Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

School of Electrical Engineering

Second Year B.Tech. (EL & CE)

Basic IoT Laboratory

Course Code: ECE2009A

Laboratory Manual

2022-23

Basic IoT Laboratory

I N D E X

Sr. No.	Name of the Experiment	Page	Date of Checking	Signature of Batch I/C
1	To introduce various hardware platforms for IoT based design. (Example platforms are Arduino Uno /Node MCU/ Raspberry Pi/ ESP8266 / Beagle board/ Tiva / MSP430 /Jetson Nano/ Intel Galileo)			
2	Understanding Arduino IDE and Interfacing Basic Sensors with hardware platforms.			
3	Interfacing stepper motor/DC Motor and relay with hardware platforms.			
4	Understanding Node MCU as development platform and connecting to Wi-Fi network through Arduino IDE.			
5	Exploring cloud infrastructure for connecting IoT devices and sending and visualizing sensor data to open source cloud via Arduino IDE.			
6	Understanding Raspberry-Pi as single board Computer and exploring GPIO.			
7	Sensor and Actuator interfacing with Raspberry-Pi and development of simple application.			
8	Data Analysis for IoT. Upload and retrieve sensor data using Thing Speak or similar Platform.			
9	Study and interfacing of domain specific sensors and actuators with hardware platform.			
10	Project based on real life IoT applications and report submission.			

CERTIFICATE

Certified that Mr./Ms.____of Class **Second Year B. Tech. (EL & CE)** Division______Roll No._________has completed the laboratory work in the subject **Basic IoT Laboratory** during the Semester **IV** in the School of Electrical Engineering during the Academic Year 2022-2023.

Signature of the Faculty **Head of the Department**

Laboratory Instructions

Internet of Things

1. In addition to the instructions in this laboratory manual, follow all verbal and written instructions provided by the instructional staff.
2. Before coming to the laboratory, students should carefully read the sections of text which are covered in the assigned experiment. If the topic has been covered in lectures, the lecture notes should be studied as well.
3. Students should wear a valid ID card before entering the lab.
4. Use of mobile phone in the lab is strictly prohibited. Each student must keep mobile phones in "Switched Off" mode while entering and working in the Lab.
5. If any problem arises, please bring the same to the notice of lab in-charge.
6. Handle all instruments softly and gently.
7. In case of theft / destruction of the instruments in the lab, double cost of the lost/destructed instrument will be charged from the student/user.
8. Lab Assistants are available to assist basic instrument handling problems.
9. Do not exceed voltage ratings than that of provided in the manual.
10. Before leaving the laboratory, disconnect all electrical connections and return equipment to Lab Assistant.
11. It is mandatory to get checked the experiments write-ups in the next practical turn.

Second Year B. Tech (EL & CE)

Sememester: IV **Subject:** Basic IoT Lab

Name: **Class:**

Roll No: **Batch:**

Experiment No: 01

Name of the Experiment: To introduce various hardware platforms for IoT based design. (Example platforms are Arduino Uno /Node MCU/ Raspberry Pi/ ESP8266 / Beagle board/ Tiva / MSP430 /Jetson Nano/ Intel Galileo)

Performed on:

Submitted on:

Marks	Teacher's Signature with date

Aim: To introduce various hardware platforms for IoT based design. (Example platforms are Arduino Uno /Node MCU/ Raspberry Pi/ ESP8266 / Beagle board/ CC3200 Tiva / MSP430 /Jetson Nano/ Intel Galileo)

Pre-requisite: Knowledge of Electronics and circuits, Knowledge of Assembly and C Programming language.

Objective:

1. To introduce various hardware platforms for IoT based design.
2. To understand pin configuration of various hardware platforms.
3. To compare various hardware platforms

Components and equipment required:

Arduino Uno Board, NodeMCU, Raspberry Pi Board

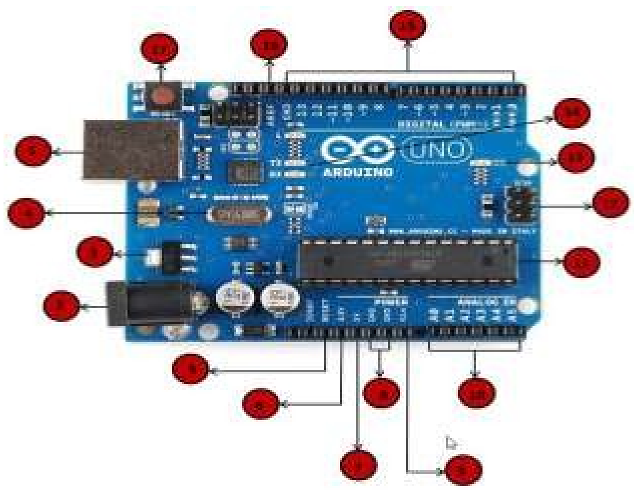
Theory:

IoT hardware platforms are considered as the most significant component of the IoT environment. One IoT device can be connected to other IoT device to enable the information exchange using suitable Internet protocols. IoT platforms works as a bridge between the device sensors and data networks. It gives data connectivity to the sensor system and helps to understand using back-end applications and helps to analyse the huge data generated by sensors.

Arduino

Reference Link: https://www.tutorialspoint.com/arduino/arduino_overview.htm
Arduino is a single-board microcontroller intended to make the application more manageable which are interactive objects and its surroundings. The hardware features with an open-source hardware board designed around an 8-bit Atmel AVR microcontroller or a 32-bit Atmel ARM. Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog input pins, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery.

Pin Diagram of Arduino Uno:



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. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.
5, 17: 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, 7, 8, 9: 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.

10: Analog pins

The Arduino UNO board has six 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.

11: Main Microcontroller

Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.

12: ICSP Pins

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.

13: 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.

14: TX and RX LEDs

On your board, you will find two labels: 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.

15: 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 labelled "~" can be used to generate PWM.

16: 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.

Raspberry Pi

Reference Link: https://www.electronicwings.com/raspberry-pi/raspberry-pi-introduction

Raspberry Pi is popularly used for real time Image/Video Processing, IoT based applications and Robotics applications. Raspberry Pi is slower than laptop or desktop but is still a computer which can provide all the expected features or abilities, at a low power consumption. Raspberry Pi Foundation officially provides Debian based Raspbian OS. Also, they provide NOOBS OS for Raspberry Pi. We can install several Third-Party versions of OS like Ubuntu, Arch linux, RISC OS, Windows 10 IOT Core, etc.

Raspbian OS is official Operating System available for free to use. This OS is efficiently optimized to use with Raspberry Pi. Raspbian have GUI which includes tools for Browsing, Python programming, office, games, etc. We should use SD card (minimum 8 GB recommended) to store the OS (operating System). Raspberry Pi is more than computer as it provides access to the on-chip hardware i.e. GPIOs for developing an application. By accessing GPIO, we can connect devices like LED, motors, sensors, etc and can control them too.

It has ARM based Broadcom Processor SoC along with on-chip GPU (Graphics Processing Unit). The CPU speed of Raspberry Pi varies from 700 MHz to 1.2 GHz. Also, it has on-board SDRAM that ranges from 256 MB to 1 GB. Raspberry Pi also provides on-chip SPI, I2C, I2S and UART modules.

Pin Diagram of Raspberry Pi 3 Model:

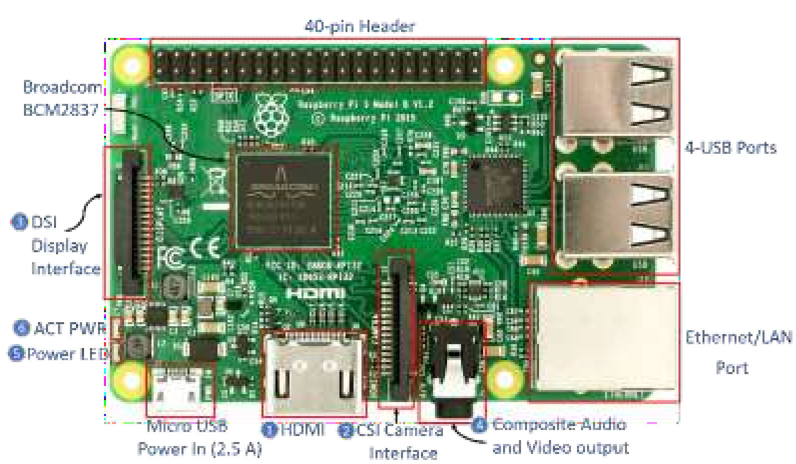


Figure 1.2: Raspberry Pi 3 Model B Hardware

1. **HDMI (High-Definition Multimedia Interface):** It is used for transmitting uncompressed video or digital audio data to the Computer Monitor, Digital TV, etc. Generally, this HDMI port helps to connect Raspberry Pi to the Digital television.
2. **CSI Camera Interface:** CSI (Camera Serial Interface) interface provides a connection in between Broadcom Processor and Pi camera. This interface provides electrical connections between two devices.
3. **DSI Display Interface:** DSI (Display Serial Interface) Display Interface is used for connecting LCD to the Raspberry Pi using 15-pin ribbon cable. DSI provides fast High-resolution display interface specifically used for sending video data directly from GPU to the LCD display.
4. **Composite Video and Audio Output:** The composite Video and Audio output port carries video along with audio signal to the Audio/Video systems.
5. **Power LED:** It is a RED coloured LED which is used for Power indication. This LED will turn ON when Power is connected to the Raspberry Pi. It is connected to 5V directly and will start blinking whenever the supply voltage drops below 4.63V.
6. **ACT PWR:** ACT PWR is Green LED which shows the SD card activity.

GPIO Pinout of Raspberry Pi 3 Model:

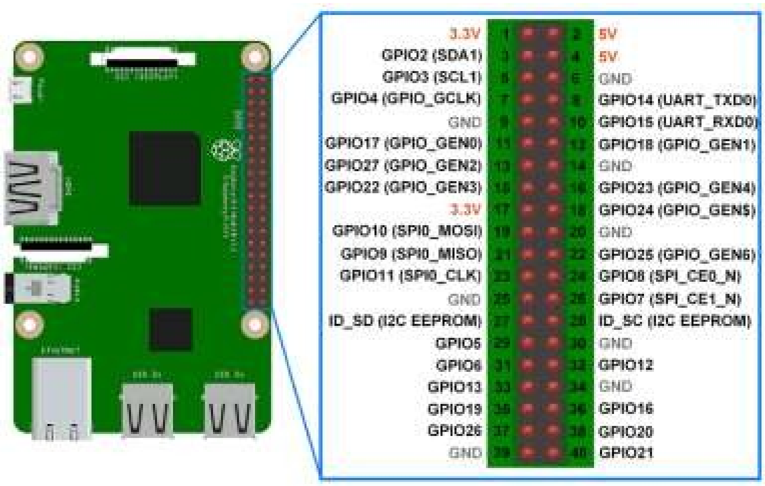


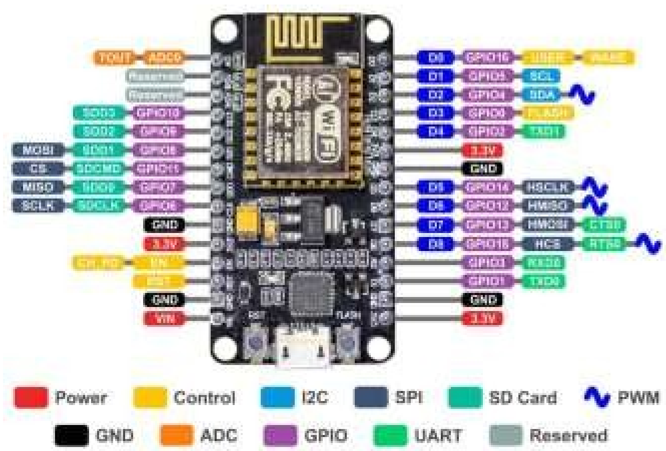
Figure 1.3: Raspberry Pi 3 Model B GPIO Pinout

Node MCU ESP8266

Reference Link: https://www.electronicwings.com/nodemcu/introduction-to-nodemcu

The NodeMCU (Node MicroController Unit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK. That makes it an excellent choice for the Internet of Things (IoT) projects of all kinds. But, what about Arduino? The Arduino project created an open-source hardware design and software SDK for their versatile IoT controller. Similar to NodeMCU, the Arduino hardware is a microcontroller board with a USB connector, LED lights, and standard data pins. It also defines standard interfaces to interact with sensors or other boards. But unlike NodeMCU, the Arduino board can have different types of CPU chips (typically an ARM or Intel x86 chip) with memory chips, and a variety of programming environments. There is an Arduino reference design for the ESP8266 chip as well. However, the flexibility of Arduino also means significant variations across different vendors. For example, most Arduino boards do not have WiFi capabilities, and some even have a serial data port instead of a USB port.

Pin Diagram of NodeMCU:



Power Pins: There are four power pins. VIN pin and three 3.3V pins.

- VIN can be used to directly supply the NodeMCU/ESP8266 and its peripherals. Power delivered on VIN is regulated through the onboard regulator on the NodeMCU module – you can also supply 5V regulated to the VIN pin
- 3.3V pins are the output of the onboard voltage regulator and can be used to supply power to external components.

I2C Pins: are used to connect I2C sensors and peripherals. Both I2C Master and I2C Slave are supported. I2C interface functionality can be realized programmatically, and the clock frequency is 100 kHz at a maximum. It should be noted that I2C clock frequency should be higher than the slowest clock frequency of the slave device.

GPIO Pins: NodeMCU/ESP8266 has 17 GPIO pins which can be assigned to functions such as I2C, I2S, UART, PWM, IR Remote Control, LED Light and Button programmatically. Each digital enabled GPIO can be configured to internal pull-up or pull-down, or set to high impedance. When configured as an input, it can also be set to edge-trigger or level-trigger to generate CPU interrupts.

ADC Channel: The NodeMCU is embedded with a 10-bit precision SAR ADC. The two functions can be implemented using ADC. Testing power supply voltage of VDD3P3 pin and testing input voltage of TOUT pin. However, they cannot be implemented at the same time.

UART Pins: NodeMCU/ESP8266 has 2 UART interfaces (UART0 and UART1) which provide asynchronous communication (RS232 and RS485), and can communicate at up to 4.5 Mbps. UART0 (TXD0, RXD0, RST0 & CTS0 pins) can be used for communication. However, UART1 (TXD1 pin) features only data transmit signal so, it is usually used for printing log.

SPI Pins: NodeMCU/ESP8266 features two SPIs (SPI and HSPI) in slave and master modes. These SPIs also support the following general-purpose SPI features:

- 4 timing modes of the SPI format transfer
- Up to 80 MHz and the divided clocks of 80 MHz
- Up to 64-Byte FIFO

SDIO Pins: NodeMCU/ESP8266 features Secure Digital Input/Output Interface (SDIO) which is used to directly interface SD cards. 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 are supported.

PWM Pins: The board has 4 channels of Pulse Width Modulation (PWM). The PWM output can be implemented programmatically and used for driving digital motors and LEDs. PWM frequency range is adjustable from 1000 μs to 10000 μs (100 Hz and 1 kHz).

Control Pins: are used to control the NodeMCU/ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.

- **EN:** The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
- **RST:** RST pin is used to reset the ESP8266 chip.
- **WAKE:** Wake pin is used to wake the chip from deep-sleep.

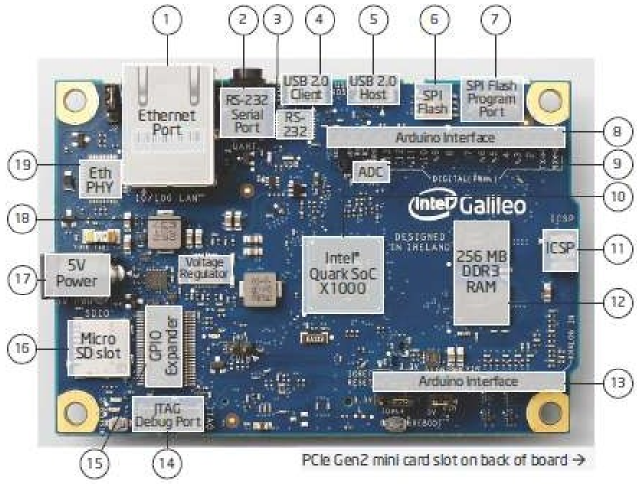
Intel Galileo

Reference Link: https://www.arduino.cc/en/ArduinoCertified/IntelGalileo

Intel Galileo is a Microcontroller board based on the Intel Quark SoC X1000 Application Processor, a 32-bit Intel Pentium class system on a chip. It's the first board based on an Intel architecture designed to be hardware and software pin-compatible with Arduino Shields designed for the UNO R3. Galileo is designed to support Arduino shields that operate at either 3.3V or 5V. The core operating voltage of Galileo is 3.3V. In spite of this, a jumper on the board can convert the voltage up to 5V at the I/O pins.

In addition to supporting the Arduino shield ecosystem, the Intel development board comes with many computing industry standard I/O interfaces, including ACPI Express, PCI Express, 10/100 Mbit Ethernet, Micro SD or SDHD, USB 2.0 device and EHCI/OHCI USB host ports, high-speed UART, RS-232 serial port, programmable 8 MB NOR flash and a JTAG port for easy debugging.

Pin Diagram of Arduino Intel Galileo:



1. Ethernet Port: 10/100 Ethernet connector
2. RS-232 Serial Port: 3-pin 3.5mm jack (not audio)
3. RS-232: RS-232 transceiver
4. USB 2.0 Client: USB Client connector (Micro-USB Type B)
5. USB 2.0 Host: USB 2.0 Host connector (Micro-USB Type AB)
6. SPI Flash: 8 MByte Legacy SPI Flash to store the firmware (or bootloader) and the latest sketch.
7. SPI Flash Program Port: 7-pin header for Serial Peripheral Interface (SPI) programming
8. Shield Interface: Complies with Arduino Uno Revision 3 shield pinout
9. ADC: Analog to Digital converter
10. Intel® Quark SoC X1000: 400 MHz 32-bit Intel® Pentium processor
11. ICSP: 6-pin in-circuit serial programming (ICSP) header
12. 256 MB DDR3 RAM: 256 MByte DRAM, enabled by the firmware by default
13. Arduino Interface: Complies with Arduino Uno Revision 3 pinout
14. JTAG Debug Port: 10-pin standard JTAG header for debugging
15. GPIO Expander: GPIO pulse width modulation provided by a single I2C I/O expander
16. Micro SD slot: Supports micro SD card up to 32 GBytes (Optional)
17. 5V Power: The board is powered via an AC-to-DC adapter
18. Voltage Regulator: Generates 3.3 volt supply
19. Eth PHY: Ethernet Physical layer transceiver

Jetson Nano

NVIDIA Jetson Nano is an embedded system-on-module (SoM) and developer kit from the NVIDIA Jetson family, including an integrated 128-core Maxwell GPU, quad-core ARM A57 64-bit CPU, 4GB LPDDR4 memory, along with support for MIPI CSI-2 and PCIe Gen2 high-speed I/O. There is also the Jetson Nano 2GB Developer Kit with 2GB memory and the same processing specs.

Useful for deploying computer vision and deep learning, Jetson Nano runs Linux and provides 472 GFLOPS of FP16 compute performance with 5-10W of power consumption. NVIDIA Jetson today is widely used in diverse fields such as robotics, retail, industrial, agriculture, and AIoT.

Pin Configuration of Jetson Nano:

20. microSD card slot for main storage
21. 40-pin expansion header
22. Micro-USB port for 5V power input, or for Device Mode
23. Gigabit Ethernet port
24. USB 3.0 ports (x4)
25. HDMI output port
26. DisplayPort connector
27. DC Barrel jack for 5V power input
28. MIPI CSI-2 camera connectors

Conclusion:

Post Lab Questions:

1. List out different Arduino flavours with features
2. Mention different Arduino shields with their use
3. Compare various versions of Raspberry Pi with respect to different parameters
4. Different Models of the NodeMCU with Specifications
5. Brief use cases of any one IoT hardware platform

Additional links for more information:

1. **Getting started with Arduino with Spoken-Tutorial**
   https://spoken-tutorial.org/tutorial-search/?search_foss=Arduino&search_language=English
2. **Getting started with NodeMCU**
   https://www.electronicshub.org/getting-started-with-nodemcu/
3. **Introduction to Raspberry Pi**
   https://www.udemy.com/course/introduction-to-raspberry-pi-4/
4. **Building IoT with Raspberry Pi**
   https://www.blemobileapps.com/blog/building-internet-things-iot-raspberry-pi/
5. **An overview of IoT Hardware Development Platforms**
   Dhawan Singh, Amanpreet Sandhu, Aditi Thakur and Nikhil Priyanka, "An overview of IoT Hardware Development Platforms", International Journal on Emerging Technologies 11(5):155-163(2020)