Internet of Things

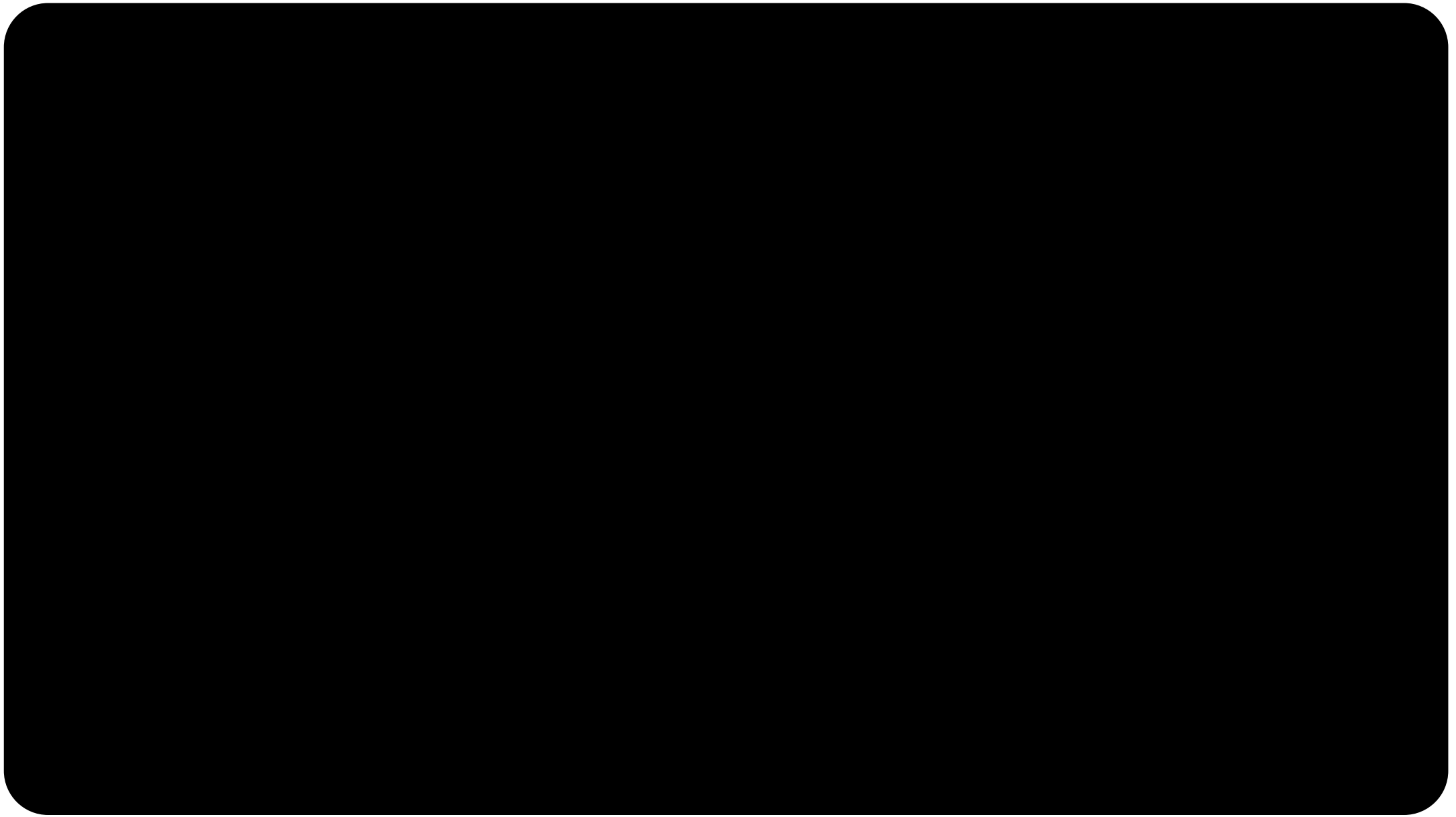
IE 4030

Uditha Dharmakeerthi

Module Content- In Details

CONTENTS OF THE MODULE	
Topic	**Aligned learning outcomes**
1. The Internet of Things: An overview • **Introduction to IoT systems** • **Cognitive systems**	**LO1**
2. IoT eco system • **Layers in IoT** • **Services in IoT** • **Services in big data management**	**LO1, LO2**
Design principles for connected devices (MQTT Protocol, Node-Red)	
4. IoT Data gathering and warehousing •	**LO2 – LO4**
5. Embedded systems for IoT • **Introduction to microcontrollers** • **Low power embedded systems** • **Introduction to real-time operating systems**	**LO2 – LO4**

Are you ready to embrace digital transformation?



Apita ekama device eke idan device keepayak ekka communicate karanna puluwan MQTT platforme eka haraha.

Introducing MQTT

MQTT (***Message Queuing Telemetry Transport***)is a Client Server publish/subscribe messaging transport protocol.

Api yawana message eka MQTT protocol eka haraha api anith side ekata data send karanawa.

It is lightweight, open, simple, and designed to be easy to implement.

Brocker kenek use karala api mulin connection hadaaganna one. After Topic ekak yatathe yawana data eka ee topic eka subscribe karala tiyena ayata witharai pennanne.

These characteristics make it ideal for use in many situations, including constrained environments such as for communication in Machine to Machine (M2M) and Internet of Things (IoT) contexts where a small code footprint is required and network bandwidth is at a premium

MQTT ekedi api subscribe karala tiyena data witharai display wenne. But HTTP ekedi api subscribe nokarapu data pawaa apita display wenawa. ethakota eka HTTP eken athulata awata passe thama data filter karaganna wenne.

Reference :official MQTT 3.1.1 specification

Introducing MQTT

The MQTT protocol was invented in 1999 by Andy Stanford-Clark (IBM) and Arlen Nipper (Arcom, now Cirrus Link).

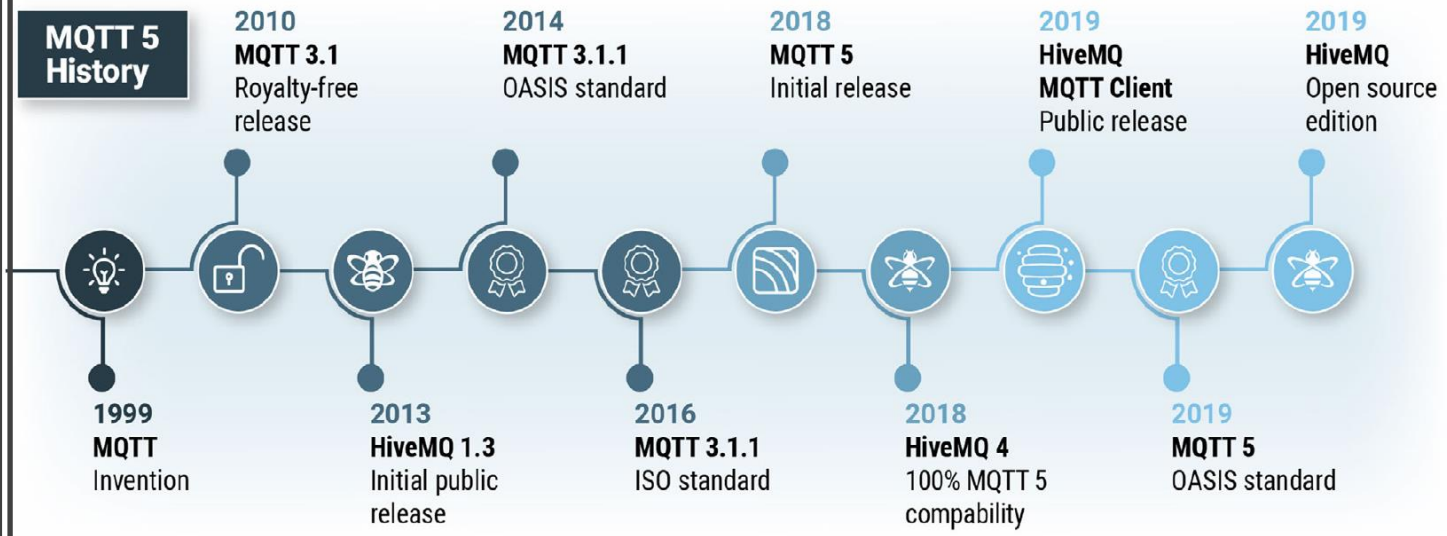
They needed a protocol for **minimal battery** loss and **minimal bandwidth** to connect with oil pipelines via satellite.

The two inventors specified several requirements for the future protocol:

- Simple implementation
- Quality of Service data delivery
- Lightweight and bandwidth efficient
- Data agnostic
- Continuous

Reference :official MQTT 3.1.1 specification

Introducing MQTT



Reference :official MQTT 3.1.1 specification

What is Publish / Subscribe Architecture?

The pub/sub model removes direct communication between the **publisher of the message** and the **recipient/subscriber**.

The filtering activity of the **broker makes it possible** to control which client/subscriber receives which message.

mulin check karala balanawa eyata mokak hari topic ekak awashai data eka yawanna. mulin eya topic eka publish karanawa. ita passe e topic eka subscribe kare kawda balala eka subscribe karapu kenata witharak data eka yawanawa.

The decoupling has three dimensions:

- **Space decoupling**:

Publisher and subscriber do not need to know each other (for example, no exchange of IP address and port).

MQTT ekedi devices deka interconnect karanna one ne. HTTP nam same port same name wenna one

- **Time decoupling**:

Publisher and subscriber do not need to run at the same time.

device dekama ekama time zone wenna one ne.

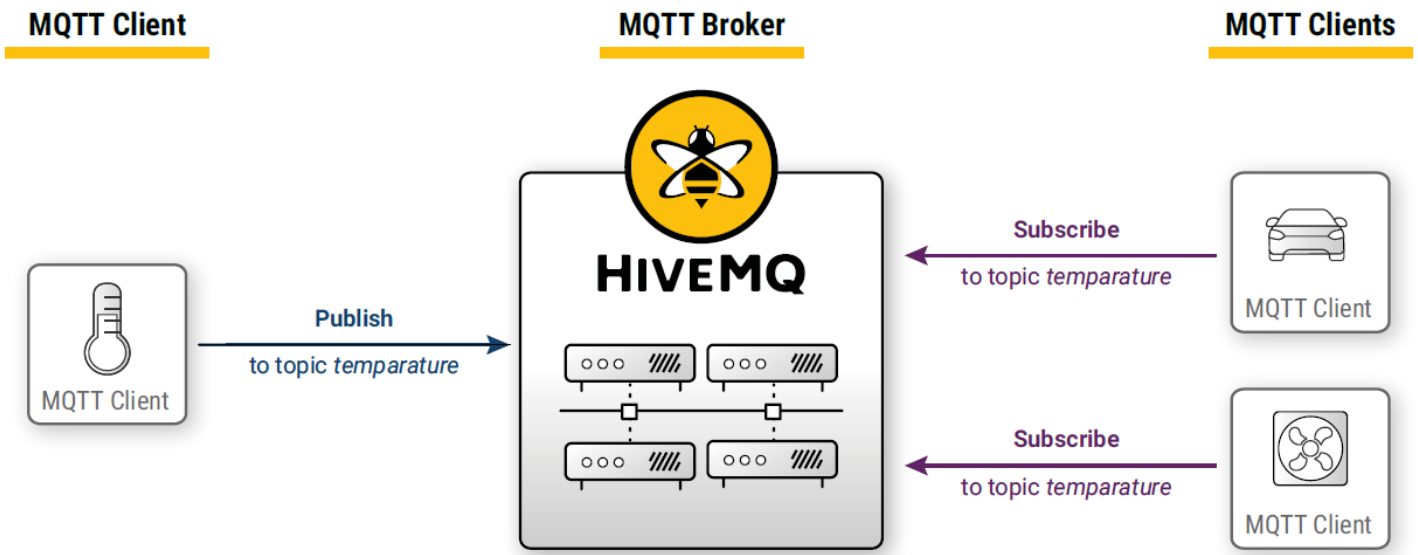
- **Synchronization decoupling**:

Operations on both components do not need to be interrupted during publishing or receiving.

publishing or receiving dekedima weda karana service eka stop nokara, running wena service eka nawaththanne nathuwa data read karanna puluwan

Reference :official MQTT 3.1.1 specification

Introducing MQTT

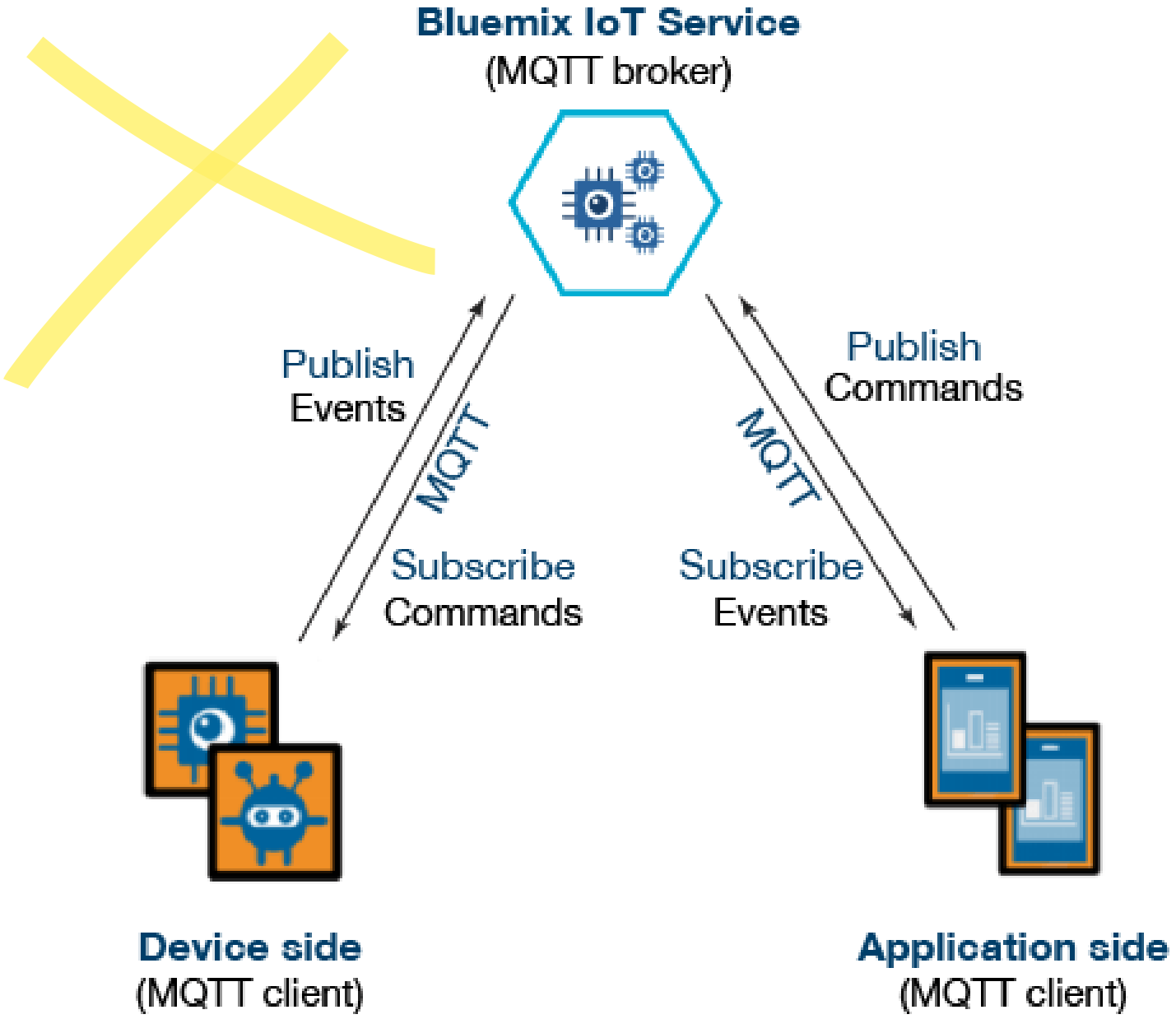


Topic Send -->

Store the Topic and find the who is subscriber or not to send the data

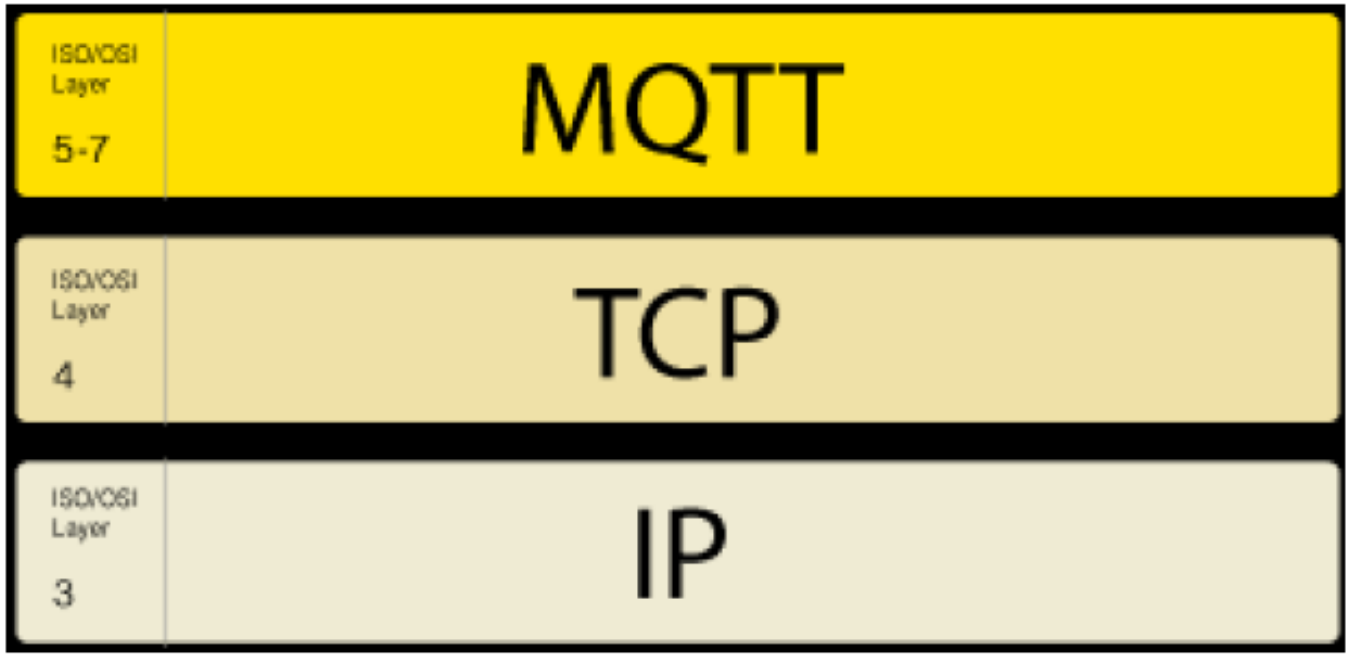
Reference :official MQTT 3.1.1 specification

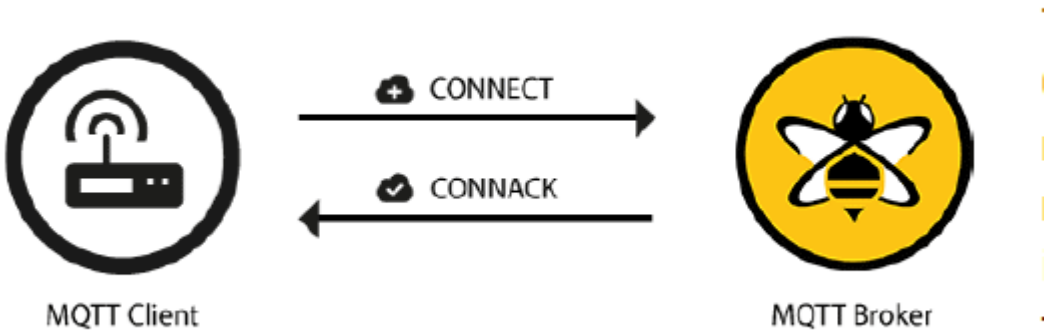
MQTT Architecture in Details



MQTT Connection

The MQTT protocol is based on TCP/IP. Both the client and the broker need to have a TCP/IP stack.





Reference :official MQTT 3.1.1 specification

MQTT connection through a NAT

- MQTT client is located behind a router that uses network address translation (NAT) to translate from a private network address (like 192.168.x.x, 10.0.x.x) to a public facing address

- The MQTT client initiates the connection by sending a CONNECT message to the broker. As the broker has a public address and keeps the connection open to allow bidirectional sending and receiving of messages (after the initial CONNECT), there is no problem at all with clients that are located behind a NAT

Reference :official MQTT 3.1.1 specification

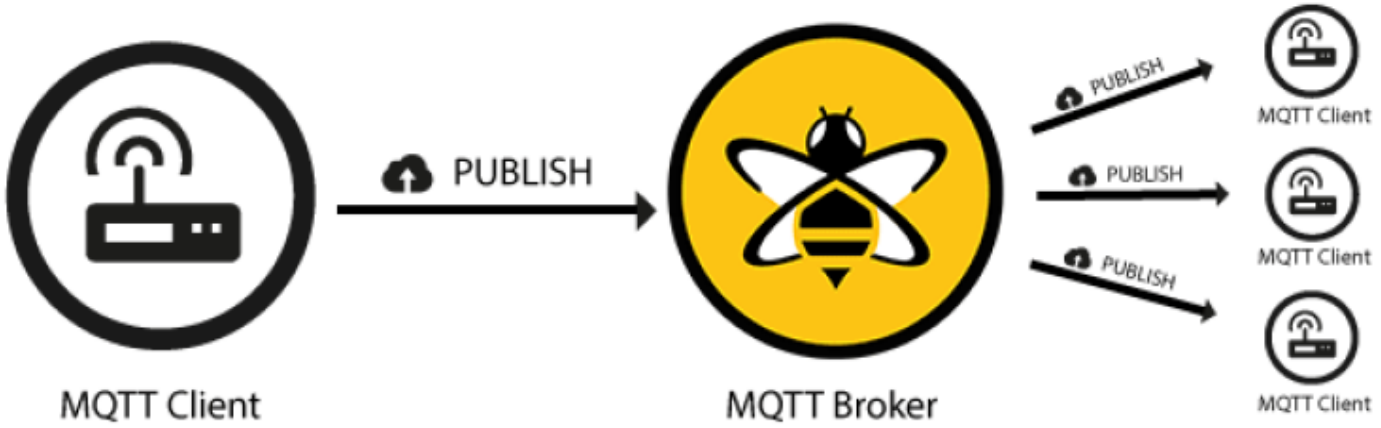
MQTT Publish, Subscribe & Unsubscribe

Publish

- An MQTT client can publish messages as soon as it connects to a broker.

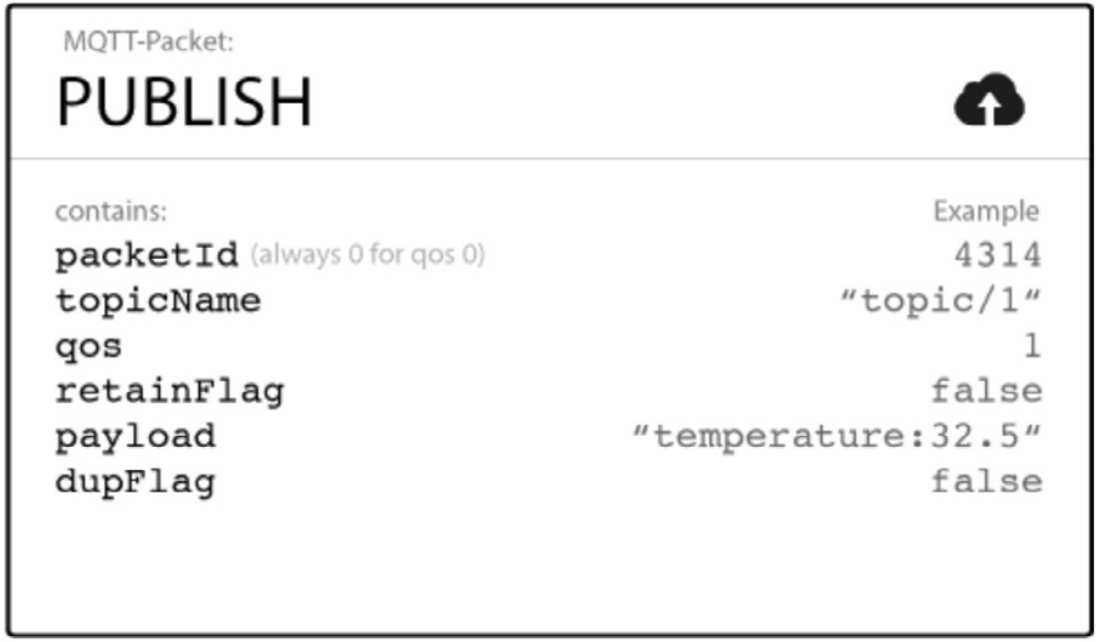
MQTT utilizes topic-based filtering of the messages on the broker. Each message must contain a topic that the broker can use to forward the message to interested clients. Typically, each message has a payload which contains the data to transmit in byte format.

The broker reads the message, acknowledges the message (according to the QoS Level), and processes the message.



Reference :official MQTT 3.1.1 specification

MQTT Publish



QoS: This number indicates the Quality of Service Level (QoS) of the message.
There are three levels:

0, 1, and 2. The service level determines what kind of guarantee a message has for reaching the intended recipient (client or broker).

Retain Flag: This flag defines whether the message is saved by the broker as the last known good value for a specified topic. When a new client subscribes to a topic, they receive the last message that is retained on that topic.

Reference :official MQTT 3.1.1 specification

MQTT Publish

Payload: This is the actual content of the message. MQTT is data-agnostic. It is possible to send images, text in any encoding, encrypted data, and virtually every data in binary.

Packet Identifier: The packet identifier uniquely identifies a message as it flows between the client and broker. The packet identifier is only relevant for QoS levels greater than zero. The client library and/or the broker is responsible for setting this internal MQTT identifier.

DUP Flag: The flag indicates that the message is a duplicate and was resent because the intended recipient (client or broker) did not acknowledge the original message.

Reference :official MQTT 3.1.1 specification

MQTT Subscribe

Subscribe

Publishing a message **doesn't make sense if no one ever receives** it. In other words, if there are no clients to subscribe to the topics of the messages.

To receive messages on topics of interest, the client sends a SUBSCRIBE message to the MQTT broker.

This subscribe message is very simple, it contains a unique packet identifier and a list of subscriptions.

MQTT Subscribe Attributes
Packet Identifier

The packet identifier uniquely identifies a message as it flows between the client and broker.The client library and the broker is responsible for setting this internal MQTT identifier

List of subscriptions
A SUBSCRIBE message can contain multiple subscriptions for a client

Reference :official MQTT 3.1.1 specification

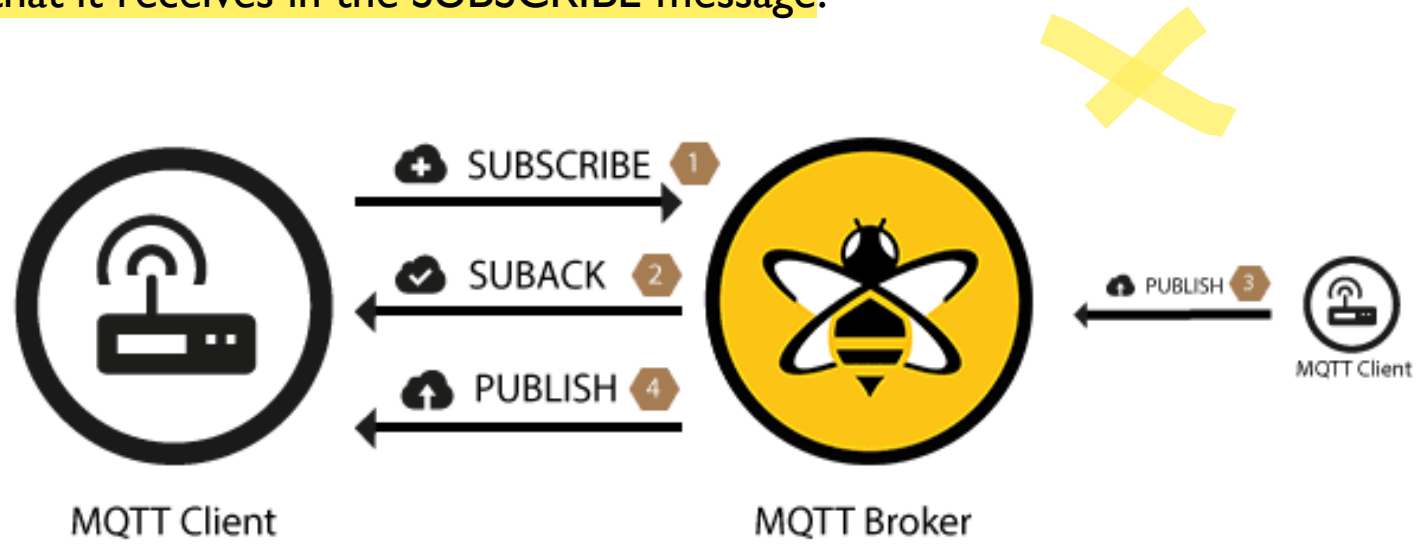
MQTT Suback

Suback

To confirm each subscription, the broker sends a SUBACK acknowledgement message to the client. This message contains the packet identifier of the original Subscribe message (to clearly identify the message) and a list of return codes.

Packet Identifier: The packet identifier is a unique identifier used to identify a message. It is the same as in the SUBSCRIBE message.

Return Code: The broker sends one return code for each topic/QoS-pair that it receives in the SUBSCRIBE message.



Reference :official MQTT 3.1.1 specification

MQTT Unsubscribe

Unsubscribe

The counterpart of the SUBSCRIBE message is the UNSUBSCRIBE message. This message deletes existing subscriptions of a client on the broker.

Packet Identifier: The packet identifier uniquely identifies a message as it flows between the client and broker. The client library and the broker is responsible for setting this internal MQTT identifier

Reference :official MQTT 3.1.1 specification

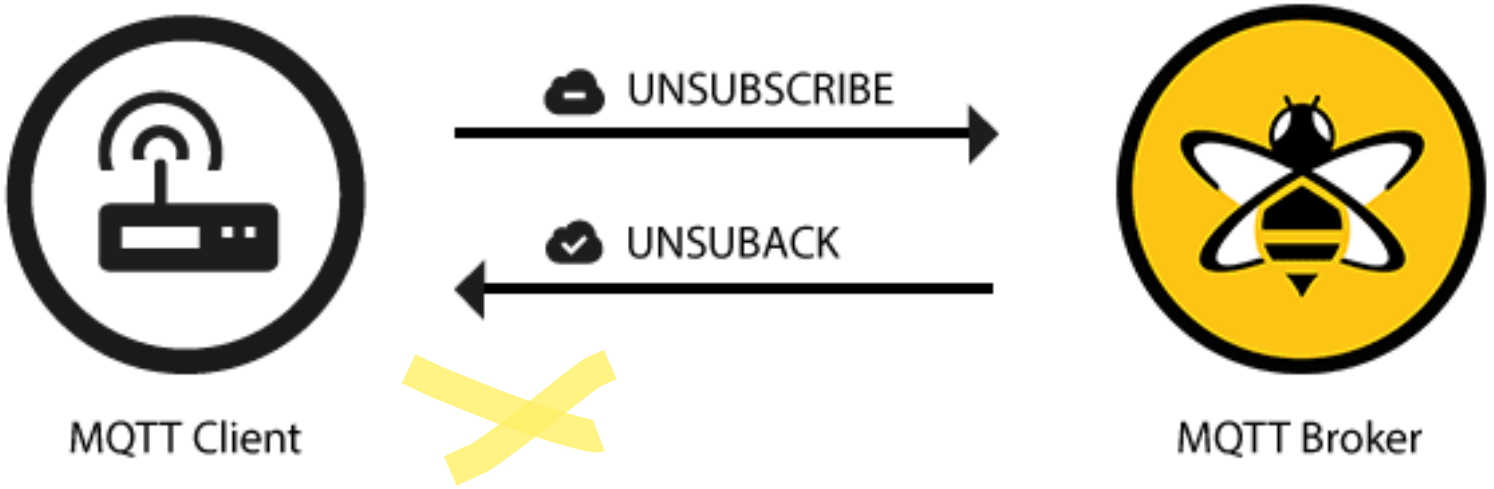
MQTT Unsuback

Unsuback

To confirm the unsubscribe, the broker sends an UNSUBACK acknowledgement message to the client. This message contains only the packet identifier of the original UNSUBSCRIBE message (to clearly identify the message).

Packet Identifier: The packet identifier uniquely identifies the message. As already mentioned,this is the same packet identifier that is in the UNSUBSCRIBE message.

After receiving the UNSUBACK from the broker, the client can assume that the subscriptions in the UNSUBSCRIBE message are deleted



MQTT Topics

Topics

In MQTT, the word **topic refers to an UTF-8 string that the broker uses to filter messages for each connected client.** The topic consists of one or more topic levels. Each topic level is separated by a forward slash (topic level separator).

Wildcards

When a client subscribes to a topic, it can subscribe to the exact topic of a published message or it can use wildcards to subscribe to multiple topics simultaneously. A wildcard can only be used to subscribe to topics, not to publish message. There are two different kinds of wildcards: single-level and multi-level.

MQTT Topics

Wilidcard

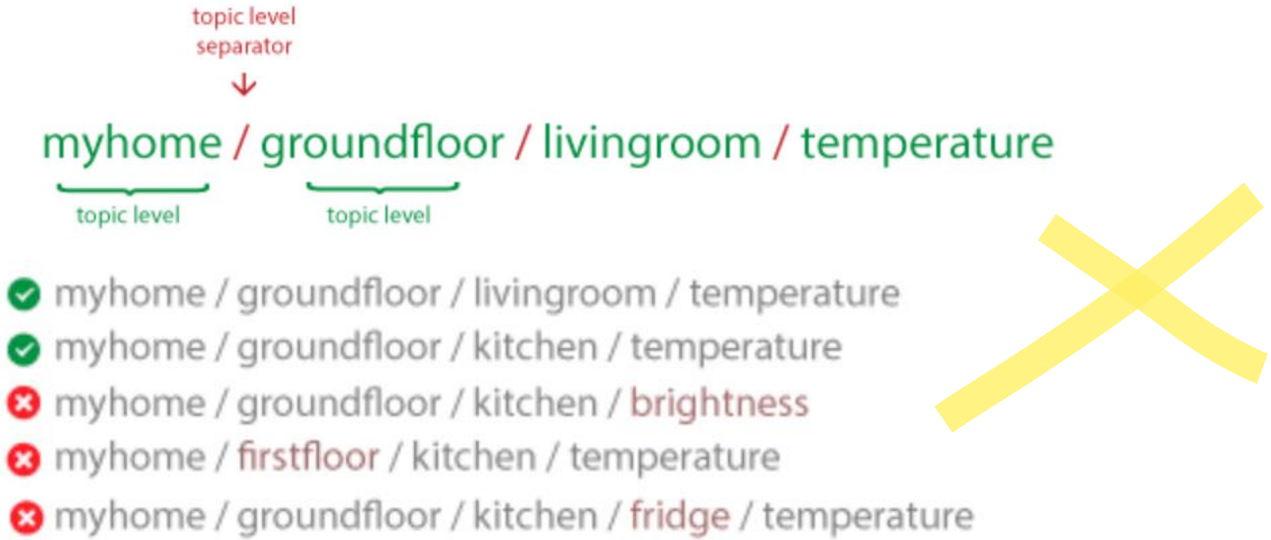
Single Level: +

As the name suggests, a single-level wildcard replaces one topic level. The plus symbol represents a single-level wildcard in a topic.

Any topic matches a topic with single-level wildcard if it contains an arbitrary string instead of the wildcard.

For example a subscription to

myhome/groundfloor/+/temperature can produce the following re:



Wildcards

single-level wildcard

myhome / groundfloor / + / temperature

only one level

multi-level wildcard

myhome / groundfloor / #

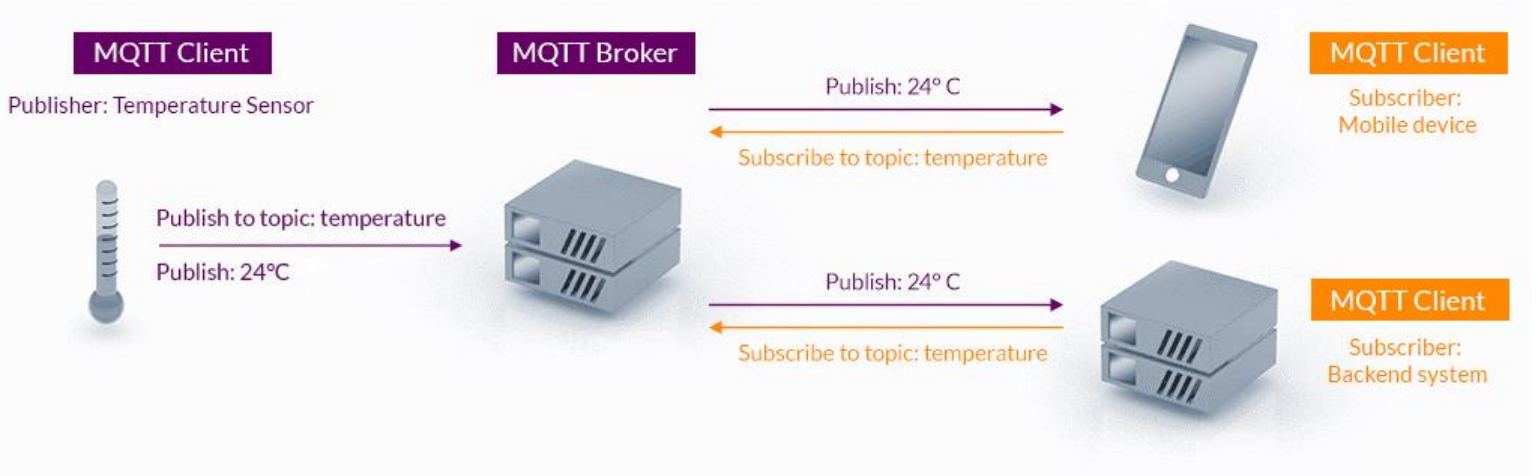
only at the end

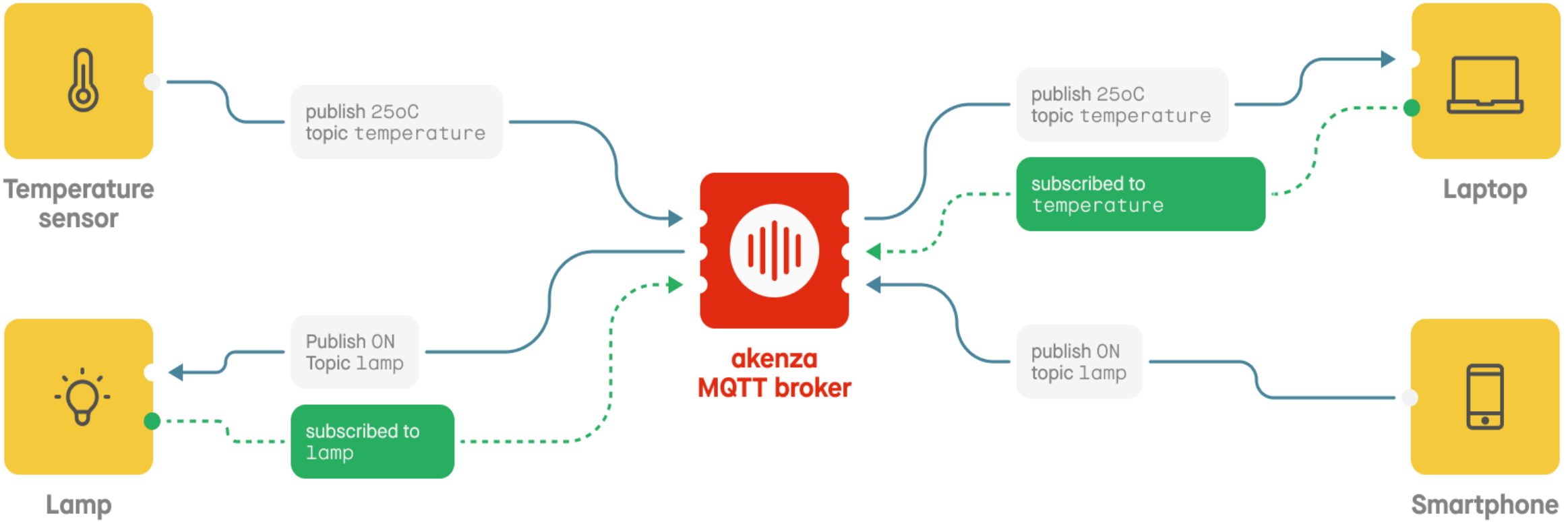
multiple topic levels

- ✓ myhome / groundfloor / livingroom / temperature
- ✓ myhome / groundfloor / kitchen / temperature
- ✗ myhome / groundfloor / kitchen / brightness
- ✗ myhome / firstfloor / kitchen / temperature
- ✗ myhome / groundfloor / kitchen / fridge / temperature

- ✓ myhome / groundfloor / livingroom / temperature
- ✓ myhome / groundfloor / kitchen / temperature
- ✓ myhome / groundfloor / kitchen / brightness
- ✗ myhome / firstfloor / kitchen / temperature

MQTT Example





MQTT Example

Types of MQTT Brokers?

Generally, there are two types of brokers:

1. Managed Brokers
2. Self-Hosted Brokers

Managed Brokers

Managed brokers don't require you to set up anything on your server to enable MQTT communication. Managed broker services let you use their hosted brokers for your system. AWS IoT Core is a good example of a managed MQTT Broker.

Self-Hosted Brokers

As the name implies, self-hosted MQTT brokers require you to install the broker on your own VPS or server with a static IP. The installation process is not difficult but managing, securing, and scaling the brokers requires in-depth knowledge of the system.

There are several open-source implementations of MQTT brokers including mosquitto and hivemq.

List of Popular MQTT Brokers

There are plenty of viable managed and self-hosted MQTT brokers available. Here is an overview of some of the most popular options.

Type	Address and Port	WebSocket Support	SSL Support	Scalability	
AWS IoT Core MQTT	Managed	Dynamically assigned	Yes, port=443	Yes, port=8883	Auto Scale
Mosquitto	Self-hosted and Managed	test.mosquitto.org	Yes, port=8081,8080	Yes, port=8883,8884	Scale horizontally making bridges
Mosca/Aedes	Self-hosted and Managed	test.mosca.io	Yes, port=3000	Yes, port=8883	Horizontally and vertically
HiveMQ	Self-hosted and Managed	broker.hivemq.com	Yes, port=8000,443	Yes, port=8883	Yes, Broker-Clustering
VerneMQ	Self-hosted and Managed	self-assigned	Yes, port=9001,9002	Yes, port=8883	Both horizontal and vertical
Azure IoT Hub	Managed	Dynamically assigned	Yes, port=443	Yes, port=8883	Auto Scale
EMQ X	Self-hosted	Self-assigned	Yes, port=8083, 8084	Yes, port=8883	Clustering
ejabberd	Self-hosted and managed	Self-assigned and dynamically allocated	Yes	Yes	Clustering

MQTT QoS

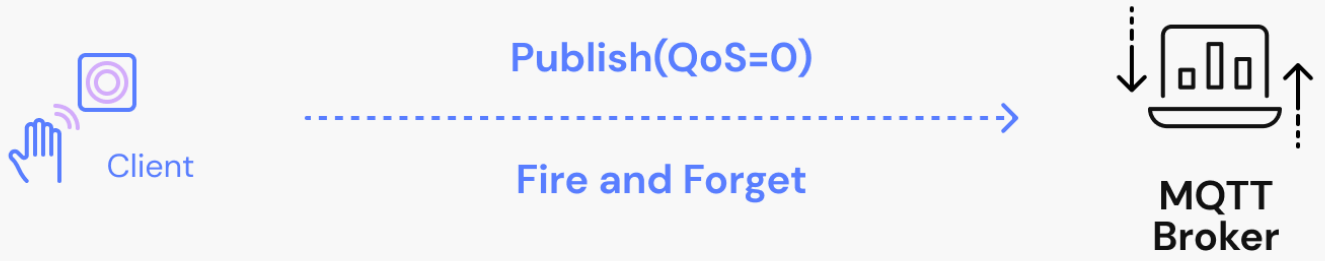
QoS or Quality of Service is a predefined agreement between publisher and subscriber of a topic.

QoS is used in MQTT to set the message delivery guarantee levels. There are 3 QoS levels in MQTT:

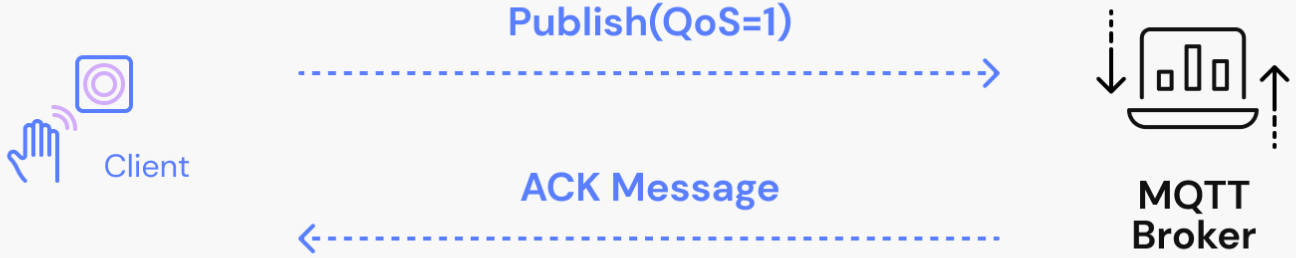
1. QoS 0 (At most once)
2. QoS 1 (At least once)
3. QoS 2 (Exactly once)

MQTT QoS

1. QoS 0 - At most Once



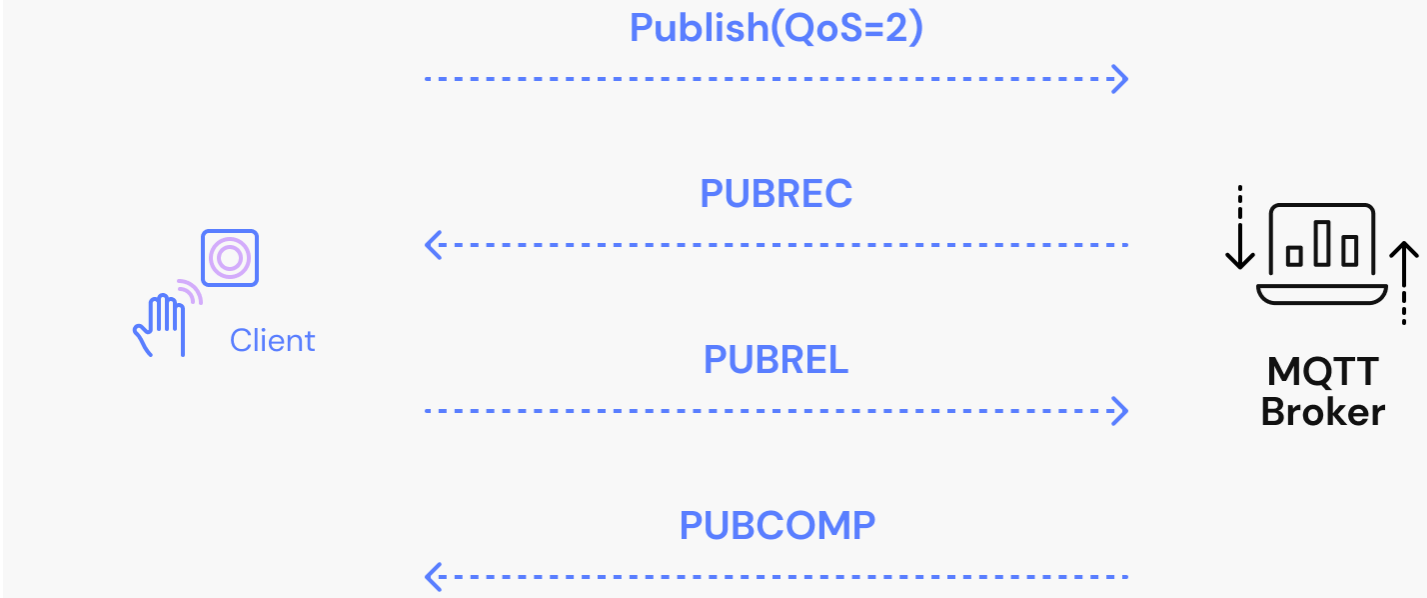
2. QoS 1 - At least once



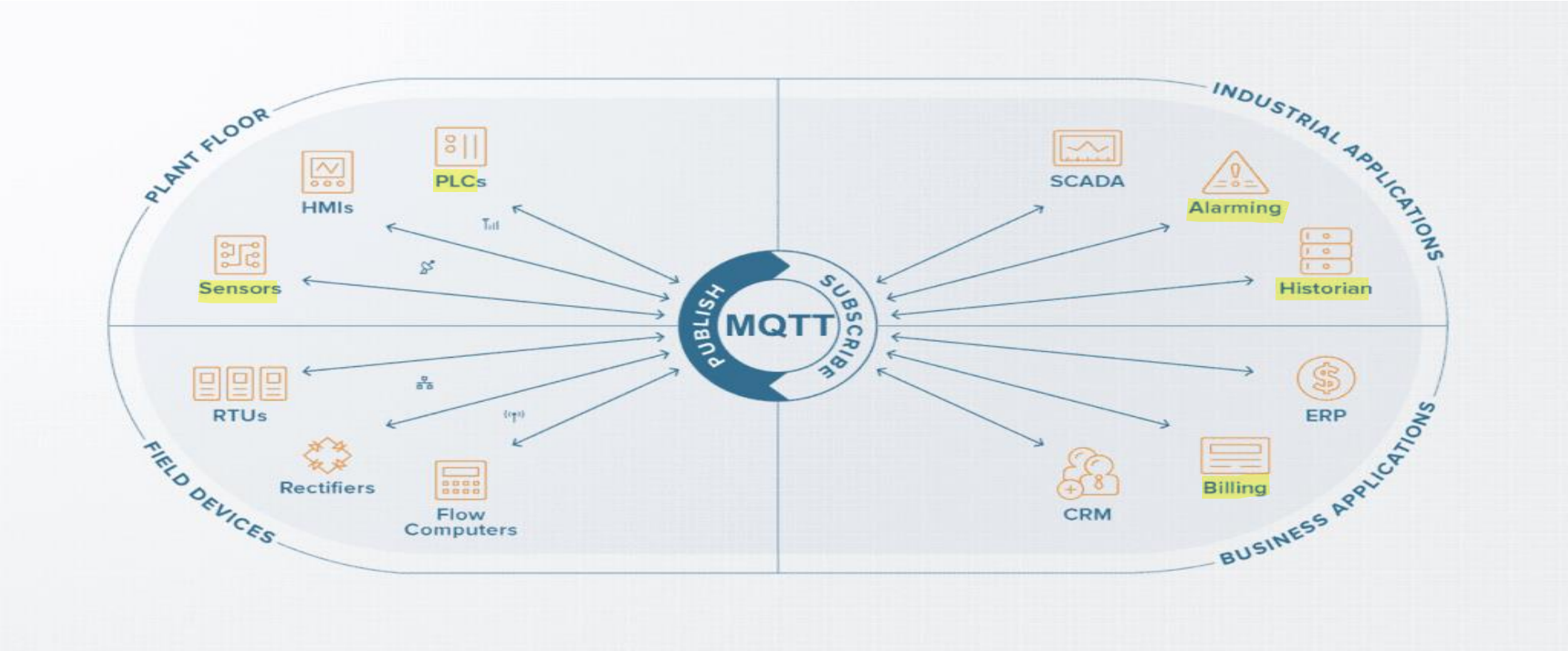
The QoS1 level requires the broker to guarantee that the message is sent at least one time to the client using an acknowledgment packet named PUBACK.

MQTT QoS

3. QoS 0 – Exactly once

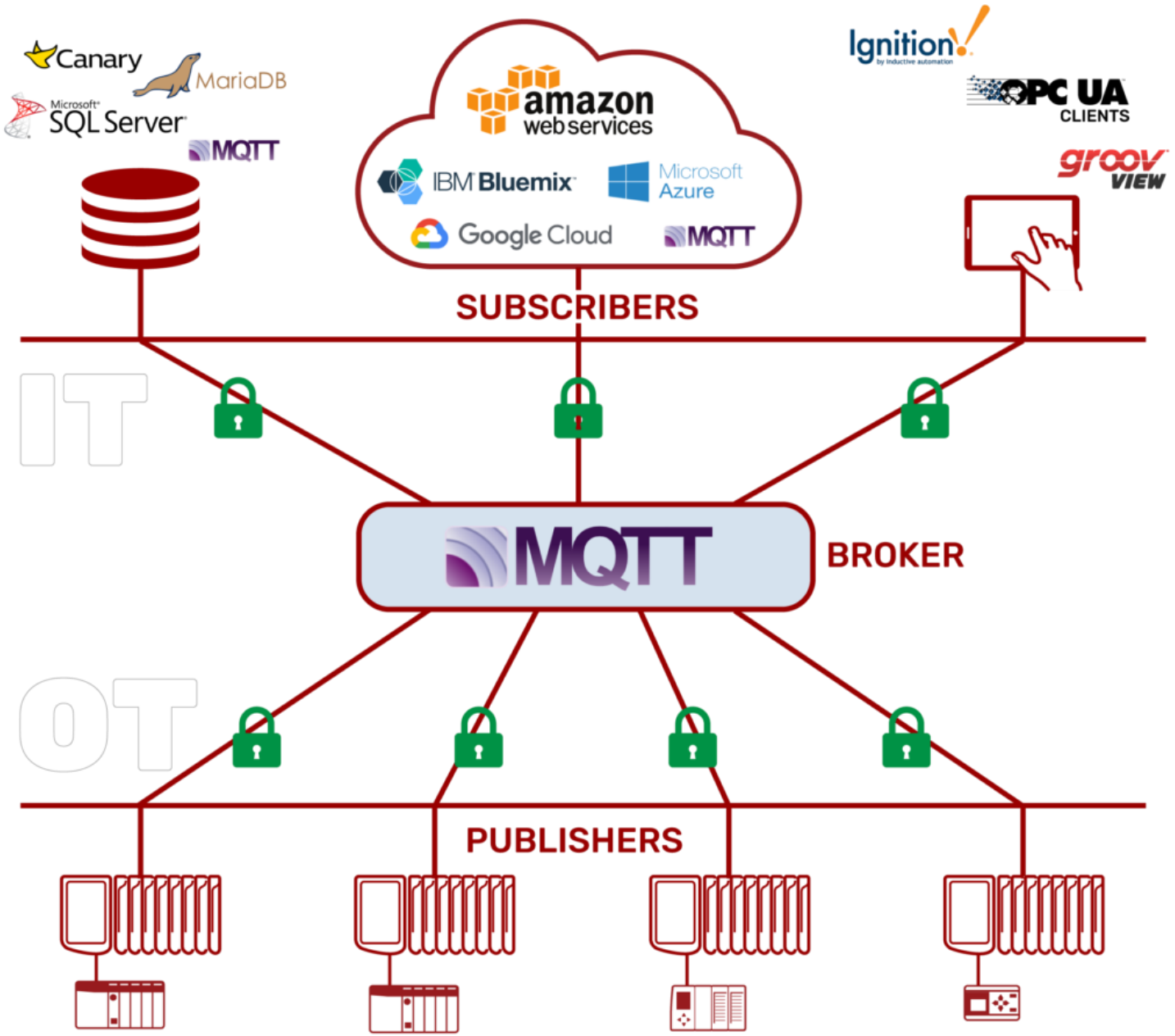


The QoS 2 level guarantees that the broker will send the published message only once. It uses multilevel acknowledgment as shown in the figure above. PUBREC is the first acknowledgment from the broker, PUBREL is the acknowledgment from the publishing client that PUBREC is received while PUBCOMP is the second and final acknowledgment from the broker.



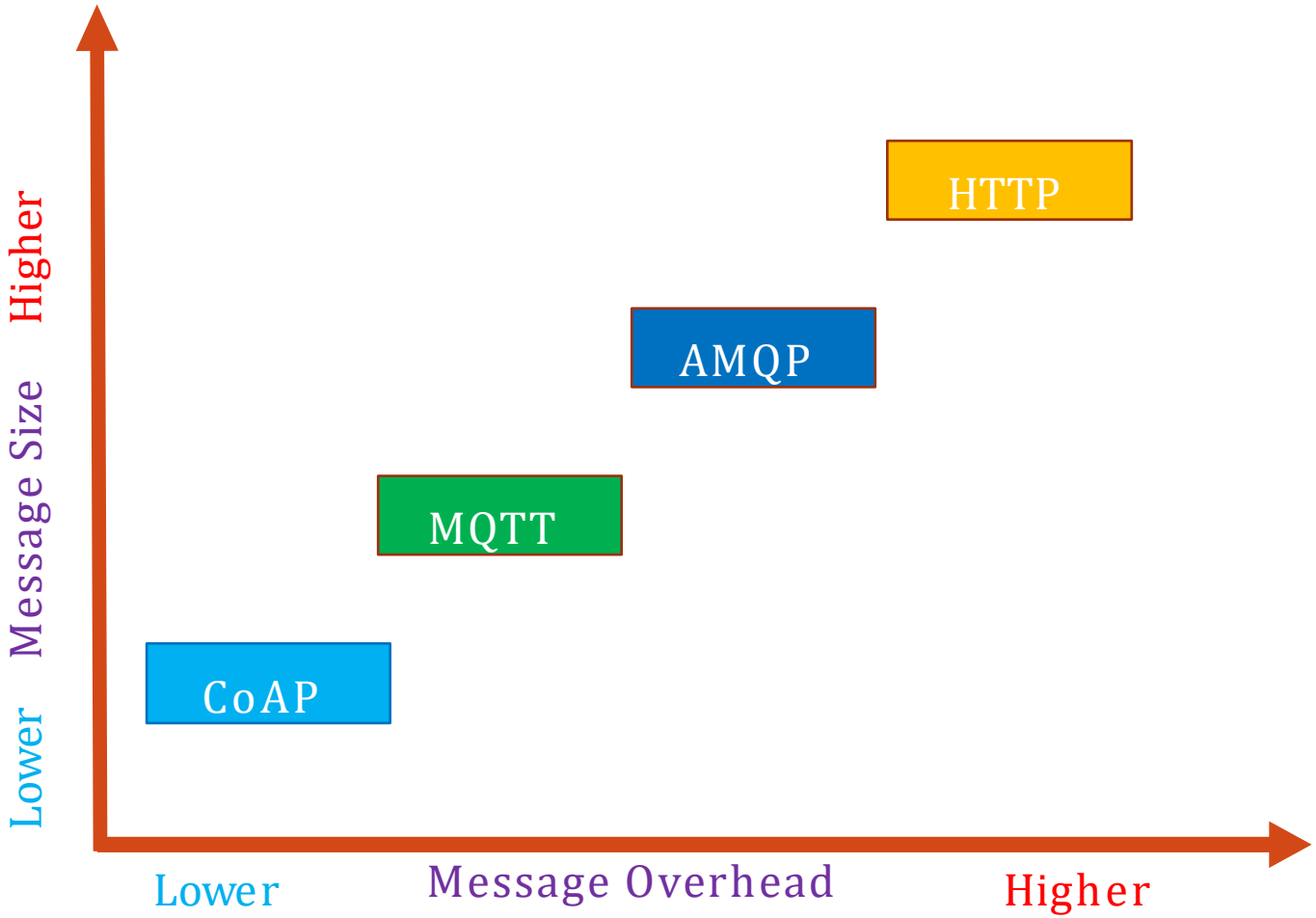
MQTT Applications

MQTT Applications

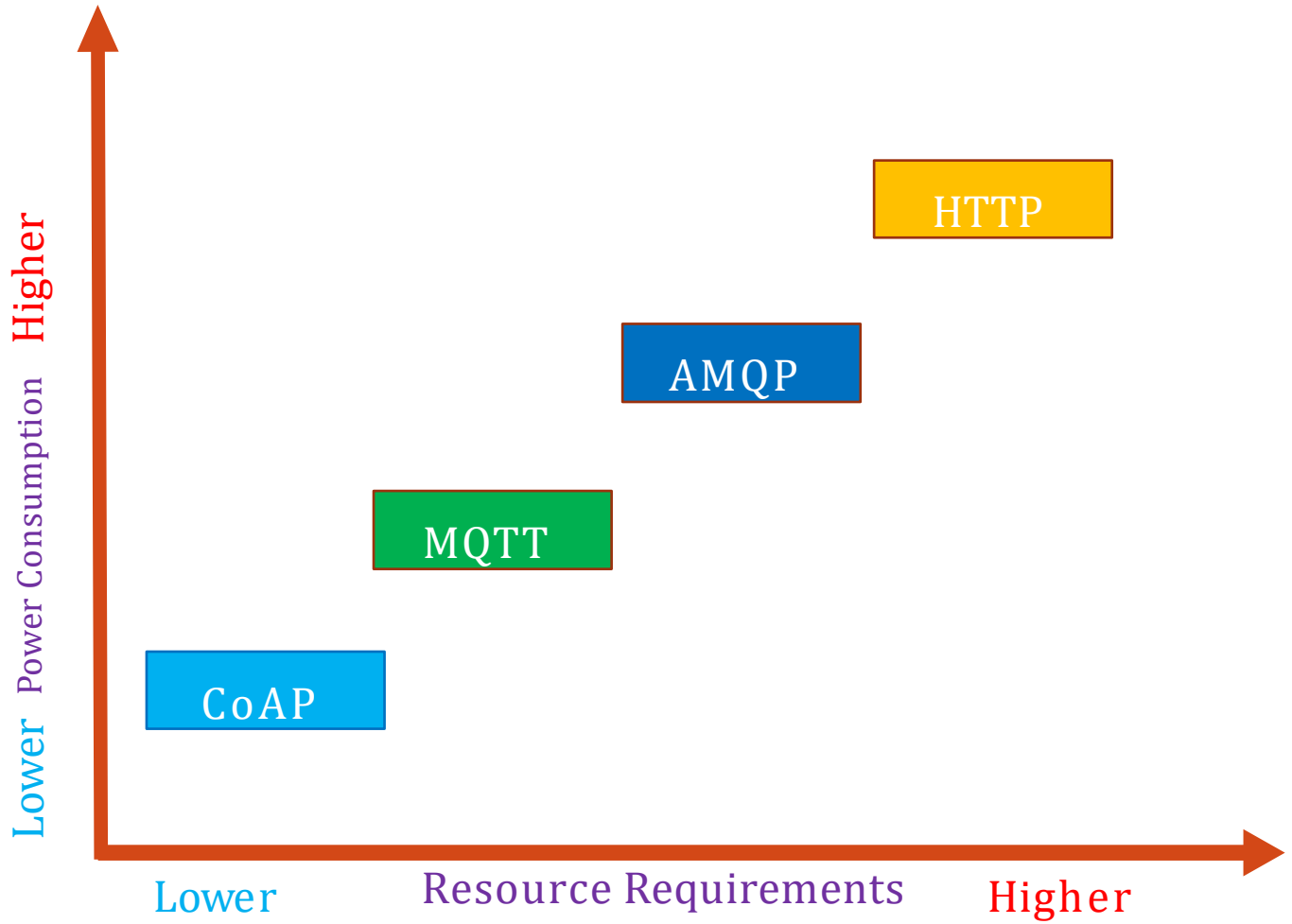


	MQTT	COAP	AMQP	HTTP
Year	1999	2010	2003	1997
Architecture	Client/Broker	Client/Server Client/Broker	Client/Server Client/Broker	Client/Server
Model	Publish/Subscribe	Publish/Subscribe Request/Response	Publish/Subscribe Request/Response	Request/Response
Header Size	2 Byte	4 Byte	8 Byte	Undefined
Message Size	Small and Undefined	Small & Undefined	Negotiable & Undefined	Large & Undefined
QoS	QoS 0 - At most once(Fire-and-Forget) QoS 1 - At least once QoS 2 - Exactly once	Confirmable Message(similar to At most once) or Non-confirmable Message (similar to At least once)	Settle Format (similar to At most once) or Unsettle Format (similar to At least once)	Limited (via Transport Protocol - TCP)
Standards	OASIS, Eclipse Foundation	IETF, Eclipse Foundation	OASIS, ISO/IEC	IETF and W3C
Transport Protocol	TCP	UDP, SCTP	TCP, SCTP	TCP
Security	TLS/SSL	DTLS, IPSec	TLS/SSL, IPSec, SASL	TLS/SSL
Default Port	1883/ 8883 (TLS/SSL)	5683 (UDP Port)/ 5684(DLTS)	5672/5671 (TLS/SSL)	80/ 443 (TLS/SSL)
Licensing Model	Open Source	Open Source	Open Source	Free

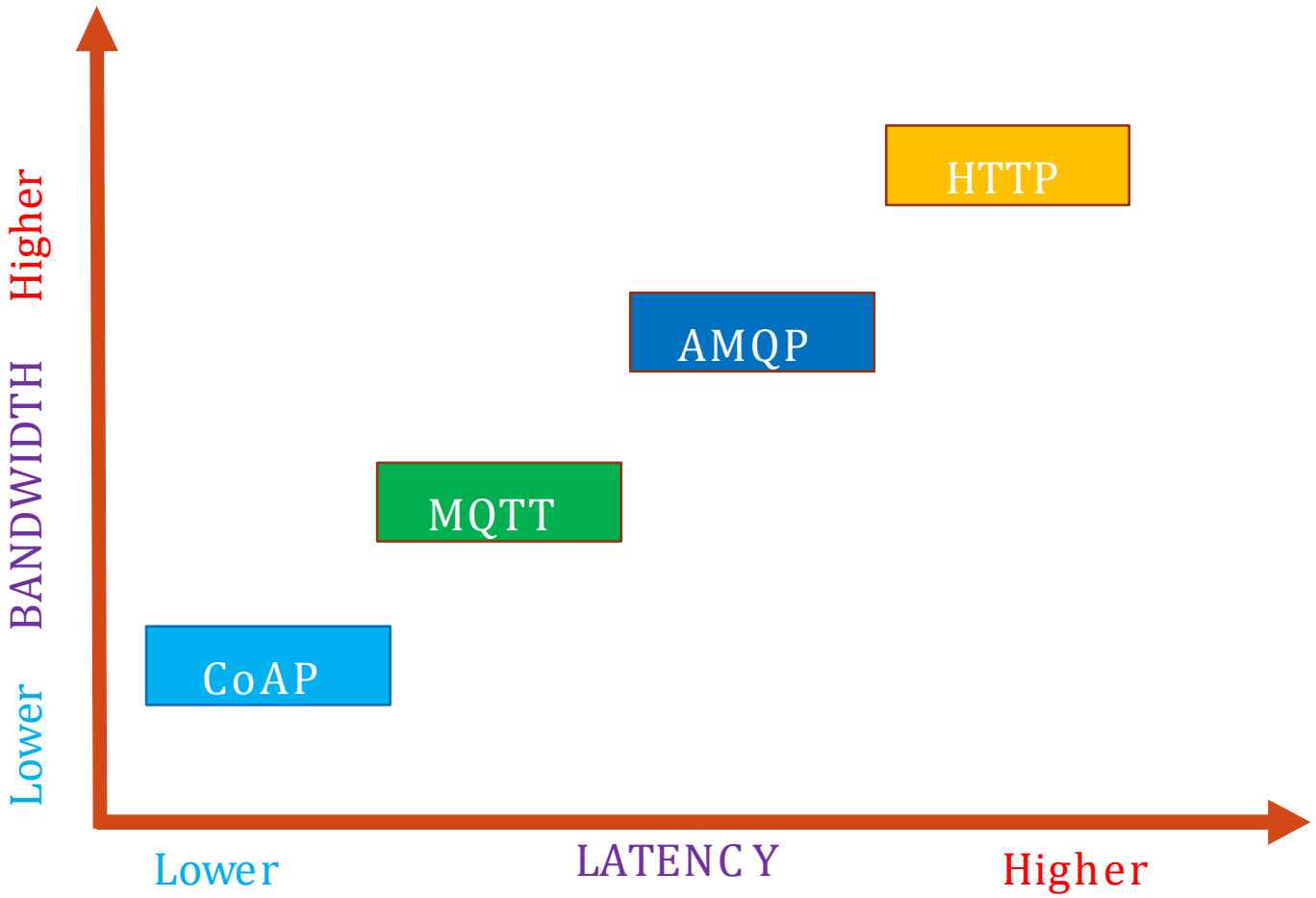
Massage Size Vs Massage Overhead



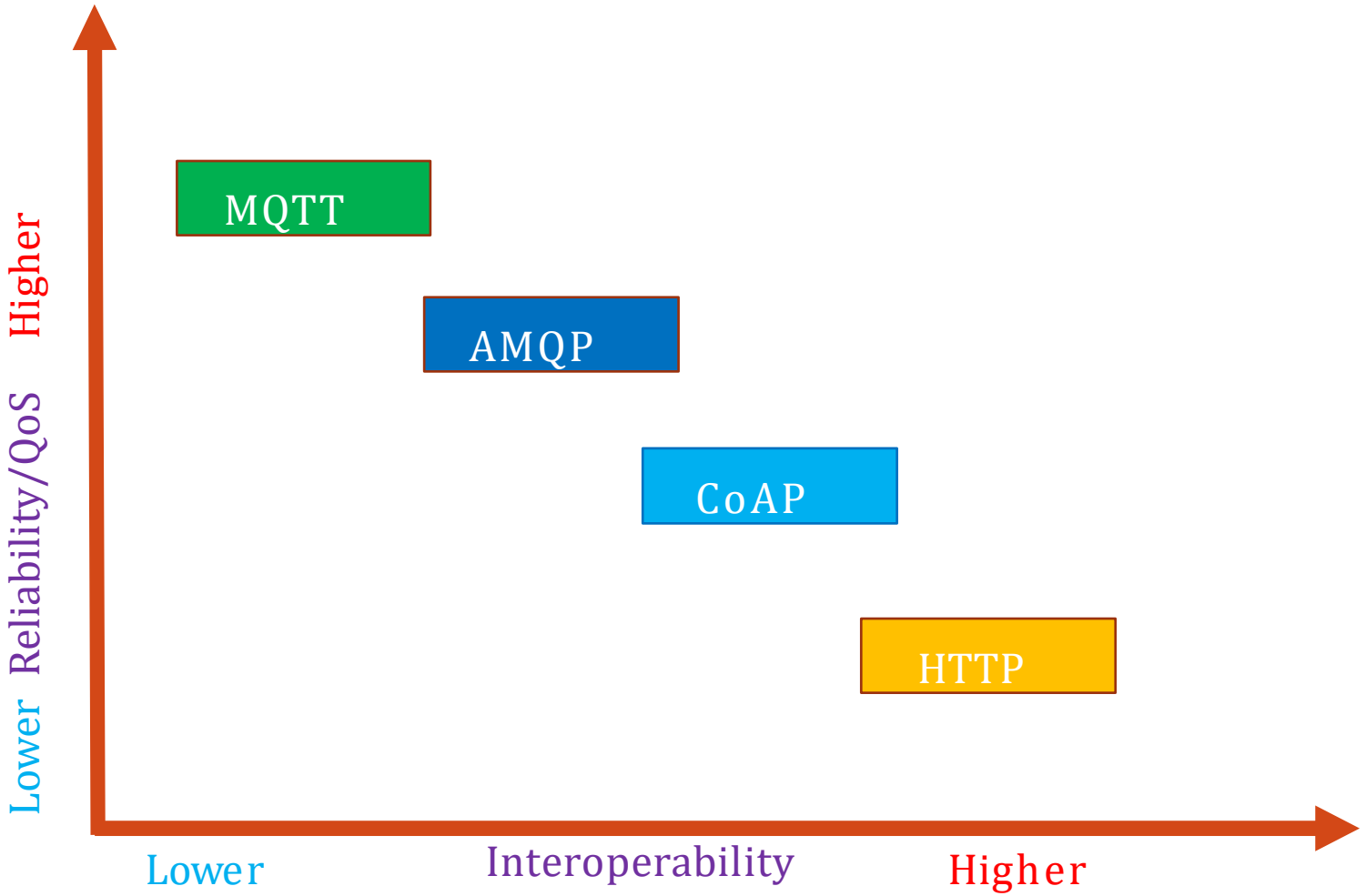
Power Consumption Vs Resource Requirement



Bandwidth Vs Latancy



Reliability/QoS Vs Intropersability



Q & A

What is MQTT in IoT?

MQTT is the standard protocol for messaging and data exchange for the Internet of Things. The protocol uses a publish/subscribe architecture. The technology provides a scalable and cost-effective way to connect devices over the Internet. It is able to deliver data over the Internet in near real-time and with guarantees of delivery. MQTT is designed for IoT devices-lightweight, which enables low-cost device communication.

Who uses MQTT?

MQTT is used by many major companies, especially in the automotive, industry 4.0, transport, and entertainment sectors.

MQTT is used for data exchange between constrained devices and server applications. It keeps bandwidth requirements to an absolute minimum, handles unreliable networks, requires little implementation effort for developers, and is, therefore, ideal for machine-to-machine (M2M) communication.

How does MQTT work?

MQTT is the standard protocol for messaging and data exchange for the Internet of Things.

The protocol uses a publish/subscribe architecture. The technology provides a scalable and cost-effective way to connect devices over the Internet. It is able to deliver data over the Internet in near real-time and with guarantees of delivery. MQTT is designed for IoT devices - lightweight, which enables low-cost device communication.

What is an MQTT client?

An MQTT client is any device (from a microcontroller to a full-fledged server) that operates an MQTT library and connects to an MQTT broker over a network. Each MQTT client can be both publisher and/or subscriber.

What does an MQTT broker do?

An MQTT broker is at the heart of any MQTT deployment. Depending on the implementation, a broker can handle up to millions of concurrently connected MQTT clients.

The broker is responsible for receiving all messages, filtering the messages, determining who is interested in each message, and sending the message to these subscribed clients.

Thank You

Internet of Things

IE 4030

Uditha Dharmakeerthi

IIOT Overview

- IIoT is a subset of IoT
- Introduction to Industrial Internet of Things (IIOT)
- Cyber Physical system

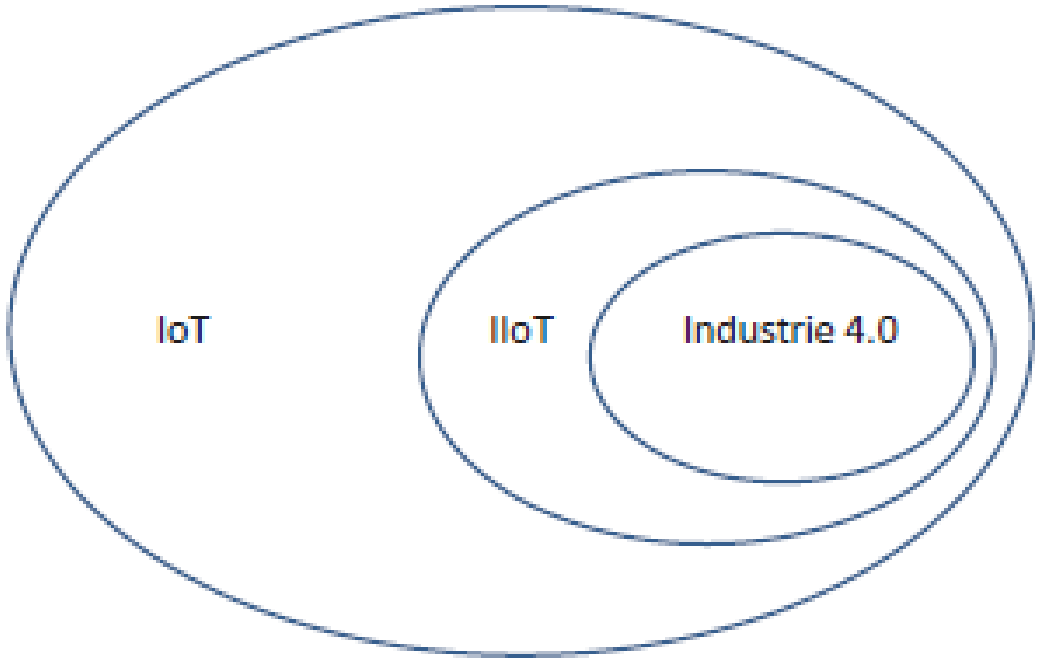
- Understanding IIoT
  - Smart factories
  - The Industry 4.0 strategy
  - Industrial Internet
  - Factories of the Future

- Notable Developments & Initiatives

Introduction to IIoT

Industrial internet of things, is part of general IoT evaluation. In the industrial environment, the effort for smart factories, the Industry 4.0 strategy , the Industrial Internet, and the European initiative for the Factories of the Future (FoF) have initiated the adoption of IoT in industry with the goals of increasing flexibility and productivity, while reducing production cost. This developing concept is termed as Industrial
IoT (IIoT)

IIoT is a subset of IOT



Cyber Physical System

CPS are physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core

Or

CSP are integrations of computation with physical processes. Embedded computers and networks monitor and control the physical processes, usually with feedback loops were physical processes affect computations and vice versa

Cyber Physical System

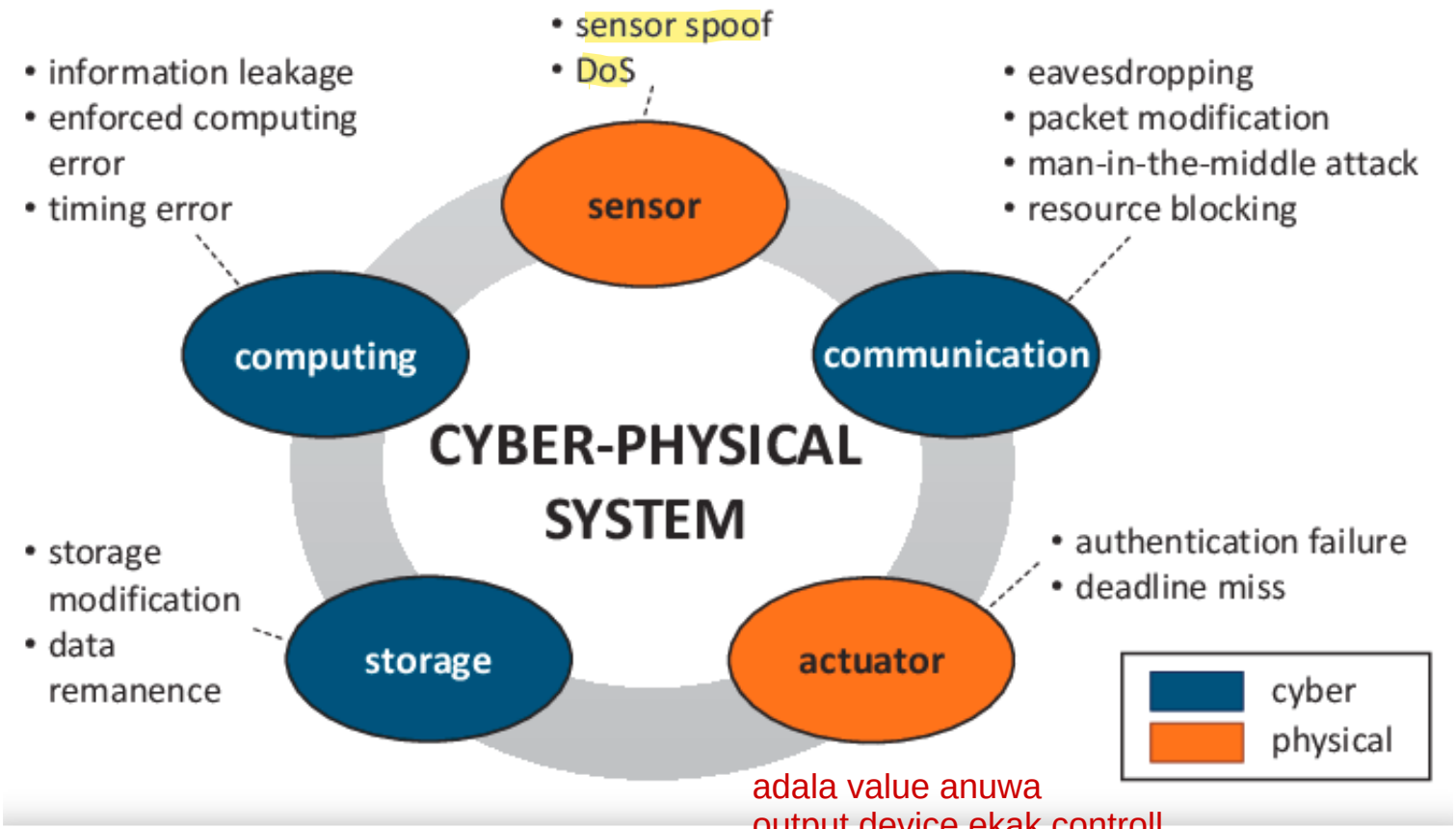
Cyber physical system (CPS)
– involves transdisciplinary approaches of
- Cybernetics
- Mechatronics
- Design and process science/embedded systems

– Higher levels of automation that enables mass customized manufacturing and production of goods and services

– Flexibility in CPS by the easily programmable, configurable, and controllable manufacturing lines.

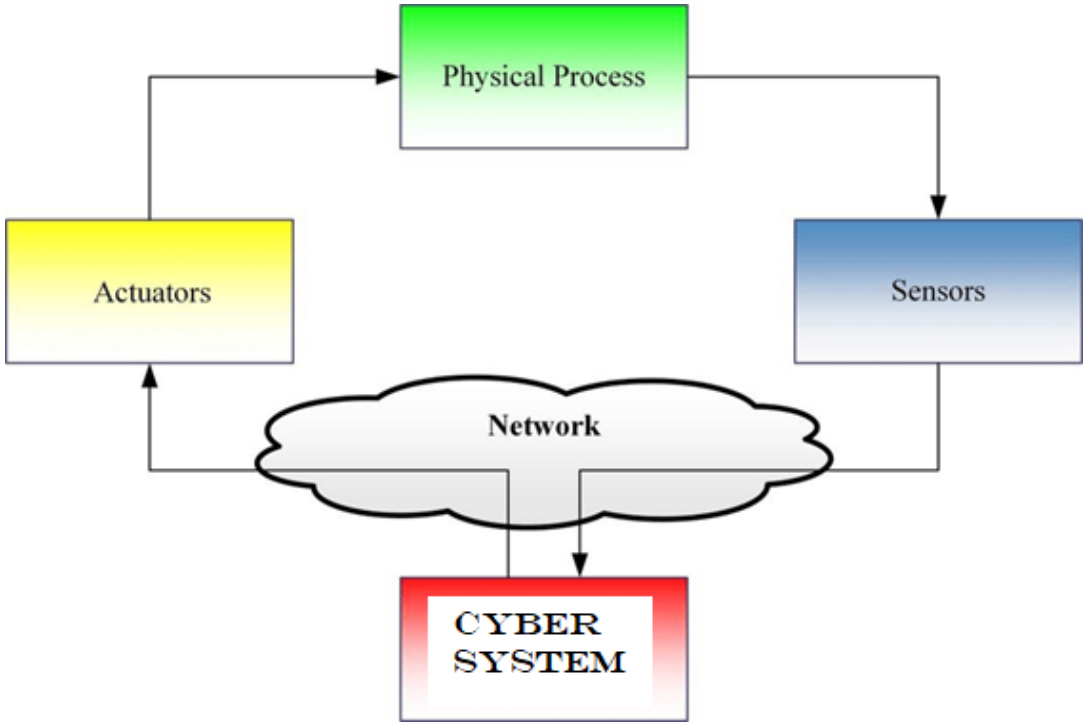
Cyber Physical System

cllect data



adala value anuwa
output device ekak controll
karana eka

Cyber Physical System



CPS is Integrations of computation with physical processes.

- Combination of physical system and Cyber system
- A physical system comprises of Actuators and sensors
- Cyber system contains on networking/communication, computing, data storage & processing,

Basic structure of IIOT

The IIoT implementation into Industrial environment are associated to the followings

- Smart factories
- The Industry 4.0 strategy
- Industrial Internet
- Factories of the Future

Smart factories

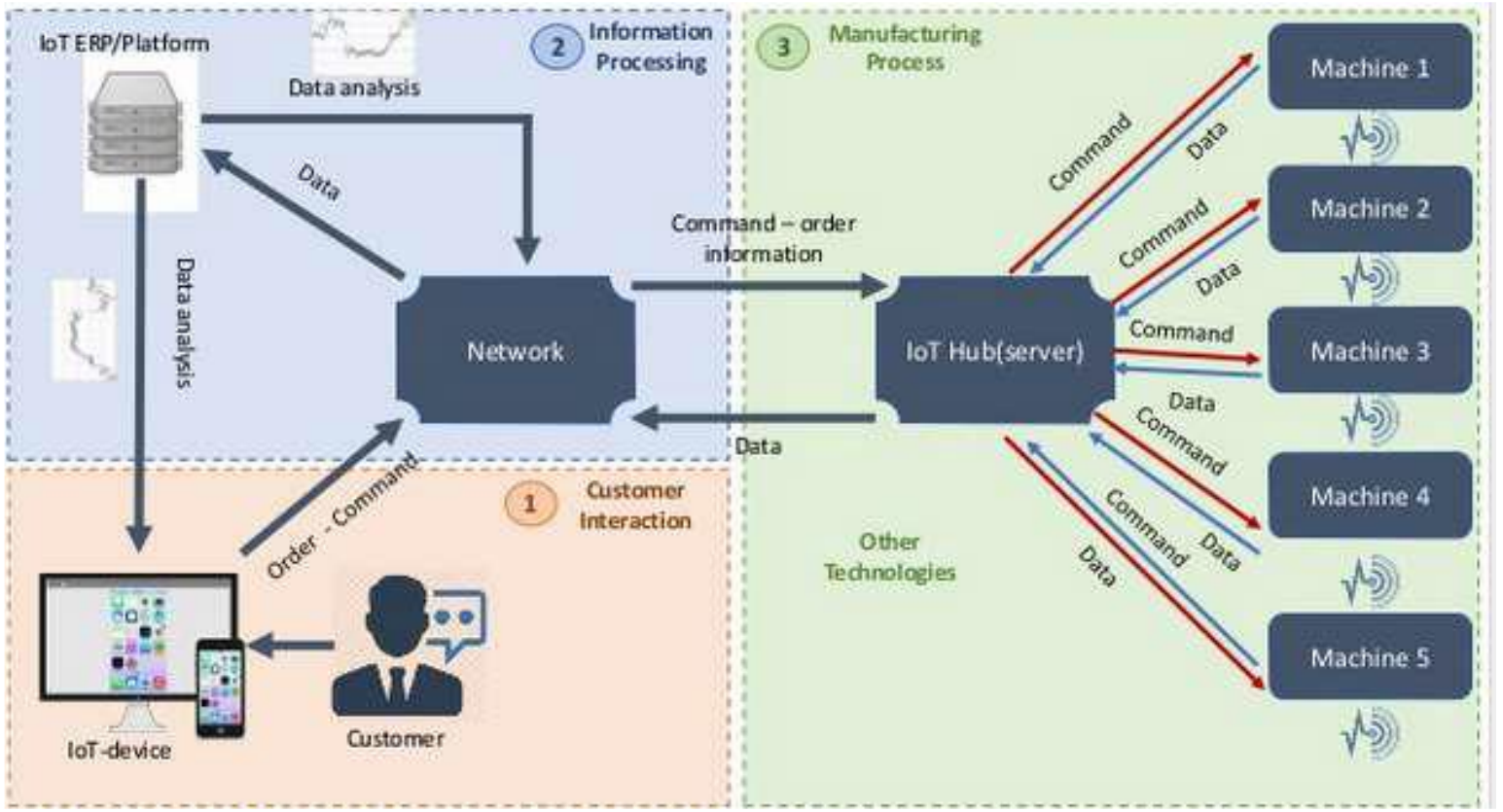
- The smart factories are based on cyber physical system.

fully internet haraha computerization wena system ekak thama smart factory kiyanne

- Do smart manufacturing

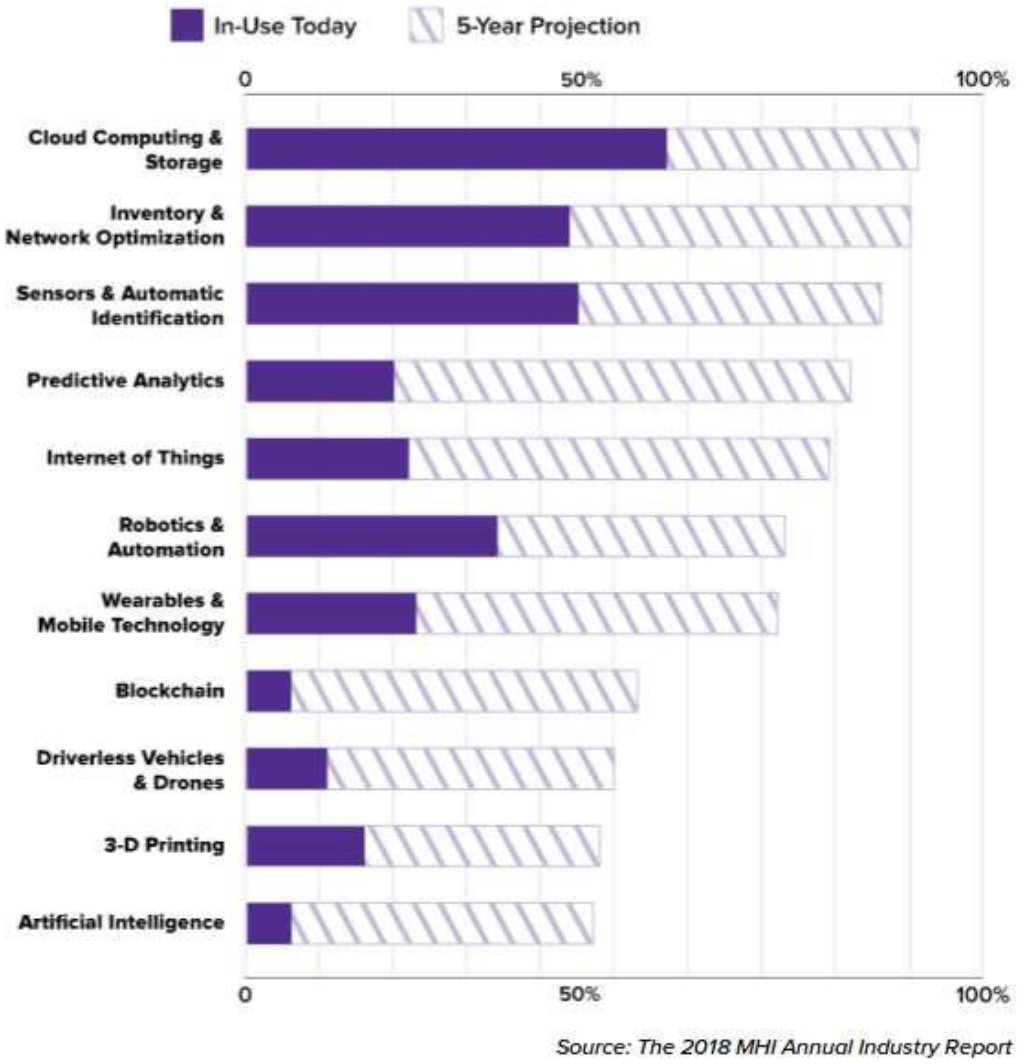
- The factories embodies the goals of industry 4.0 strategy to large scale are known as smart factories.

Smart factories



Smart Manufacturing

Rates of adoption for 11 Emerging technologies in smart manufacturing



Source: The 2018 MHI Annual Industry Report

The Industry 4.0 strategy

Initiative in Germany that target to bring IoT technology to the manufacturing and production sector

- Initiated because of
  - High quality
  - Low cost production

- Cyber physical system (CPS)

  - Higher levels of automation that enables mass customized manufacturing and production of goods and services

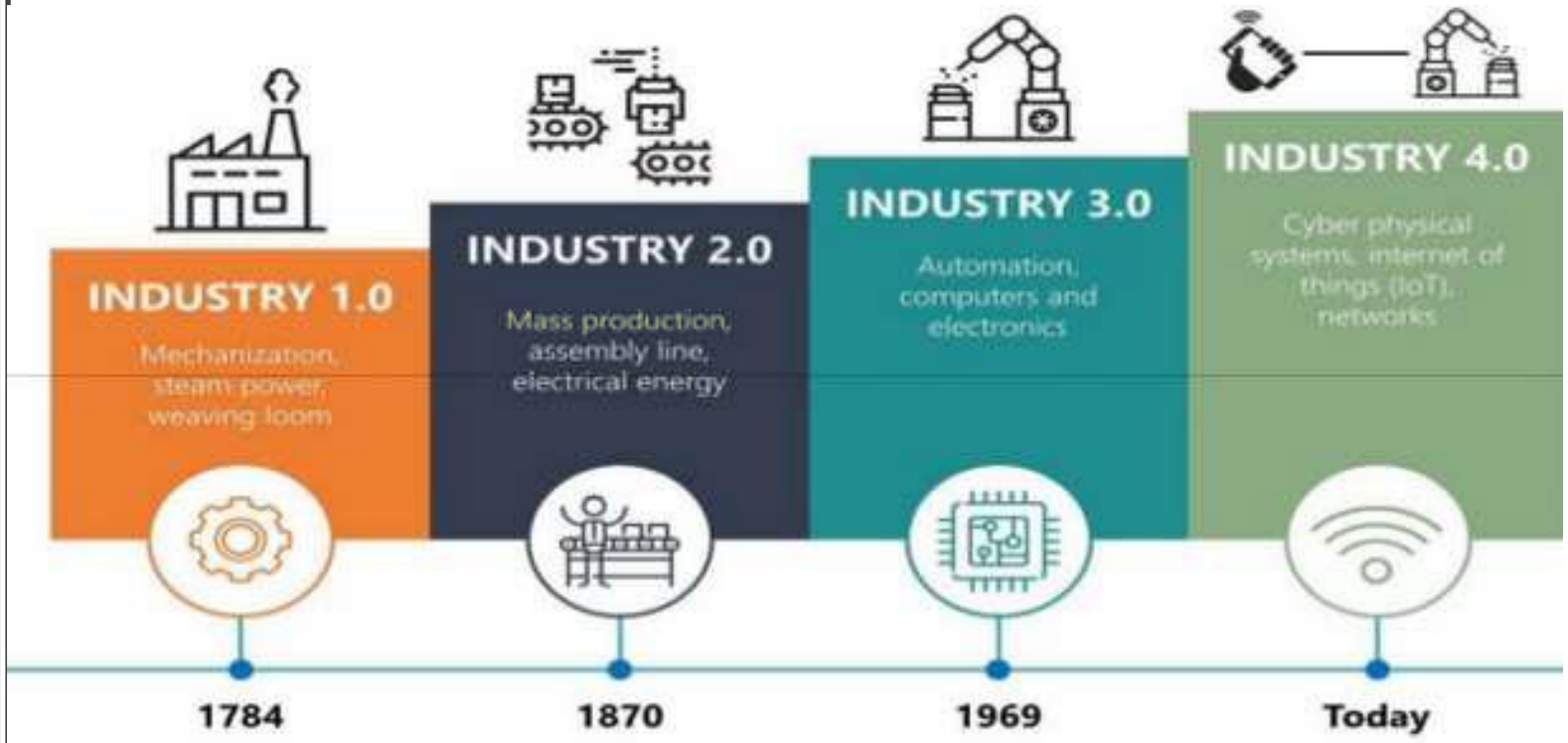
  - Flexibility in CPS by the easily programmable, configurable, and controllable manufacturing lines.

The Industry 4.0 strategy

is based on
– The widespread deployment
– Use of computational and communication resources.

- The Industrie 1.0 ☐ Steam Engine
- The Industrie 2.0 ☐ Electricity
- The Industrie 3.0 ☐ Computer, Automation, PLC, SCADA
- The Industrie 4.0 ☐ significant advances in high performance, low-power processors, memories, and communication components that enable efficient processing and networking.

The Industry 4.0 strategy



The Industry 4.0 strategy

Smart phones

– Thousands of applications specially for transportation and communication

– Mobile banking

– Health monitoring

• **Smart television**

– Customized TV control on various types of channel

– Management of internet gamming

– Various entertainment services

• **Smart home appliance**

– Home device management

– Reduced operational cost and increase living quality

Industrial internet

Heterogeneous devices are required to communicate

- In normal mode we used IPv6 for LAN and WAN

- When we talk about
  - Technologies of machine-to-machine communication,
  - SCADA, HMI, Industrial data analytics,
  - Cyber-security system
  - Supervisory controls
  - Data acquisition system (SCADA)

- Connection through wireless communication

Industrial internet

- Industrial and IP-enabled low-power wireless networking technologies needs low-power wireless solutions with

– strict reliability
– power consumption requirements of industrial applications.
– Easy to integrate in industrial environment

• These solutions are based on Time- Synchronized Channel Hopping, a medium access control technique at the heart of industrial standards such as

– WirelessHART
– ISA100.11a
– IEEE802.15.4e

Operational Technology (OT vs IT)

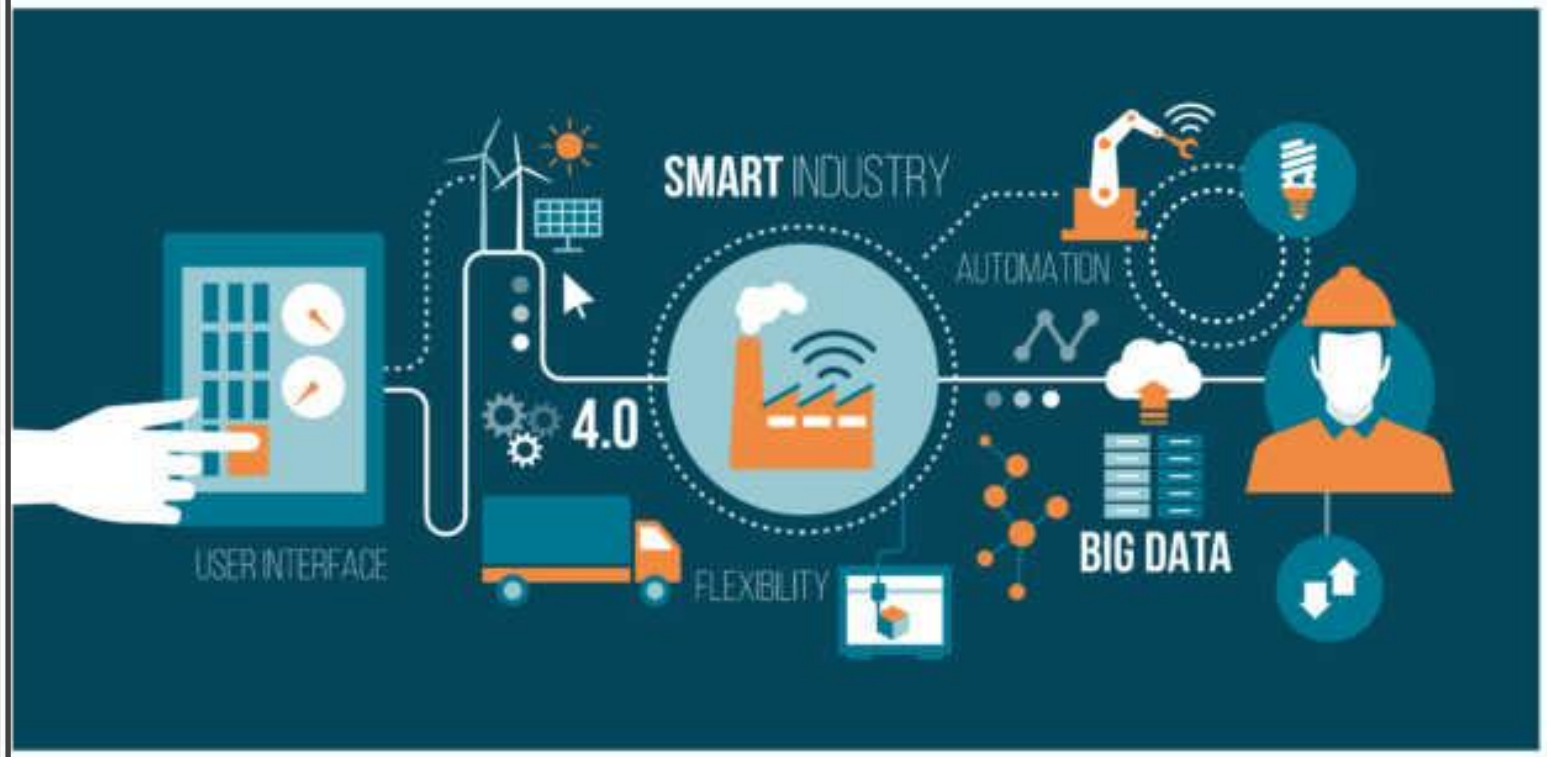
- PLC +SCADA → relationship → industrial network →

known in IoT as a OT (operational technology)

- OT has traditionally evolved independently from the typical IT technology, because OT addresses the strong requirements such as
- – continuous operation,
- – safety, internet access nathi nisa data wala safety eka wedi
- – real-time operation, etc.

- Challenge: The integration of these OT systems with
- the traditional enterprise IT systems at many levels,

Operational Technology (OT vs IT)



INTERNET OF THINGS

Factories of the future

The future factories are commonly divided into three kinds

A. Smart factories

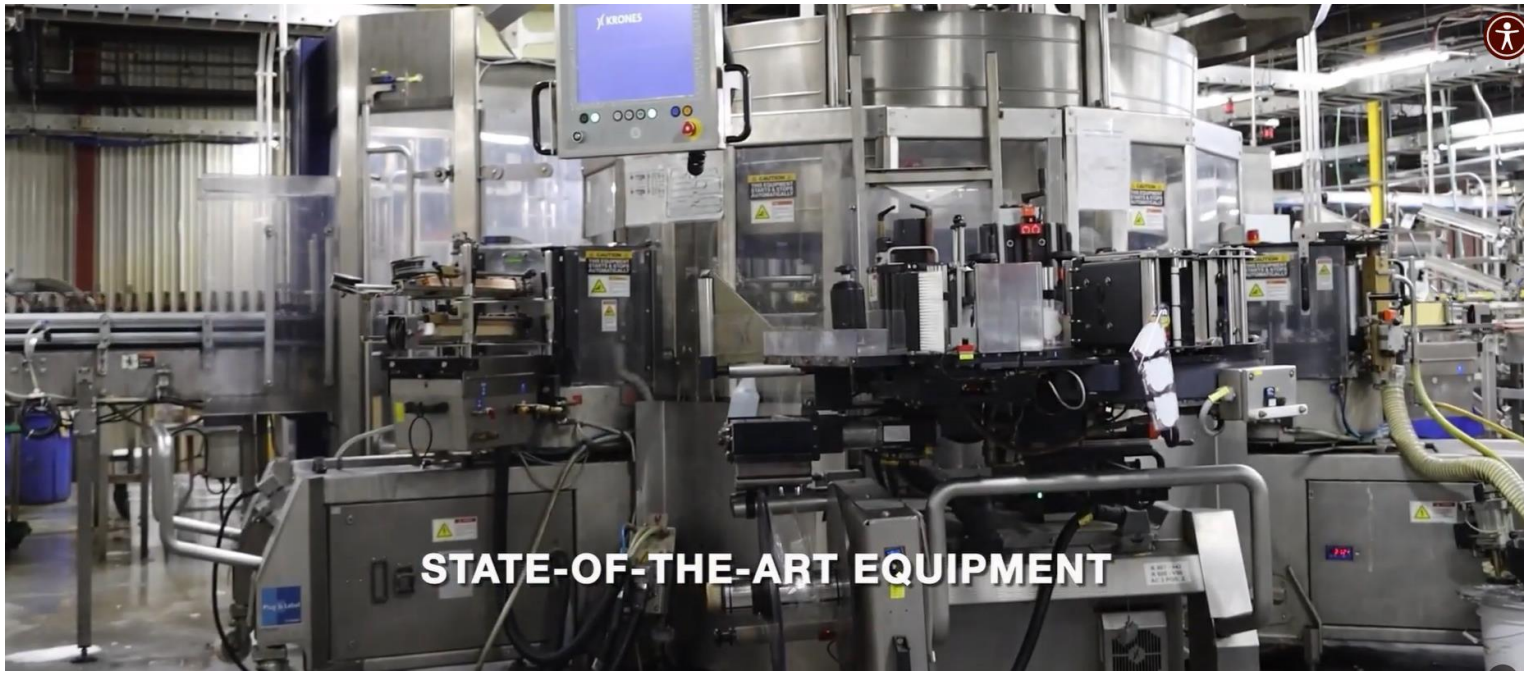
B. Connected factories

C. Virtual factories

A. Smart factories

Enabler of cyber physical system-as a tools

- AI based testing (making log and do predictions)
- Maximize Resource Utilization
- Sustainable Manufacturing
- Reconfigurable Processes and Systems
- Minimal Downtime



B. Connected factories

All devices are connected to each other

- Components include Sensors, Actuators, Transducers, Controllers, Display, Computers, Workstations
- Integration with Business Decision Making
- Examples of connecting factories: CISCO router and other products update online, IT solutions

Smart+Connected Communities

Connected Industries

Sports & Entertainment

Connected Energy

Connected Transportation

AECOM, AGT INTERNATIONAL, Honeywell, Johnson Controls, Schneider Electric, United Technologies

ABB, EMERSON, Honeywell, Rockwell Automation, YOKOGAWA

DAKTRONICS, EVS, Johnson Controls, Panasonic

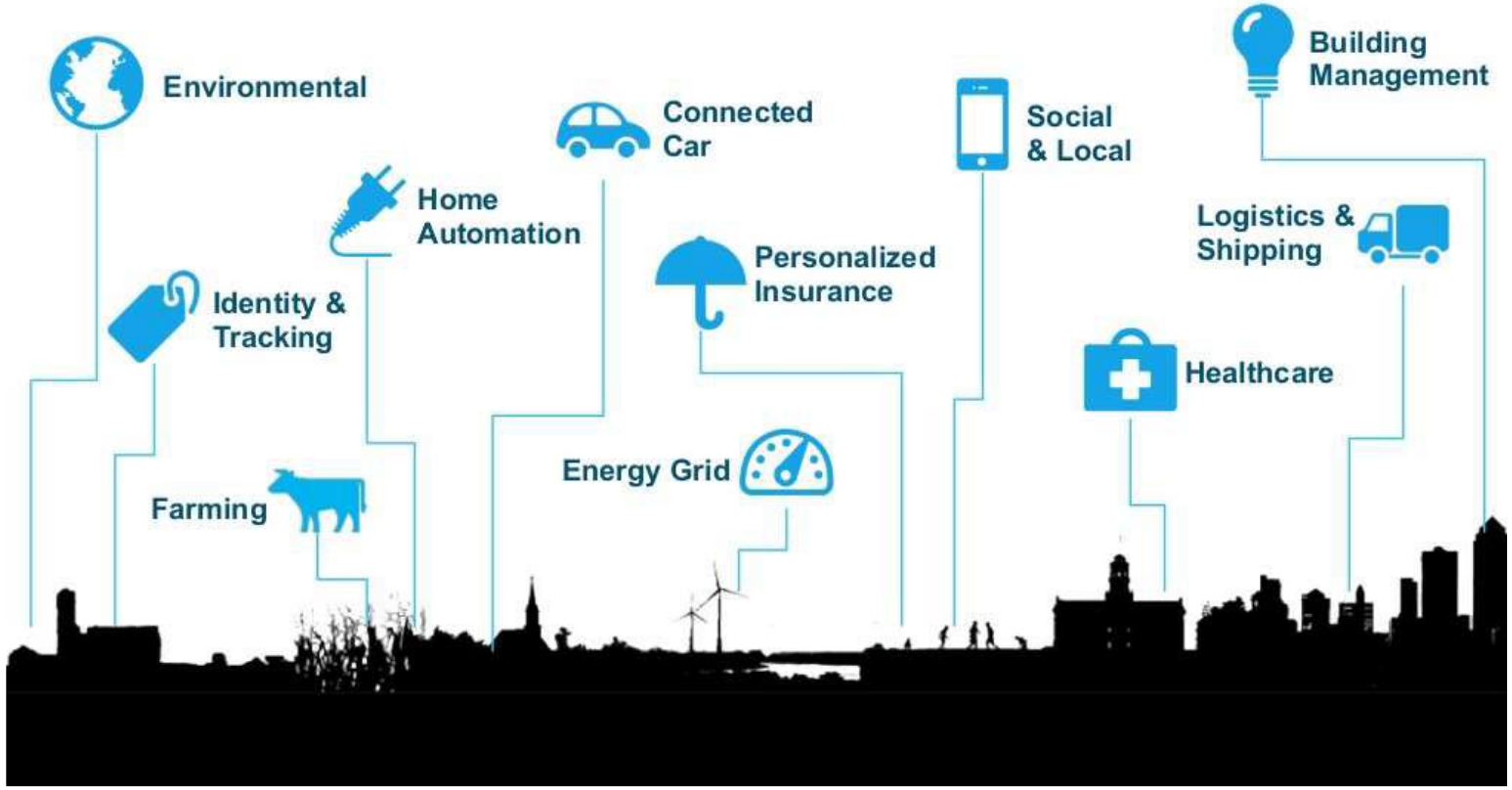
ALSTOM, BLACK & VEATCH, COOPER, EMERSON, SIEMENS, Schneider Electric

ALSTOM, AnsaldoSTS, BOMBARDIER, T

C. Virtual factories

- Advanced decision support capability
- 3D Modeling of Entire Plant
- Access to all processes from a workstation
- Real world simulation prior to implementation
- CAD/CAM designing

Connecting The physical world to the web



Development of IIoT Architecture

• ITU  International Telecommunication Union

2012 (ITU-T-Y.2060 references guidelines)
– Smart grid
– Intelligent transportation system
– E-heath

• Industrial Internet consortium (IIC) also work on reference architecture of IIoT. Published version 1.7 in 2017 known as industrial internet reference architecture.

Communication methods of IoT devices (ITU)

- Devices can be simply data-carrying communicating Storing data and
- Data-capturing interacting with the physical objects through reader and writers

- Sensing and actuating devices

- General-purpose devices with embedded processing and communication resources, such as machines, appliances, and consumer electronic products.

General Architecture requirements

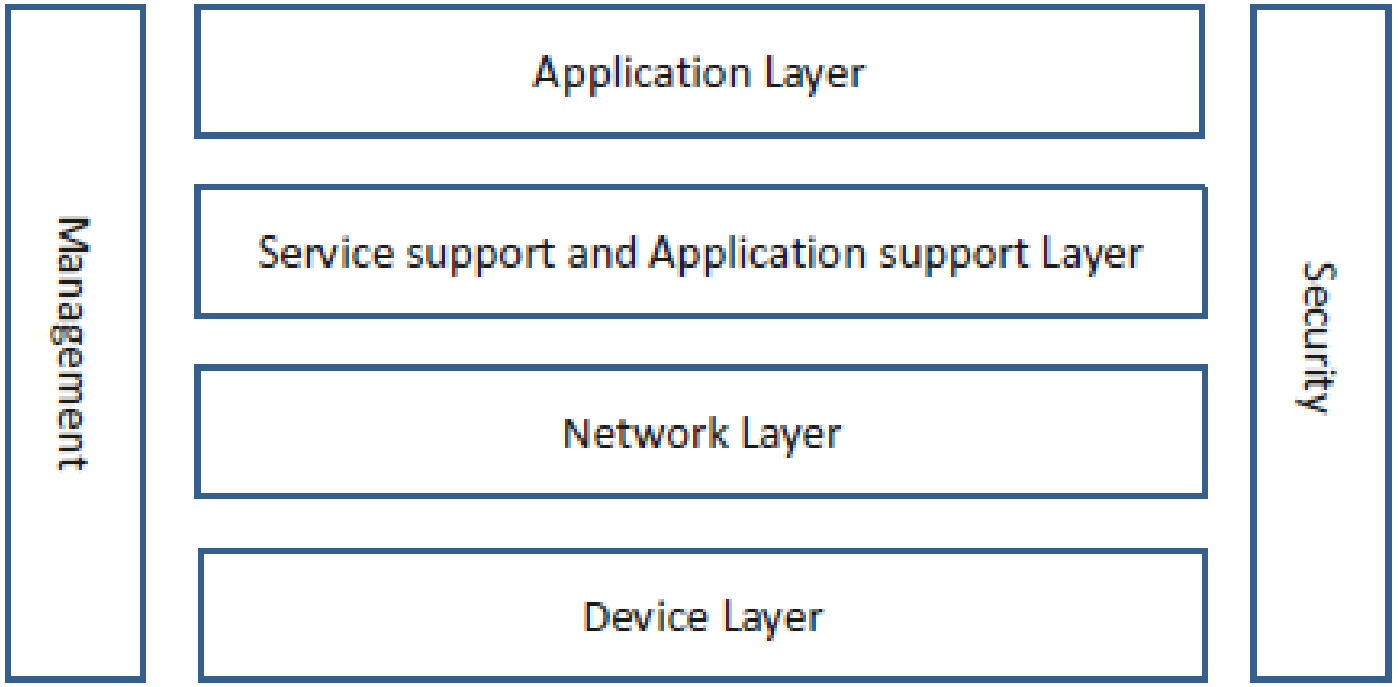
- ITU include
  - interoperability,
  - identification-based connectivity,
  - autonomy in networking and services,
  - accommodation of location-based services,

security and privacy
- capabilities for management of things and services, including plug and play.

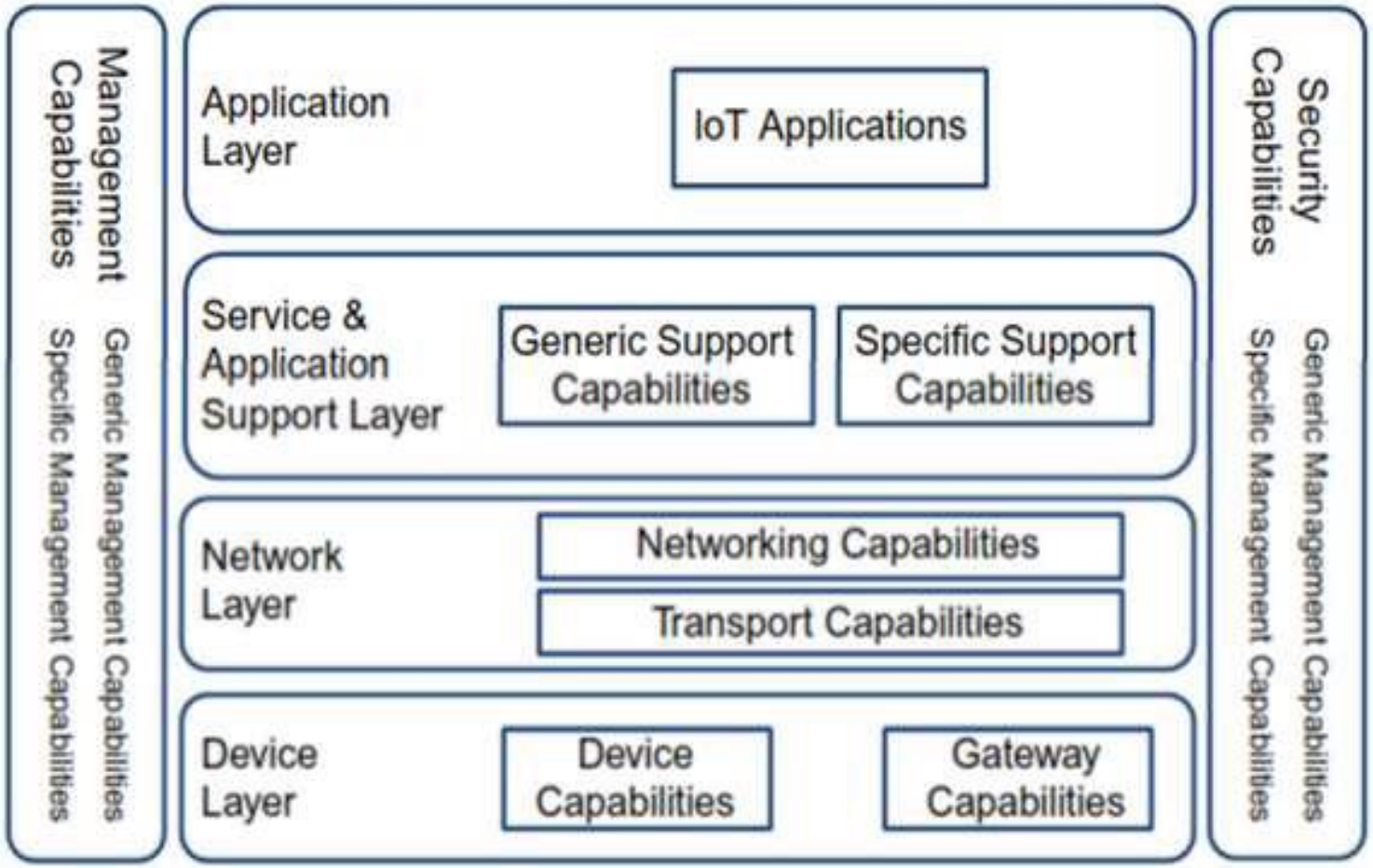
IUT Architecture of (I)IoT

- ITU accommodates fundamental characteristic of IoT such as
  – Interconnectivity, scale (devices and data issues, management, storage and processing), Heterogeneity, services for things
  – Dynamic nature of device information, and connectivity
- Almost include all "things" that means things are identifiable and able to connect to communication network

(I)IoT reference model by ITU



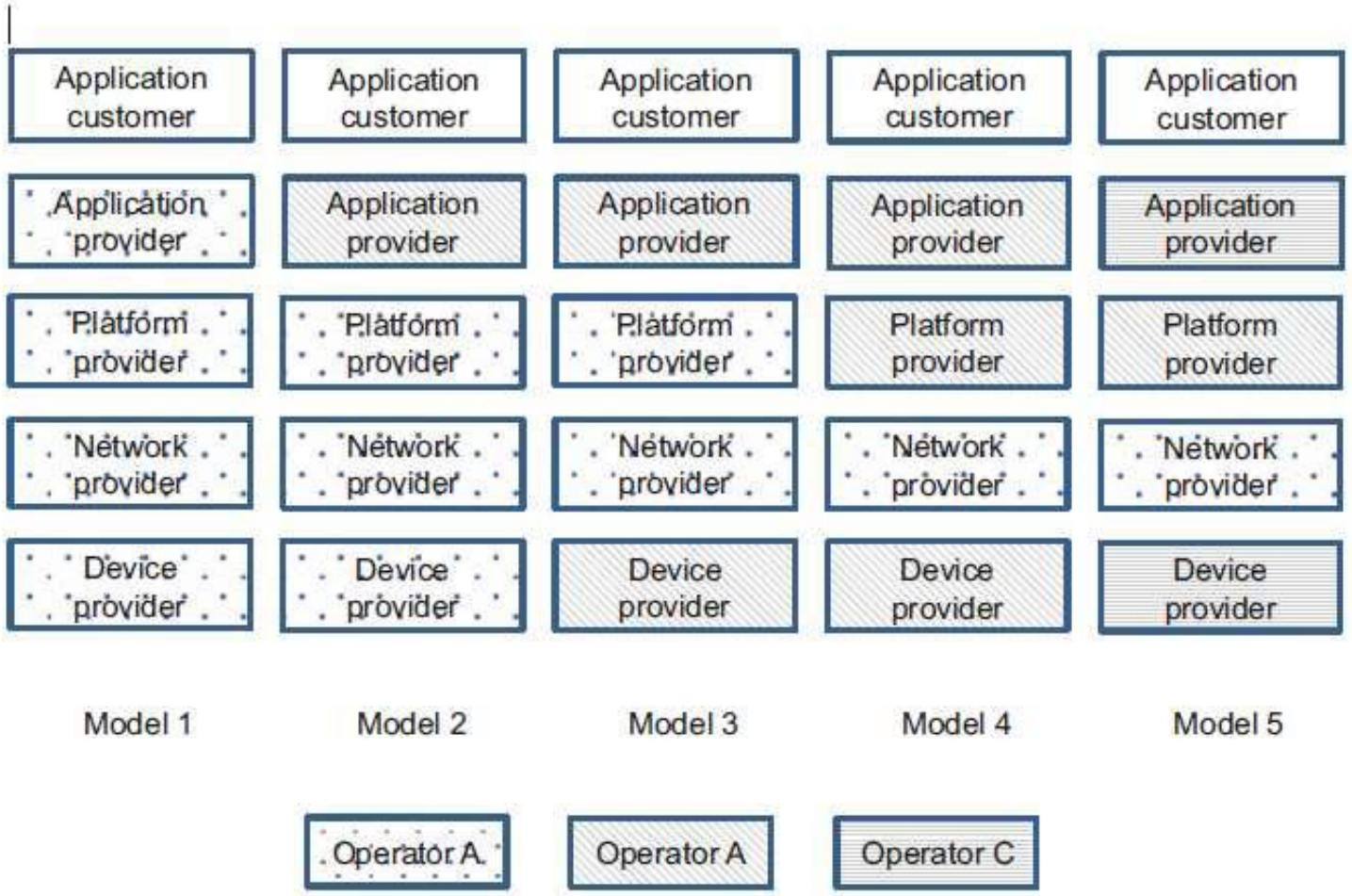
(I)IoT reference model by ITU



Five Business roles in (I)IoT

1. Device provider
2. Network provider
3. Platform provider
4. Application provider
5. Application customer

Five Business roles in (I)IoT



(I)IoT Business models-ITU

Model 1: the same organization has the roles of device, network platform, and application provider

Model 2: one stakeholder has the roles of device, network, and platform provider and another stockholder has the role of the application provider.

Open nature of (I)IoT

Number of connecting devices increases day by day increase complexity

- ITU Architecture vision is to increase communication of "any things","any time" and "anyplace"

- Decision support system of machines not present in ITU

- All stakeholders that are • ITU Not involved all stakeholders such as system engineer, product manager, end user etc

- Integration of IT and OT

IIoT Architecture (IIC17)

Industrial Internet Consortium (IIC) focuses on similar concepts (like ITU) and develops a reference IIoT architecture that has several
similarities with the ITU approach and reference model.

- IIC-17 is Specialized evolution of the ITU Model

- Focused on stakeholders needs and interest

IIC Architecture approach

- IIC approach to see the interests and concerns of all types of stakeholders in an integrated way, (focus on complete business models and applications at all levels, from devices to IIoT services.)

- Different stakeholders who need to make different decisions at different levels of abstraction.

- Stakeholders are focus on the parameters of interest

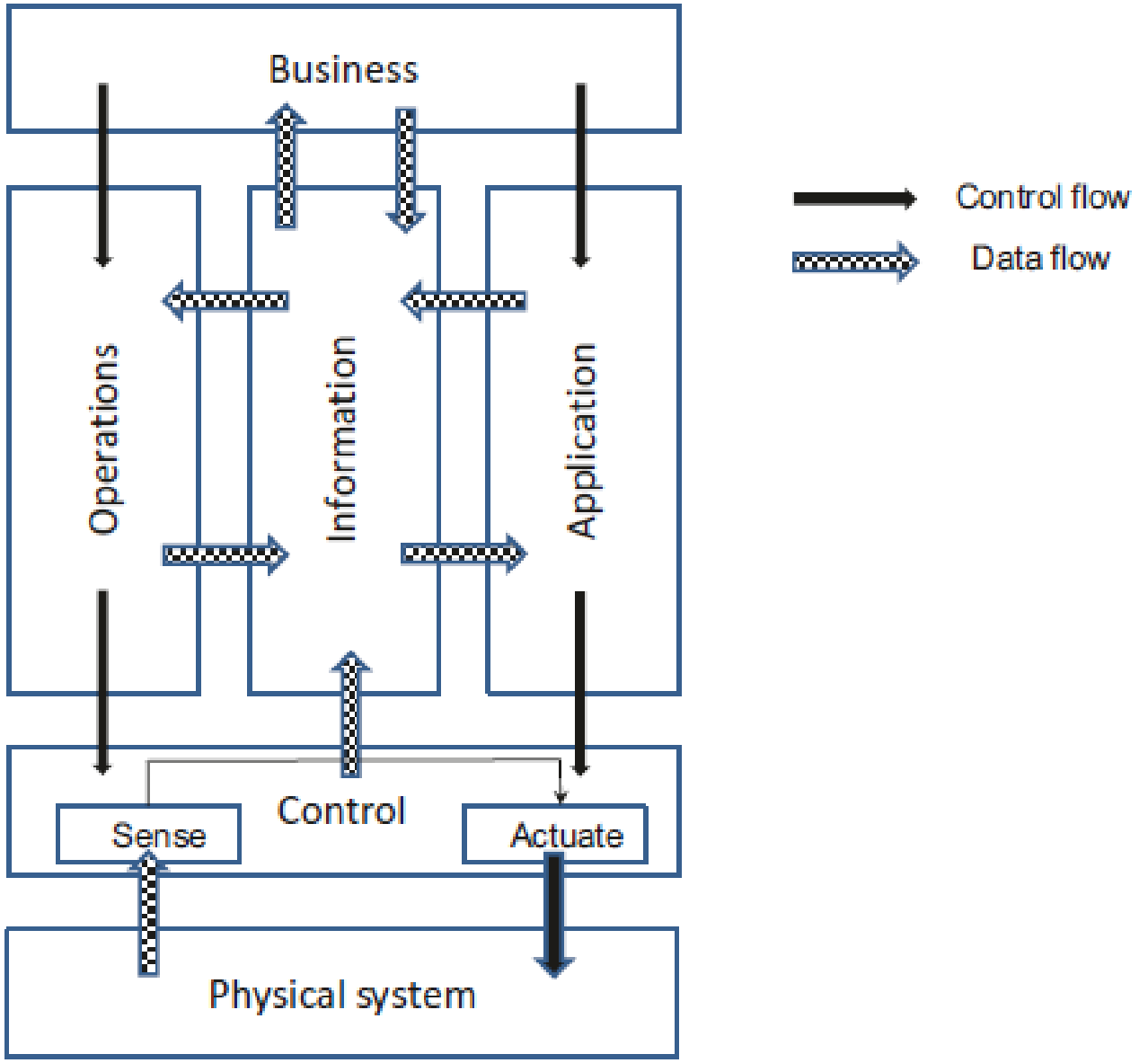
- For this purpose, IIC has identified four different viewpoints:

(a) Business,

(b) Usage,

(c) Functional

IIC reference architecture



IIC reference architecture

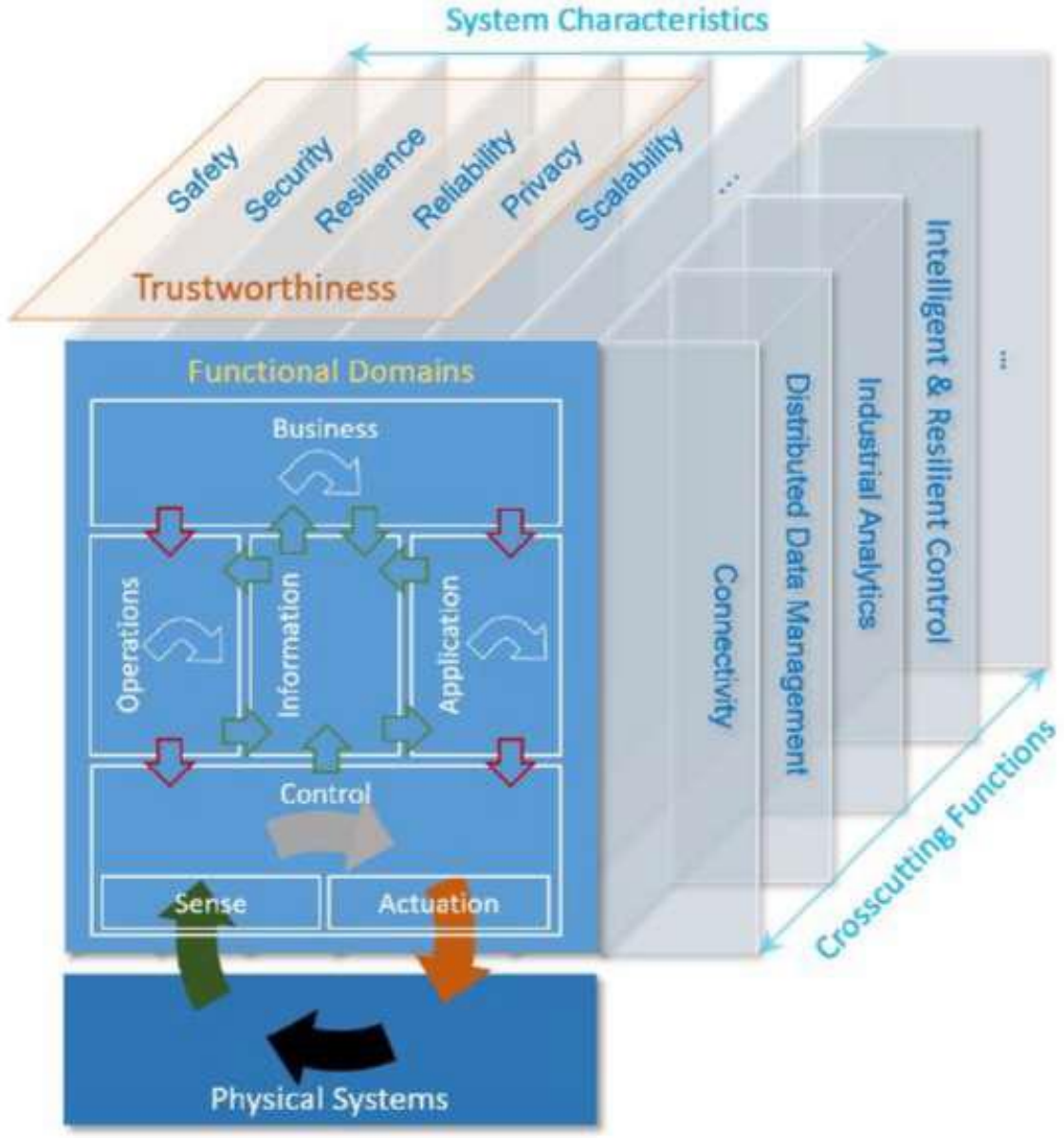


Figure 6-5: Functional Domains, Crosscutting Functions and System Characteristics

IIC reference architecture domains

The control domain is realized by industrial control system (actuator, sensors and physical system loop)

- The operation domain includes system monitoring and management as well as optimization for the efficient operation of the systems.

- Information domain is responsible for collecting data from all domains and analyzing them to enable high-level decisions for the system

The application domain is the set of APIs and user interfaces so that other applications or human users can use the application effectively.

- The business domain related to management and decision system such as enterprise resource planning systems (ERP), manufacturing execution systems (MES)

IIC reference architecture vs ITU

The diagram shows the data and control flow among the domains, as specified by IIC

- It is not a layered approach as ITU represents
- It is a logical functional decomposition within a layer or across layers in a hierarchy
- The IIC approach and the ITU approach are complementary

- **IIC is generalized reference architecture of ITU because it includes**

– crosscutting functions analogous to the ITU layers

– it enables the development of more detailed functional models per layer addressing complete control loops

– It providing support to all types of stakeholders – from device designers to business developers – for effective decision making.

Basic technologies in IIoT

Microchip's identification information to a reader over wireless media (RFID-1980's- logistics and supply chain management)

• Wireless sensor networks (WSN)-issues (range for communication , battery sensors)

– WLAN, Zigbee, Bluetooth, etc
• Application protocols that are suitable for the various IIoT application domains such as
– Constrained Application Protocol (CoAP)
– Message Queue Telemetry Transport (MQTT)
– Advanced Message Queuing Protocol (AMQP).

Applications & challenges IIoT

Wide range of IoT application domain

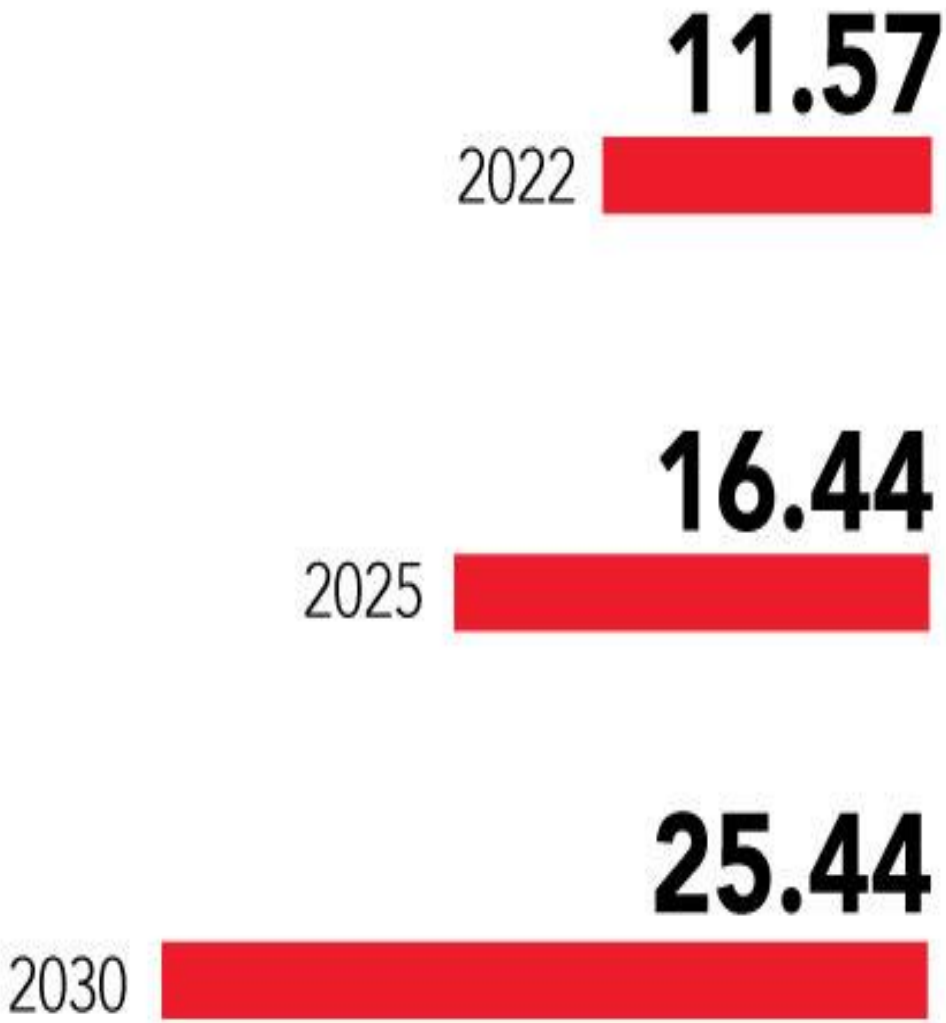
• PLC+ SCADA ICS

• Industrial control system (ICS) when add with IIoT services make critical infrastructure of industry 4.0.

• Power distribution networks
– Electrical network (wind, solar, hydro plants)
– electricity production, distribution, and consumption processes

• water distribution and management networks
• oil and gas distribution networks
– Managing pipelines and storage tanks

• Transportation
– operation and management of traffic lights, for toll payments etc



NUMBER OF IOT-CONNECTED DEVICES WORLDWIDE 2022-2030

(IN BILLIONS)

SOURCE: FINANCESONLINE.COM

Q & A

Thank You

Internet of Things

IE 4030

Uditha Dharmakeerthi

Overview

01

SECURITY BY DESIGN PARADIGM

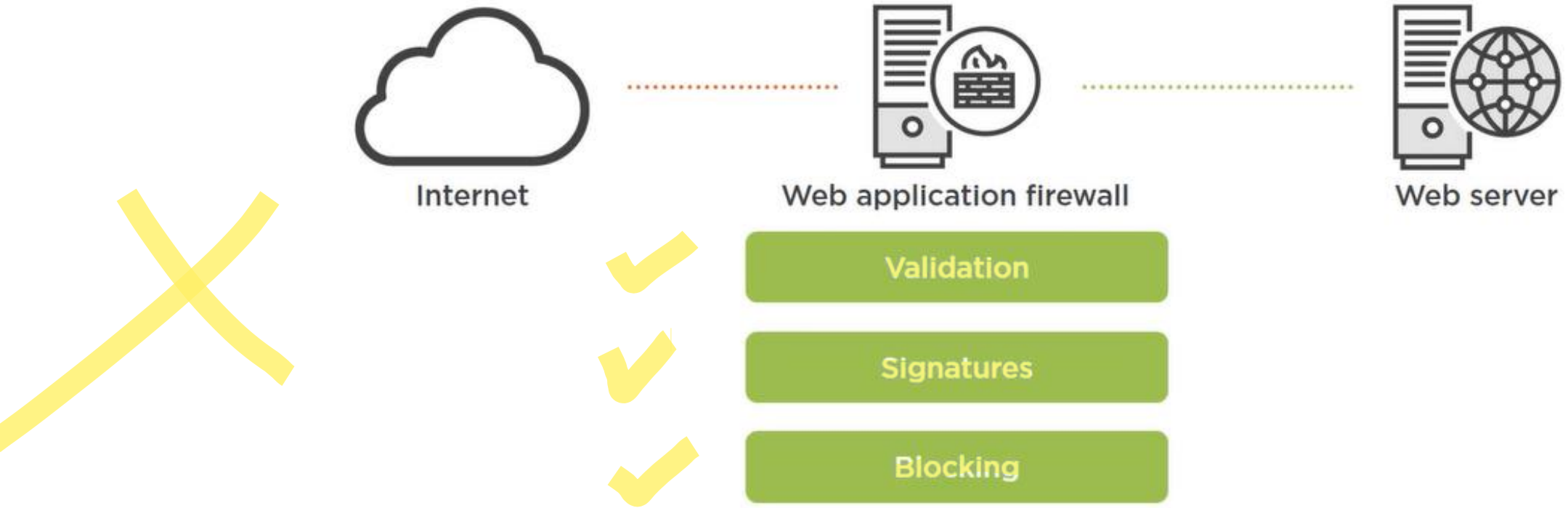
02

FRAMEWORK FOR IOT SECURITY

03

HOW CURRENT TRENDS IN IT CAN HELP IMPROVE SECURITY

Where Traditional control fall short



Where Traditional control fall short

Where New Trend of WAF

Where Traditional control fall short

Where Traditional control fall short

Device is design to be secure

Security controls are built it

Security is embedded in the entire project at each stage of deployment lifecycle

Protecting communication devices

Secure By Design

Protecting Communications

Encrypting all communications

- TLS or DTLS are proven standard

Authentication Mechanisms

- Mutual authentication

Security key Management

No Unauthorized code id executed

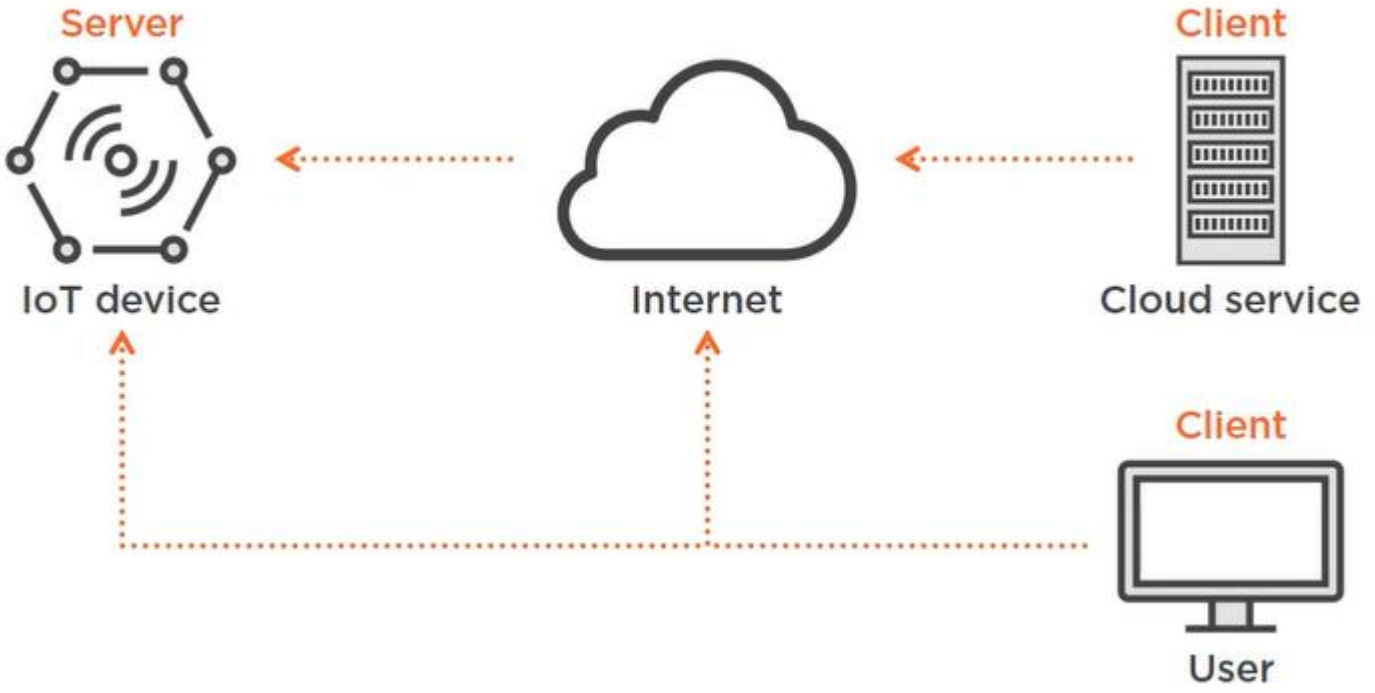
- Code signing capabilities

Secure updates

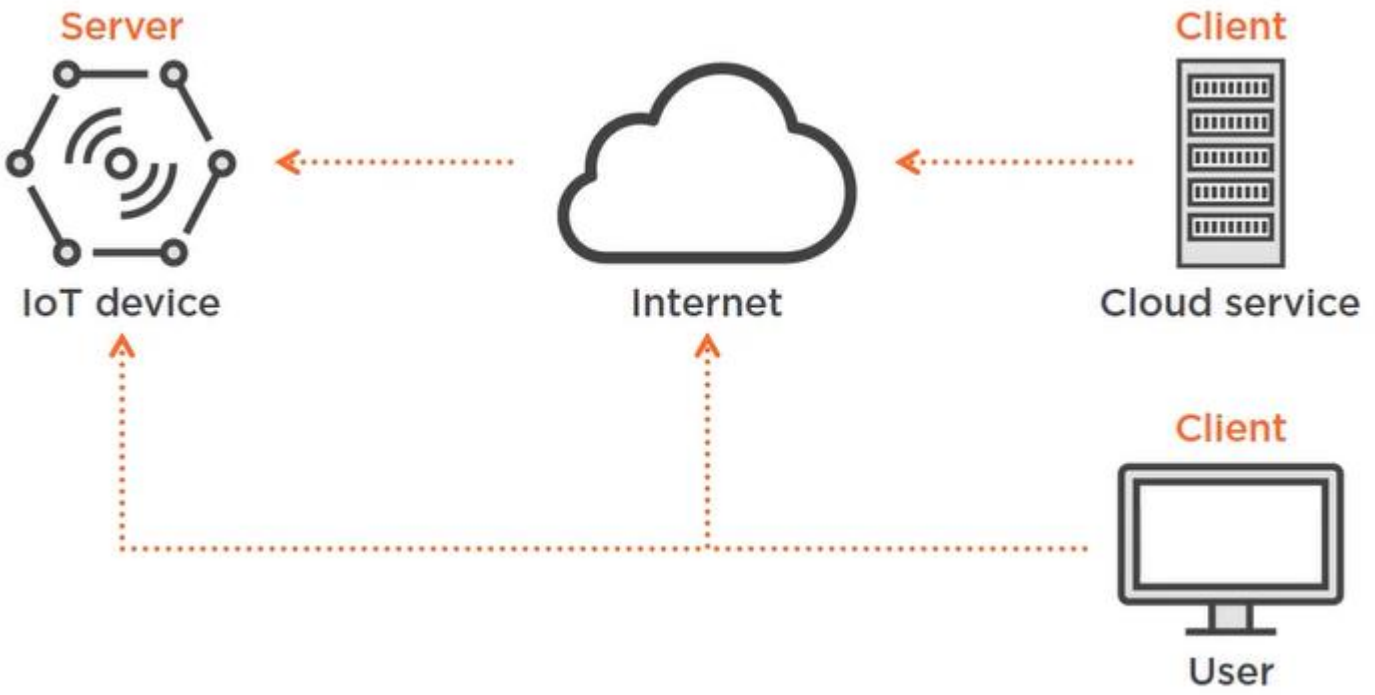
- Validation mechanism for software updates
- Over-the –air updates

Protecting Devices

Trust Model


Server
Client
IoT device
Internet
Cloud service
Client
User

Trust Model


Server
Client
IoT device
Internet
Cloud service
Client
User

Trust Model

The smart device will host services that users and other services will connect to

Increase security risks and exposure

Additional security controls are required

Constrained resources will reduce effectiveness

IOT Device as the server

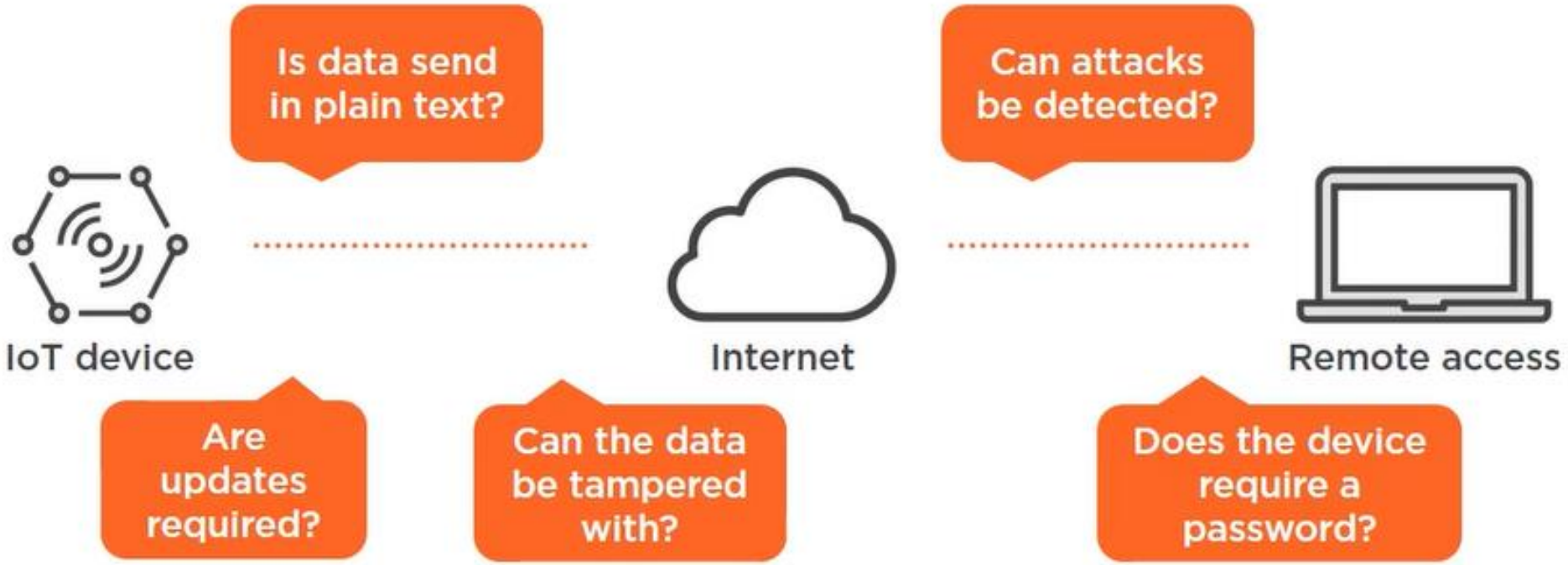
The device will connect to the cloud service to submit data

Reduce exposure and security risk

Fewer resources required

Connect to the cloud service at regular intervals

IOT Device as a client



IOT Security for Developers

- Encryption
- Authentication
- Integrity verification
- Intrusion detection
- Secure update capabilities

Security Controls

As a developer

- Principle of least privilege
  - Prevent device from being compromised
- Having an attacker mindset
  - Take into consideration concept like risk and exposure
- Use development frameworks with built –in security controls
- Join communities

Include security requirement as apart of the user stories the IOT solutions

Audit and access the security posture of the solution to find vulnerabilities

Continually access the state of the solution even after the it goes to the production

As a security professional

OWASP

NIST

Security Vendors

Government

Existing IOT Security Frameworks

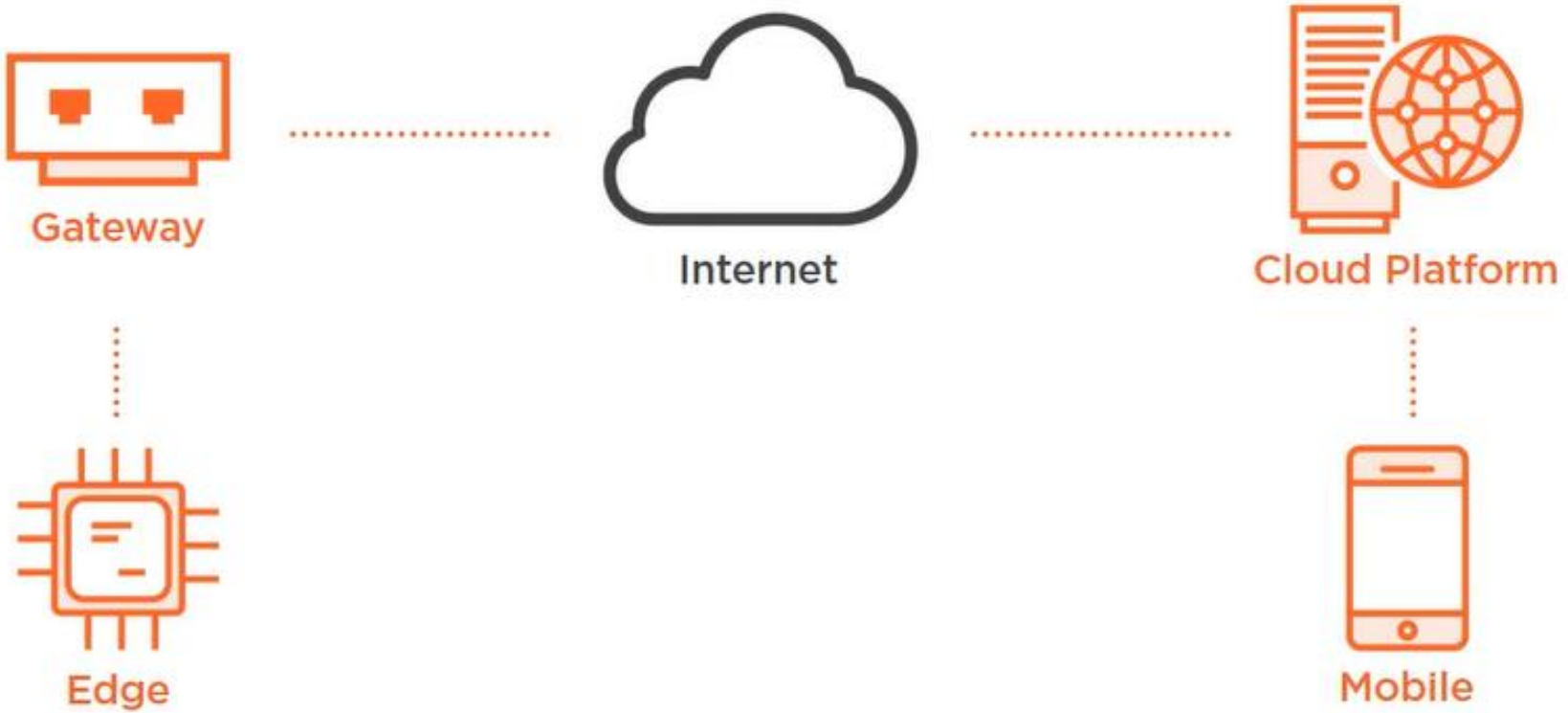
Generic security aspect of IOT

Top 10 most common vulnerabilities

Security Guidance for manufacturers and developers

Attack surface areas for auditors and security professionals

OWASP IOT Project



Secure IOT Framework


I1: Insecure web interface
I2: Insufficient authentication/authorization
I3: Insecure network services
I4: Lack of transport encryption
I5: Privacy concerns

Top Ten IOT


I6: Insecure Cloud interface
I7: Insecure mobile interface
I8: Insufficient security configurability
I9: Insecure software/firmware
I10: Poor physical security

Top Ten IOT

Focus on areas with increased risk

Solid baseline for security assessments

Measure effectiveness of security controls

Benefits of Security Framework

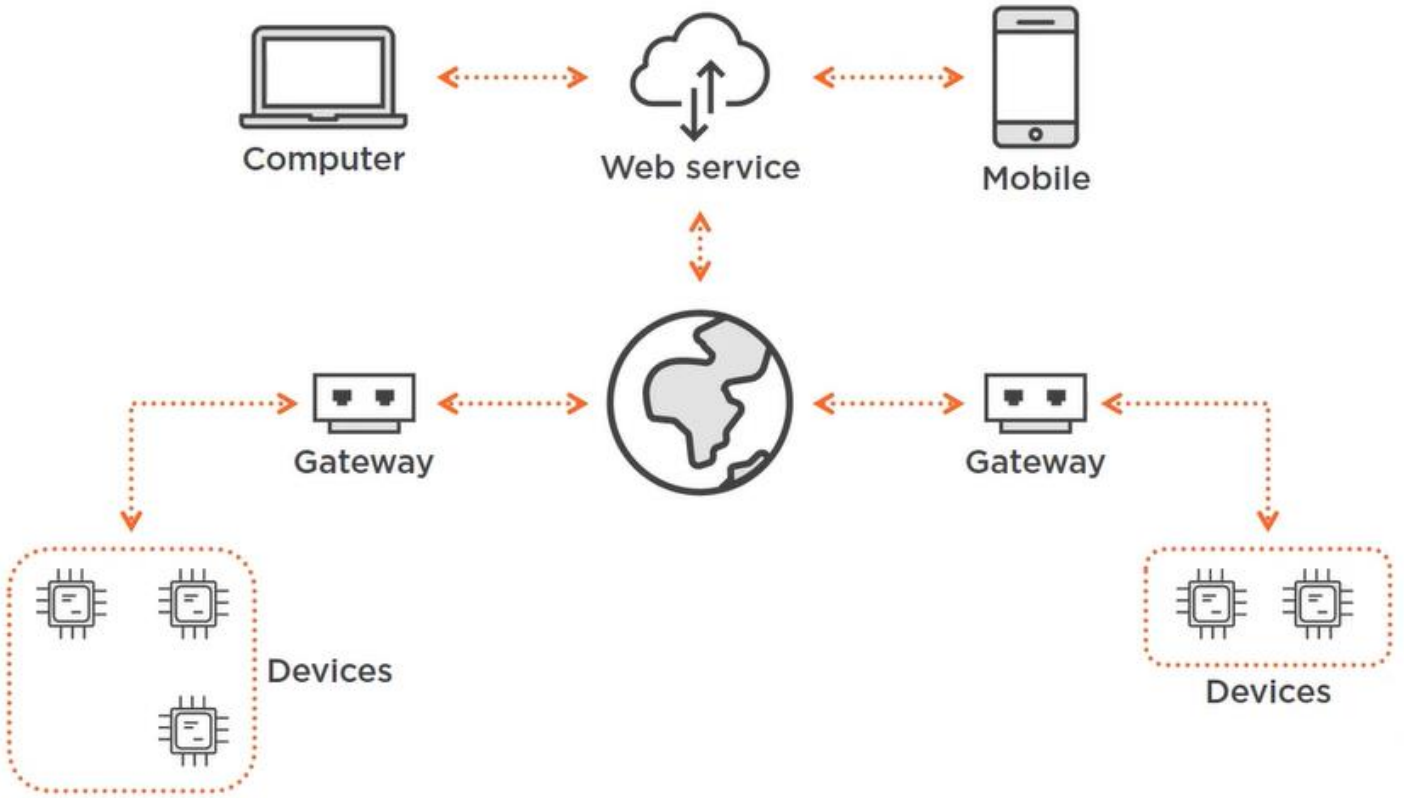
Threat modelling

Secure DevOps

Big data

Current Trends in industry


THREAT
MODELING
1
Understand the Organization and Line of Business
2
Physical and Logical Security Architecture Review
3
Define Assets
4
Identify the Vulnerabilities, Threats, and Risks
5
Review the Countermeasures and Gap Analysis
6
Integrate your Security state to SOC
7
Monitor-Tune-Mature



Current Trends in industry

Working with large data sets that are too difficult for traditional database to handle

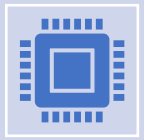
Tools and processes to capture ,store and analyze large volumes of data

Big Data enables gaining more insight from Data

-Behavioral analytics

-Predictive analytics

Big Data



Requirement for IOT solution that ingest and process large amount of data

-Applicable for hundreds of thousand of devices

Adding security into the mix

-Validating and filtering of data generated

-Monitoring and correlation of Security events

-Trend analysis

Big Data

Traditional security approach dealt with security issues when the solution was in production

Threat modeling deals with identifying potential treat at the design stage

-Security risks are identified based on internal factors

-Defenses are implemented and tested to address and mitigate risks

Reduce costs over time

Threat Modelling

Threat Modelling Process

- Focus on entire solution
- Use proven metrologies
  - STRIDE
- Efficient way to prioritize defense based on security risks
  -OWASP IOT attack surface

- Identify threats ,elaborate and validate mitigations

Threat Modelling for IOT

Device

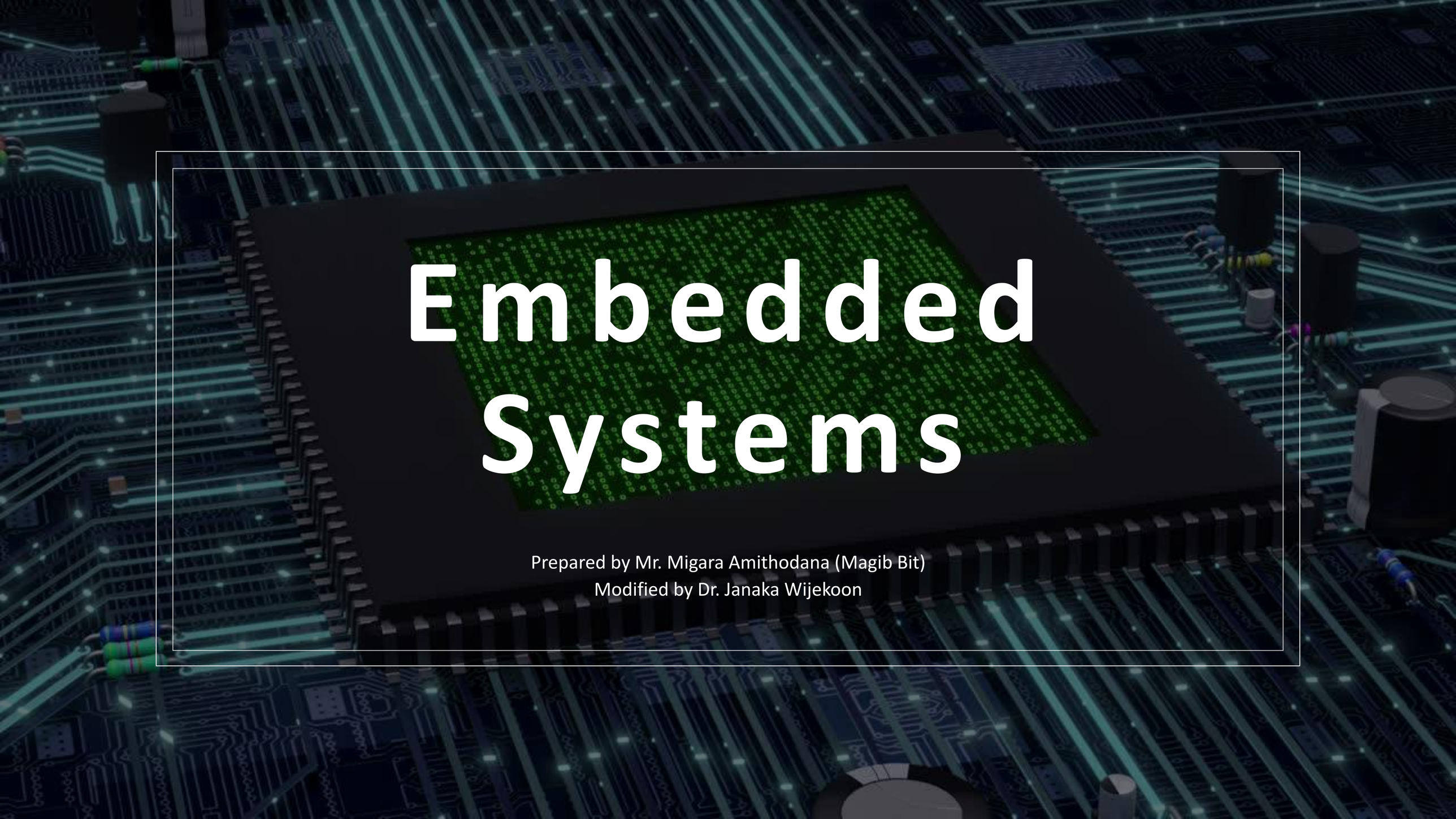
Communication

Lifecycle

Security Goals

Q & A

Thank You



Embedded Systems

Prepared by Mr. Migara Amithodana (Magib Bit)
Modified by Dr. Janaka Wijekoon


China ranks 'good' and 'bad' citizens with 'social credit' system
Share
11661
11628
11662
11667
11676
11684
11648
11653
658
11663
Watch on YouTube

Embedded system mean using micro controller , micro controller, sensor and actuators 3ma ekathuwak

Sensor user karanne thiyena physical parameter ekak digital signal ekak widiyata convert karana eka.

W h y ?

Sensors --> Sensor is used for converting a physical parameter to a digital signal

Analog to digital converts --> some of the sensors are generating analog signal as signal. then we want to use analog digital converter to identify the what are the correct value of the signal. Then analog to digital converts are used step sizes for measure the value of the sensor.

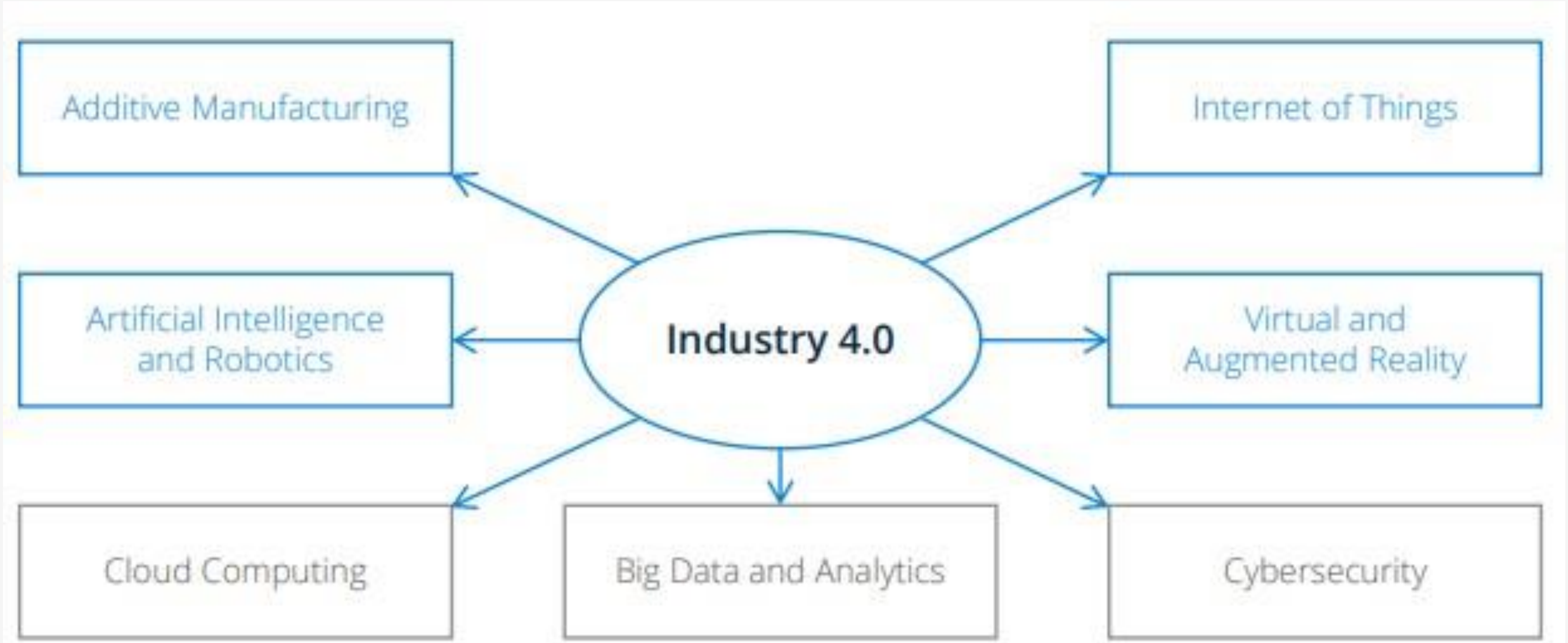
Digital to analog sensor --> PWM(puls with modulation technology) moter controlers use karaddi moter speed adu wedi karanna ana analog signal denna one. eka apita bord eken denna be.. eka api PWM widiyata denna one.

EX: Moter drivers

Actuators: Motors, Lights, fans, Pumps
Actuators mean out put devices.

Actuators aniwaryen one IOT device ekaka parameters control karanna

- Identify the working principals
- Knowledge on available devices
- Requirement specifications
- Implementations
- Vurnerability assesment



Sources: BCG, Statista

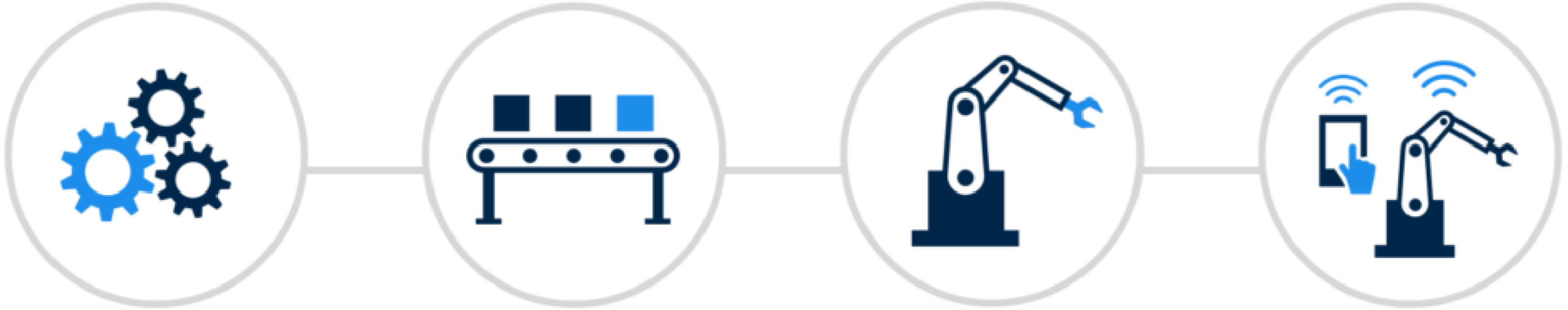
Industry 1.0

Industry 2.0

Industry 3.0


Industry 4.0

The Four Industrial Revolutions



Industry 1.0

Mechanization and the introduction of steam and water power

Industry 2.0

Mass production assembly lines using electrical power

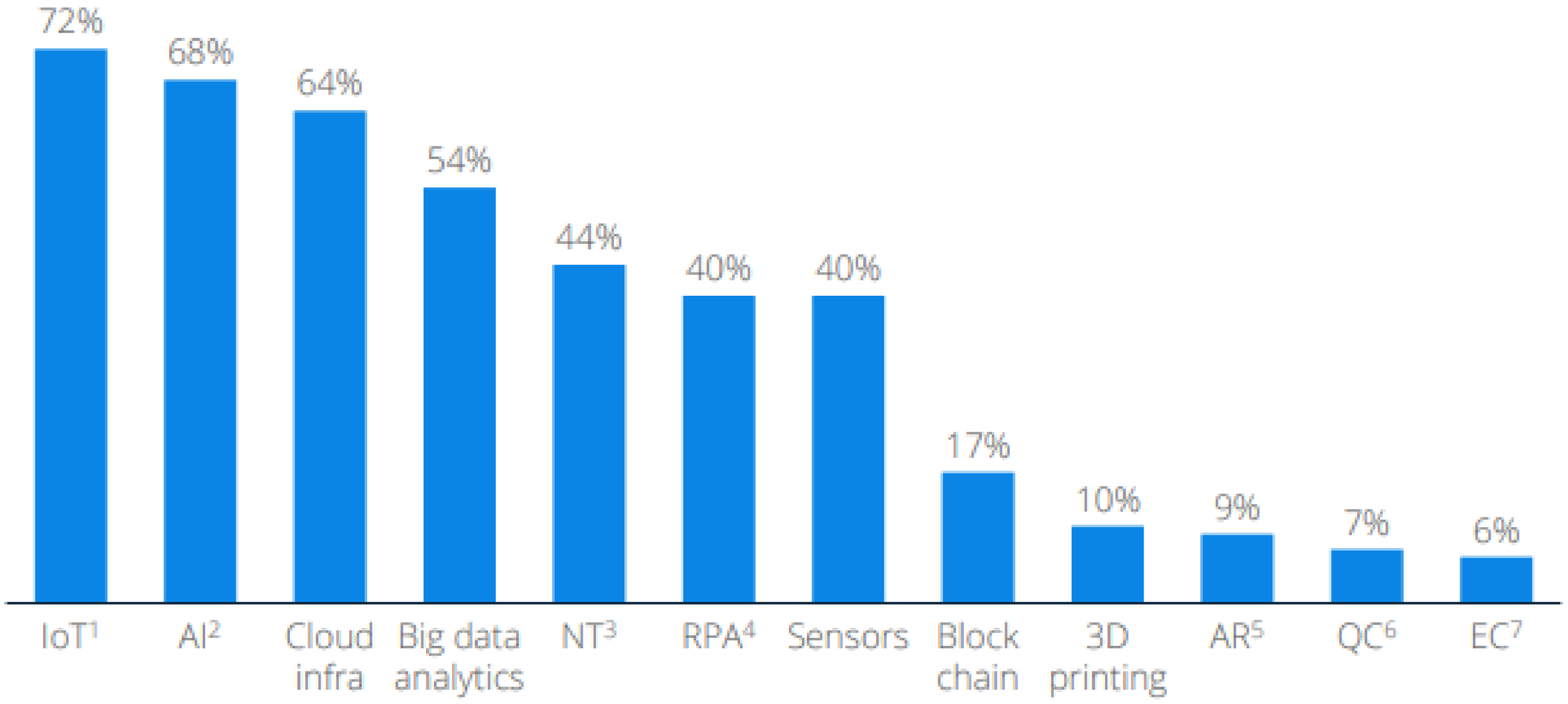
Industry 3.0

Automated production, computers, IT-systems and robotics

Industry 4.0

The Smart Factory. Autonomous systems, IoT, machine learning

Which tech impact to business?



Sources: Deloitte Survey of 2,029 business leaders in technology, as of September 2019

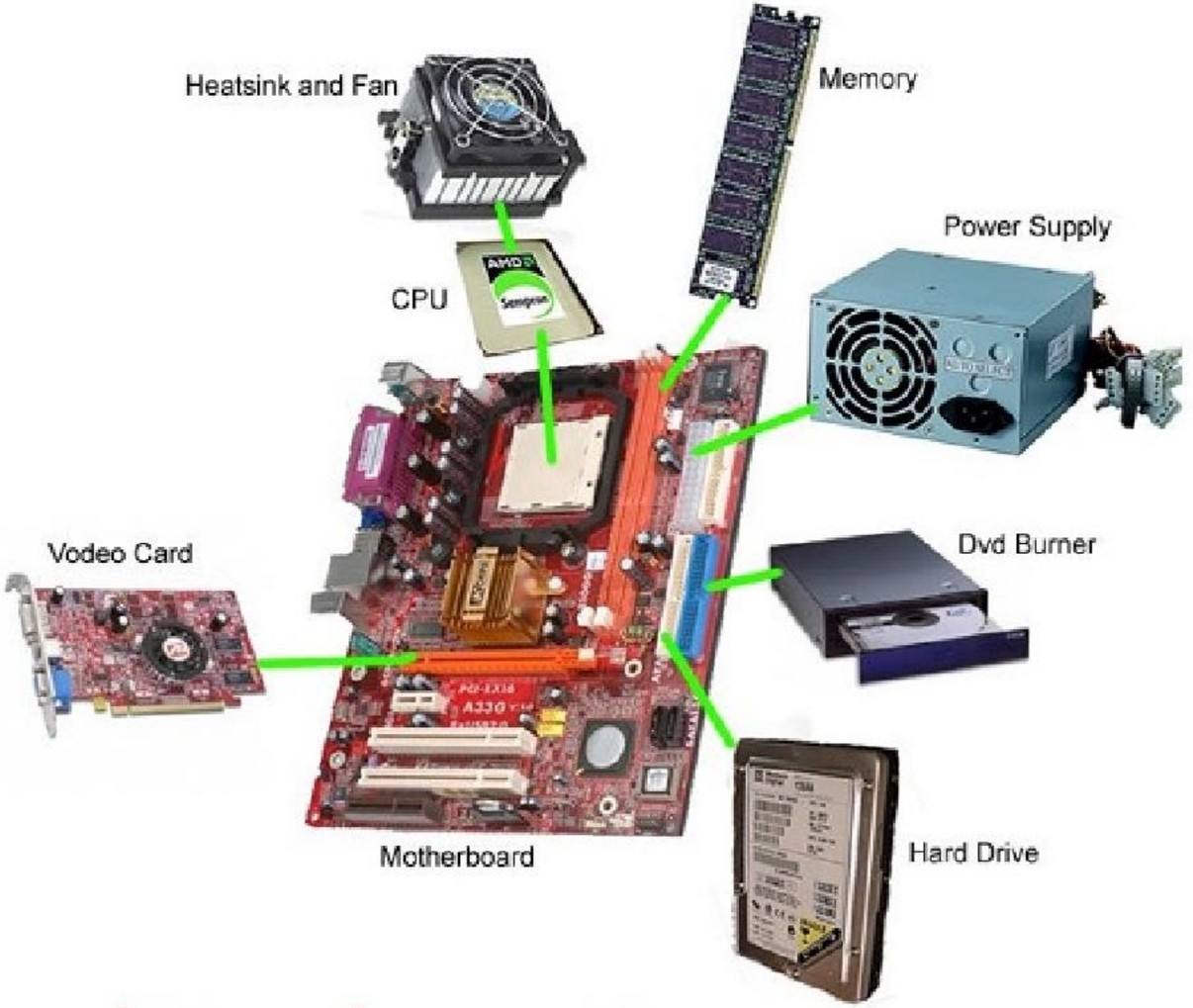
What is a computer?

an electronic device for storing and processing data, typically in binary form, according to instructions given to it in a variable program. Modern computers are,

- General purpose
- Multifunctional
- User friendly
- Portable

CPU

Matheeboad



Computer CPU and Motherboard Hardware Components

How many Embedded Systems in use today?

Embedded System

A computer system—a combination of a computer processor, computer memory, and input/output peripheral devices—that has a dedicated function within a larger mechanical or electronic system.

Characteristics

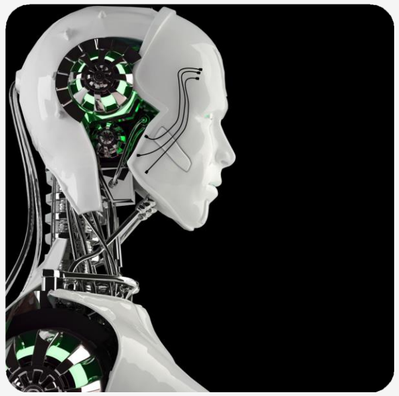
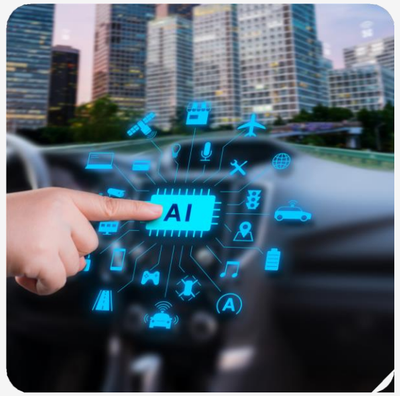
- Specific function
- Combination of hardware and software
- Component of a larger system
- Limited memory & capacity
- Firmware/instructions

History

The Apollo Guidance Computer (AGC)

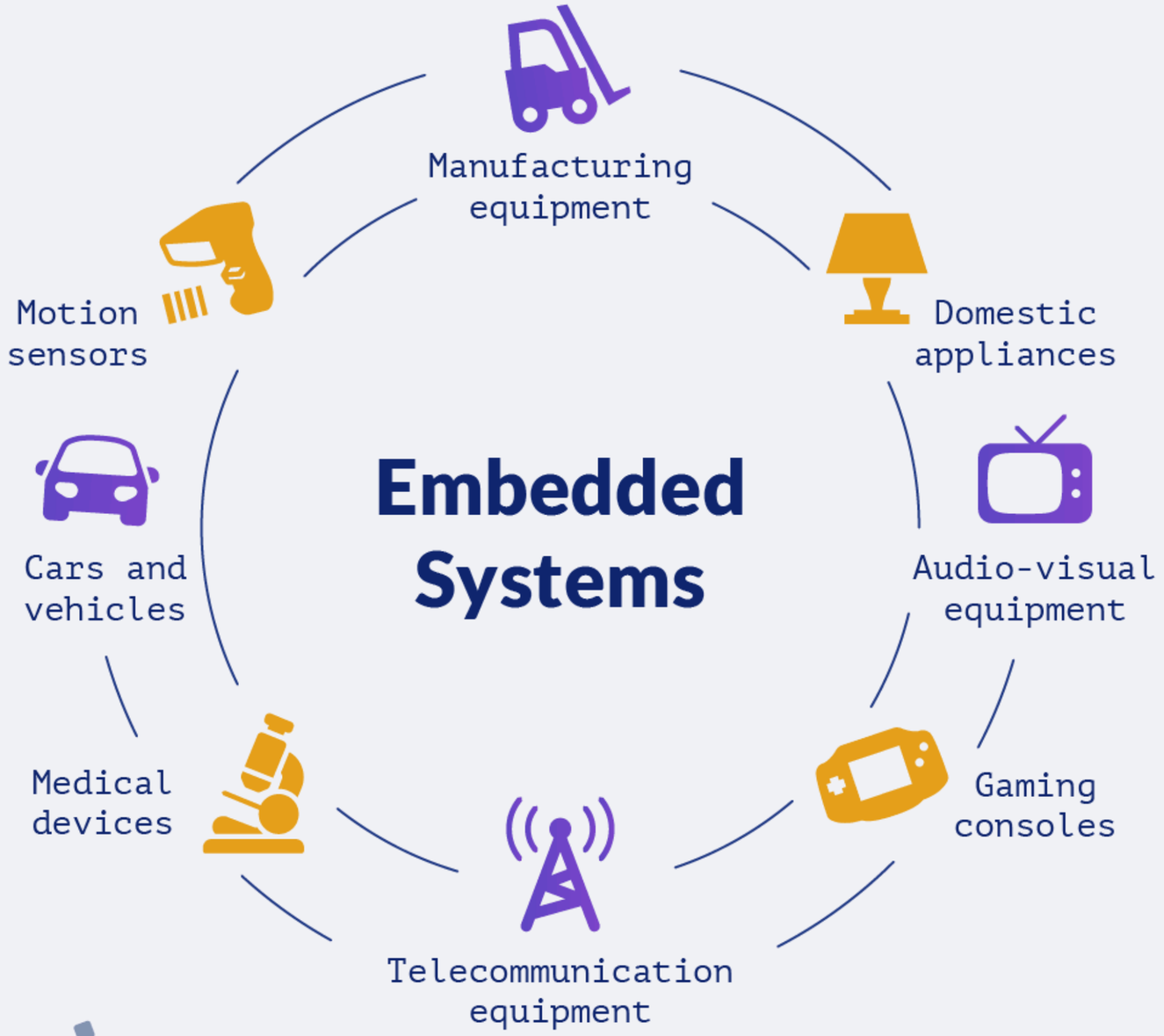
Applications





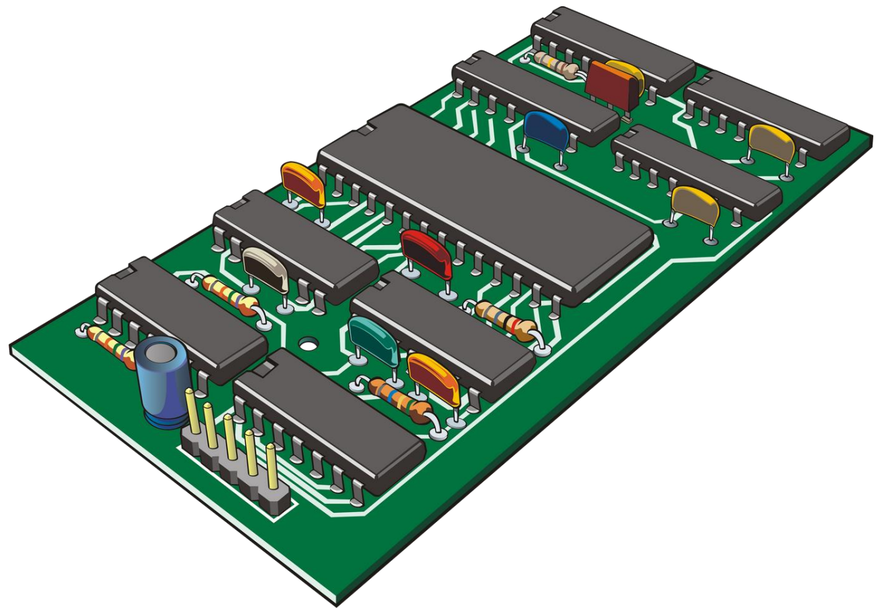

- Wearable devices
- Mobiles
- Robotics
- Automobile
- Industrial automation
- IoT applications



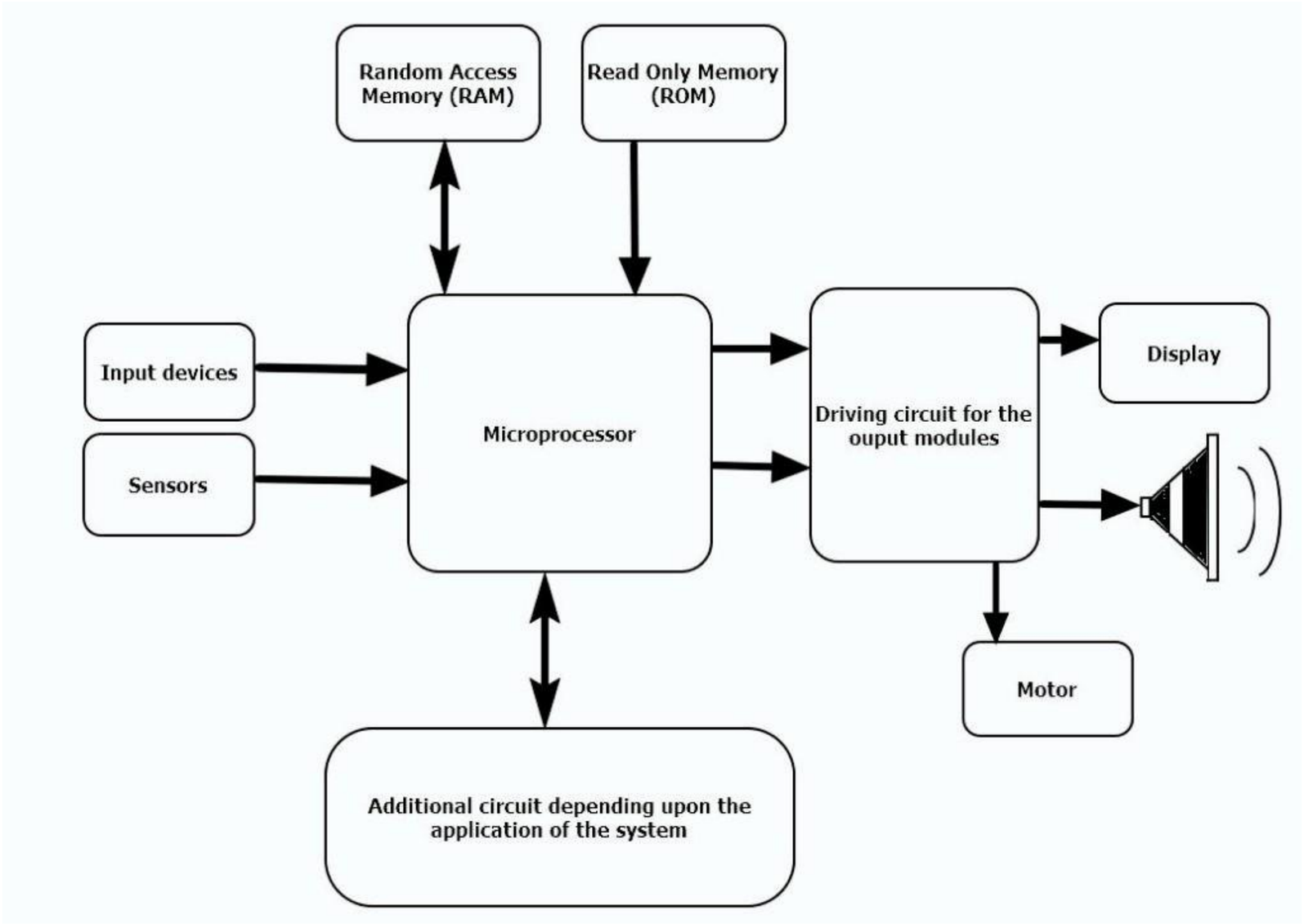
sam solutions

Purpose

- Data collection/Storage/Representation
- Data communication
- Data (Signal) processing
- Monitoring
- Control
- Application specific user interface

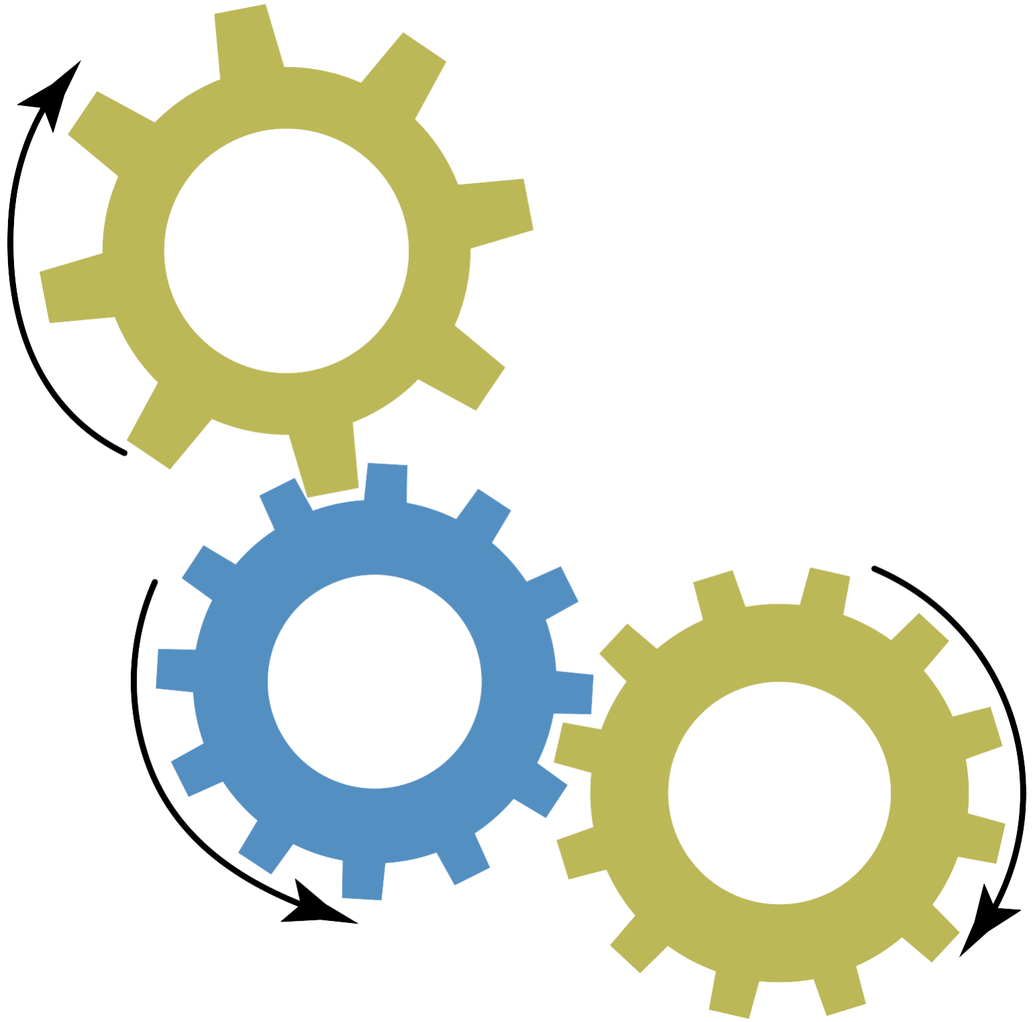


Architecture



Structure of an ES

- Sensors
- A-D Converter
- D-A Converter
- Actuators

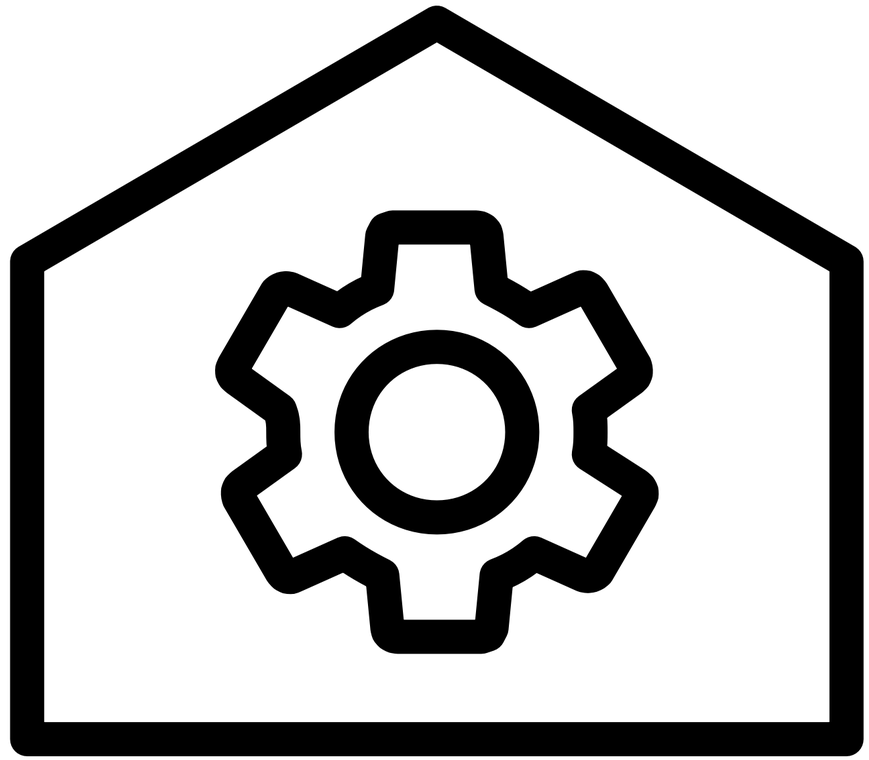


Sensors

The sensor measures and converts the physical quantity to an electrical signal

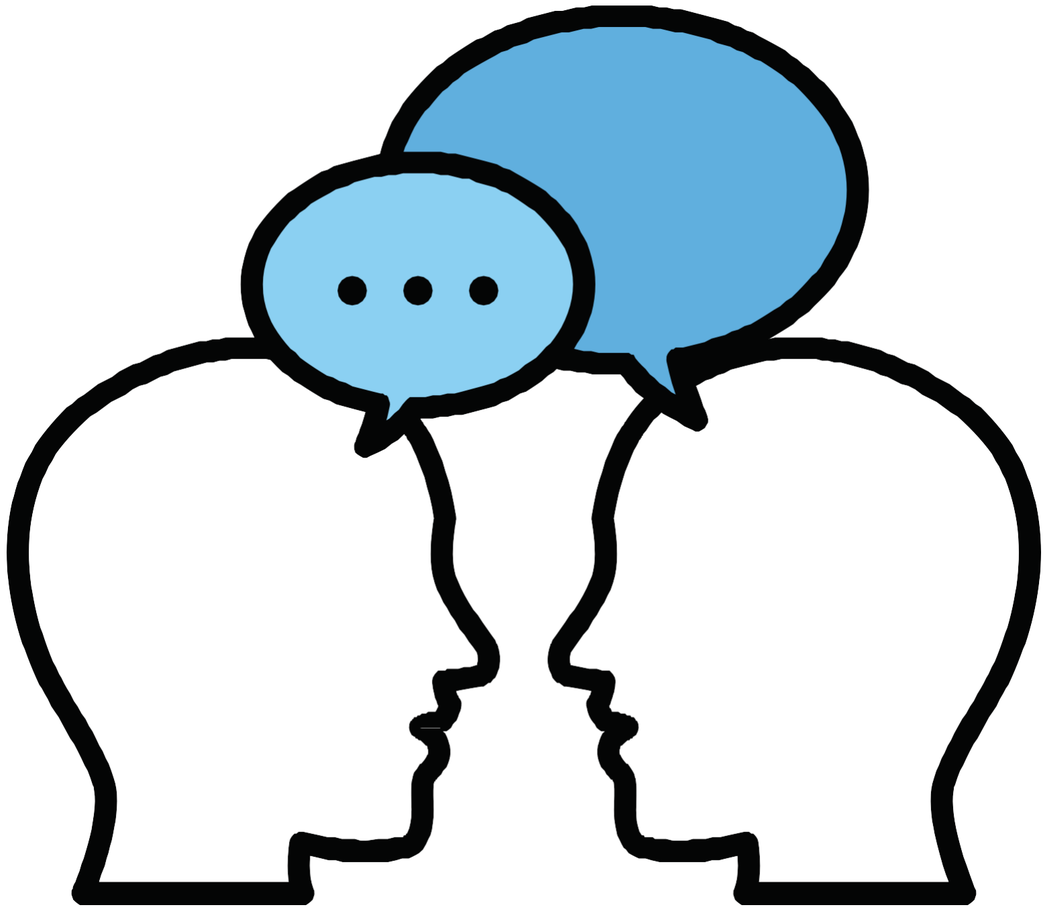
Actuators

An actuator is a device that produces a motion by converting energy and signals going into the system.



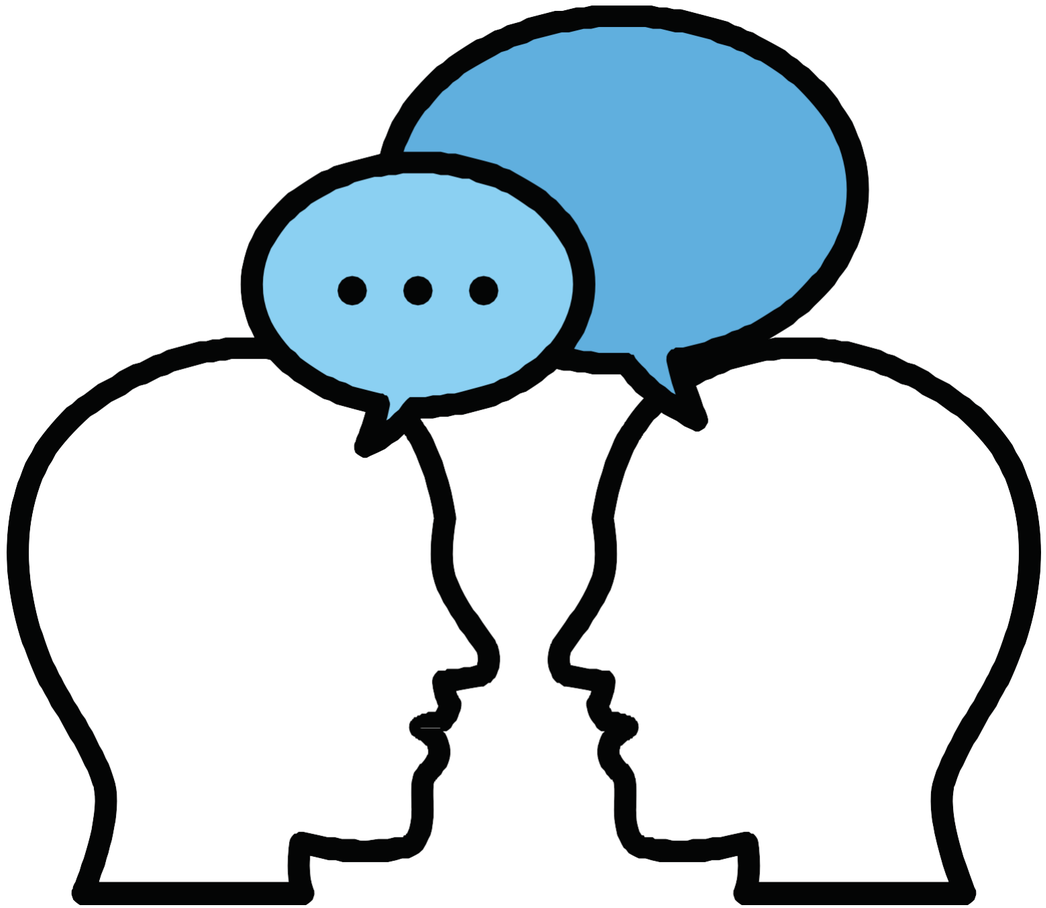
Communication - Internal

- Inter Integrated Circuit (I2C)
- Serial Peripheral Interface (SPI)
- Universal Asynchronous Receiver Transmitter (UART)



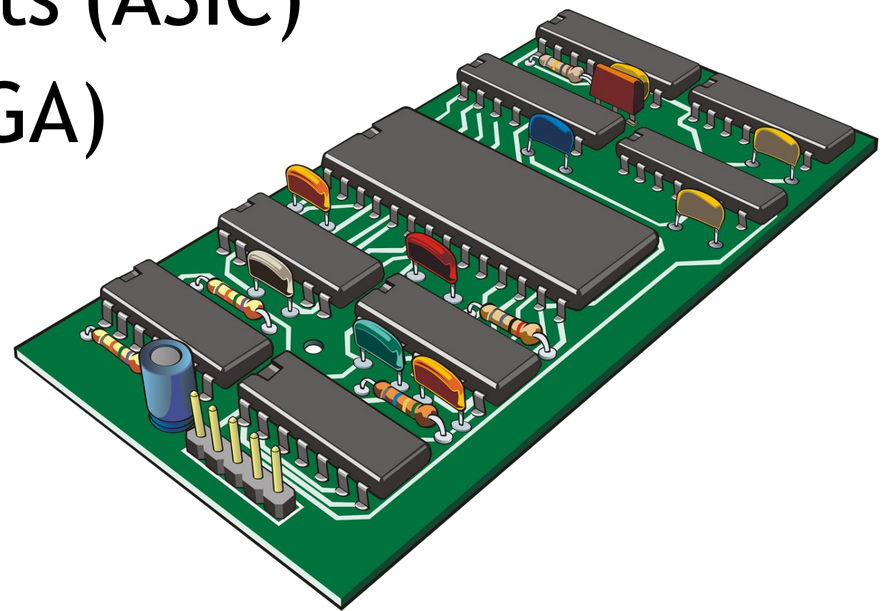
Communication - External

- RS-232 C & RS-485
- Universal Serial Bus (USB)
- Bluetooth
- Wi-Fi
- Zig Bee



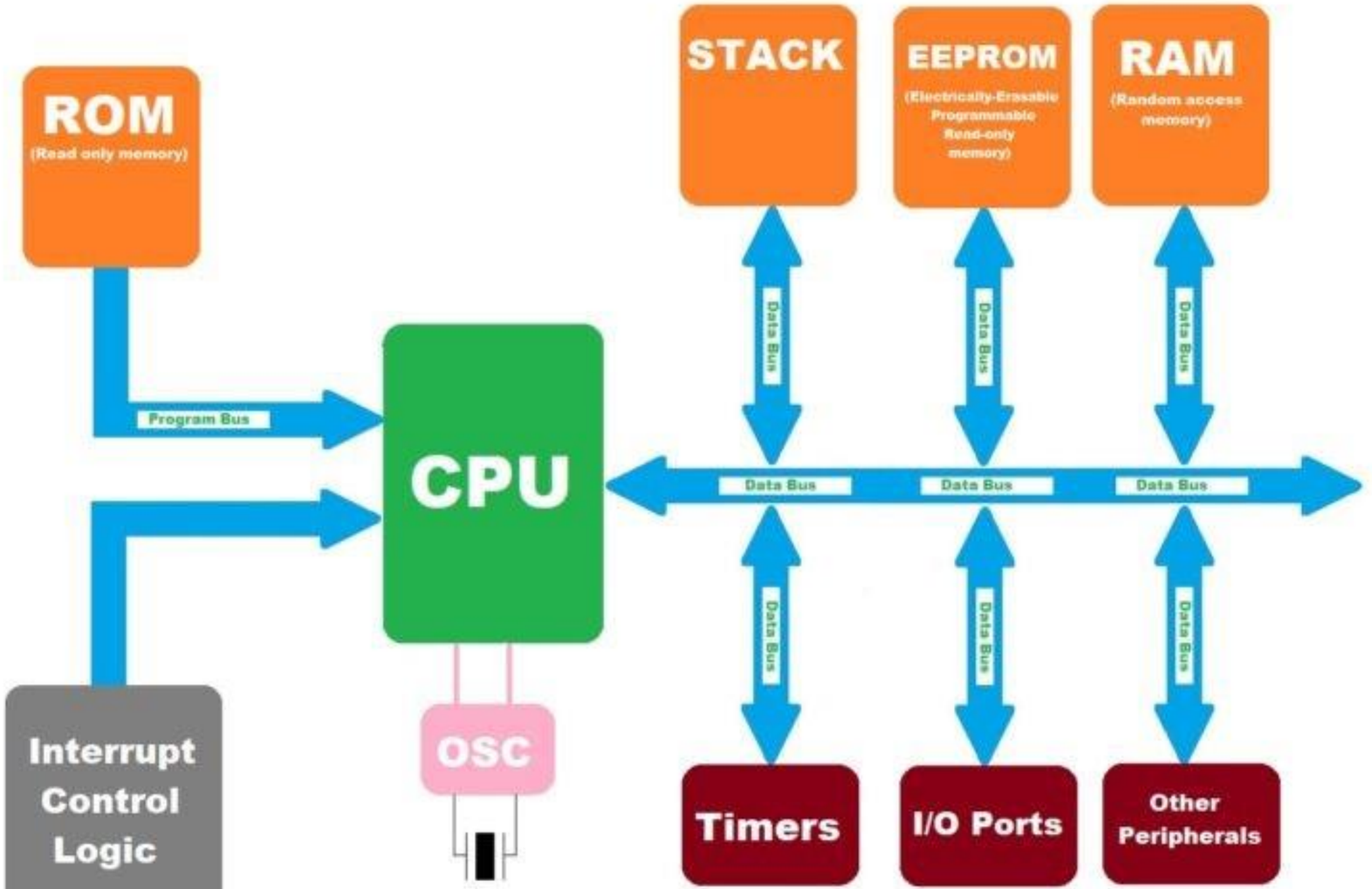
Brains

- Microcontrollers
- SoC
- digital signal processors (DSP)
- application-specific integrated circuits (ASIC)
- field-programmable gate arrays (FPGA)

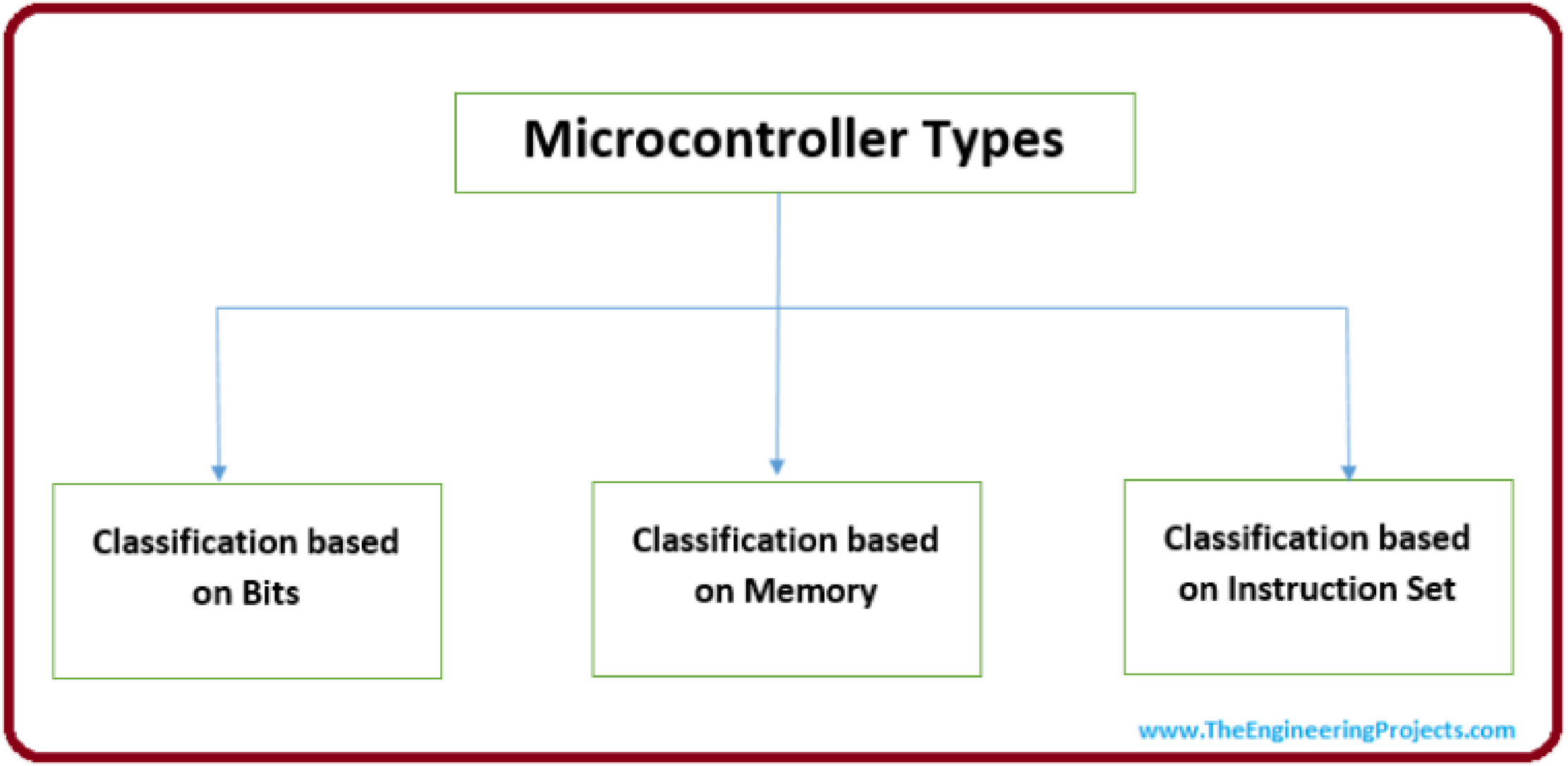


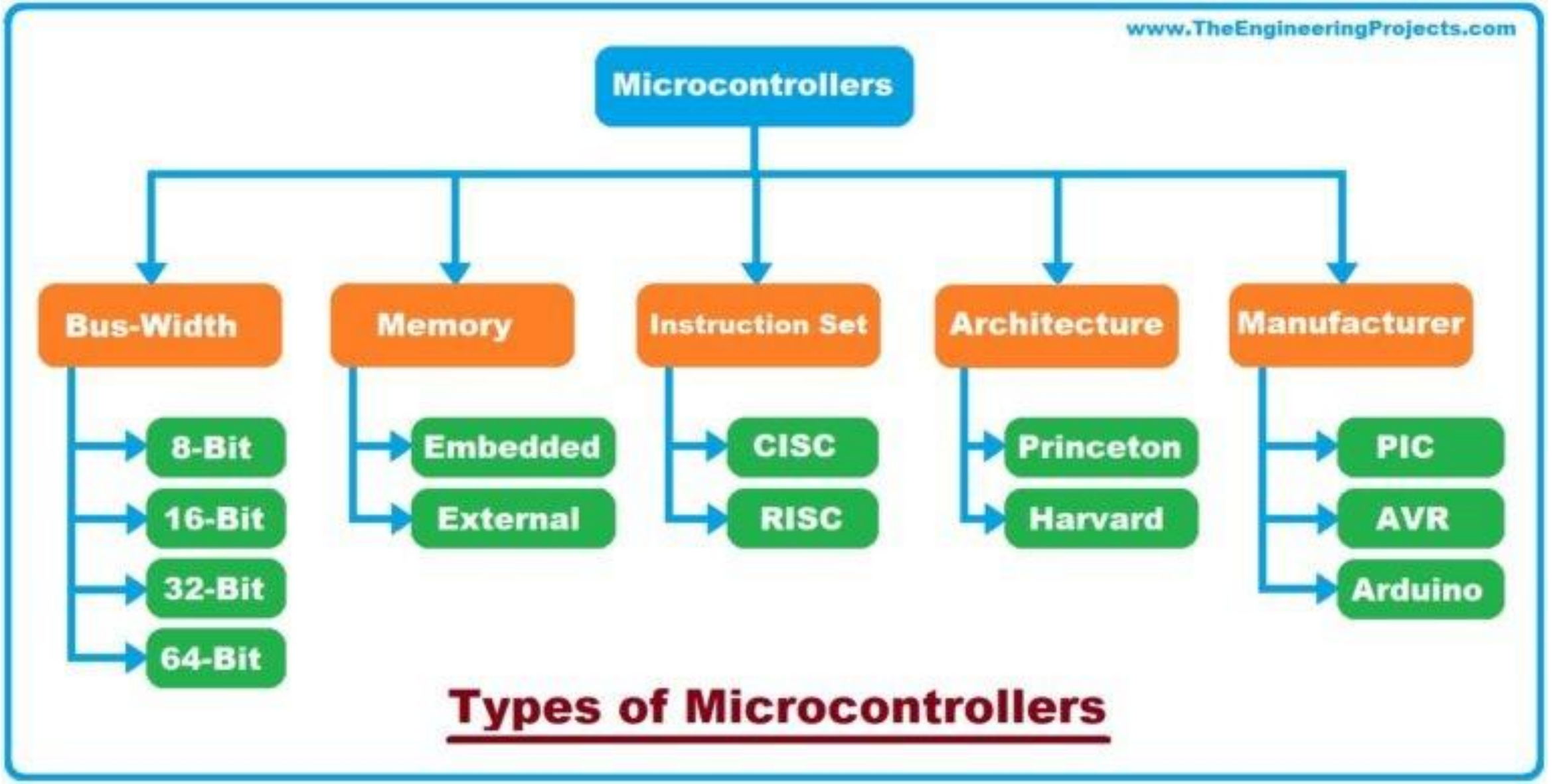
Microcontrollers Architechture

(What is a Microcontroller ?)



www.TheEngineeringProjects.com


Microcontroller Types
Classification based on Bits
Classification based on Memory
Classification based on Instruction Set
www.TheEngineeringProjects.com


www.TheEngineeringProjects.com
Microcontrollers
Bus-Width
Memory
Instruction Set
Architecture
Manufacturer
8-Bit
16-Bit
32-Bit
64-Bit
Embedded
External
CISC
RISC
Princeton
Harvard
PIC
AVR
Arduino
Types of Microcontrollers

Atmel AVR

AVR

ATX Mega

ATmega 328P

PIC 18F877A

8051

Arduino

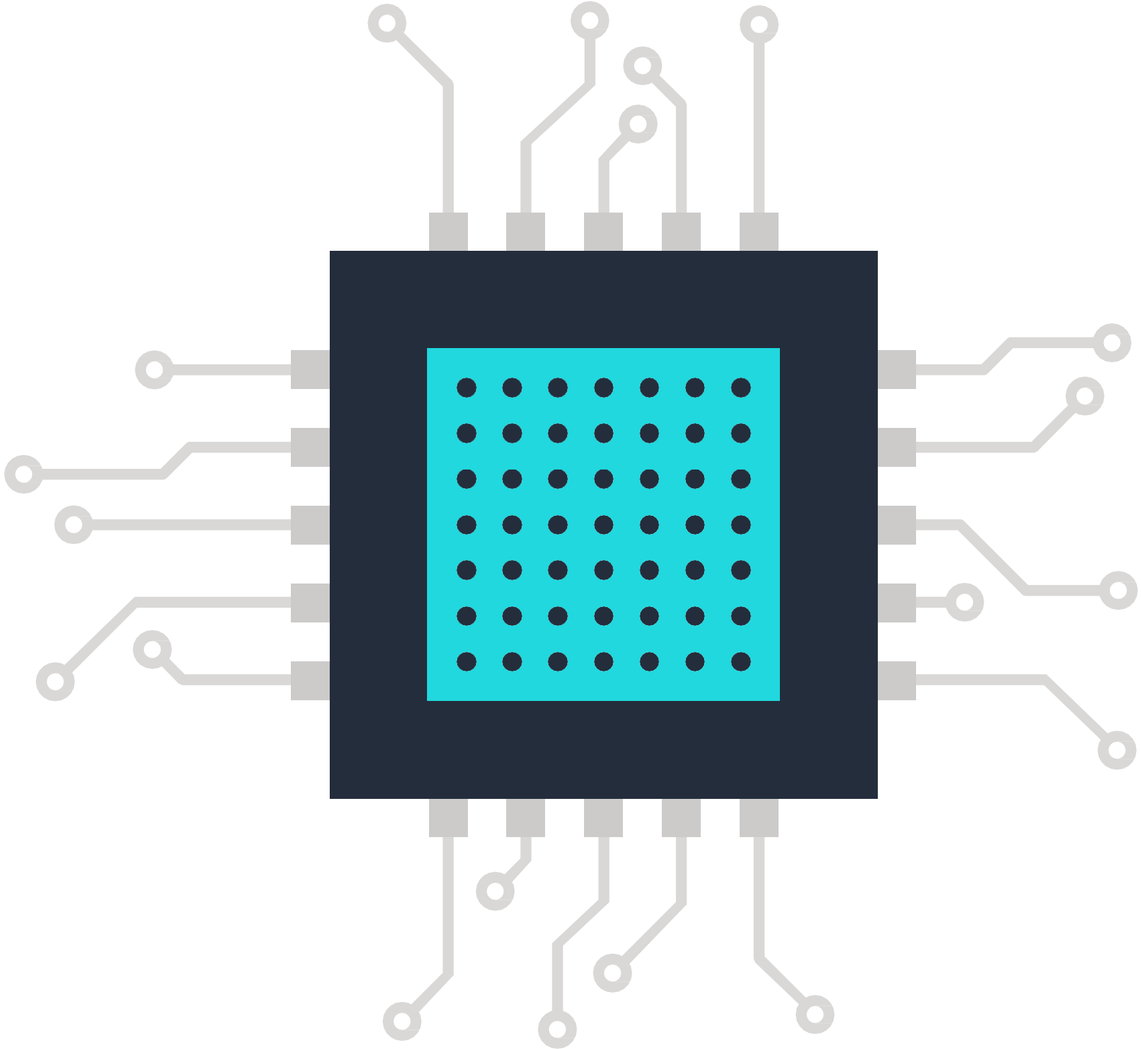
ARM

Microcontrollers Vs. Microprocessors

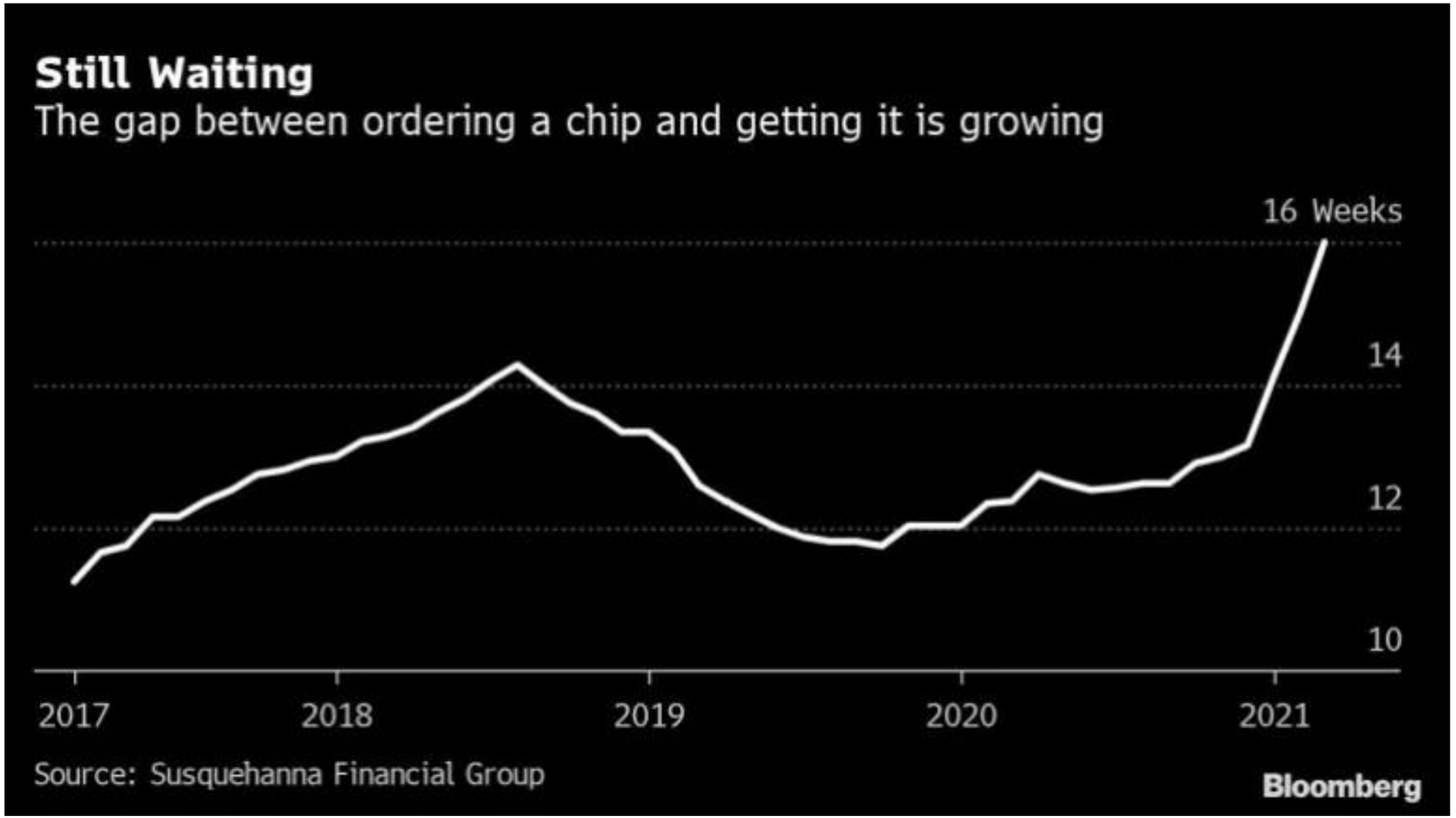
Summary	Microprocessor	Microcontroller
Applications	Advanced data processing, video, computer vision, personal computers, fast communications, multi-core computation.	Embedded devices, control systems, smartphones, consumer electronics.
Processing Power	Higher	Lower
Memory	External - Flexible	Internal – Limited Size
Power Consumption	Higher	Lower
Size	Larger	Smaller
Price	Expensive	Cheaper
I/O	Need external peripherals with I/O pins	Programmable digital and analog I/O pins

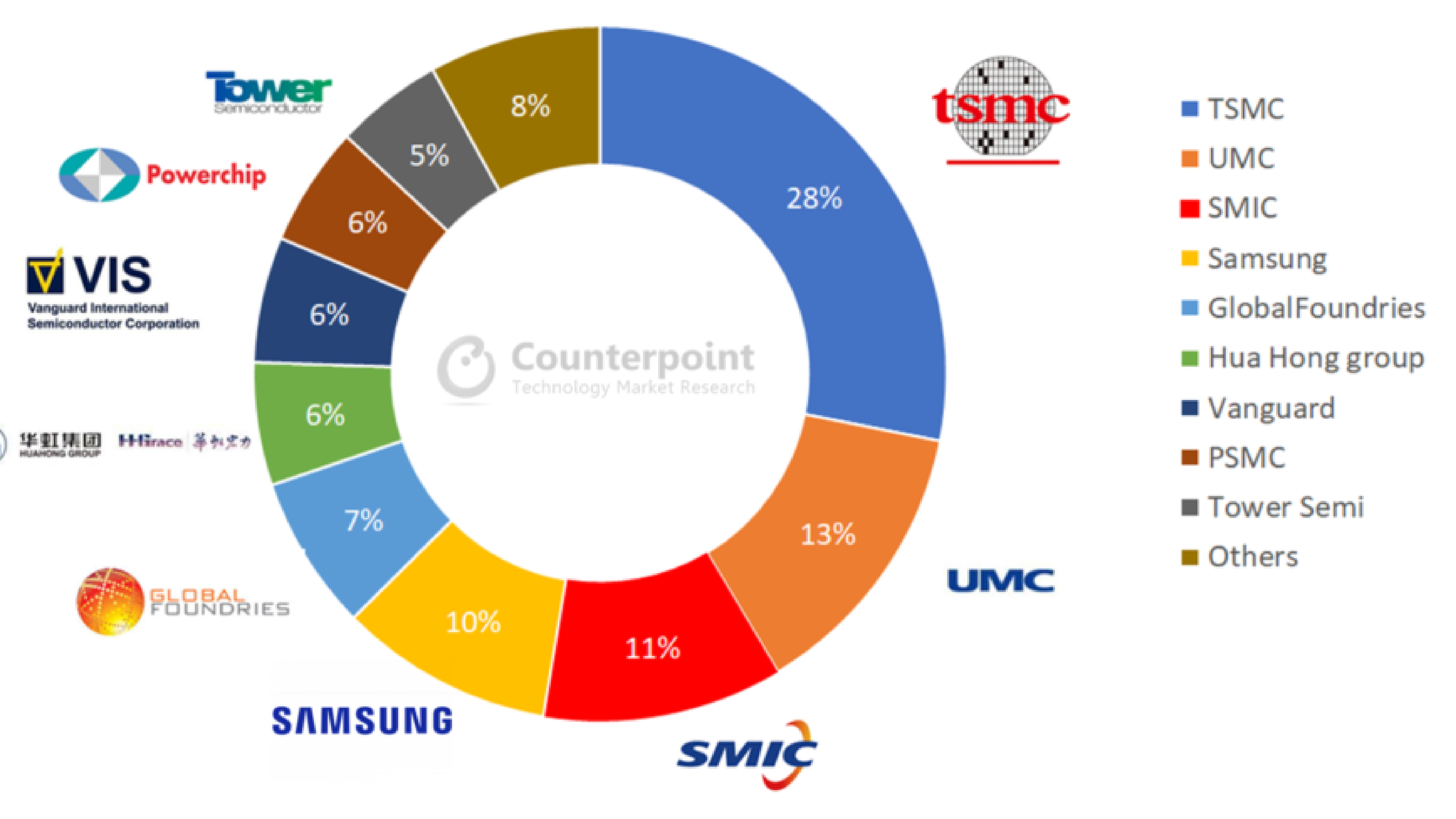
S o C

- Processor/multiple processors
- I/O Handling
- Communications
- Connectivity



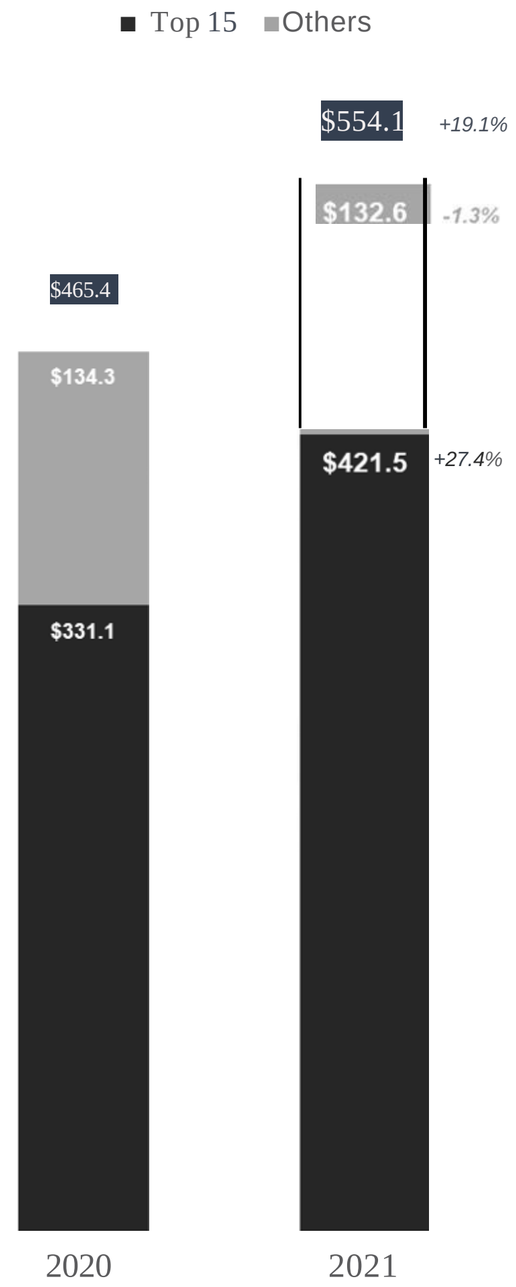
Global chip shortage




Tower Semiconductor
Powerchip
VIS
Vanguard International Semiconductor Corporation
华虹集团 HUAHONG GROUP
HHGrace 華虹宏力
GLOBAL FOUNDRIES
SAMSUNG
SMIC
tsmc
UMC
28%
13%
11%
10%
7%
6%
6%
6%
5%
8%
Counterpoint
Technology Market Research
TSMC
UMC
SMIC
Samsung
GlobalFoundries
Hua Hong group
Vanguard
PSMC
Tower Semi
Others

Total Semiconductor Industry Revenues

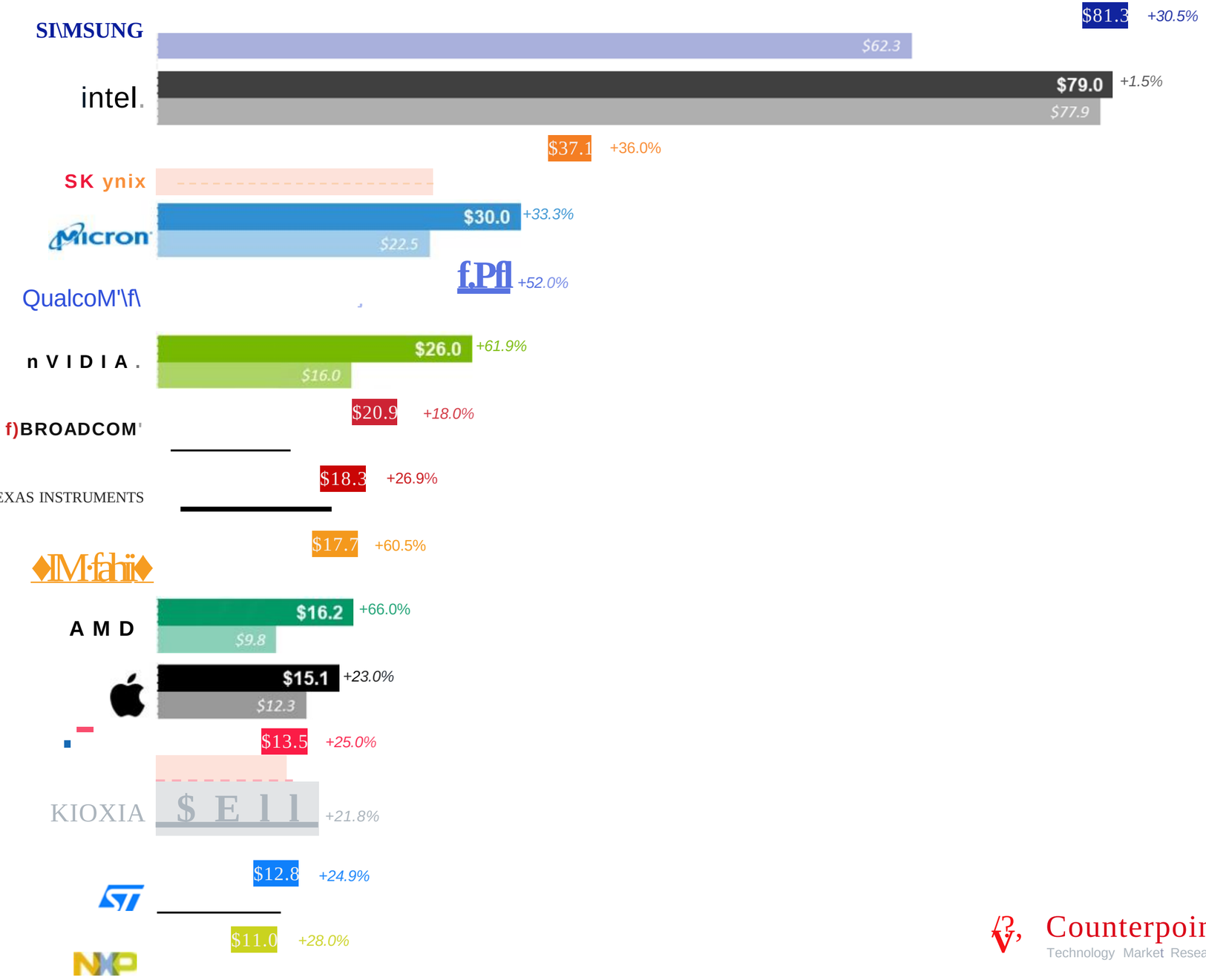
{US$, Billions}



2021 Rankings

2021 vs *2020* Semiconductor Companies Revenues

{US$, Billions}



Counterpoint
Technology Market Research

Advantages

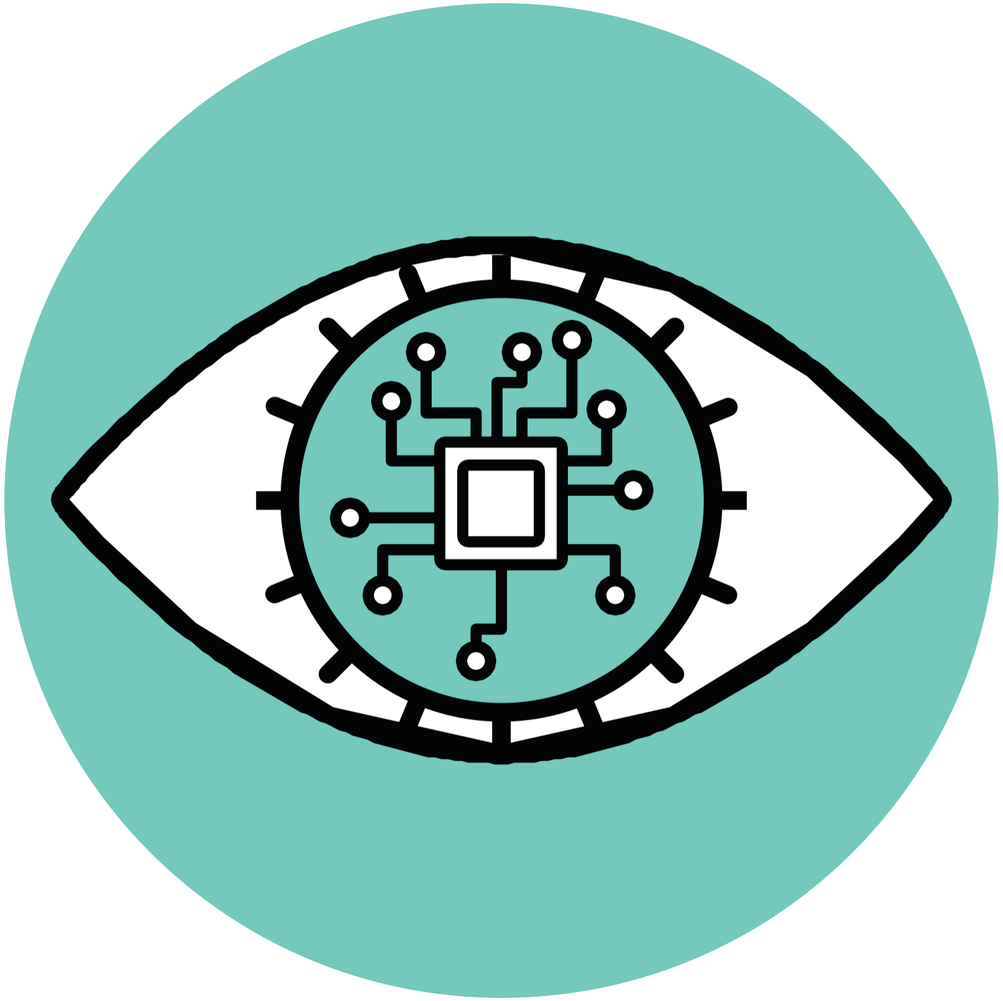
- Small size
- Many applications do not require as much computing power
- Reduced power consumption
- Reduced design cost

Threats

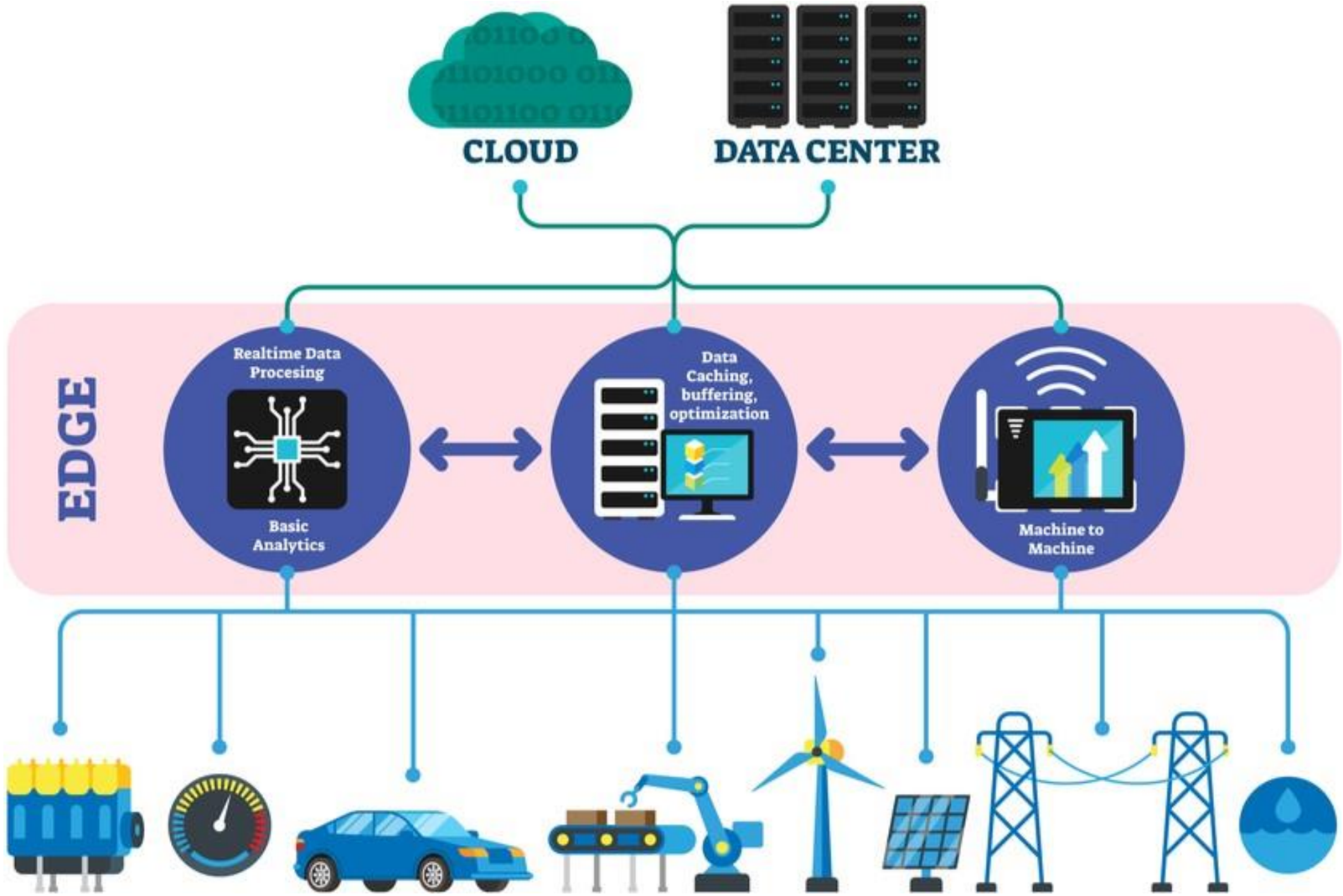
- Wired communication peripherals
- Unsecured wireless communication
- Access via internet
- Physical security
- Data storage

Future of ES

- AI
- VR
- AR
- IoT



Future of ES


CLOUD
DATA CENTER
EDGE
Realtime Data Procesing
Basic Analytics
Data Caching, buffering, optimization
Machine to Machine

100 Billion $
by 2025