BULETINUL INSTITUTULUI POLITEHNIC DIN IAŞI
Publicat de
Universitatea Tehnică „Gheorghe Asachi” din Iaşi
Volumul 66 (70), Numărul 4, 2020
Secţia
ELECTROTEHNICĂ. ENERGETICĂ. ELECTRONICĂ

REVIEW ON MICROCONTROLLER PLATFORMS AND THEIR APPLICABILITY IN IoT EMBEDDED SYSTEMS

BY

IONUȚ-ADRIAN IFTODE[*] and CRISTIAN FOȘALĂU

"Gheorghe Asachi" Technical University of Iaşi
Faculty of Electrical Engineering

Received: October 12, 2020
Accepted for publication: December 15, 2020

Abstract. Since nowadays there is a large variety of embedded platforms available for technicians and researchers to use for their projects in the Internet of Things (IoT) domain, it is thought that it would be much useful to have a comparative synthesis in order to outline the most common microcontroller platforms features from a practical point of view to be used throughout this field of research. Therefore, a comparison among different microcontroller platforms based on a set of parameters and criteria such as: data acquisition and control capabilities, processing power, power management, flashing process, connectivity interfaces and cyber security level is provided in the paper. The microcontroller embedded platforms taken into consideration in this study are the most commonly used ones in IoT applications: MSP430G2x MCU from Texas Instruments, STM32F103 MCU from STMicroelectronics and PIC32MZ MCU from Microchip Technology. The comparison may be very useful for the practitioners who are in situation to choose the type of device needed to be employed as computing, command and control unit according to the criteria imposed by the application in order to optimize the performance-to-cost ratio.

Keywords: embedded microcontroller platforms; Internet of Things; MSP430; STM32; PIC32.

[*]Corresponding author; *e-mail*: iftodeionutadrian@gmail.com

1. Introduction

The 21st century technological revolution has provided a heavy advancement on the design of an embedded system (Marwedel, 2017), considering the global smart sensor market rapid growth and the expansion of the Internet of Things (IoT) concept. IoT provides a vision of a hyper-connected world in which almost all physical objects and people can be seamlessly interconnected and can exchange data and use artificial intelligence to make insightful decisions that individuals and society benefit. Technological advancements in the electronics, telecommunications and software industries have laid the foundation for the launch of IoT. Expanded wireless connections, smaller, faster, more power-efficient electronic components and more powerful software are driving the highly competitive IoT device market. Developers must understand the technical building blocks provided to them and make the best choice to ensure technical and commercial success, which is crucial.

Other considerations such as the large-scale deployment of IoT-like devices, the ability of certain devices to automatically connect to other devices and the possibility of deploying these devices in unsafe environments have exacerbated this challenge (Daniel *et al*., 2017). In principle, developers, and users of IoT devices and systems have a collective obligation to ensure that they do not expose users and the Internet itself to potential harm. Therefore, a security collaborative approach will be needed to develop effective and appropriate solutions to meet the IoT security challenges, which is well suited to the scale and complexity of the problem (Zhi-Kai *et al*., 2014). Security must be rooted in the architecture of the complete solution and the threats must be identified in order to correctly design and implement the countermeasures in synergy with other security features.

IoT devices often have demanding requirements for high-performance and low-power microcontroller units (MCU) that provide the highest level of integration with available peripherals and software layers. The IoT devices integrate a variety of embedded devices which aim to fulfil certain goals such as: health and telemedicine (Ray *et al.*, 2019 ; Gope *et al.*, 2020; Venkata Virajit Garbhapu, 2017), home solar panel management (López-Vargas *et al.*, 2019), big-data management within smart buildings (Plageras *et al.*, 2018), weather station (Sahu, 2016) and so on.

This paper presents a comparative analysis on three MCU families: MSP430G2x, STM32F103Cx and PIC32MZ manufactured by Texas Instruments, STMicroelectronics, and Microchip Technology, respectively. The analysis proposed in this paper aims to create an overview regarding the MCU's capabilities on which researchers and engineers need to take into account within their work.

The article is divided in nine sections, as follows: introduction, current situation and theoretical overview are being described within sections 1 and 2, whereas a wide description based on a set of MCU parameters is presented in sections 3 to 8. Finally, section 9 summarizes the conclusions of the paper.

2. Embedded Systems

The embedded systems are designed to perform special functions in a larger mechanical or electrical system (Oyetoke, 2015). These systems are typically designed to perform a single or multiple repeated functions. However, regardless of the function involved, they are rarely required to do something more than the assigned task.

Taking into consideration these facts, the embedded system architecture contains several main components that are bundled together to fulfil its purpose (Ti-Yen Yen, 1995). Fig. 1 describes the working principle of such a system, starting with the continuous acquisition and processing of input parameters by using the microcontroller, based on software and hardware sub-components. After acquiring, processing, and managing the data within the input parameters, the microcontroller triggers certain actions integrated as output parameters.

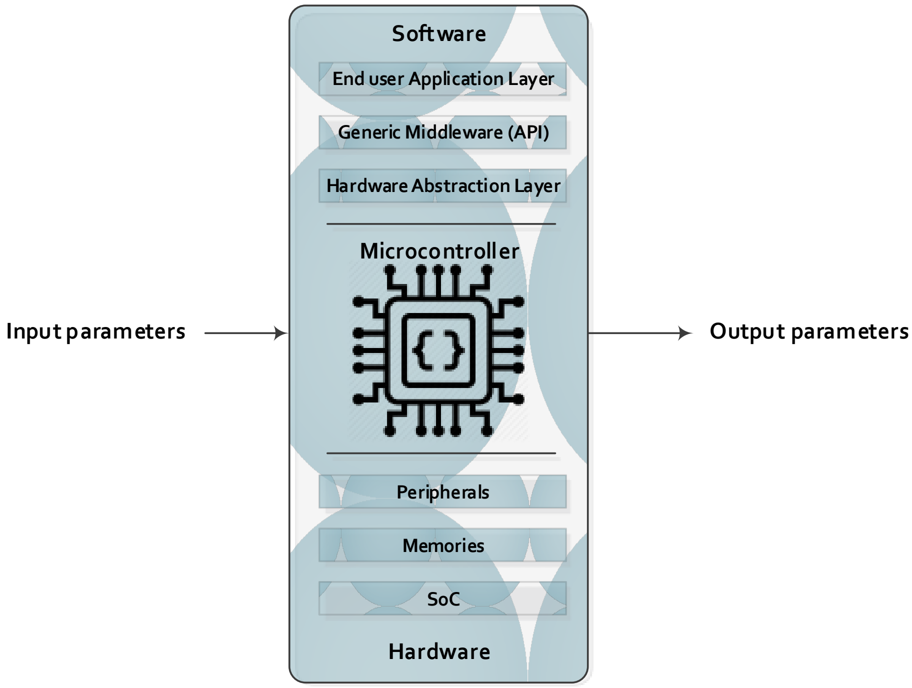

Fig. 1 – Embedded systems workflow.

In this paper, we have made a brief comparison of three microcontroller series (Texas Instruments, 2020; STMicroelectronics, 2020; Microchip Technology, 2020), which engineers, researchers or enthusiastic people can choose according to their knowledge level and application type. Table 1 describes the aforementioned microcontroller platforms, which can be integrated into an embedded system considering a set of criteria such as: data acquisition and control, processing power, power management, communication interfaces, security level and price.

Table 1

Microcontroller Comparison Based on Specified Parameters

Parameters	Characteristics	MSP430G2x	STM32F103Cx	PIC32MZ
Data acquisition and control	*I/O pins*	*Up to 24*	*Up to 37*	*Up to 120*
	ADC	*10-bit SAR*	*12-bit SAR*	*12-bit SAR*
	DMA	*x*	*x*	*x*
	Timer resolution	*16-bit*	*16-bit*	*16-bit & 32-bit*
Processing power	*CPU Clock (MHz)*	*16*	*72*	*200*
	Flash Memory	*Up to 16kB*	*Up to 128kB*	*Up to 2048kB*
	IDE	*IAR, CCS, Keil uVision, Energia*	*STM32CubeIDE, STM32CubeMx, Keil uVision*	*MPLAB-X*
Power Management	*Operating voltage*	*2.5 – 5.5 V*	*2.0 – 3.6 V*	*2.5 – 5.5 V*
	Typical current consumption (mA)	*Up to 5mA*	*Up to 50mA*	*Up to 120 mA*
Flashing process	*Debug & program interface*	*JTAG*	*Serial wire (SWD) & JTAG*	*JTAG*
	SPI	*x*	*x*	*x*
	I2C	*x*	*x*	*x*
	CAN	-	*x*	*x*
	UART	*x*	*x*	*x*
	USART	-	-	-
	USB	-	*x*	*x*
Cyber level	*Cryptographic algorithms*	*x*	*x*	*x*
	Software IP protection	*x*	*x*	*x*
	Physical security	*x*	*x*	*x*
Others	*Temperature operation*	*-40℃ to +85℃*	*-40℃ to +85℃*	*-40℃ to +85℃*
	Cost (1x qty)	*~3$*	*~5$*	*~7$*
	*MCU Packages type ****	*TSSOP, PDIP,QFN*	*VFQFPN, LQFP*	*QFN, TQFP, LQFP*

3. Data Acquisition and Control

Data acquisition (DAQ) represents the process in which analog information (as samples) are collected at fixed time intervals (data sampling rate) and further transmitted in digital format (Maurizio Di Paolo, 2015). DAQ may include signal conditioning (used to process and scale raw sensor readings), and analog-to-digital converters, which convert analog sensor readings into digital values so that they can be processed and analyzed in a digital computing unit. As stated in the above section, the sensors are represented by input parameters to the embedded system architecture.

Fig. 2 describes an overall comparison between the selected microcontrollers based on a set of parameters related to data acquisition and control. The general-purpose inputs/outputs (GPIO) parameter offers a wide view on the MCU`s capabilities related to software pin configuration as input (with or without pull-up or pull-down), as output (push-pull or open-drain) or alternative function. As it can be seen from the graph presented in Fig. 2, the STM32F103x MCU has a larger GPIO pin number (up to 37) compared with MSP430G2x with up to 24 and PIC31MZ with only up to 16 GPIO pins. Both analog channels and analog-to-digital converter (ADC) resolution parameters are related to the analog to digital converting capabilities. All the three MCU analog channels are based on successive approximation register (SAR) ADC type with 10-bit resolution (MSP430G2x) and 12-bit resolution (STM32F103Cx, PIC32MZ). All the three MCUs have a unique number of analog channels from a more capable one up to 48 channels (PIC32MZ) to medium capable ones up to 8 and 10 channels (MSP430G2x, STM32F103x). The timing and controlling features are characterized within MCU by the number of timers and timer resolution.

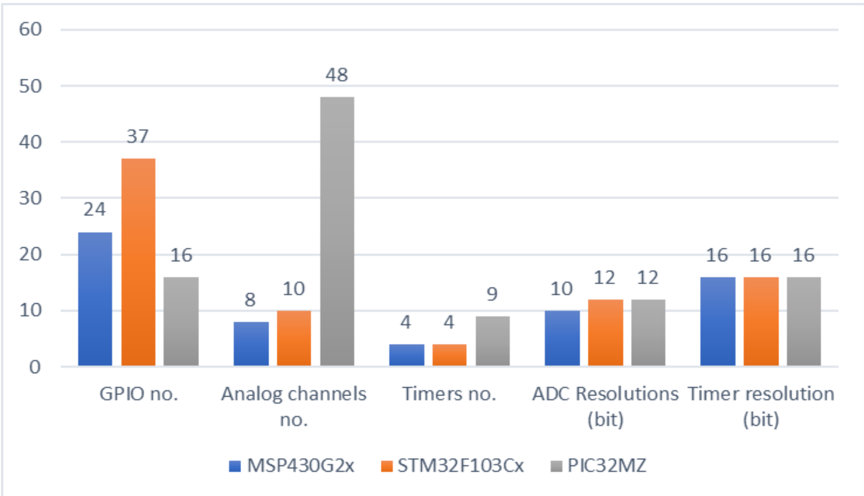

Fig. 2 – DAQ and control parameters.

In general, the timers are capable of handling input and output captures, PWM generation and interrupts. Within the selected MCUs for this paper, there is a similarity related to the timer resolution which is 16-bit and a difference related to the number of timers from a more capable MCU having up to 9 timers (PIC32MZ) to low-medium capable MCUs having up to 4 timers (STM32F103Cx, MSP430G2x). In conclusion of this criterion, if the applications require a larger number of analog quantities to be measured, PIC32MZ would be more appropriate to be employed, while if the application is based on many sensors communicating by interfaces like SPI, I2C, UART, etc., STM32F103Cx is the best. On the other hand, whether timing is the main quantity governing the application, PIC32MZ is also the best candidate.

4. Processing Power

Embedded devices require specific data processing, achieving data manipulation, transmission, and analysis. Some embedded devices are integrated into IoT and can directly process data, while other IoT devices transmit the data to other devices, gateways or cloud applications for further manipulation and analysis. Edge analysis is the type of data analysis that is performed at the edge of the network rather than at a centralized location. In this way, the data can be analysed in real time on the embedded device itself or on a nearby gateway (such as a router) connected to the IoT, instead of transmitting a large amount of data upstream to a cloud or database server. Edge analysis reduces upstream processing while storage requirements reduce the network load.

The processing power needed by the embedded applications depends on the data processing required by the application that uses that data. Available memory, processor specifications (word length) and clock speed parameters mainly determine the data processing rate of the device.

The capacity of non-volatile flash memory (used to store data persistently until transmission upstream) determines how much data can be stored on the device and the size of the user application. Considering the comparison scenario that we have selected, the PIC32MZ has a 2MB flash memory to store programs and data, while the other two MCUs, viz. STM32F103Cx, MSP430G2x, have a flash memory capacity comprised between 64 and 128 Kbytes. Considering the CPU architecture, the PIC32MZ is different than the other two MCUs since it has a 252 MHz clock speed with a floating-point unit 32-bit RISC processor, compared with the STM32F103Cx, which has 72 MHz ARM Cortex M3 32-bit RISC processor and with the MSP430G2x, which is endowed with a 16 MHz – 16-bit RISC processor (Fig. 3). Taking into account the programming capabilities of each MCU, it can be seen that the MCUs provided by STMicroelectronics and Texas Instruments can be programmed, debugged and uploaded by using different IDEs, compared with Microchip MCU which is programmed only in MPLAB-X.

Fig. 3 – Processing power parameters.

5. Power Management

The MCU power management is characterized by a wide variety of parameters, but this paper is more focused on the operating voltage and power consumption during different MCU operating modes. The MCU operating voltage determines the voltage level of the application supply and I/O. As it can be seen from Fig. 4, the minim operating voltage can vary from 1.8 V on MSP430G2x to 2.1V on PIC32MZ, whilst the maximum operating voltage on all three study cases is 3.6V.

Taking into account that the MCU platforms integrated within the embedded systems can have a different power consumption during their lifetime, being directly dependent on the user application, a set of main power consumption operating modes can be defined, such as: sleep power mode, standby (typical) power mode, peripheral power mode and data logging power mode. Fig. 4 describes the typical power consumption of the 3 MCUs. As it can be noticed from the figure, the MSP430G2x ultra-lower power MCU has consumption of about 100 µA/MHz, while the STM32F103 and PIC32MZ MCUs need between 50 and 100 mA for their operation. It is known, however, that the microcontrollers of MSP430 family are characterized by low current consumption, even in standby mode (less than 1 µA), being appropriate to be employed in portable applications or to be embedded in IoT nodes working in the field, where no power network is available, and the device supplying must be assured from different harvesting solutions.

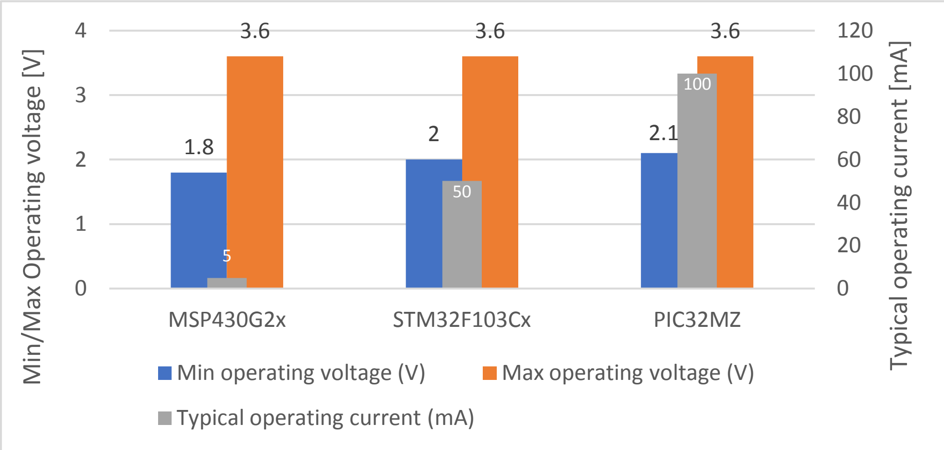

Fig. 4 – Power management parameters.

6. Communication Interfaces

On one hand, the communication interfaces are used for inter-system communication like between PC and the embedded system (MCU). On the other hand, they are used for intra system communication between MCU and other integrated circuits. Table 2 presents a set of parameters of the main inter system communication interface. As it can be observed in the table, the UART and USB are working based on a synchronous communication compared with the USART, which is working on an asynchronous mode.

Table 2
Inter Communication Interfaces

Interface	Sync Type	Typical Speed	Max Connected Devices	Communication type	Wiring
UART	*Sync*	*up to 1 Mbit/s*	*2*	*Half/Full-Duplex*	*1*
USART	*Async*	*up to 1 Mbit/s*	*2*	*Full-Duplex*	*1*
USB	*Sync*	*20-90 Mbit/s*	*Up to 128*	*Full-Duplex*	*2*

Taking into consideration the communication types, the USART and USB are working on a full-duplex communication type compared with the UART which is a half-duplex-based communication type. From the speed point of view, the USB protocol is superior to the UART and USART interfaces since it can speed up to 90 Mbit/s (depending on USB version and MCU compatibility, the speed can go even higher) compared with up to maximum 1 Mbit/s. From the maximum number of devices and wiring perspective, the UART & USART communication interfaces are used to interact with maximum 2 devices on one single wire compared with the USB, which can be used to interact with multiple devices (up to 128) on two differential wires. Table 3 describes the set of parameters of the intra system communication interfaces. As it can be seen, all the three communication interfaces are synchronized and can drive several devices on either two wires based (I2C and CAN) or 4 wires (SPI). From the communication type point of view I2C is different than SPI and CAN, since it is based on a half-duplex protocol mode compared with SPI and CAN, which are characterized on full-duplex protocol mode. On the other hand, I2C and CAN are multi-master communication interfaces compared with SPI which is a single-master protocol.

Table 3
Intra Communication Interfaces

Parameters	**Sync Type**	**Typical Speed**	**Max Devices**	**Communication Type**	**Wiring**
I2C	*Sync*	*up to 3.4Mbit/s*	*up to 128*	*Half-Duplex*	*2*
SPI	*Sync*	*up to 12MBit/s*	*Many*	*Full-Duplex*	*4*
CAN	*Sync*	*up to 8MBit/s*	*Many*	*Full-Duplex*	*2*

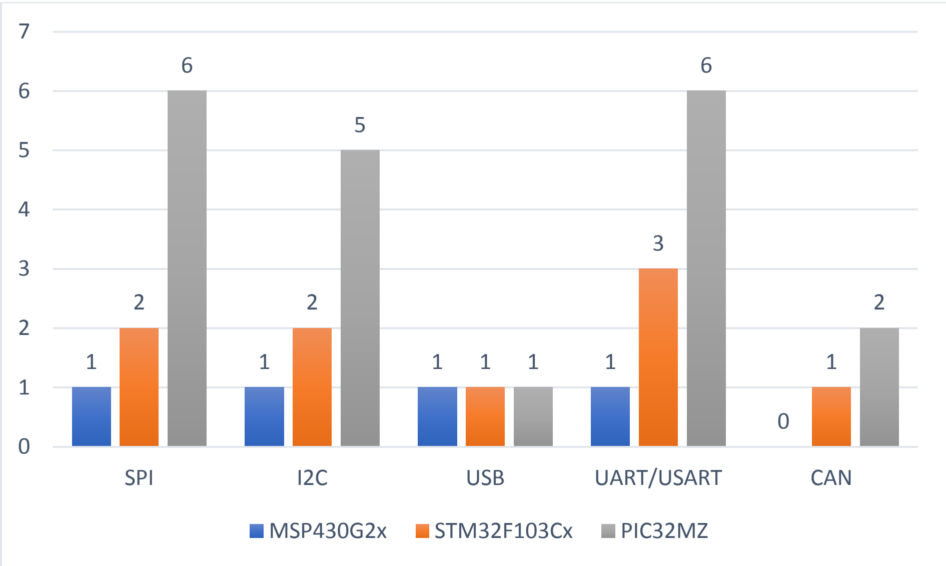

Fig. 5 – Communication interfaces capabilities.

Fig. 5 describes an overall comparison between the three selected MCUs from the communication interfaces capabilities point of view. As it can be observed from the figure, the MSP430G2x has lower capabilities compared with the STM32F103Cx, which has a medium capability. PIC32MZ has the highest capability regarding the communication interfaces.

7. Security Level

Security is a key element on IoT/embedded systems. It must be considered at all stages of design and development, even during prototyping. The integrity and security of the data acquired by any device must theoretically remain unchanged. Security requirements apply to the embedded system device itself, to the network it is included in, and cloud, mobile and web service applications. Related safety requirements include:

• Ensure that each device has enough processing power and memory to be able to encrypt and decrypt data and messages at the rate at which they are sent and received.

• Ensure that the embedded software development library supports any authorization and access control mechanisms used to verify upstream services and applications.

• Choose to use off-the-shelf devices that implement device management protocols to safely register new devices when they are added to the network to prevent spoofing.

• Choose devices that have firmware functions to support secure firmware updates by using over the air method.

All the three MCUs analysed in this paper use security key algorithms (cryptography algorithms) as security measure since it ensures confidentiality, integrity and authentication of both data and user application code.

Regarding the on-site firmware and software updates, all three MCUs can provide a secure firmware and software update based on secure bootloader and readout protection. While the bootloader is protecting the IP, assets and from remote attacks the readout protection it refers to a global flash memory protection allowing the embedded firmware code to be protected against copy, reverse engineering and dumping. Another important security level is given by the software IP encapsulation, which all selected MCUs can provide.

8. Price Level

Pricing level of microcontrollers is crucial considering that the current integrated circuit market is growing and the number of applications that need them is drastically expanding and increasing. Looking towards to the three selected MCUs as per 2021 price, the MSP430G2x is rated with a lower price per unit (~3$) compared with STM32F103Cx (~5$) and with PIC32MZ having a 7$ price per unit reflecting the high processing power and the increased number of I/O pins. The price level decreases by increasing the quantity amount.

9. Conclusions

The paper was intended to systematize the main features encountered in designing and deploying IoT application based on three commonly used commercially available embedded MCU platforms made by the well-known manufacturers: Texas Instruments, STMicroelectronics, and Microchip Technology. The criteria considered in the study were: data acquisition and control capabilities, processing power, power management, communication interfaces, security level and price. According to our study, if the application you are to develop employs a bigger number of analog quantities or is critical time dependent, using a PIC32MZ would be a good solution, whereas if the communication is performed mainly through the digital ports as well as by the serial interfaces, the STM32F103x MCU would be a good choice. Both MCUs offer a good help if the application requires a good resolution for ADC conversion. PIC32MZ provides, concurrently, the highest processing power at the cost, however, of the highest price of the three units analysed. On the other hand, if your application needs to operate under harsh supplying conditions, the MSP430G2x MCU is the best candidate to be employed, this feature being corroborated with the cheapest price.

In conclusion, choosing of a certain MCU platform for the design of IoT nodes depends usually on the application, but an optimization of the main target that is performance-to-cost ratio is advised to be performed using the results of this study.

REFERENCES

Daniel M., Kazem S., Jacob, *KIoT Security (IoTSec) Considerations, Requirements, and architectures*, 14th IEEE Annual Consumer Communications & Networking Conference (CCNC), 2017, 10006-10007.

Gope P., Gheraibia Y., Kabir S., Sikdar B., *A Secure IoT-based Modern Healthcare System with Fault-tolerant Decision Making Process*, IEEE Journal of Biomedical and Health Informatics, 2020, 1-1.

López-Vargas A., Fuentes M., Vivar M., *IoT Application for Real-Time Monitoring of Solar Home Systems Based on Arduino™ With 3G Connectivity*, IEEE Sensors Journal, 2017, 19, 679-691.

Marwedel P., *Embedded System Design* (3rd Ed.), Springer International Publishing AG., 2017.

Maurizio Di Paolo E., *Embedded Systems Design for High-Speed Data Acquisition and Control*, Springer, 2015.

Oyetoke O.O., *Embedded Systems Engineering, the Future of Our Technology World; A Look Into the Design of Optimized Energy Metering Devices*, International Journal of Recent Engineering Science, 2015.

Plageras A.P., Psannis K.E., Stergiou C., Wang H., Gupta B.B., *Efficient IoT-Based Sensor BIG Data Collection–Processing and Analysis in Smart Buildings*, Future Generation Computer Systems, 2018, 82, 349-357.

Ray P.P., Dinesh D., Debashis D., *Internet of Things-Based Real-Time Model Study on e-Healthcare: Device, Message Service and Dew Computing*, Computer Networks, 2019, 149, 226-239.

Sahu R.K., *An IoT Based Weather Information Prototype Using WeMos*, International Conference on Contemporary Computing and Informatics, 2016, 612-616.

Ti-Yen Yen, *Hardware-Software Co-Synthesis of Distributed Embedded Systems.* Springer, 1995.

Venkata Virajit Garbhapu S.G., *IoT Based Low Cost Single Sensor Node Remote Health Monitoring System*, *Procedia Computer Science*, 2017, 113, 408-415.

Zhi-Kai Z., Michael Cheng Yi C., Chia-Wei W., Chia-Wei H., Chong-Kuan C., Shiuhpyng S., *IoT Security: Ongoing Challenges and Research Opportunities*, IEEE 7th International Conference on Service-Oriented Computing and Applications, 2014, 230-234.

*** Microchip Technology, *PIC32 Family Reference Manual.* Retrieved from www://www.microchip.com, 2020.

*** STMicroelectronics, *STM32f103xx Reference Manual*, https://www.st.com, 2020.

*** Texas Instruments, *MSP430™ MCUs Development Guide Book,* https://www.ti.com, 2020.

O SINTEZĂ COMPARATIVĂ ASUPRA PLATFORMELOR CU MICROCONTROLER ȘI A APLICABILITĂȚII LOR ÎN SISTEME DE TIP IoT

(Rezumat)

Intrucât în prezent există disponibilă pe piață o largă varietate de platforme de dezvoltare cu microcontroler pe care proiectanții și dezvoltatorii de sisteme Internet of Things (IoT) le utilizează, existența unui studiu comparativ asupra acestor platforme care ar putea ajuta la alegerea judicioasă a acestora din punctul de vedere al unor criterii specifice aplicației ar fi binevenită. Lucrarea de față prezintă un astfel de studiu asupra a 3 tipuri de platforme fabricate de cei mai renumiți producători de pe piață la ora actuală și anume: MSP430G2x MCU de la Texas Instruments, STM32F103 MCU de la STMicroelectronics și PIC32MZ MCU de la Microchip Technology. Parametrii și criteriile de selecție avute în vedere sunt următoarele: capabilitățile de achiziție și control al semnalelor de intrare, puterea de prelucrare a datelor, consumul, capabilitățile de comunicare și nivelul de securitate al datelor. Rezultatele acestui studiu pot fi utilizate de tehnicienii și cercetătorii interesați de alegerea unei platforme potrivite pentru dezvoltarea sistemelor de tip IoT în scopul maximizării raportului performanță/cost al aplicației.