The Integrity of Things

for

Innovate UK

AIN3830

30/04/2019

	Name	Signature
Author:	Rikki Coles	
Technical review:	Gavin Hill	
Authorised:	Michael Lewis	
Authorised distribution:		

AMRC proprietary information.

Executive summary

Industry 4.0 powered by the Industrial Internet of Things is revolutionising manufacturing across the world. This is being catalysed, in part, by the proliferation of performant, low power and low cost single board computers. Combined with an explosion in ‘maker’ culture, open online communities, and mature open-source software the route to a fully digitally connected shop floor can be both fast and affordable. Through experiment, this project aims to objectively compare the suitability of several devices for industrial deployment to inform AMRC staff and visitors of the appropriate options by considering reliability, cost, quality of documentation, and ease of development.

List of Abbreviations

DC – Direct Current

FPGA – Field Programmable Gate Array

MQTT – Message Queue Telemetry Transport

NTP – Network Time Protocol

OEM – Original Equipment Manufacturer

OPC-UA - Object Linking and Embedding for Process Control Unified Architecture

PLC – Programmable Logic Controller

REST – Representational State Transfer

TCP – Transmission Control Protocol

TLS – Transport Layer Security

Introduction

The networking of machinery, sensors, and controls systems on the factory floor can enable rapidly reconfigurable manufacturing for real-time optimization of equipment and production efficiency. The Industrial Internet of Things (IIoT) is enabled by a number of key technologies such as cyber security, cloud computing, mobile and WiFi, real-time analytics, machine learning, and big data: but this report will focus on the "things" in IIoT. These "things" typically consist of a networked device running an application which subscribes to and publishes information about its machine or sensor, to and from any other equipment on the shop floor. The information about the local machine, which generally constitutes sensor readings of operational and environmental parameters, is obtained either via a native interface with the PLC or via direct sensor measurements. As OEMs embrace Industry 4.0 principles, new manufacturing equipment is emerging with a greater exposure of this information to the network using open and standardised network communication protocols such as MQTT, OPC-UA and REST. However there are many manufacturers in the UK who may require an entirely novel solution, or for whom it is not cost effective to upgrade or purchase proprietary IIoT equipment. Therefore IIoT devices which are widely available, well documented, robust, and cost-effective are needed for engineers to fill the gap.

The devices to be tested were down-selected from a representative cross-section of commonly available devices in the UK. Popular options with hobbyists are the Raspberry Pi and Arduino development boards. These are very cost effective devices with extensive online documentation and active open source communities. The Arduino-compatible Nucleo development boards from ST Microelectronics are gaining traction amongst researchers and engineers as more performant alternatives to Arduino development boards. Perhaps stimulated by the success of the aforementioned options, more industrial offerings, such as the IoT2020 from Siemens (based on the hobbyist-orientated Intel Galileo platform) have become available. Another popular option, particularly for when high fidelity sensor readings are required, is the FPGA based CompactRIO from National Instruments (NI). This list is not exhaustive, but is intended to be representative of the common "go-to" devices at the time of the experiment.

Experiment

To provide a fair comparison which is representative of an actual industrial deployment, it was decided that a number of prerequisites must first be stated to aide with device selection. The devices must have at least one Ethernet port, analogue and digital I/O ports, and must be capable of SSL encryption/decryption. Physically the devices must be sufficiently enclosed to be installed safely in a typical electrical/control cabinet, and must accept either 5V or 24V DC power supply. These requirements lead to the specific selection of the Raspberry Pi Model 3B+, Arduino Yun Rev. 2, ST Microelectronics Nucleo F767ZI, Siemens Simatic IoT2020, and NI CompactRIO 9063 as the most suitable offerings from each manufacturer.

To provide a standard comparison of a representative workload, a synthetic workload was designed for all devices. These tasks were: reading of internal case temperature (Digital Input), reading of real-time power consumption (Analog Input), animating an addressable RGB LED strip (Digital Output), PWM speed control of a servo motor (Pseudo-Analog Output), and publishing of sensor readings using a TLS encrypted MQTT connection every second.

Each published MQTT message contained a timestamp of when the message was sent from the device. This was picked up by a central webserver, where if the device timestamp was within 5 seconds of the server time the device was considered to be online. The device uptime is calculated as

$$Device\ Uptime\ (\%) = 100\left(1 - \frac{Device\ Downtime}{Total\ Server\ Uptime}\right),$$

where the device downtime is counted as the time since the last MQTT message was received, and the total server uptime represents the full experiment run time, measured by MQTT messages published from the server. The device downtime is used to calculate this metric to account for slight variations in the publishing interval of the MQTT messages from each device, which would lead to discrepancies in the uptime due to accumulated errors.

The devices were all mounted to a display board along with a touch screen monitor, which sat in the reception of the AMRC's Factory 2050 for 6 months at the time of writing. A dashboard on the touchscreen display allowed visitors and staff to interact with the demonstrator allowing them to view current performance metrics, additional device information, and comparison tables. The setup is shown in Figure 1. To gain additional information about how people were interacting with the demonstrator, anonymous statistics were recorded to see which aspects of the dashboard visitors found most interesting. Included in the dashboard was a 'pulse' button which sent an MQTT message to all devices, who respond by animating a different coloured LED propagating from the screen to each device and back again, to emulate data flow. This feature was included to allow the visitor to see the delay with which each device responds to network stimulus.

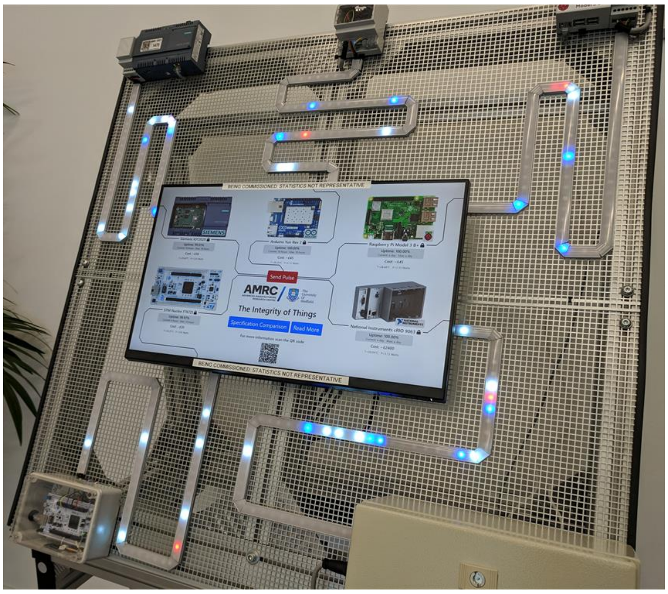

Figure 1. Experiment display board and touchscreen dashboard during commissioning phase.

Results

At the time of writing, the uptime metrics are summarised in Table 1 below. The total cost includes that of any required enclosures to bring the device up to the prerequisite standards.

Table 1 – Device uptime metrics over 5 month test period sorted by uptime percentage. Note that these results are indicative only, as other factors such as power cuts and network dropouts will have affected the exact uptime metrics.

Device	Total Cost (GBP)	Uptime (%)	Maximum Uptime (days)	Average Power Consumption (Watts)	Temperature (°C) [Min/Max]
Siemens IoT2020	**50**	**99.95**	**30**	**6.1**	**35.8/25.6**
Raspberry Pi	**60**	**93.19**	**22**	**2.3**	**34.8/24.8**
NI Compact RIO	**2400**	**89.81**	**30**	**2.8**	**26.4/16.7**
STM32 Nucleo F767ZI	**35**	**84.72**	**7**	**0.7**	**26.5/16.9**
Arduino Yun Rev. 2	**60**	**58.06**	**7**	**0.7**	**34.4/25.1**

Unfortunately, it was found that the design of the uptime calculations is affected by relative network and power failures as the server and devices are physically located in different buildings. The impact on total uptime has been small, as the top performing device lost only 0.05% overall, however the maximum uptime will be reset by even a brief outage. Relative variations are still observable between devices however, so clearly other factors are involved. In future experiments, all services must be brought onto the same power supply and local network to mitigate this.

It is unlikely that hardware issues have contributed to the sub-optimal uptimes, as the environmental conditions have been well within the device tolerances and many of the devices continue to operate at the time of writing. The authors speculate that many of the devices actual hardware uptimes are close to, if not exactly, 100%. What is likely then is that these are due to software issues. Some of which have been identified are:

Nucleo F767ZI – The library used for network time updates would sometimes cause a hard fault on the device. Due to the watchdog timer on this device being disabled out of the box on the Mbed-OS platform, the device would hang until the fault was noticed. The cause of this issue has not yet been identified, but by enabling the watchdog timer the issue has been mitigated to some extent.

Arduino Yun – The Yun circumvents the shortcomings of the ATMega microcontroller by including a microprocessor which runs a linux OS, handles encryption, and the higher level application logic. However, the bridge software that allows data to pass between the two chips exhibits some instabilities which leads to the MQTT client on the linux OS disconnecting when the 'pulse' button is pressed too quickly.

IOT2020 – No issues observed during the experiment however a memory leak was identified in the Intel MRAA library, used for input/output (I/O) control, prior to the start of the experiment which would cause the device to crash after several hours.

Rasperry Pi – No software library issues observed, but due to an unhandled exception when verifying the NTP update the master python script would stop, and hence would appear offline until restarted. This was patched during the course of the experiment.

Compact RIO – Occasionally, an unhandled exception would take this device offline but the cause of this is unknown. The authors have no reason to suspect that this is hardware or software library (LabVIEW) related.

In terms of visitor interaction, we have seen 429 page clicks to open the main dashboard page (a placeholder is shown on screen, when idle, which explained the purpose of the demonstrator). This represented good engagement with visitors to the demonstrator who wish to learn more about the IoT devices. Of those, 81 people went on to look at a side-by-side comparison of the board specifications, and then between 37 and 56 clicks to look at each specific device. 42 people loaded up a page which gives further detail about the experiment itself. The 'pulse' feature proved very popular, with over 1000 clicks to initiate the LED pulse animation.

Conclusions

Due to the aforementioned experimental artefacts, it is difficult to objectively recommend the best overall device(s). However as this is the first comparison of its kind, the uptime figures do provide a degree of qualitative description, the main conclusion of which is thus: with exception to the Arduino Yun, there is not a large variation in uptime under these experimental conditions that is inherent to the hardware. Due to much lower uptime than all other devices in the experiment, the authors cannot recommend the Arduino for anything other than an educational or basic prototyping tool.

The next consideration is cost, which is where the NI CompactRIO stands out at over 40x more expensive than the next device. The high fidelity sensor measurements offered by this FPGA based device are simply overkill for this kind of workload. However, in applications which demand these kind of measurements the CompactRIO is by far the best performer and fully justifies its cost.

Of the remaining three devices, the Raspberry Pi and STM32 Nucleo boards require additional hardware to create a safely enclosed device. If a custom enclosure is being made anyway, then these two boards make attractive options for a prototype, but this extra time and cost must be factored into the development and ongoing support time. For the Raspberry Pi many options are emerging based on the compute module such as the Kunbus RevPi series and the Brainboxes NeuronEdge. The STM32 Nucleo board is inherently a prototyping device and is a more performant "drop-in" replacement for many Arduino based projects, but offers much more capability thanks to its Cortex-M series MCU and Arm Mbed-OS platform APIs.

Based on the Intel Galileo platform, the Siemens IoT2020 inherits many of the benefits of this open-source platform but wraps them up in a rugged package, from a well-known and established OEM. This device had the best uptime, was among the cheapest options, and was among the fastest to develop reliabel software for. Unfortunately, the

Siemens IOT2000 series is deprecated as of summer 2019 due to Intel discontinuing production of the Quark chip on which it is based. At the time of writing the authors are not aware of any similar replacement at this price point.

Recommendations

The primary conclusion from this experiment is that low cost and open-source IoT devices can offer a greater degree of freedom for developing new sensor-driven applications, and can provide real-time insight for a fraction of the cost of an enterprise grade solution. These devices give full development control to the engineer which is beneficial as it gives the most flexibility for new functionality, but also places a greater responsibility on the engineer to avoid common low-level programming risks. In addition, the temptation to save money upfront by buying cheaper hardware is often outweighed by the additional time required for testing and integration at the hardware level. It is these points that have likely lead to the proliferation of many of the opinions of the suitability of these devices to industry. However, with appropriate design considerations, these devices can be extremely flexible and powerful tools.

There is increasing appearance of devices from Kunbus, Brainboxes, and Harting (amongst others) using a model of standardised application containers which can be deployed to the device. This allows for the engineer to focus on the final application, to speed up the development time and improve quality, whilst maintaining a degree of development flexibility. Combined with the availability of low cost networked sensors (e.g. Modbus TCP), these higher-level options may be the better option for applications evolving from concept into production as the IIoT becomes more commonplace. These offerings necessarily cost much more than the candidates in this experiment to support the additional testing, certification, and support but the benefits that accompany them justify the additional expense in the long term.

Benefits

Further to the principle aims outlined at the beginning of this report, several additional benefits have been identified to The AMRC and wider community. Our engineers have gained a greater understanding of the inner workings of a broad range of devices and technologies and thus are better prepared for future projects in a more diverse range of applications. As this experiment is the first of its kind known to the authors, this exercise has helped to dispel many personal preferences and opinions. We have seen stimulated debate and interest from visitors, many of whom are delighted that the benefits of IoT in manufacturing can be so readily and cheaply enabled with existing or developing skillsets. Furthermore, this project has prompted the development of similar comparison testbeds for other aspects of IIoT including integration buses, cloud platforms, and cyber security which will hopefully continue to demystify the enabling technologies for the fourth industrial revolution.

For more information please contact Rikki Coles (*r.coles@amrc.co.uk).*