Enabling ESP32-based IoT Applications in Building Automation Systems

Carlo Guarnieri Calò Carducci, Antonello Monti
Institute for Automation of Complex Power Systems
RWTH Aachen University
Aachen, Germany
CGuarnieri,AMonti@eonerc.rwth-aachen.de

Markus Hans Schraven, Markus Schumacher, Dirk Mueller
Institute for Energy Efficient Buildings and Indoor Climate
RWTH Aachen University
Aachen, Germany
MSchraven,MSchumacher,DMueller@eonerc.rwth-aachen.de

***Abstract*—Low-cost system-on-a-chip (SoC) microcontrollers with integrated wireless network capabilities represent a constantly growing and reliable solution for interfacing industrial sensors and actuators commonly used in Building Automation Systems (BAS). These cheap and powerful devices are expediting the development of IoT applications and this process is even accelerated by the forthcoming roll-out of the 5G technology. However, built-in analog-to-digital (ADC) and digital-to-analog (DAC) converters are often characterized by low accuracy and non negligible gain and offset error that actually turn out to increase the final cost and complexity by requiring additional hardware.**

In this work, we analyze the requirements of a wireless interface for BAS and we discuss some key design considerations arising from the use of a ESP32 microcontroller. The scope of such adapter is to provide retrofit compatibility to the already installed devices, by supporting the migration to IoT platforms and by providing at the same time flexible 5G-ready capabilities. The performed analysis shows that the adoption of proper measures to compensate the errors of built-in converters, effectively enables the use of ESP32 in low-cost applications with industrial-grade performance.

***Index Terms*—Building Automation, IoT, 5G, ESP32, ADC, DAC, Calibration, Gateway**

I. Introduction

The Internet of Things has made its way into many applications over various fields already, one of them being building automation systems (BAS) [1]–[3]. In such BAS, IoT integration is achieved by adding Internet-ability to sensors and actuators, by either direct connection to the network [4], [5] or interfacing specific gateways [6], [7] for data acquisition (DAQ) and control. The number of connected devices constantly increases, so does the requested data bandwidth and the demand for flexible wireless technology. In this scenario, the 5G technology is approaching industrial utilization with a set of new technologies that can respond to the latest challenges.

At the same time, managing the Industry 4.0 [8] transition of the pre-existing BAS infrastructures to IoT platforms ready for the next communication technology is yet challenging [9]: the diversity of the installation scenarios and applications, the fragmentation of the communication protocols, the country-dependent available spectrum and supply constraints that require custom designs based on application-specific energy budget analysis, make it difficult to implement generic solutions.

Several commercial solutions [10]–[12] are available in the market of industrial devices. However, they mostly target new installations or otherwise would require the replacement of already installed devices, even if still properly working. Recent works [13] try to tackle this problem by developing network adapters, however the support is extended only to digital interfaces. Our solution, on the contrary, a wireless interface for sensors and actuators previously developed in a preliminary prototype version [14], is capable of supporting many physical interfaces commonly used in BAS and providing modular support for the integration in different network architectures.

In this work, we first provide our analysis of the interface requirements for a wireless adapter in BAS. We then describe an architecture based on the recently released in the market but already widely used ESP32 micro-controller [15]. Finally, we analyze the performance of such SoC and we provide useful considerations to enable its use in reliable industrial applications.

II. Wireless Interface

The market of sensors and actuators for industrial and building automation is characterized by a high fragmentation in terms of physical interfaces, protocols and standards. However, our investigation of open and license-free protocols summarized in Table I, led to identification of the most recurrent subset of interfaces.

TABLE I
Common wired interfaces in BAS

Type	Port	Application	Trend
Analog	2/3/4 wire	Resistive measurement	2.7%
		0(2) - 10 V voltage signals	7.4%
		0(4) - 20 mA current signals	14.8%
Digital	RS485	Modbus	58.3%
	Ethernet	BACnet	11.4%
	EN 13757-2	M-Bus	5.4%

Trend data are the results of worldwide searched keywords by Google Trends

Analog two-wire interfaces for active devices are either based on voltage or current signals. The former is the simplest

978-1-7281-0429-4/19/$31.00 ©2019 IEEE

and oldest interface, the output from a sensor or the input to an actuator is mapped to a fixed range. A more reliable analog interface encodes the transducers' IO in a current signal flowing in an isolated (2-wire) or grounded loop (3-wire). This requires a more complex circuitry then a voltage interface, but also provides improved noise rejection and no attenuation over long distances. Many ranges have been used over time for these two interfaces, but 0-10 V and 0-20 mA have become widely adopted standard ranges. Both are also often used in the live-zero configuration with ranges of 2-10 V and 4-20 mA respectively, where the offset is used for fault detection or for supplying slave devices.

Similarly, passive sensors are based on 2/3/4-wire interfaces, depending on the application and on the requested accuracy. An example are resistance temperature detectors (RTD), force sensitive resistors (FSR) and strain gauges, just to name a few. Aside from the basic 2-wire configuration, where the terminals are used to both biasing the resistor with a known current and reading the induced voltage drop, additional terminals can be used for accurate voltage sensing, especially when the resistance of the sensing element is small and comparable with the parasitic resistance of the wires.

Digital interfaces have enabled the coexistence of multiple devices on the same bus and the use of more complex network protocols. Common hardware layers are RS485 and Ethernet, which respectively allows for cheap low-speed multi-drop topologies that only need two wires in half-duplex configuration, and more expensive TCP/IP based high-speed communication. On top of these, several application layers have been developed to implement Industrial Control Systems. Modbus and BACnet are among the most common network protocols for industrial and building automation respectively. Similarly to Modbus, the Profibus protocol can be considered a younger and improved version of the same that extend the support to multiple master devices, however Modbus is more suitable for simple point-to-point communication. KNX, LonWorks and OPC, even if belonging to the category of the open protocols, still require fees [16] for the industrial use and thus are not taken into account in this survey. Finally, M-Bus (Meter-bus) is one of the most adopted standard for the remote reading of metering devices (eg. energy, gas).

Using the results of our survey (Table I) as hardware interface requirements and with the aim to target the wide class of BAS devices working with 24 V power supply, we developed a cheap and versatile board that can support all the previous mentioned interfaces. This interface adapter is intended to replace a previously existing point-to-point wired data connection by plugging the input/output signal wires of a device directly into the interface adapter, without requiring any specialized technical intervention.

The proposed architecture is presented in Fig. 1, where the white blocks are currently still under development. The board is based on the ESP32 by Espressif Systems, a cost-efficient ($\approx$ 2.5 €/kunit) and industrial environment compliant SoC that integrates a 32-bit microprocessor and both Wi-Fi and dual-mode Bluetooth BR/LE (Basic Data Rate and Low Energy).

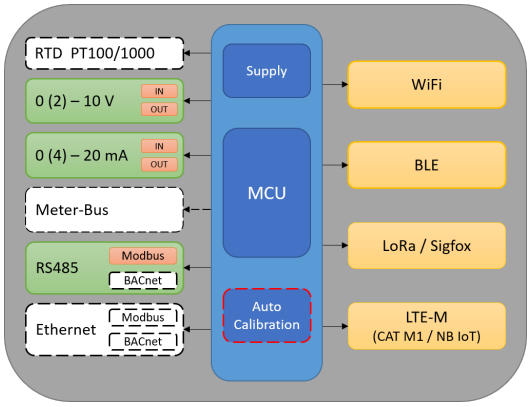

Fig. 1. Conceptual architecture of the proposed wireless adapter. Transducer (green) and network (yellow) interfaces

The high number of embedded peripheral interfaces, together with the large family of supported programming languages that range from the classical C\C++ to Java and Python, and the cryptographic hardware acceleration features make this module well-suited for implementing IoT solutions [17], [18].

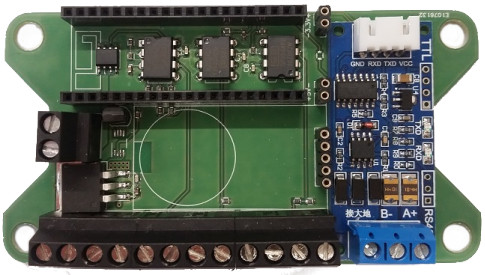

Fig. 2. Prototype v1: Maximum 36 V input power block (left), ESP32 board slot and analog front-end (top), RS485 module (right) and I/O terminal block (bottom). [Dimensions 106x56 mm]

In the investigated design, of which a first developed prototype is presented in Fig. 2, the ESP32 is not directly integrated in the board as standalone chip module, but rather embedded in the development boards produced by Pycom Ltd [19]. This solution is more expensive, but suits perfectly for investigation purposes, resulting extremely practical for testing different radio technologies according to the scenario. In fact, the manufacturer provides a wide variety of fully interchangeable development boards that, beside integrating Wi-Fi and Bluetooth as mentioned before, also extend the range of available LPWAN (Low Power Wide Area Network) radio technologies to both sub-GHz technologies like LoRa and Sigfox, and to cellular standards like LTE-M (Machine Type Communication) and Narrowband IoT (NB-IoT). In the former case mainly to exploit the deep signal penetration that low frequencies offer inside buildings and in the latter case to take advantage of higher bandwidth and connection density. Especially this latter case is worth to emphasize because these cellular technologies are based on the LTE 3GPP (3rd Generation Parternship Project) standard and as such 5G-

ready. This means that these modules will only need a software update to work in the new networks.

However, the ESP32 exhibits considerable limits when dealing with analog signals like those requested for voltage and current loop control. The built-in 18 channels (2 physical) 12-bit successive approximation (SAR) ADC and the 2 channels 8-bit DAC have very poor specifications. Although the manufacturer provides software tools for supporting the calibration in the form of API, still additional external circuitry is often needed. Depending on the desired accuracy, the additional cost resulting from the increased design complexity might become significant. Therefore, a clear analysis of the calibration requirements is an essential step for any further improvement based on cost-benefit trade-off analysis.

III. Performance Evaluation

In the following, the performance of the built-in ADC and DAC blocks of the ESP32 modules are examined separately for sake of clarity. As most of the BAS application involve slow varying signals, the tests are performed in static conditions.

The tests are conducted using a NI USB-6009 14-bit DAQ board by National Instruments and five Espressif ESP32-PICO-KIT V4 modules for compared statistical results. All the statistics provided in this Section are estimated over a set of 1000 samples.

A. Analog to Digital Converter

As first step, we measured the reference voltage V_{ref} of the internal 12-bit ADC. This is possible thanks to a useful feature of the ESP32 that allows to internally route V_{ref} to any GPIO pin. Hence, we can determine the correct value of the least significant bit size as $LSB = V_{ref}/2^{12}$.

Subsequently, for several known voltages, we acquired the corresponding binary code and used the correct LSB value to calculate the equivalent acquired voltage. Hence, the input/output transfer function for each examined ADC is obtained by linear regression ($\overline{R^2} = 0.999977$) of the acquired voltages with respect to the known ones. Measured values of V_{ref}, together with the ADC gain and offset, are summarized in Table II. For each measured parameter, the mean value and the relative uncertainty with 95% confidence interval is presented as well.

The results show that the internal voltage reference is affected by a significant variability. Considering a *t*-Student distribution with $\nu = N - 1$ degrees of freedom, the expanded uncertainty associated to V_{ref} is equal to 37.8 mV, more than the 3% of the full scale (FS). Similarly, the ADC transfer function exhibits a substantial negative offset with an average value of almost 80 mV, nearly the 6.8% of the FS range, and with a variation among modules that approaches the 19%. The examined modules also show a positive gain error, which is conversely affected by a very small variability within the 0.36%. Therefore, we conclude that the internal ADC is affected by significant errors and that a calibration is mandatory for reliable results.

TABLE II
ADC Parameters

ESP32	V_{ref} (V)	Gain (V/V)	Offset (mV)
#1	1.1614	1.1584	-83.71
#2	1.1439	1.1545	-84.37
#3	1.1641	1.1575	-77.73
#4	1.1306	1.1565	-71.95
#5	1.1504	1.1573	-75.10
mean	1.1501	1.1569	-78.57
[a]$U_{r_{\%}}$	3.29	0.36	19.08

[a]Coverage factor $k_4 = 2.77$

The knowledge of the correct value of V_{ref} would be sufficient to compensate the gain error. In fact, the residuals dispersion shows that the gain error can be largely considered as a systematic error after compensating for the correct LSB size. The further compensation of the offset error based on the available statistical information would lead to a residual offset error equal to the 1.3% of the FS. Unfortunately, a numerical compensation can reduce the final offset error but cannot recover signals smaller than the physical offset. If by contrast V_{ref} is unknown, then the ADC gain error would be linked to it by an inverse linear relationship, which is evident from a calculated Pearson coefficient $\rho_{(V_{ref},ADC_G)}$ equal to -0.9952 and a *p*-value of 0.04%, far below the test significance level of 5%.

B. Digital to Analog Converter

The performance of the two built-in 8-bit DAC is evaluated by conducting a similar analysis. The output voltage of each examined DAC is measured for a given set of binary codes and the input/output transfer function is obtained by linear regression ($\overline{R^2} = 0.999960$) of the measured voltages with respect to the expected ones. Measured values of the DAC gain and offset are summarized in Table III, together with the supply voltage V_{sup} provided by the on-board load regulator (LDO).

TABLE III
DAC Parameters

ESP32	V_{sup} (V)	Gain (V/V)		Offset (mV)	
		DAC_1	DAC_2	DAC_1	DAC_2
#1	3.3021	0.9476	0.9328	53.25	92.00
#2	3.3054	0.9402	0.9368	95.13	88.34
#3	3.2917	0.9370	0.9326	104.62	104.49
#4	3.3059	0.9418	0.9340	69.96	72.74
#5	3.2599	0.9294	0.9418	98.54	55.90
mean	3.2930	0.9392	0.9356	84.30	82.69
$U_{r_{\%}}$	1.63[a]	1.32[b]		52.03[b]	

[a,b]Coverage factors respectively $k_4 = 2.77$ and $k_9 = 2.26$

No information is provided by the manufacturer regarding the internal DAC architecture. However, the poor results in terms of observed gain uncertainty and output offset, together with the 8-bit resolution and the cheap cost of the module itself, suggests the use of a binary-weighted DAC architecture.

The gain error of the DAC transfer function exhibits an uncertainty slightly above 1%. In order to reduce this variability, the correlation $\rho_{(V_{sup},DAC_G)}$ between V_{sup} and the DAC transfer function gain was investigated as possible means of calibration. However, the resulting 49.7% p-value suggests a lack of evidences to reject the hypothesis of no correlation. The worst results emerge from the transfer function offset measurement: with an average value of almost 83 mV, the DAC offset is affected by considerable variations, with an expanded uncertainty that exceed 50%, thus preventing any kind of calibration performed on the basis of statistical observations. Therefore, the offset compensation can only be achieved through the implementation of a specific hardware solution, which would also help in recovering the lower 2.5% of the full scale output (FSO) otherwise not accessible.

Alternatively, the synthesis of analog signals can be more easily achieved by means of pulse-width modulation (PWM). The ESP32 has 16 PWM channels that can be configured independently for a different frequency and duty cycle resolution, provided that their product is less than the clock speed. The advantage of the PWM modulation resides in the extremely high linearity, impossible to achieve with cheap DACs, with the clear dependence of the maximum output voltage on the power supply, and with the ability to substantially swing the output between the positive and the negative rails.

TABLE IV
PWM PARAMETERS

ESP32	V_{sup} (V)	Gain (V/V)	Offset (mV)
#1	3.3021	0.9895	7.52
#2	3.3054	0.9890	7.14
#3	3.2917	0.9894	7.08
#4	3.3059	0.9894	7.06
#5	3.2599	0.9892	7.24
mean	3.2930	0.9893	7.20
[a]$U_{r_{\%}}$	1.63	0.05	7.17

[a]Coverage factor $k_4 = 2.77$

The performance of this technique was measured by setting the PWM frequency at 100 kHz, way above the normal bandwidth of signals used in BAS. The signal is then filtered with a simple RC low-pass filter with $\tau = 100s$, large enough to attenuate the modulating signal by 76 dB ($\approx$ 500 μV) and small enough to provide a bandwidth of 16 Hz. As for the case of the DAC, the output is acquired for different values of duty cycle, and the transfer function gain and offset are estimated by linear regression ($\overline{R^2} = 0.999972$) after compensating for the measured V_{sup}.

The results reported in Table IV show that a calibration of the analog signal synthesis by means of PWM is possible with the only knowledge of V_{sup}. In this case, the gain error shows a negligible variation among different modules, which on the contrary would match the uncertainty of V_{sup} if the latter is unknown. Similarly, an average transfer function offset of almost 7 mV is observed, approximately 0.2% of the FSO, low enough to not require any kind of compensation in most of the applications.

IV. DEVICE CALIBRATION

Tests have shown that accurate analog signal generation and measurement is possible using the ESP32 if the following arrangements are considered:

- *ADC* the transfer function offset is properly compensated and the internal ADC V_{ref} voltage is known
- *DAC* the transfer function offset is properly compensated and the internal LDO V_{sup} voltage is known
- *PWM* the internal LDO V_{sup} voltage is known

In order to achieve acceptable performance, we provided our board with a simple but effective auto-calibration procedure.

A. Offset compensation

The measured ADC and DAC offset reported in Section III, reduce the effective dynamic of the converters. As a result, the designed adapter is not able to fully swing the output voltage/current close to the negative rail and similarly is not able to read small signals. Several elegant architectures can be implemented, however our goal is to keep minimal the number and the cost of the adopted components.

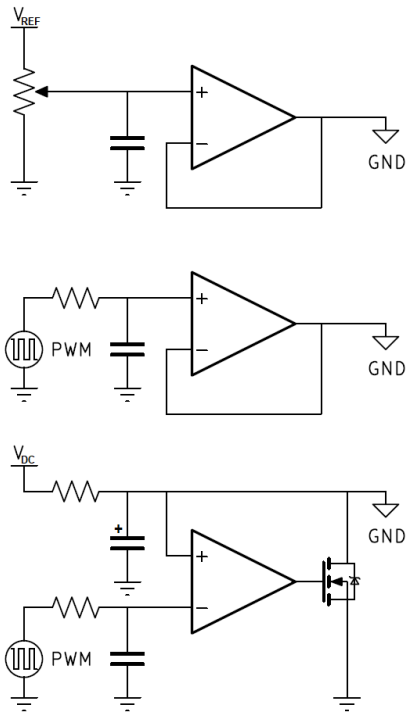

Fig. 3. Static (top), precision (center) and high-load (bottom) dynamic offset compensation scheme.

The simplest way of providing such a compensation is by using a level shifter. For this purpose, we identified two opamp-based solutions (Fig. 3) to shift the signal ground:

- *static* compensation based on the average offset value determined via statistical analysis
- *dynamic* compensation via closed-loop control for auto-calibration
  - *precision* can swing very close to the rail while providing a negligible noise
  - *high-load* can swing very close to the rail while sinking a significant current

The former solution is not applicable to the DAC because of its significant offset variability, whereas it can be used in principle to compensate the ADC negative offset, leading to a 1.3% residual offset uncertainty. However, this would require the use of 0.1% resistors, of a zero-drift amplifier and a known voltage reference. If the internal V_{ref} is used for this purpose, then the two uncertainties would combine in a 3.5% accuracy, since no correlation is found between the internal reference and the offset of the converters.

Conversely, the dynamic solution would permit to compensate the offset by varying linearly the PWM duty cycle. This would require only a generic amplifier with output swing capabilities close to the negative rail and no precision passive component. A linear regression of the ADC reading with respect to the duty cycle percentage value would easily provide the offset compensation transfer function

$$V_{OS} = m \cdot D_{\%} + n \tag{1}$$

where V_{OS} is the ADC offset and $D_{\%}$ is the PWM duty cycle. The correct duty cycle value to apply for an effective offset compensation is then found by solving equation (1) for a null output, as shown in a real case in Fig 4.

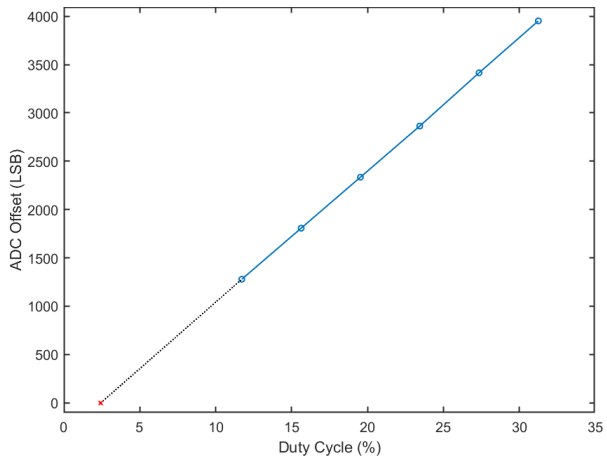

Fig. 4. Example of offset compensation transfer function (blue) and calculated dynamic compensation value (red).

Alternatively, the DAC output followed by a resistor divider can be used to generate the reference voltage for the dynamic compensation. This would result in a reduction of the voltage ripple on the reference and prevent eventual noise coupling of the PWM signal. However, the DAC regulation is characterized by lower resolution and generally higher non-linearity.

If the loop current sensing is carried out by reading the voltage generated on a shunt resistor, then the high-load configuration in Fig. 3 should be adopted. In fact, because of the non-negligible output impedance, even an opamp with driving capability within 1 mV from the negative rail would inevitably see this offset dramatically increase hundred times when the output current exceeds few milliamperes.

B. Gain compensation

Once the ADC offset has been compensated, a simple circuit like the one shown in Fig. 5 can be used to find the actual value of V_{ref} and V_{sup}.

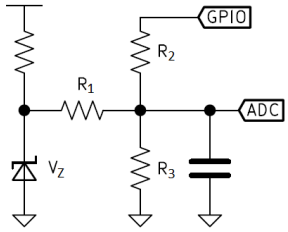

Fig. 5. Circuit for gain compensation: $R_1 = 1k\Omega$, $R_{2,3} = 820\Omega$

The cheap (≈ 0.25 €/kunit) LM4040CIM3X-2.0 voltage reference by Texas Instruments, provides a fixed voltage of 2.048 V with 0.5% accuracy. This voltage V_Z is then attenuated and combined with a GPIO pin in a voltage adder, composed of three 0.1% resistors (≈ 0.05 €/kunit). The calibration steps are summarized in the following:

1) GPIO is configured in Hi-Z mode and the voltage of the reference is split between R_1 and R_3
2) GPIO is configured as output with LOW state and the voltage of the reference is split between R_1 and $R_2 \parallel R_3$
3) The ADC gain is calculated by linear regression of the voltages acquired in the steps 1 and 2
4) GPIO is configured as PWM and the duty cycle is varied in the range 0-30%
5) The PWM gain is calculated by linear regression of the voltages acquired in step 4

This simple and cheap, but at the same time effective technique can be used to calibrate both the ADC and the PWM with a resulting accuracy that is strictly dependent on the accuracy of the selected voltage reference. For the proposed configuration, the final uncertainty of the gain is 0.52% at 25°C and 1.13% at 65°C.

V. Conclusion

The performance of a ESP32 as core unit of a wireless adapter for sensors and actuators in BAS has been analyzed. Despite of its low cost, this resourceful SoC is versatile, robust and integrates native network capabilities, potentially well-suited for industrial applications. However, the numerous built-in signal conversion blocks exhibit low accuracy, especially in

relation to the voltage of the internal regulators and to the offset of their characteristic.

Our statistical analysis highlights the accuracy limit of these devices by quantifying the variability of important parameters among different modules. Nevertheless, as shown by the authors, the obtained statistical information can be used for important design considerations. Furthermore, based on these considerations, simple measures can be implemented to achieve industrial-grade performance in analog signal applications, without increasing either cost or circuit complexity. These results are important for developing cheap but at the same time reliable industrial solutions based on SoC.

ACKNOWLEDGMENT

EUROPEAN UNION
Investing in our Future
European Regional
Development Fund

We thank the ERDF for their financial support.

REFERENCES

[1] M. Jung, C. Reinisch and W. Kastner, "Integrating Building Automation Systems and IPv6 in the Internet of Things," 2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, Palermo, 2012, pp. 683-688.

[2] M. Jain, N. Kaushik and K. Jayavel, "Building automation and energy control using IoT - Smart campus," 2017 2nd International Conference on Computing and Communications Technologies (ICCCT), Chennai, 2017, pp. 353-359.

[3] S. Spinsante et al., "A LoRa enabled building automation architecture based on MQTT," 2017 AEIT International Annual Conference, Cagliari, 2017, pp. 1-5.

[4] T. Supramaniam, R. Ibrahim, S. M. Hassan and K. Bingi, "Development of WirelessHART adapter with industrial transmitter for process monitoring," 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), Kuching, 2017, pp. 294-298.

[5] F. D. Arcari et al., "Development of a WirelessHART - EnOcean Adapter for Industrial Applications," 2017 VII Brazilian Symposium on Computing Systems Engineering (SBESC), Curitiba, 2017, pp. 181-186.

[6] B. Kang, D. Kim and H. Choo,"Internet of Everything: A Large-Scale Autonomic IoT Gateway," in IEEE Transactions on Multi-Scale Computing Systems, vol. 3, no. 3, pp. 206-214, 1 July-Sept. 2017.

[7] J. Huang and H. Hsieh, "Design of Gateway for Monitoring System in IoT Networks," 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, Beijing, 2013, pp. 1876-1880.

[8] VDMA-Forum Industrie 4.0 (2018), "Guideline Sensors for Industrie 4.0," [Online]. Available: https://industrie40.vdma.org/documents/4214230/25447567/Guideline+Sensors+for+Industrie+40.pdf

[9] J. Weiss, R. Yu, "Wireless Sensor Networking for the Industrial Internet of Things," Analog Devices White Paper WP007. [Online]. Available: https://www.analog.com/media/en/technical-documentation/white-papers/wp007fa.pdf

[10] Industrial Wireless, Phoenix Contact. [Online]. Available: https://www.phoenixcontact.com/assets/downloads_ed/global/web_dwl_promotion/52001683_EN_HQ_Industrial_wireless_LoRes.pdf

[11] Wireless Industrial Sensor Interface, BEI Sensors. [Online]. Available: https://www.mouser.com/pdfdocs/bei-sensors_wireless_industrial_sensors_interface_swiftcomm.pdf

[12] Cloud Connection Guide, Belimo Automation. [Online]. Available: https://www.belimo.com/mam/americas/pictures_and_graphics/marketing/teaser_images/iot/MAN18_Belimo_Connection_Guide_EN-US.pdf

[13] F. Wang, Z. Wang, K. Chang, J. Huang, L. Li, Q. Wang, "New Multifunctional Industrial Wireless Network Adapter," Elsevier Energy Procedia, Volume 118, 2017, Pages 95-103.

[14] M.H. Schraven, C. Guarnieri Calo' Carducci, M.A. Baranski, D. Müller, A. Monti, "Designing a Development Board for Research on IoT Applications in Building Automation Systems," 36th International Symposium on Automation and Robotics in Construction (ISARC 2019), Banff, Canada, 2019, in press.

[15] ESP32 Overview by Espressif Systems (Shanghai) Co., Ltd. [Online]. Available: https://www.espressif.com/en/products/hardware/esp32/overview

[16] Guide to Open Protocols in Building Automation, Schneider Electric, 2015. [Online]. Available: https://download.schneider-electric.com/files?p_enDocType=Brochure&p_File_Name=SE+Protocols+Guide.pdf

[17] A. Maier, A. Sharp and Y. Vagapov, "Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things," 2017 Internet Technologies and Applications (ITA), Wrexham, 2017, pp. 143-148.

[18] D. Ghosh, A. Agrawal, N. Prakash and P. Goyal, "Smart Saline Level Monitoring System Using ESP32 And MQTT-S," 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom), Ostrava, 2018, pp. 1-5.

[19] Hardware modules by Pycom Ltd. [Online]. Available: https://pycom.io/webshop/#dev