INTERNET OF THINGS (IOT)

Asst. Prof. DR. MUHANED TH. M. AL-HASHIMI

Tikrit University

Collage Of Computer And Mathematical Science

2024 - 2025

IOT PROCESSING TOPOLOGIES *AND* TYPES

LECTURE (5)

2204 - 2025

18 Of March

Lecture outline

1. Data Format
2. Importance of Processing in IoT
3. Processing Topologies
4. IoT Device Design and Selection Considerations
5. Processing Offloading

Data Format

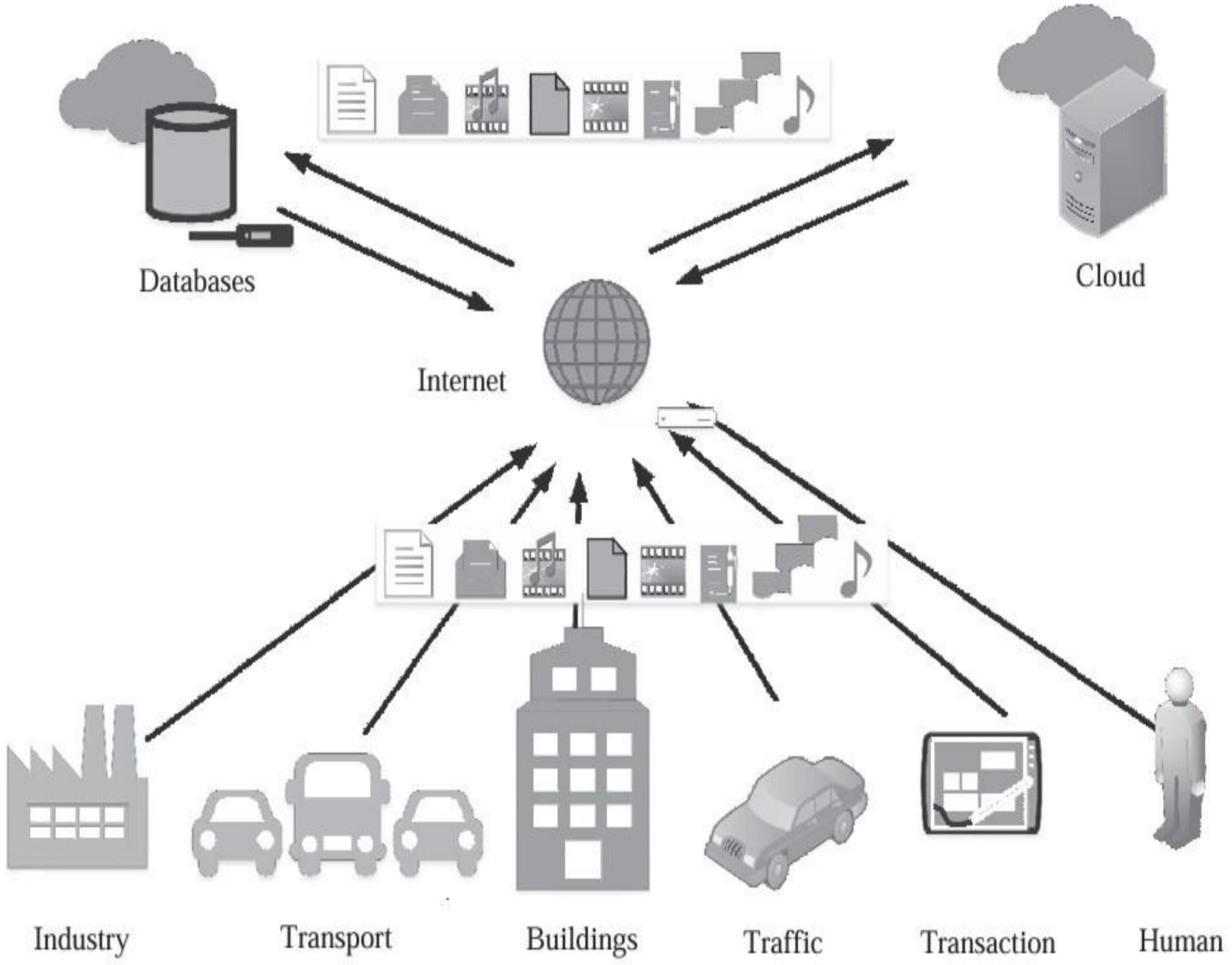
- The **Internet** is a vast space where **huge quantities and varieties of data are generated** regularly and flow freely.
- The **massive volume of data generated** by this huge number of users is further **enhanced by the multiple devices utilized by most users**.
- In addition to these data-generating sources, non-human data generation sources, such as sensor nodes and automated monitoring systems, further add to the data load on the Internet.
- This **huge data volume is composed of a variety of data** such as e-mails, text documents (Word docs, PDFs, and others), social media posts, videos, audio files, and images.

Data Format

- However, these **data** can be broadly <u>grouped into two types </u>based on **how they can be accessed** and **stored**:

  1) Structured data
  2) unstructured data

  as shown in the Figure below.



- The **figure** illustrates the various data-generating and storage sources connected to the Internet and the plethora التنوع الهائل of data types contained within it

Data Format

1) Structured data

- These are typically **text data** that have a pre-defined structure.
- Structured data are associated with relational database management systems (**RDBMS**).
- These are primarily **created** by using **length-limited data fields** such as phone numbers, social security numbers, and other such information.
- Even if the data is **human** or **machine generated**, these data are **easily searchable** by querying algorithms as well as human-generated queries.
- **Common usage** of this type of data is associated with **flight or train reservation systems**, **banking systems**, **inventory controls**, and other similar systems.
- Established **languages** such as Structured Query Language (**SQL**) are used for accessing these data in RDBMS.
- However, in the context of IoT, structured data holds a **minor share** of the total generated data over the Internet.

Data Format

2) Unstructured data

- In simple words, **all the data on the Internet**, which is **not structured**, is categorized as unstructured.
- These data types have **no pre-defined structure** and can **vary** according to applications and data-generating sources.
- Some of the common **examples** of human-generated unstructured data include text, e-mails, videos, images, phone recordings, chats, and others.
- Some common examples of machine-generated unstructured data include sensor data from traffic, buildings, industries, satellite imagery, surveillance videos, and others.
- As already evident from its examples, this data type **does not have fixed formats** associated with it, which makes it **very difficult for querying** algorithms to perform a look-up.

Importance of Processing in IoT

- The vast **amount and types of data** flowing through the Internet **require intelligent processing techniques**.
- This requirement has **become even more crucial** with the rapid advancements in IoT, which is **laying enormous pressure** on the **existing network** infrastructure globally.
- Given these **urgencies**, it is **important to decide**:
  - ✓ when to process and
  - ✓ what to process?
- Before deciding upon the processing to pursue, we first divide the data to be processed into three types based on the **urgency of processing**:

1) Very time critical.
2) time critical.
3) normal.

Importance of Processing in IoT

- Data from sources such as flight control systems, healthcare, and other such sources, which need **immediate decision support**, are deemed as very critical. These **data have a very low threshold of processing latency**, typically in the range of a **few milliseconds**.

- Data from sources that can **tolerate normal processing latency** are deemed as time critical data. These data, generally associated with sources such as vehicles, traffic, machine systems, smart home systems, surveillance systems, and others, which can tolerate a latency of a **few seconds**, fall in this category.

- Finally, the last category of data, normal data, can **tolerate a processing latency** of a **few minutes to a few hours** and are typically associated with less data-sensitive domains such as agriculture, environmental monitoring, and others.

Importance of Processing in IoT

- Considering the requirements of data processing, the **processing requirements** of data from very time-critical sources are exceptionally **high**.
- ✓ Here, the need for processing the data **in place** or **almost nearer to the source** is crucial in achieving the deployment success of such domains.

- Similarly, considering the requirements of processing from category 2 data sources (time-critical), the processing requirements allow for the transmission of data to be processed to **remote locations/processors** such as clouds or through collaborative processing.

- Finally, the last category of data sources (normal) typically have no particular time requirements for processing urgently and are **pursued leisurely على مهل as such**.

Processing Topologies

- The identification and intelligent selection of the processing requirement of an IoT application are one of the crucial steps in deciding the architecture of the deployment.

- A properly designed IoT architecture would result in massive **savings in network bandwidth** and **conserve** significant amounts of overall **energy** in the architecture while **providing the proper and allowable** processing **latencies** for the solutions associated with the architecture.

Processing Topologies

- Regarding the importance of processing in IoT, we can divide the various **processing solutions** into two large topologies:

1) On-site.
2) Off-site.

- The off-site processing topology can be **further divided** into the following:

1) Remote processing.
2) Collaborative processing.

Processing Topologies / On-site

- **The on-site processing topology** signifies that the **data is processed at the source itself**.

- ✓ This is crucial in **applications** that have a very low tolerance for latencies.
- ✓ These **latencies may result** from the processing hardware or the network (**during transmission of the data** for processing **a way from** the processor).

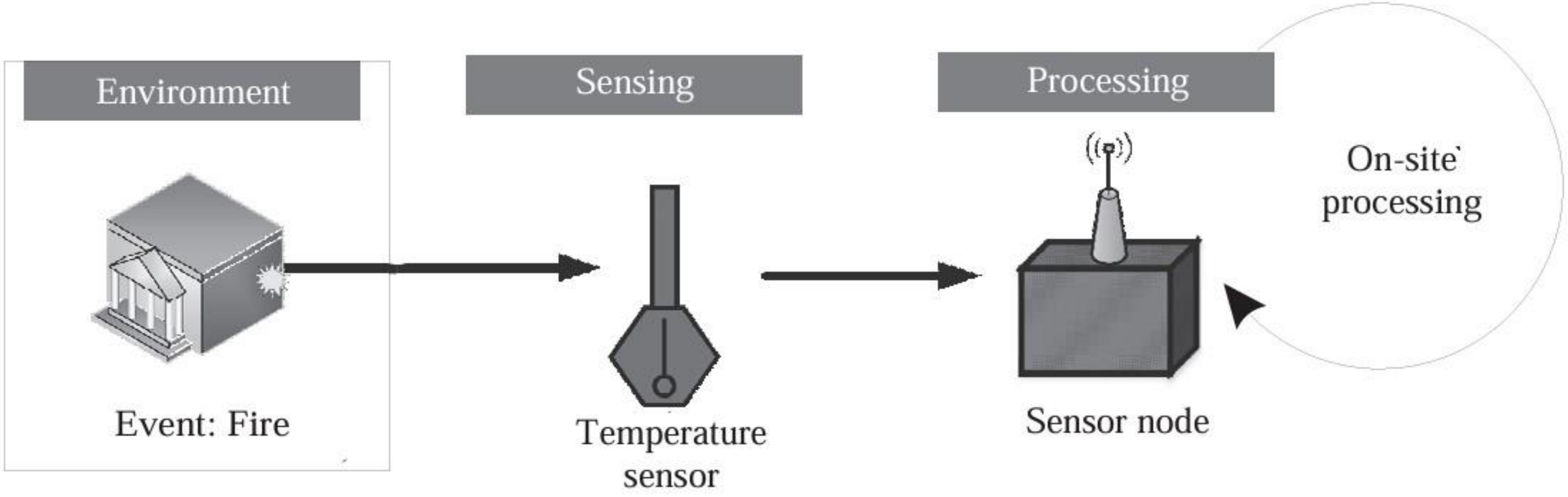
- Applications such as those associated with **healthcare** and **flight control systems** (real-time systems) have a breakneck data generation rate.

- ✓ These additionally show rapid temporal changes that can be missed (leading to catastrophic damages) unless the processing infrastructure is fast and robust enough to handle such data.

Processing Topologies / On-site

- The **figure** shows the **on-site processing topology**, where an event (**fire**) is detected utilizing a temperature sensor connected to a sensor node.

- The sensor node **processes** the **information** from the **sensed** event and **generates** an **alert**.
  - ✓ The node additionally has the option of forwarding the data to a remote infrastructure for further analysis and storage.



Processing Topologies / Off-site

- **The off-site processing paradigm**, as **opposed** to the on-site processing paradigms, **allows for latencies** (due to processing or network latencies); it is significantly **cheaper** than on-site processing topologies.

✓ This **difference in cost** is mainly due to the **low** demands and **requirements** of processing at the **source itself**.

✓ Often, **the sensor nodes are not required to process data on an urgent basis**, so having a dedicated and expensive on-site processing infrastructure is not sustainable for large-scale deployments typical of IoT deployments.

Processing Topologies / Off-site

- **In the off-site processing topology**,

- ✓ The sensor node is **responsible** for the collection and framing of data that is **transmitted to another location for processing**.
- ✓ Unlike the on-site processing topology, the off-site topology **has a few dedicated high-processing enabled devices,** which can be borrowed by multiple simple sensors to accomplish their tasks.
- ✓ At the same time, **this arrangement keeps the costs** of large-scale deployments extremely **manageable**.

Processing Topologies / Off-site

- **In the off-site topology**,

- ✓ The data from these sensor nodes (data-generating sources) is **transmitted** either to a **remote location** (which can either be a server or a cloud) or to **multiple processing nodes**.

- ✓ Multiple nodes can come together to **share their processing power** in order to collaboratively process the data (which is important in case a feasible **communication pathway or connection** to a remote location **cannot** be **established by a single node**).

Processing Topologies / Off-site / Remote processing

- Off-site / Remote processing is **one of the most common processing topologies** prevalent in present-day IoT solutions.

✓ It **senses data** by various sensor nodes; the **data** is then forwarded to a remote server or a cloud-based infrastructure for further processing and analytics.

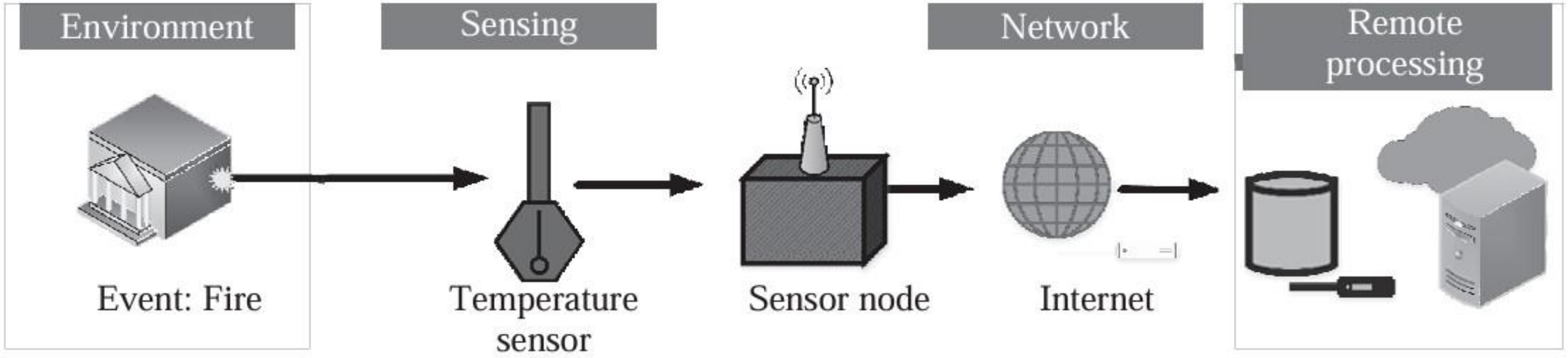
✓ The **processing of data** from hundreds and thousands of **sensor nodes** can be **simultaneously offloaded** to a single, powerful **computing platform.**

✓ This **results in massive cost and energy savings** by enabling the reuse and reallocation of the same processing resource while also enabling the deployment of smaller and simpler processing nodes at the site of deployment.

✓ This setup also **ensures massive scalability of solutions without significantly affecting the cost** of the deployment.

Processing Topologies / Off-site / Remote processing

- **Figure** shows the outline of one such paradigm, where the **sensing** of an event is performed **locally**, and the **decision-making** is **outsourced** to a remote processor (here, cloud).
- However, this paradigm tends to **use up a lot of network bandwidth** and relies heavily on the presence of **network connectivity between the sensor nodes and the remote processing infrastructure**.



Processing Topologies / Off-site / Collaborative processing

- Off-site / Collaborative processing topology typically finds use in scenarios with **limited or no network connectivity**, especially **systems lacking a backbone network**.

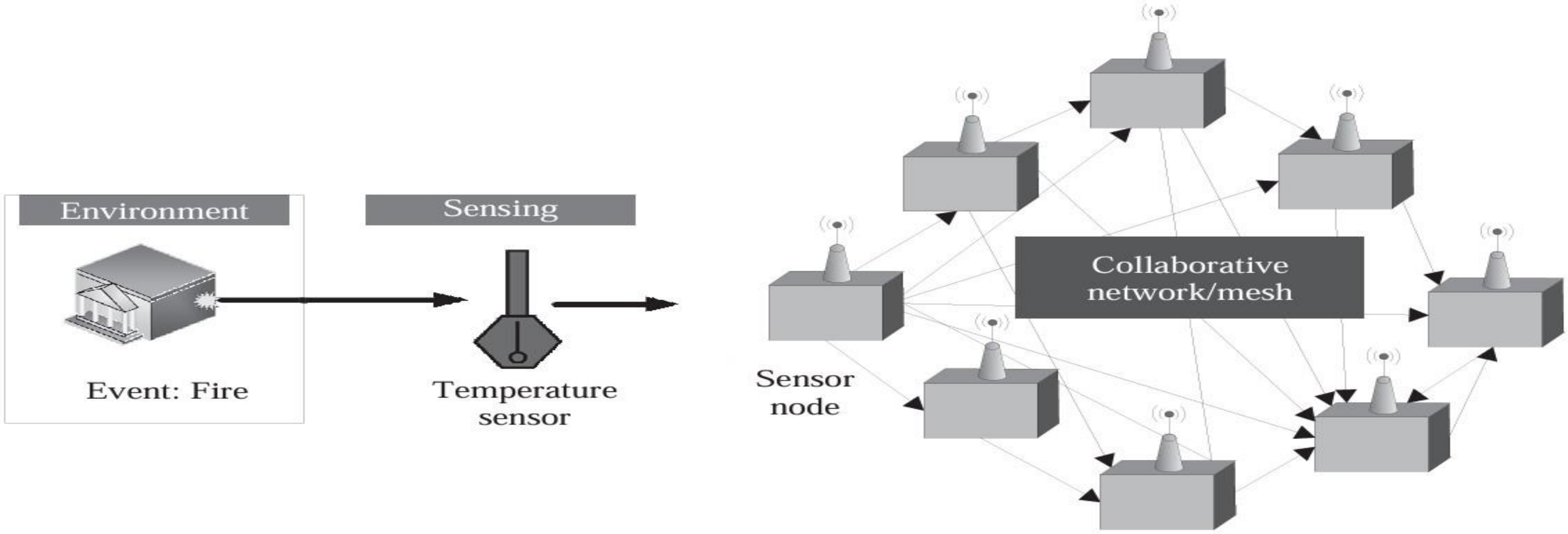
✓ This topology **can be quite economical** for large-scale deployments spread over vast areas, where providing networked access to a remote infrastructure is not viable. In such scenarios, the simplest solution is to **club together the processing power of nearby processing nodes and collaboratively process** the data in the vicinity of the data source itself.

✓ This approach also **reduces latencies due to the transfer of data over the network**.

✓ It **conserves the bandwidth** of the network, especially ones connecting to the Internet.

Processing Topologies / Off-site / Collaborative processing

- The **figure** shows the **collaborative processing topology** for collaboratively processing data **locally**.
- This topology **can be quite beneficial for applications** such as agriculture, where an intense and temporally high frequency of data processing is not required, as agricultural data is generally logged after significantly long intervals (in the range of hours). One important point to mention about this topology is the preference of mesh networks for easy implementation of this topology.



IoT Device Design and Selection Considerations

- The main consideration **of minutely defining an IoT solution** is the **selection of the processor** for developing the sensing solution (i.e., the sensor node).

- The main factor **governing the IoT device design and selection** for various applications is the **processor**.

✓ However, the other important considerations are as follows.

1) **Size**.
2) **Energy**.
3) **Cost**.
4) **Memory**.
5) **Processing power**.
6) **I/O rating**.
7) **Add-ons**.

IoT Device Design and Selection Considerations

1) **Size:**

This is one of the crucial factors for deciding the form factor and the energy consumption of a sensor node. It has been observed that larger the form factor, larger is the energy consumption of the hardware. Additionally, large form factors are not suitable for a significant bulk of IoT applications, which rely on minimal form factor solutions (e.g., wearables).

IoT Device Design and Selection Considerations

2) **Energy**:

The energy requirements of a processor are the most important deciding factor in designing IoT-based sensing solutions. The higher the energy requirements, the higher the energy source (battery) replacement frequency. This principle automatically lowers the long-term sustainability of sensing hardware, especially for IoT-based applications.

IoT Device Design and Selection Considerations

3) Cost:

The cost of a processor, besides the cost of sensors, is the driving force in deciding the density of deployment of sensor nodes for IoT-based solutions. The cheaper cost of the hardware enables a much higher density of hardware deployment by users of an IoT solution. For example, cheaper gas and fire detection solutions would enable users to include much more sensing hardware for a lesser cost.

IoT Device Design and Selection Considerations

4) Memory:

The memory requirements (both volatile and non-volatile memory) of IoT devices determines the capabilities the device can be armed with. Features such as local data processing, data storage, data filtering, data formatting, and a host of other features rely heavily on the memory capabilities of devices. However, devices with higher memory tend to be costlier for obvious reasons.

IoT Device Design and Selection Considerations

5) Processing power:

As covered in earlier sections, processing power is vital (comparable to memory) in deciding what type of sensors can be accommodated with the IoT device/node, and what processing features can integrate on-site with the IoT device. The processing power also decides the type of applications the device can be associated with. Typically, applications that handle video and image data require IoT devices with higher processing power as compared to applications requiring simple sensing of the environment.

IoT Device Design and Selection Considerations

6) **I/O rating**:

The input–output (I/O) rating of an IoT device, primarily the processor, is the deciding factor in determining the circuit complexity, energy usage, and requirements for support of various sensing solutions and sensor types. Newer processors have a meager I/O voltage rating of 3.3 V, as compared to 5 V for the somewhat older processors. This translates to requiring additional voltage and logic conversion circuitry to interface legacy technologies and sensors with the newer processors. Despite low power consumption due to reduced I/O voltage levels, this additional voltage and circuitry not only affect the complexity of the circuits but also affect the costs.

IoT Device Design and Selection Considerations

7) Add-ons:

The support of various add-ons a processor or for that matter, an IoT device provides, such as analog to digital conversion (ADC) units, in-built clock circuits, connections to USB and ethernet, inbuilt wireless access capabilities, and others helps in defining the robustness and usability of a processor or IoT device in various application scenarios. Additionally, the provision for these add-ons also decides how fast a solution can be developed, especially the hardware part of the whole IoT application. As interfacing and integration of systems at the circuit level can be daunting to the uninitiated, the prior presence of these options with the processor makes the processor or device highly lucrative to the users/ developers.

Processing Offloading

- The processing offloading paradigm is **important** for the development of densely deployable, energy-conserving, miniaturized, and cheap IoT-based solutions for sensing tasks.

- **Building** upon the basics of the **off-site processing topology**, we delve a bit further into the various nuances of **processing offloading in IoT**.

✓ **Data offloading is divided into three parts:**

1) offload location (which outlines where all the processing can be offloaded in the IoT architecture).
2) offload decision making (how to choose where to offload the processing to and by how much).
3) offloading considerations (deciding when to offload).

Acknowledgment

- **These lecture slides are based on:**

1) **Chapter 6 (P 115-127)** from the book "Introduction to IoT" by (Sudip Misra, Anandarup Mukherjee, Arijit Roy).

Basics of Networking

END OF LECTURE (3)

Keep connected with the classroom

btukscx

THANK YOU FOR YOUR ATTENTION