

Configurable device connected to IoT cloud service for managing and visualizing sensors and actuators

Author: Ing. Martin Žingor

Supervisor: doc. Ing. Peter Papcun, PhD.

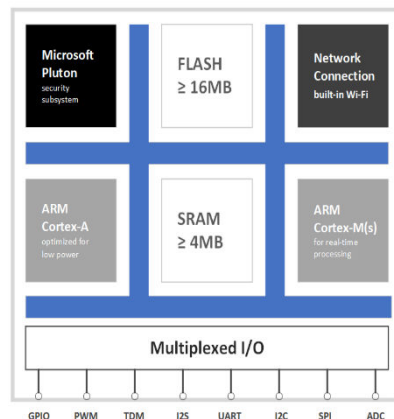
Introduction

Our goal was to create a web application that allows the user (with minimal IT skills) to manage configurable Azure Sphere MT3620 devices from anywhere. Management means adding sensors, actuators and simple automations. Part of this work is also processing of incoming data from connected devices. Our goals:

- Configurable device
- Service for device-application communication
- Web application

Device

As a device we chose Azure Sphere MT3620, which is manufactured by Microsoft corporation. It is a secured, high-level application platform with built-in communication and security features for internet-connected devices. As a most interesting feature of the device is **Hardware-based root of trust**. This feature ensures, that the device and its identity cannot be separated, thus preventing device forgery or spoofing.



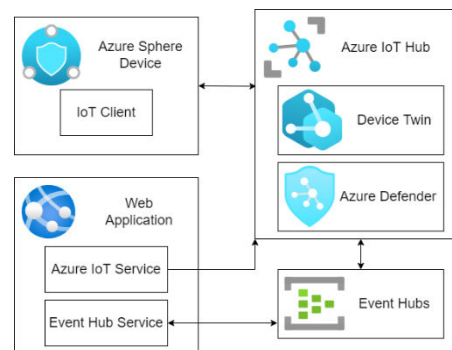
Microsoft I/O
Firewalls

Architecture

To achieve the goals we defined earlier, it is necessary to work on 3 separate parts of the solution simultaneously.

These parts are:

- **Device codebase** – Microsoft has a public repository, where you can download and update code samples, that are written in C.
- **Frontend of the web application** – For the web application, we started from the scratch. We used React open-source library and as a programming language, we used Typescript.
- **Backend of the web application** – This part is written in C#, with use of .NET 6 framework, Entity framework (communication with database) and local PostgreSQL database.



As we can see in the second picture, we focused to create modern IoT solution, that takes advantage of all the newest IoT technologies: device twin, Over-the-air updated, error reporting, renewable security.

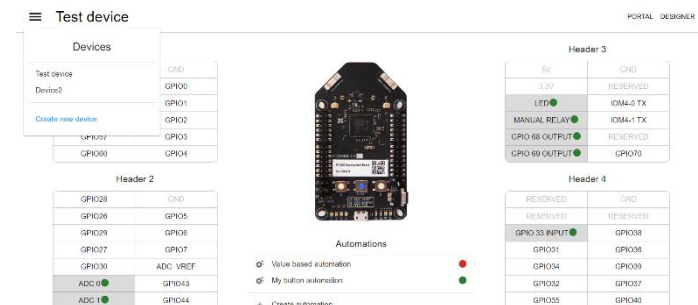
IoT services we used:

- **Device twin** – JSON representation of the device
- **IoT Hub** – library for communication
- **Storage blob** – unprocessed messages storage

- **Event Hub** – processor for incoming and outgoing messages

Results

We successfully created solution, where user can easily create and update device and its pins, create automatizations, turn on/off actuators, see device telemetry. Device states and settings are stored in device twin, that way we eliminate all inconsistency that can occur. All pending messages (offline device) are stored for further completion. We can see main screen of the application in the picture below.



Conclusion

Solution is unique, because we used nontraditional device, which is relatively new. We achieved: global accessibility, read/write actions for actuators and sensors, automatizations, advanced security, reliable and instant communication. Solution is generic, can be used with different devices connected to IoT Hub. We don't recommend device for school purposes, because this development kit can't be used in production. For production use, MT3620 chip has to be integrated to our own circuit board.