© 2018 IJSRSET | Volume 4 | Issue 7 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section : Engineering and Technology

A Comparative Study of Arduino Uno, NodeMCU in IoT

P. Ashwini, Pavan. S, Roja Ramani A

Assistant Professor, Department of CSE, Teegala Krishna Reddy Engineering College, Hyderabad, Telangana, India

ABSTRACT

IOT Hardware components can include low-power boards; single-board processors like the Arduino Uno and Node MCU. An Arduino Uno is a microcontroller motherboard. A microcontroller is a simple computer that can run one program at a time, over and over again and it is very easy to use. But the structure of Arduino is its disadvantage, the easy to use hardware/software of Arduino unable a person to learn the basics of many things likes Serial communication, ADC, I2C and Arduino libraries are not very efficient in certain parts and waste RAM and CPU cycles. NodeMCU is a development board featuring the popular ESP8266 WiFi chip. Its obvious advantage over the Arduino is that it can readily connect to the Internet via WiFi. However, the ESP8266 breakout board has limited pins although the chip itself has a lot of output ports. The NodeMCU solves this problem by featuring 10 GPIO pins each capable of using PWM, I2C and 1-wire interface.This paper provides Internet of Things oriented comparison of Arduino Uno and Node MCU boards with suitable selection of the hardware development platforms that are capable enough to improve the understanding of technology and also summarizes various capabilities of available hardware development platforms for IoT.

Keywords : IOT, Arduino Uno, NodeMCU, ESP8266, Microcontroller.

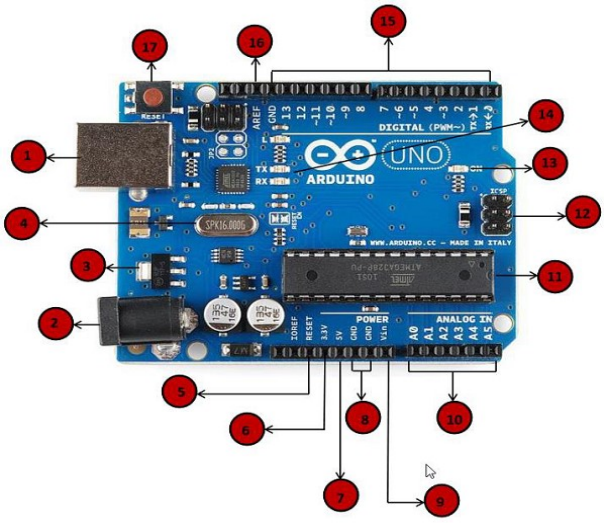
I. INTRODUCTION

IOT is growing rapidly and our machines can interact with each other and act accordingly. Internet of Things (IoT) devices enables formerly unimaginable levels of remote monitoring and control. The building blocks of an IoT device are remarkably similar[1]. Here are the three main components-Sensors, Microcontrollers, Transmission for a wireless sensor node. Sensors – gather information about the environment and condition signals before transmitting to the microprocessor. Sensors and actuators connect to the microcontroller through digital or analog *General Purpose Input/output* (GPIO) pins or through a hardware bus. Standard communication protocols like I2C and SPI are used for intra-device communication with the components that are connected with the bus[2]. Adopting standards makes it easier to add or swap out components that are connected with the bus. **Microcontrollers** – process the signal from sensors, determine appropriate responses, and manage power consumption and local memory. Microcontrollers contain a processor core (or cores), memory (RAM), and *erasable programmable read-only memory* (EPROM) for storing the custom programs that run on the microcontroller. *Microcontroller development boards* are PCBs with additional circuitry to support the microcontroller to make it more convenient to prototype with and program the chip. A *microcontroller* is a SoC that provides data processing and storage capabilities. The components in an IoT node will vary in sophistication, depending on the application. But the basic topology of a wireless sensor node always includes these elements[3].

Transmission –wireless chips, radio modules and protocols needed to transmit the information between devices and to the cloud[12].

Arduino

The Arduino is an incredibly flexible micro-controller and development environment that cannot only be used to control devices, but can also be used to read data from all kinds of sensors. Its simplicity and extensibility, in addition to its great success and adoption by users, has led to the development of a variety of hardware extensions and software libraries that enable wired and wireless communication with the Internet. Power consumption is extremely low, which is a result of their CPUs usually running at 8 to 16 MHz[4][11]. This makes them an ideal solution for places where there's no way to get external power: with aggressive enough power saving, an Arduino board can run for months on several AA batteries. Arduino is the ideal open hardware platform for experimenting of the Internet of Things and incredibly popular hardware/software platform for creating interactive IoT objects and devices. **Arduino platform**, which includes a physical board processor, shields with individual libraries of C code, and an integrated development environment (IDE) for writing, compiling, and uploading code, can also be used to read data from all kinds of sensors.field-programmable gate arrays (FPGA)[5][6].



1. **Power USB:**Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).

2. **Power (Barrel Jack):**Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).

3. **Voltage Regulator:**The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

4. **Crystal Oscillator:**The crystal oscillator helps Arduino in dealing with time issues.

5. **Arduino Reset:**You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).

6. **Pins (3.3, 5, GND, Vin):**

- 3.3V (6) – Supply 3.3 output volt
- 5V (7) – Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- GND (8)(Ground) – There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- Vin (9) – This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

7. **Analog pins:**The Arduino UNO board has five analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor

8. Each Arduino board has its own microcontroller (11):

9. **ICSP pin:**Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus.

10. **Power LED indicator:**This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.

11. **TX and RX LEDs:**TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board. RX flashes during the receiving process.

12. **Digital I/O:** The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled "~" can be used to generate PWM.

13. **AREF:** AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

Disadvantages:

1. The structure of Arduino is its disadvantage.

2. The easy to use hardware/software of Arduino unable a person to learn the basics of many things likes Serial communication, ADC, I2C etc.

3. The Arduino libraries are not very efficient in certain parts and waste RAM and CPU cycles.

4. No integrated support for WIFI network.

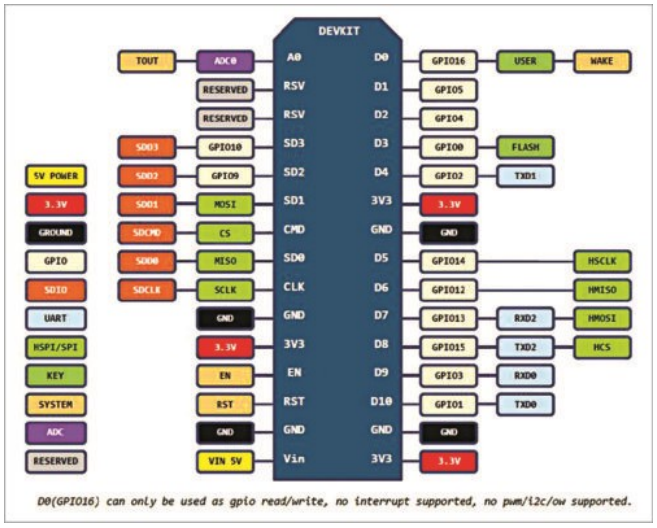
5. Energy consumption is high.

NodeMCU:

The NodeMCU (Node MicroController Unit) is an open source software and hardware development environment for IOT that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains all crucial elements of the modern computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK [7].It uses the Lua scripting language. It is based on the eLua project, and built on the ESP8266 SDK 1.4. NodeMCU is a development board featuring the popular ESP8266 WiFi chip. Its obvious advantage over the Arduino or PIC is that it can readily connect to the Internet via WiFi[8]. However, the ESP8266 breakout board has limited pins although the chip itself has a lot of output ports. The NodeMCU solves this problem by featuring 10 GPIO pins each capable of using PWM, I2C and 1-wire interface. NodeMCU comes with 128KB RAM and UNO it's just 2kB RAM so NodeMCU is having more ram space.ESP8266 is a low-cost, WiFi Module chip that can be configured to connect to the Internet for Internet of Things (IoT)[9]. **NodeMCU is a Firmware on ESP8266**. It is basically an SoC (System on Chip).System on Chip (SoC) is an integrated circuit that integrates all components of a computer or other electronic systems. Programs for the NodeMCU are written in Lua, which is an interpreted programming language similar to Python and Ruby. In many respects, it's probably easier to learn than the variant of C used by the Arduino. It has a *much* simpler syntax[10][13].

Advantages of NodeMCU platform relative to the Arduino

- Low cost
- Integrated support for WIFI network
- Reduced size of the board
- Low energy consumption



Comparison between Arduino and NodeMCU

1. NodeMCU is having 4MBytes of ROM (flash) and Arduino UNO is just 32 KB,

2. NodeMCU can store more code compare to Arduino UNO.

3. NodeMCU comes with micro USB port and Arduino UNO is comes with USB type B connector, micro USB cable is easy available compare to USB type B.

4. NodeMCU development board is smaller in size compared to Arduino UNO.

5. NodeMCU and Arduino UNO priced almost same, so you can opt to buy NodeMCU.

Parameters	Arduino Uno	NodeMCU
Processor	ATMega328P	ESP8266
Operating voltage	5V	5V
Clock speed	16 MHz	80 MHz – 160 MHz
System memory	2kB	128kB
Flash memory	32 kB	4MB
EEPROM	1 kB	-
Communication supported	IEEE 802.11 b/g/n IEEE 802.15.4 433RF BLE 4.0 via Shield	IEEE 802.11 b/g/n
Development environments	Arduino IDE	Arduino IDE, Lua Loader
Programming language	Wiring	Wiring, C, C++
I/O Connectivity	SPI I2C UART GPIO	UART, GPIO

II. CONCLUSION

The comparative study of Arduino and NodeMCU shows how these platforms are promoting the growth of IoT by utilizing the specific board as per the intended application. The detailed analysis show that NodeMCU have higher performance in comparison with Arduino in terms of its storage and computing speeds. The Arduino libraries are not very efficient in certain parts and waste RAM and CPU cycles and there is no integrated support for WIFI network.Where as NodeMCU equipped with inbuilt wifi to connect to internet and store the data to the cloud servers if required for further processing.NodeMCU is a low cost device is a first choice for implementing sensor networks in an IoT scenario.

III. REFERENCES

[1]. Ray, P. P. "A survey on Internet of Things architectures." Journal of King Saud University-Computer and Information Sciences (2016).

[2]. https://www.farnell.com/datasheets/1682209.pdf

[3]. https://www.raspberrypi.org/documentation/hardware/computemodule/RPI-CM-DATASHEET-V1_0.pdf

[4]. http://espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf

[5]. Waher, Peter, et al.
[6]. Maksimović, Mirjana, et al. "Raspberry Pi as Internet of things hardware: performances and constraints." design issues 3 (2014)
[7]. Di Nisio, A., et al. "Design of a low cost multipurpose wireless sensor network." Measurements & Networking (M&N), 2015 IEEE International Workshop on. IEEE, 2015.
[8]. Morais, R., Valente, A., Couto, C., Correia, J. H., "A Wireless RF CMOS Mixed-Signal Interface for Soil Moisture Measurements", Sensors and Actuators A (Physical), Elsevier, volume 115, pp: 376-384, 2004.
[9]. Lei Yuan, Xiong construction, Zhao Xiaohui. "Wi-Fi-based wireless sensor network design and research"Modern electronic technology, volume 18, pp.192-197, 2009.
[10]. Garrity, D. P.,Akinnifesi, F. K., Ajayi, O. C, Evergreen Agriculture: a robust approach to sustainable food security in Africa. Food Security, volume 2, pp. 197- 214, 2010
[11]. J. Domingues, A..Damaso, N. Rosa, WISeMid: Middleware for IntegratingWireless Sensor Networks and the Internet, Distributed Applicationsand Interoperable Systems Lecture Notes in Computer ScienceVolume 6115, pp 70-83, 2010.
[12]. Wu Hao, "Internet of Things wireless mobile communication andapplication analysis", Computer Knowledge and Technology, Volume 6, pp.5205-5206, 2010.
[13]. R. Roman, P. Najera, and J. Lopez, "Securing the Internet of Things," IEEE Computer, volume 44, pp. 51- 58, 2011.

AUTHOR DETAILS:

Mrs.P. Ashwini, M.tech(CSE), Asst. Professor, Department of Computer Science& Engineering, Teegala Krishna Reddy Engineering College, Hyderabad, T.S-500 097, India.

Mr.Pavan Sista, Department of Computer Science& Engineering, Teegala Krishna Reddy Engineering College, Hyderabad, T.S-500 097, India.

Mrs.A.RojaRamani ;M.Tech(CSE), Asst.Professor, Department of Computer Science Engineering, Teegala Krishna Reddy Engineering College, Hyderabad, T.S-500 097, India.