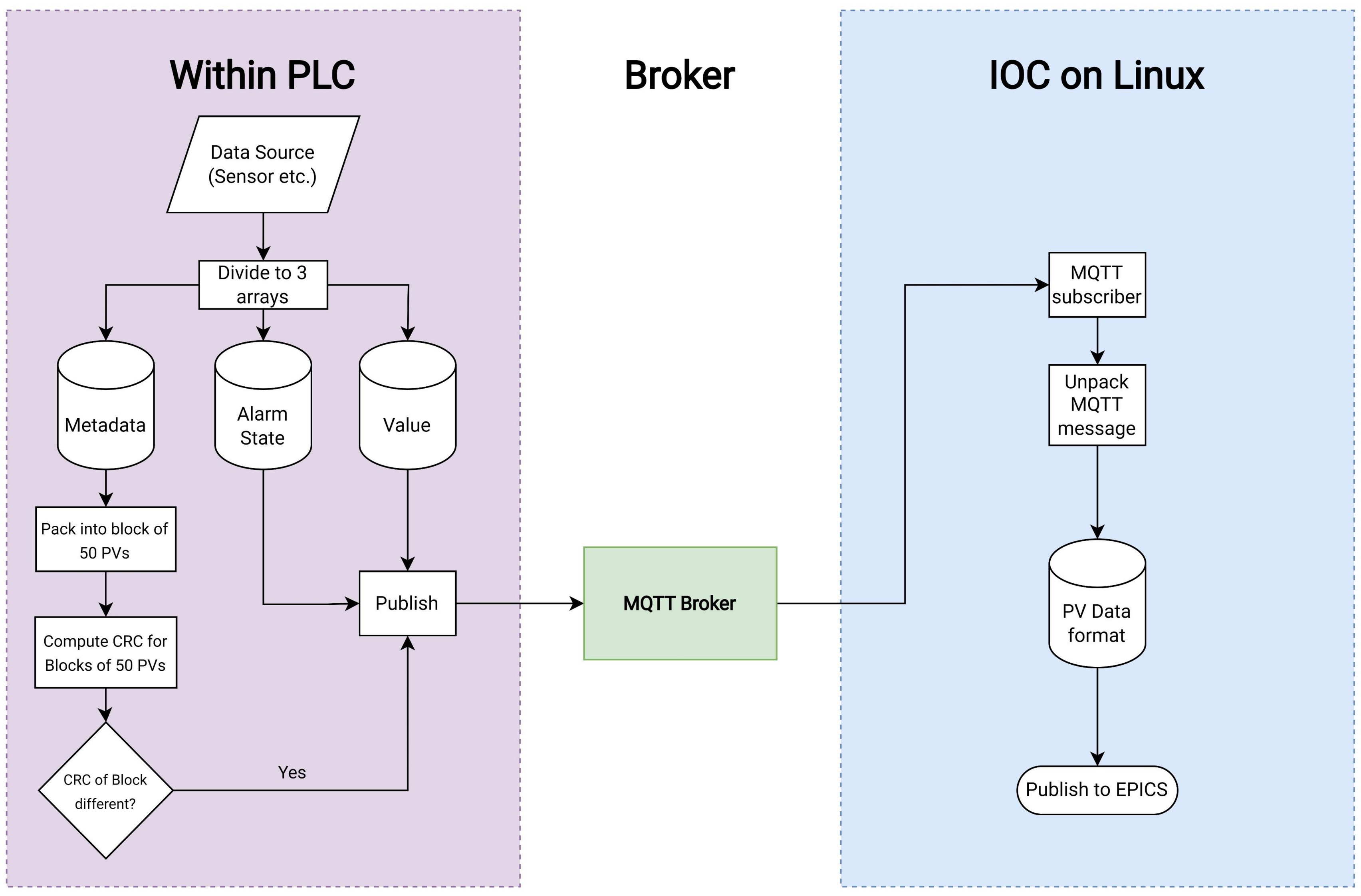

Figure 2: MQTT to EPICS interface logic flow chart

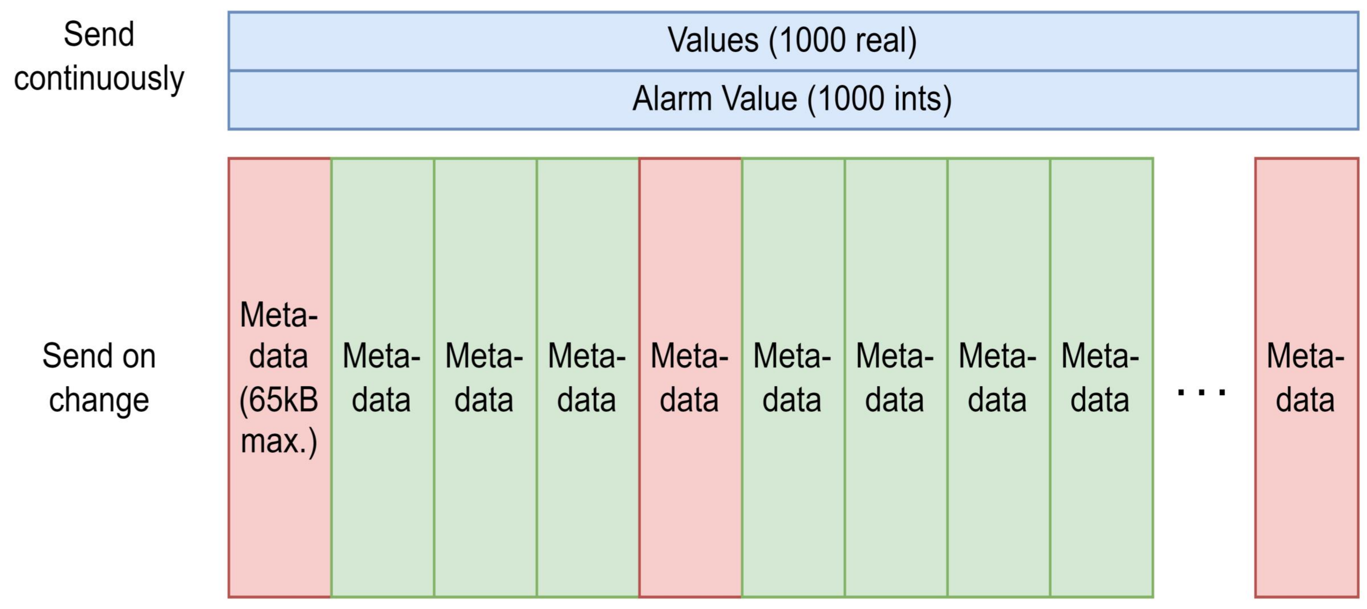

Figure 3: Illustration of the update process, red shows a change in the CRC whilst green is unchanged.

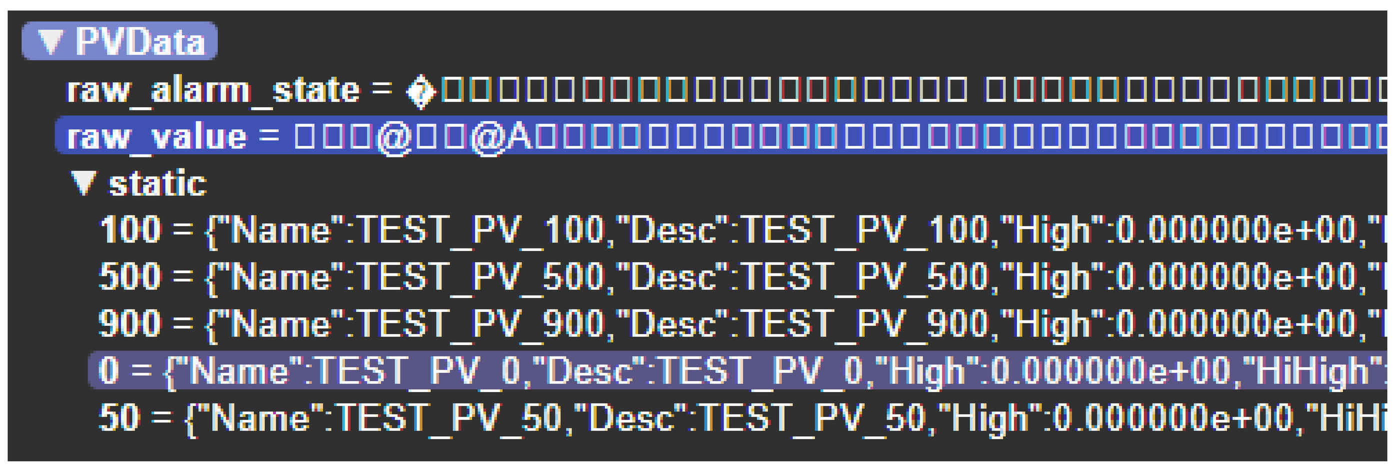

Figure 4: Screenshot of the structure as received by the MQTT broker

The PLC dictates the number of PVs, the values, alarm values and other data within the structure of the PV. The alarm state and the PV values are sent as a single array as fast as the PLC allows.
The IOC uses the position of the data within the received array as well as the name of the metadata topic within MQTT to decide which values the PV metadata belongs to.
This way the metadata can be sent asynchronously and can be fully defined and configured within the PLC logic, the only configuration required within the IOC is the root device topic within MQTT. The IOC can also request a refresh of all metadata by publishing a flag which the PLC is subscribed to.

The PLC code comes in two variants:

- Generic with support for both NJ and NX models
- Optimised NX specific version.

The NJ variant code is based on a slower MQTT implementation. The table below shows the rate of updates per second achieved for different number of PVs

Model variant	50 PVs	1000 PVs
NX	33 updates/s	13 updates/s
NJ	20 updates/s	8 updates/s

The difference in the rate of updates per second is attributed to the NX specific implementation being more optimised. It should be noted that the majority of applications in ISIS use less than 50 PVs, with notable exceptions requiring more than 100 PVs.

Overall, the system has a relatively low development overhead as most of the libraries used are high level and supported by manufacturers and/or large organisations (see Paho-MQTT, OMRON). As such, the interface can be also easily adapted to work as a generic EPICS to MQTT interface.

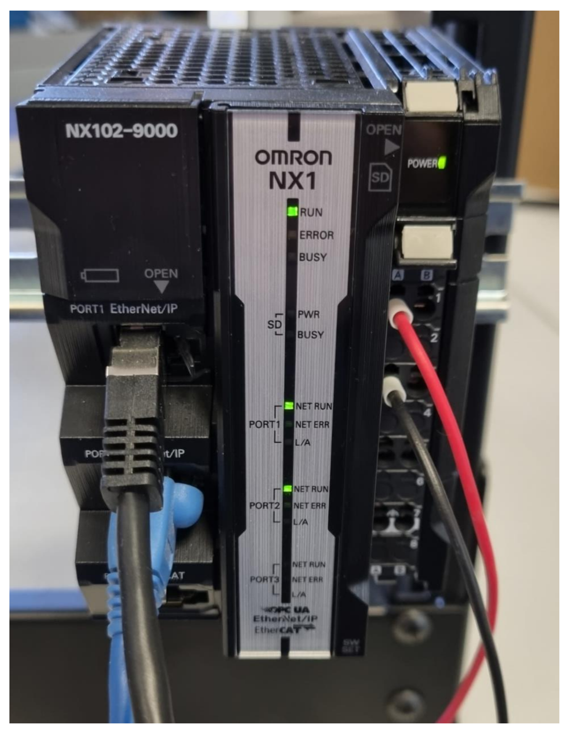

Figure 5: Example PLC

Conclusion

We demonstrated a successful adoption of MQTT as an EPICS IOC, leveraging existing developer experience and vendor support for MQTT. We also show that the system allows a high refresh rate with a 1000 PVs by intelligently updating the metadata only on change of variables. We also discussed that due to the modular implementation of the IOC a possible generic MQTT-EPICS adapter may be developed

Find out more about ISIS