|  |
| --- |
| Close-up image showing the leaf-sides of two oversized books side-by-side on a bookshelf, with additional books in soft focus background |
| IOT REDBOOK  INTRODUCTION |
| |  |  |  | | --- | --- | --- | | Navneet Mishra, Jacob Victor |  |  | |

**Table of Contents**

[Introduction 2](#_Toc534891380)

[IoT Challenges 2](#_Toc534891381)

[IoT Architecture 3](#_Toc534891382)

[IOT Architecture Diagram 4](#_Toc534891383)

[IoT Attack Surface 4](#_Toc534891384)

[Hardware Basics 5](#_Toc534891385)

[Resistor 5](#_Toc534891386)

[Capacitor 5](#_Toc534891387)

[Transistor 6](#_Toc534891388)

[Dual in line package (DIP) 7](#_Toc534891389)

[Surface-Mount (SMD/SMT) Packages 8](#_Toc534891390)

[Quad Flat Packages 8](#_Toc534891391)

[Quad-flat no-leads (QFN) 9](#_Toc534891392)

[Ball grid array (BGA) 9](#_Toc534891393)

[Computer memory 9](#_Toc534891394)

[PCB 10](#_Toc534891395)

[Multimeter 11](#_Toc534891396)

[Continuity Test 12](#_Toc534891397)

[Datasheet 12](#_Toc534891398)

[Bootloaders 12](#_Toc534891399)

[IOT operating system 12](#_Toc534891400)

[File Systems 13](#_Toc534891401)

[Serial and parallel communication 13](#_Toc534891402)

[Reference 13](#_Toc534891403)

# Introduction

Internet of Things (IoT) is an ecosystem of connected physical objects that are accessible through the internet. The ‘thing’ in IoT could be a person with a heart monitor or an automobile with built-in-sensors, i.e. objects that have been assigned an IP address and have the ability to collect and transfer data over a network without manual assistance or intervention.

IoT involves extending Internet connectivity beyond standard devices, such as desktops, laptops, smartphones and tablets, to any range of traditionally dumb or non-internet-enabled physical devices and everyday objects. Embedded with technology, these devices can communicate and interact over the Internet, and they can be remotely monitored and controlled.

The extensive set of applications for IoT devices is often divided into consumer, commercial, industrial, and infrastructure spaces.

* Consumer IoT includes the connected devices such as smart cars, phones, watches, laptops, connected appliances, and entertainment systems.
* Commercial IoT includes things like inventory controls, device trackers, and connected medical devices.
* Industrial IoT covers such things as connected electric meters, waste water systems, flow gauges, pipeline monitors, manufacturing robots, and other types of connected industrial devices and systems.

# IoT Challenges

**Governmental Regulatory Standards** - Regulations should do more than tell companies what they can’t do - rules should help guide corporate players through minefields of uncertainty. It’s a lot of responsibility, especially when it comes to still-developing IoT technology that holds great promise—and real risks.

**Multiple protocols and standards** - It leads to compatibility issue between vendors. compatibility refers to the ability of disparate IoT devices and services to exchange information and to use the information that has been exchanged. Current IoT initiatives focus largely on applications and devices that address disparate needs, but provide little scope for interoperation and connection.

**Privacy and security issues** -  IoT has already turned into a serious security concern that has drawn the attention of prominent tech firms and government agencies across the world. The hacking of baby monitors, smart fridges, thermostats, drug infusion pumps, cameras and even the radio in your car are signifying a security nightmare being caused by the future of IoT. So many new nodes being added to networks and the internet will provide malicious actors with innumerable attack vectors and possibilities to carry out their evil deeds, especially since a considerable number of them suffer from security holes.

**Connectivity challenges** - Connecting so many devices will be one of the biggest challenges of the future of IoT, and it will defy the very structure of current communication models and the underlying technologies. At present we rely on the centralized, server/client paradigm to authenticate, authorize and connect different nodes in a network.

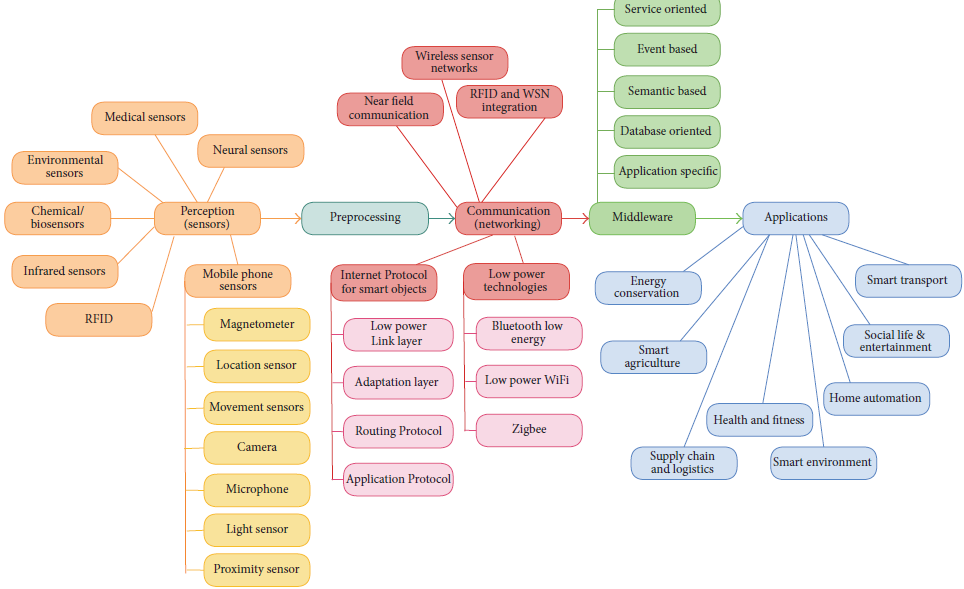
This model is sufficient for current IoT ecosystems, where tens, hundreds or even thousands of devices are involved. But when networks grow to join billions and hundreds of billions of devices, centralized systems will turn into a bottleneck. Such systems will require huge investments and spending in maintaining cloud servers that can handle such large amounts of information exchange, and entire systems can go down if the server becomes unavailable.

# IoT Architecture

There is no single consensus on architecture for IoT, which is agreed universally. Different architectures have been proposed by different researchers. Given below is a generalized 5 – layer architecture.

Sensors, actuators, compute servers, and the communication network form the core infrastructure of an IoT framework. All IoT applications need to have one or more sensors to collect data from the environment. Sensors are essential components of smart objects. One of the most important aspects of the Internet of Things is context awareness, which is not possible without sensor technology. IoT sensors are mostly small in size, have low cost, and consume less power.

# IOT Architecture Diagram



# IoT Attack Surface

* Mobile
  + Storage
  + Communication
  + Authentication
  + Encryption
  + Generic Mobile vulnerabilities
  + OWASP Mobile Top 10
* Device
  + Sensing
  + Storage
  + Communication
  + Authentication
  + Encryption
  + Hardware
  + Network Services – HTTP, SSH, Telnet...
* Web/Cloud
  + Storage
  + Communication
  + Authentication
  + APIs
  + Encryption
  + Generic Web/cloud vulnerabilities
  + OWASP Top 10
* Network
  + Custom IoT Protocols
  + Radio analysis/replay/command
  + Injection
  + Improper Encryption
  + Insecure Protocol implementation
  + Protocol vulnerabilities

# Hardware Basics

## Resistor

Resistors are electronic components, which offer resistance against the current flow, or speaking at a deeper level, against the electrons' flow. Resistors, denoted by R, are passive components (they only consume power and can’t generate it) but rather reduce voltage and current by dissipating power in the form of heat.

The unit of resistance is ohms (Ω) and resistors are usually built using carbon or metal wire. It is also worth noting that there could be two different categories of resistors-fixed and variable. As the name implies, a fixed resistor is where the resistance is fixed and cannot be changed, whereas in a variable resistor, the resistance can be varied using certain techniques. One of the most popular examples of a variable resistor would be a potentiometer.

You will also find the resistors being color-coded in order to help convey the value of resistance they offer.



## Capacitor

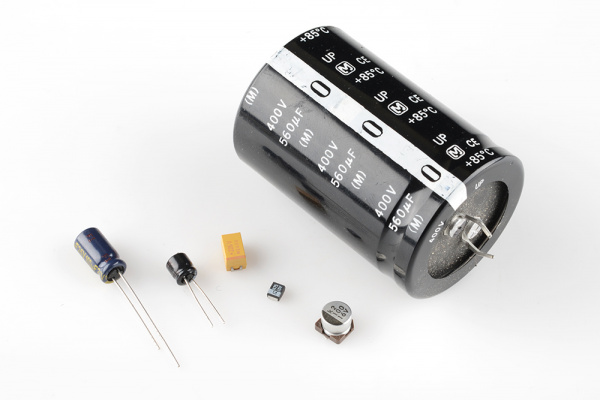
Capacitors are one of the other most common components found in almost all embedded devices. As the name suggests, one of their primary tasks is to hold energy in the form of an electric charge.

Inside a capacitor are two oppositely charged plates, which hold the electric charge when connected to a power source. Some of the other usages of a capacitor are acting as filters, reducing electrical noise affecting other chips on the device, separating AC and DC components (AC coupling), and so on. The unit of capacitance is the faraday denoted with an *F.*

Based on the type of material, Capacitors can be of following types -



Ceramic Capacitor



Electrolytic capacitor

Note – In the figure above the small capacitors which don’t have the protruding pins are the surface mount capacitors.

## Transistor

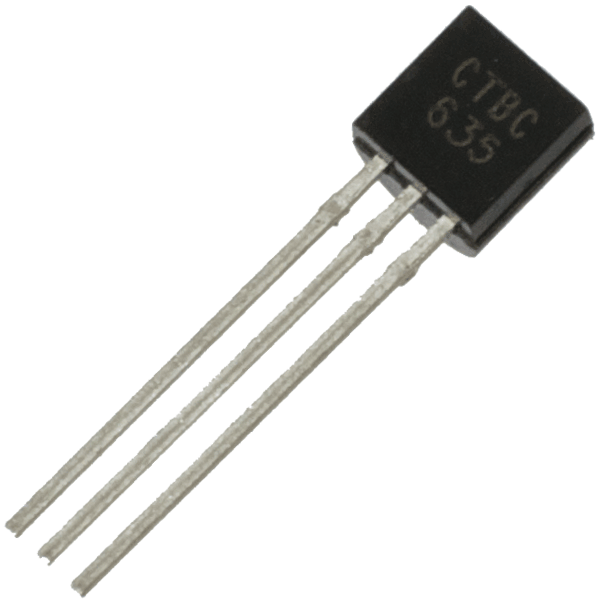
Transistors are fundamentally three-terminal devices. the most common transistor around is the bi-polar junction transistor (BJT). On a bi-polar junction transistor (BJT), those pins are labeled **collector** (C), **base** (B), and **emitter** (E). The circuit symbols for both the NPN and PNP BJT are below. The only difference between an NPN and PNP is the direction of the arrow on the emitter. The arrow on an NPN points out, and on the PNP it points in. A useful mnemonic for remembering which is which is: **NPN:**N**ot**P**ointing i**N.



They serve a number of purposes by acting as both switches and amplifiers. As an amplifier, it can take in a small current and amplify it to produce a much bigger output current. One of the examples of this could be in a microphone connected to a loudspeaker, where the mic takes in a small sound input and amplifies it to produce a much louder sound when it comes out via the loudspeaker.

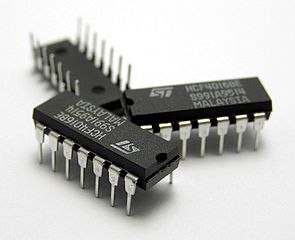
Similarly, as a switch, it can take a small current input and use it to allow a much larger current to flow by activating the new current flow.

This is what a transistor looks like:

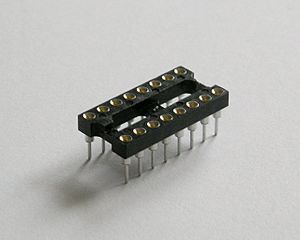


## Dual in line package (DIP)

A dual in-line package (DIP or DIL), or dual in-line pin package (DIPP) is an electronic component package with a rectangular housing and two parallel rows of electrical connecting pins. A DIP is usually referred to as a DIP*n*, where *n* is the total number of pins. For example, a microcircuit package with two rows of seven vertical leads would be a DIP14. Most DIP packages are secured to a PCB by inserting the pins through holes in the board and soldering them in place. Where replacement of the parts is necessary, such as in test fixtures or where programmable devices must be removed for changes, a DIP socket is used.



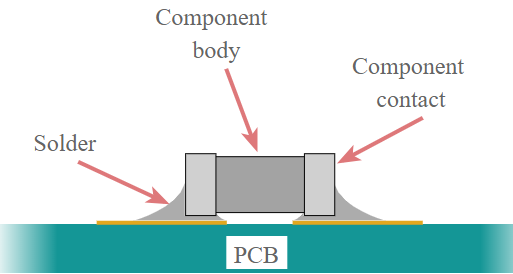
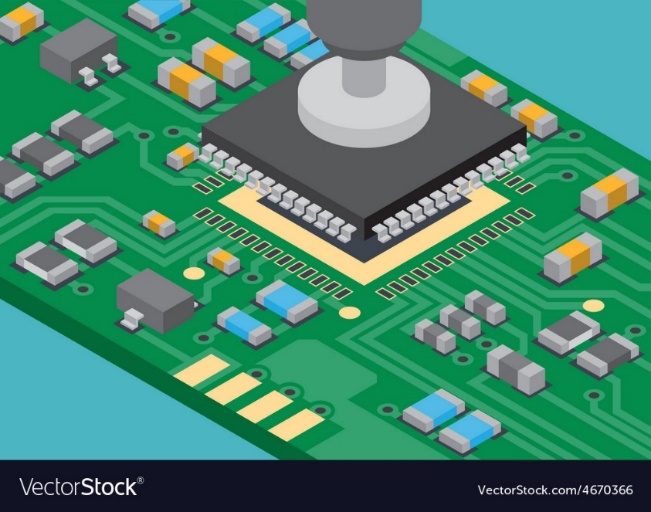
**DIP14**

**DIP Socket**

## Surface-Mount (SMD/SMT) Packages

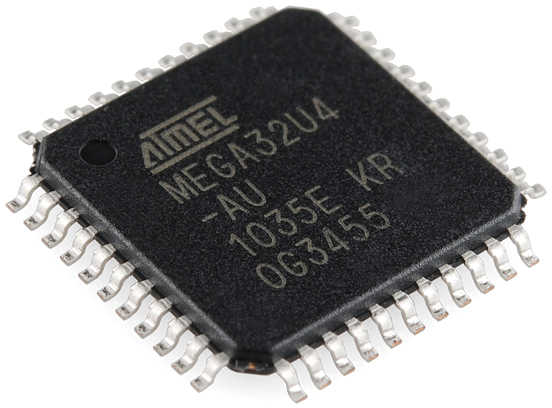
It is a method for producing [electronic](https://en.wikipedia.org/wiki/Electronics) circuits in which the components are mounted or placed directly onto the surface of [printed circuit boards](https://en.wikipedia.org/wiki/Printed_circuit_board) (PCBs). it has largely replaced the [through-hole technology](https://en.wikipedia.org/wiki/Through-hole_technology) construction method of fitting components with wire leads into holes in the circuit board. SMT component is usually smaller than its through-hole counterpart because it has either smaller leads or no leads at all.

Surface mount technology IC placement

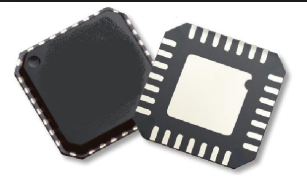
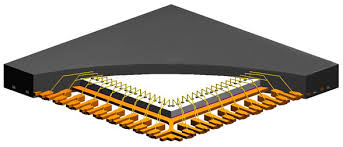
## Quad Flat Packages

A QFP or Quad Flat Package is a [surface mount](https://en.wikipedia.org/wiki/Surface-mount_technology) [integrated circuit](https://en.wikipedia.org/wiki/Integrated_circuit) [package](https://en.wikipedia.org/wiki/Integrated_circuit_packaging) with "gull wing" leads extending from each of the four sides.[[1]](https://en.wikipedia.org/wiki/Quad_Flat_Package#cite_note-Greig07-1)Socketing such packages is rare and through-hole mounting is not possible.



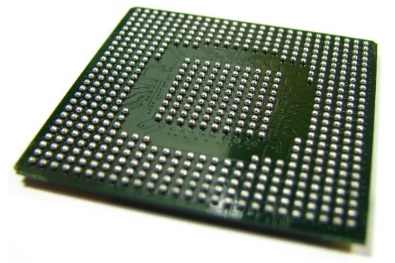
## Quad-flat no-leads (QFN)

The QFN package is similar to the [Quad Flat Package](https://en.wikipedia.org/wiki/QFP), but the leads do not extend out from the package sides. It is hence difficult to hand-solder a QFN package. The QFN (Quad Flat No-lead) package is probably the most popular semiconductor package today because of four reasons: low cost, small form factor and good electrical and thermal performance. Like any other semiconductor package, a QFN package functionality is to connect (both physically and electrically) silicon dies (the ASIC) to a printed circuit board (PCB) using surface-mount technology. QFN is a lead frame-based package which is also called CSP (Chip Scale Package) with the ability to view and contact leads after assembly.



## Ball grid array (BGA)

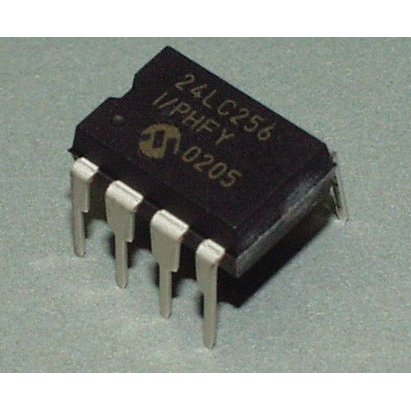
It is a type of [surface-mount](https://en.wikipedia.org/wiki/Surface-mount) packaging (a [chip carrier](https://en.wikipedia.org/wiki/Chip_carrier)) used for [integrated circuits](https://en.wikipedia.org/wiki/Integrated_circuit). BGA packages are used to permanently mount devices such as [microprocessors](https://en.wikipedia.org/wiki/Microprocessor). A BGA can provide more interconnection pins than can be put on a [dual in-line](https://en.wikipedia.org/wiki/Dual_in-line_package) or [flat package](https://en.wikipedia.org/wiki/Quad_Flat_Package). The whole bottom surface of the device can be used, instead of just the perimeter. The leads are also on average shorter than with a perimeter-only type, leading to better performance at high speeds. The BGA is a solution to the problem of producing a miniature package for an integrated circuit with many hundreds of pins



## Computer memory

This is where you will be able to find things such as the firmware and application programming interface (API) keys. It is any physical device capable of storing information temporarily like RAM (random access memory), or permanently, like ROM (read-only memory). Memory devices utilize integrated circuits and are used by operating systems, software, and hardware. Memory can be either volatile and non-volatile memory. Volatile memory is a memory that loses its contents when the computer or hardware device loses power. Computer RAM is an example of a volatile memory and is why if your computer freezes or reboots when working on a program, you lose anything that hasn't been saved. Non-volatile memory, sometimes abbreviated as NVRAM, is a memory that keeps its contents even if the power is lost. EPROM is an example of a non-volatile memory.

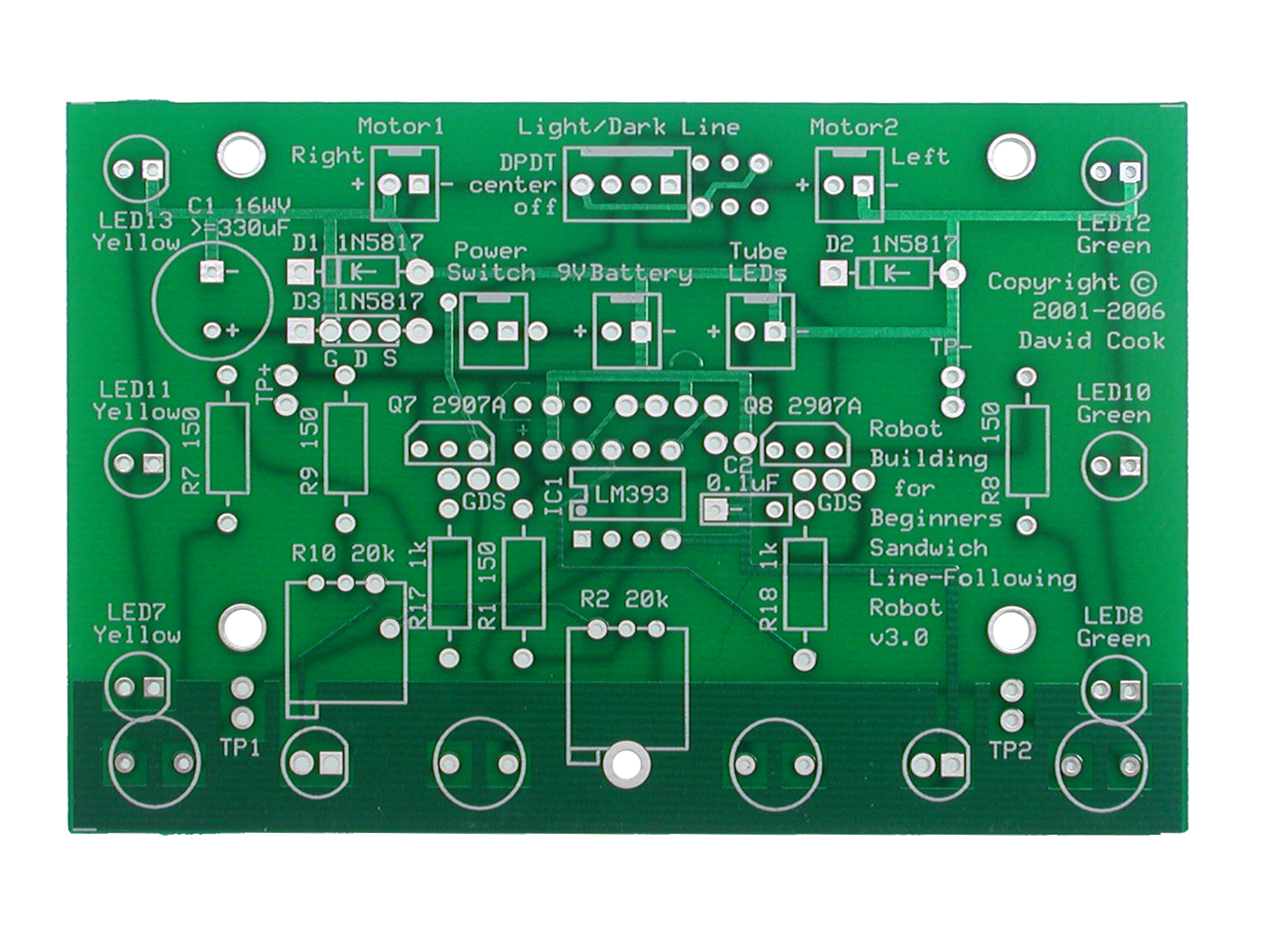
The EEPROM is a non-volatile storage location which is read and writable as single blocks of bytes. The EEPROM can be erased by electrical charges or UV exposure. Similar to other flash storage types, EEPROM allows a limited number of write cycles. EEPROM is a chip of interest, as firmware may be loaded on an EEPROM and can be removed from the PCB to an EEPROM reader for further analysis:



NAND flash memory is written and read in blocks, which are commonly found in USB drives but are also in IoT devices as well as game consoles. The NAND flash typically contains a device's bootloader which follows various instructions to start the operating system and can be manipulated.

## PCB

Device hardware starts with the Printed Circuit Board (PCB), which is comprised of fiberglass, copper, the solder mask, silkscreen, traces, and pads. Components such as resistors, capacitors, chips for Wi-Fi, EEPROMs, and serial and microcontrollers are soldered onto the PCB. There are various layers of thin copper foil that make a PCB conductive and also insulated layers that make it non-conductive. It's important to identify components of interest when looking at a PCB. Components of interest include sources of input into the device firmware either directly or indirectly. Components such as the EEPROM, NAND flash, Universal Asynchronous Receiver/Transmitter (UART), and Joint Test Action Group (JTAG) are some of the most common components to focus on for testing purposes.



## Multimeter

The digital multimeter (DMM) is a vital instrument. We can use it to measure DC and AC voltage, DC current, resistance, continuity, and test diodes etc. Two probes are plugged into two of the **ports** on the front of the unit. **COM** stands for common and is almost always connected to Ground or ‘-’ of a circuit. The **COM** probe is conventionally black but there is no difference between the red probe and black probe other than color. **mAVΩ** is the port that the red probe is conventionally plugged in to.



### Continuity Test

This test helps insure that connections are made correctly between two points. It helps us check the continuity of any wire or circuit. To check continuity between two points A and B,

* set the multimeter to ‘Continuity’ mode (look for symbol like sound coming from a speaker)
* place the black probe on the COM slot and the Red probe on the mAVΩ slot.
* place one probe (any probe) on point A and the other on point B.
* If there is connection between point A and point B, then the meter will read zero and you will also get a “beep” sound. If there is no connection, you won’t get the beep sound.

To learn more about how to use a multimeter visit - <https://learn.sparkfun.com/tutorials/how-to-use-a-multimeter>

# Datasheet

The datasheet is your complete encyclopedia on a part. The most obvious thing the datasheet will give you the part's pinout, so that you know how to connect it. It will let us know voltage and current ratings. Absolute Maximum Ratings, typical application, basic specifications, mechanical drawings, timing diagrams, graphs, example schematics, packaging information and many things. A datasheet is the best place to find:

* What voltage a part needs to run
* How fast a part will run
* How to communicate with a part.

# Bootloaders

A bootloader's responsibility is to initialize RAM for volatile data storage, initialize serial port(s), detect the machine type, set up the kernel tagged list, load initramfs (initial RAM filesystem), and call the kernel image. The bootloader initializes hardware drivers via a Board Support Package (BSP), which is usually developed by a third party. The bootloader resides on a separate Electrically Erasable Programmable Read-only Memory (EEPROM), which is less common, or directly on flash storage, which is more common. Think of the bootloader as a PC's BIOS upon start up. It is a program that loads an operating system when a computer is turned on. Some of the common bootloaders for ARM and MIPS architectures are: Redboot, u-boot, and barebox. Once the bootloader starts up the kernel, the filesystem is loaded.

# IOT operating system

This type of operating system is typically designed to be resource-efficient and reliable. Resource efficiency comes at the cost of losing some functionality or granularity that larger computer operating systems provide, including functions which may not be used by the specialized applications they run. Unlike a desktop operating system, the embedded operating system does not load and execute applications. This means that the system is only able to run a single application. E.g

* Linux–
* Windows CE–
* VxWorks–
* Cisco IOS / JunOS

<https://en.wikipedia.org/wiki/Real-time_operating_system>

# File Systems

We can think of the filesystem as the location that contains configuration files, services, account passwords, hashes, and application code, as well as start-up scripts. There are many filesystem types employed within the firmware, and sometimes even proprietary file types are used depending on the device. However, some of most common types of filesystems are -

* squashFS
* cramFS
* JFFS2
* YAFFS2
* ext2

The most common filesystem utilized in devices (especially consumer devices) is SquashFS.

# Serial and parallel communication

The two methods of data communication in embedded devices are serial and parallel communication. As the name suggests, serial communication sends data one bit at a time sequentially. This means that if 8 bits have to be transferred, it will send one bit after the other, and the data transfer will be complete only when all the 8 bits are received. However, in cases of parallel communication, multiple bits would be transferred at the same time, thus making the data transfer process faster compared to its serial counterpart.

You might think that parallel communication would be much better and would be used predominantly everywhere because of faster data transfer rates. This, however, is not the case because we did not consider the amount of real estate on the circuit board it would require for a parallel communication. Embedded devices are extremely low in physical space. Thus, when it comes to data transfer, faster is not always the better option, when considering the fact that a parallel data transfer would require much more data lines compared with a serial data transfer. Some of the examples of parallel data transfer communications are PCI and ATA, whereas a serial communication is undertaken using USB, UART, and SPI.

# Reference

* https://www.happiestminds.com/Insights/internet-of-things/
* https://en.wikipedia.org/wiki/Internet\_of\_things
* https://iot.ieee.org/newsletter/march-2017/three-major-challenges-facing-iot.html
* https://techbeacon.com/sites/default/files/4\_stage\_iot\_solutions\_architecture\_0.jpeg
* https://en.wikipedia.org/wiki/Dual\_in-line\_package
* https://www.electronics-notes.com/images/smt-surface-mount-technology-basic-concept-01.svg
* https://www.vectorstock.com/royalty-free-vector/surface-mount-technology-ic-placement-vector-4670366
* https://en.wikipedia.org/wiki/Quad\_Flat\_Package
* https://anysilicon.com/ultimate-guide-qfn-package/
* http://www.statschippac.com/~/media/Files/Package%20Datasheets/QFN.ashx
* https://en.wikipedia.org/wiki/Ball\_grid\_array
* https://www2.deloitte.com/insights/us/en/focus/internet-of-things/regulating-iot-technology-role-of-government.html
* https://datafloq.com/read/internet-of-things-iot-security-privacy-safety/948
* https://www.computerhope.com/jargon/m/memory.htm
* https://techterms.com/definition/firmware
* https://www.computerhope.com/jargon/f/firmware.htm
* http://downloads.hindawi.com/journals/jece/2017/9324035.pdf
* http://ec.europa.eu/information\_society/newsroom/cf/dae/document.cfm?doc\_id=1750