L3: Smart Objects Hardware

IT3779 Smart Object Technologies

Outline

- What is a Smart Object?
- Smart Object Architecture
  - Microcontroller
  - Communication sub system
  - Sensor sub system
  - Power sub system
- Smart Object Power Management Techniques

What is a Smart Object?

- A Smart Object is essentially, a tiny computer whose primary functions are:
  - Sense and collect info from the environment.
  - Communicate with other objects or systems over network
  - Report their own info and status
  - Receives information to update their own status

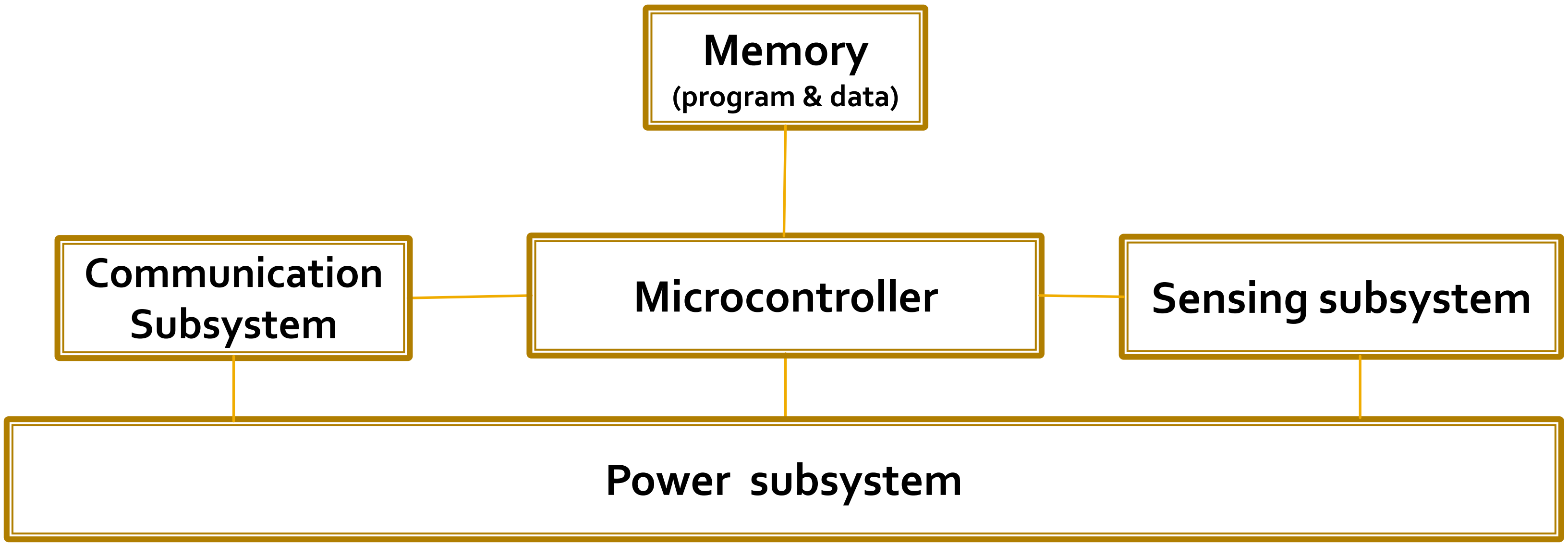
What is a Smart Object?

To be practical, Smart Objects (SO) should have the following features:

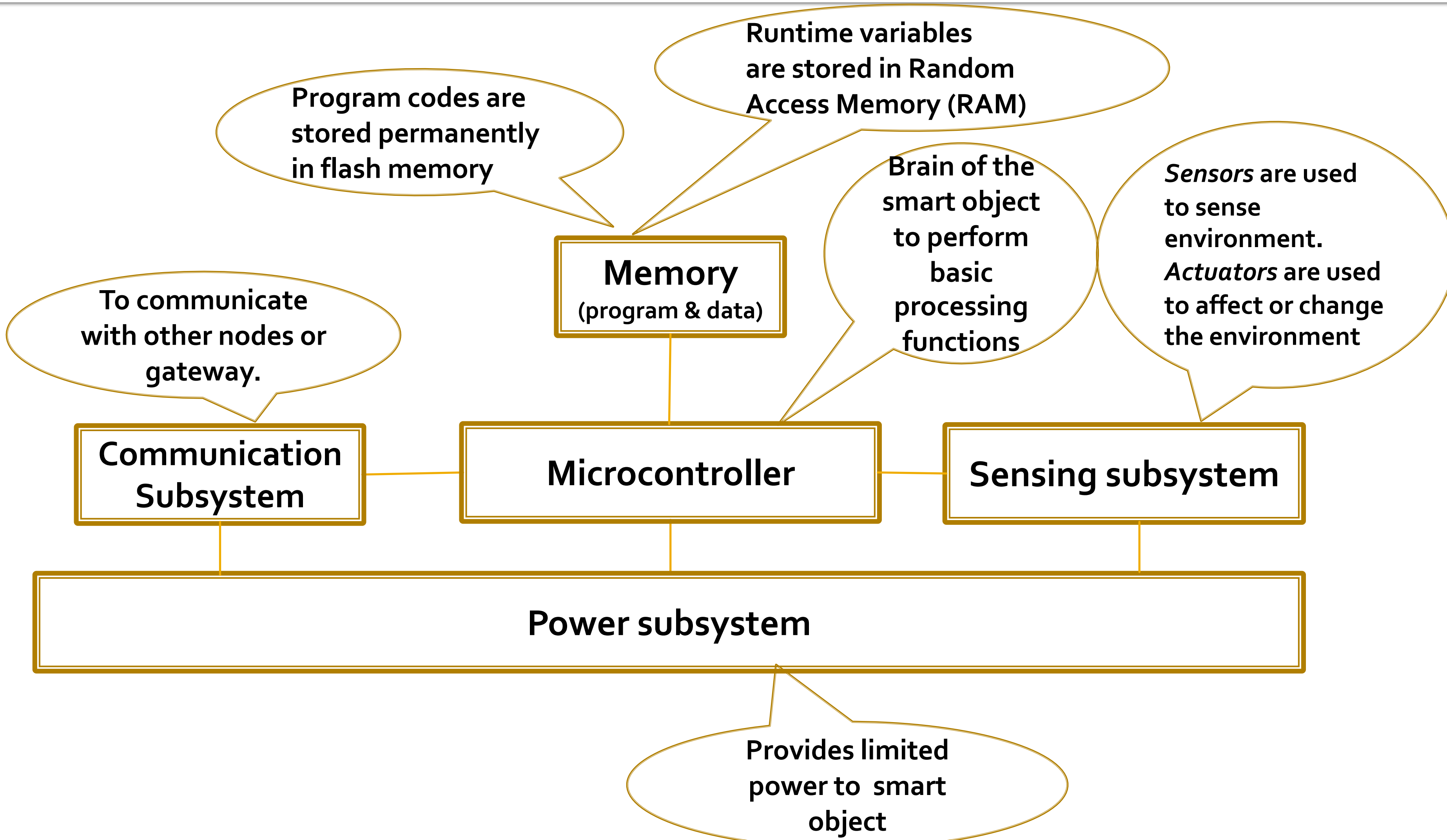
- **Tiny in size**
  - Typically installed in large-scale, deployments & have to be inconspicuous & easily deployed
- **Low cost**
  - May be in thousands & hence need to be low cost
- **Low energy consumption**
  - Designed to be disposable as it is not practical to replace batteries in the thousands in remote locations.

Typical Smart Object Architecture

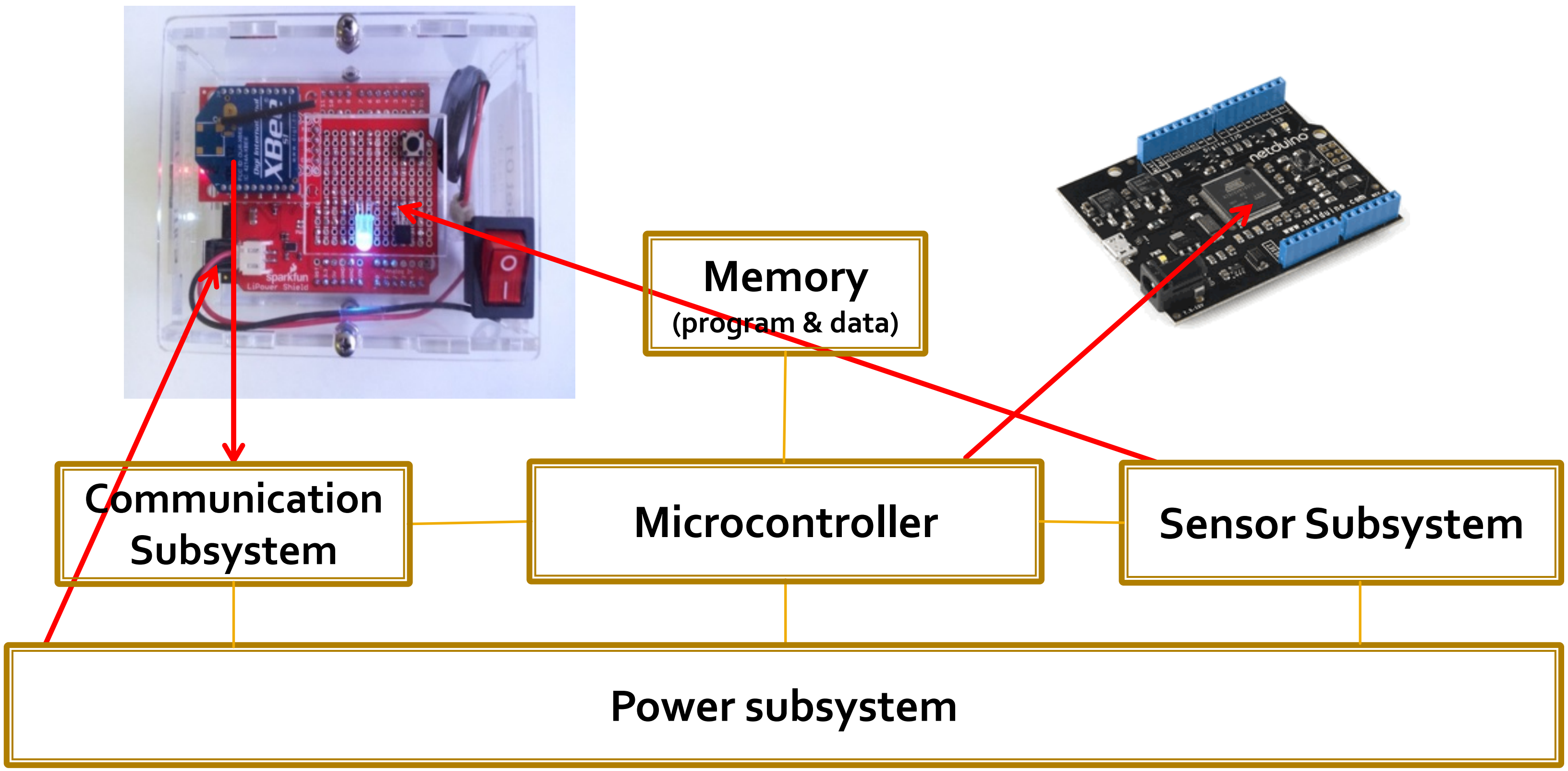
- The basic components of a Smart Object:
  - Microcontroller with memory
  - Sensing subsystem
  - Communication subsystem



Typical Smart Object Architecture



Example of Smart Objects platform : Netduino Hardware



Smart Object implementation using Netduino Hardware

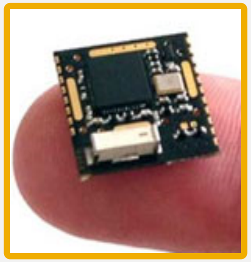
Examples of Smart Objects platforms

Libelium Waspmote

An open source wireless sensor platform specially focused on the implementation of low consumption modes to allow the sensor nodes to be completely autonomous and battery powered

Raspberry Pi

A single-board computer developed in the UK by the Raspberry Pi Foundation. The Raspberry Pi is a credit-card sized computer that plugs into your TV and a keyboard



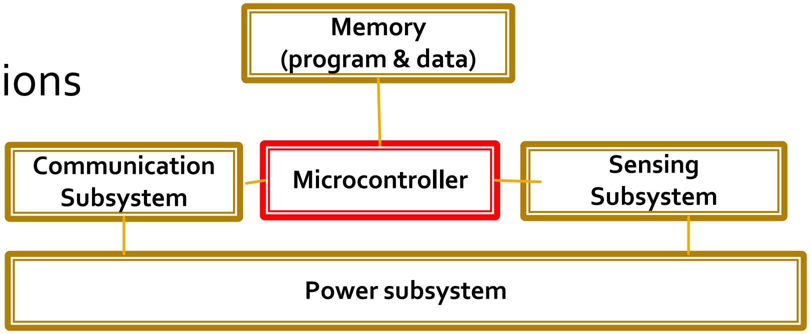
RFduino

A finger-tip sized wireless enabled microcontroller

Examples of Other Smart Object Hardware

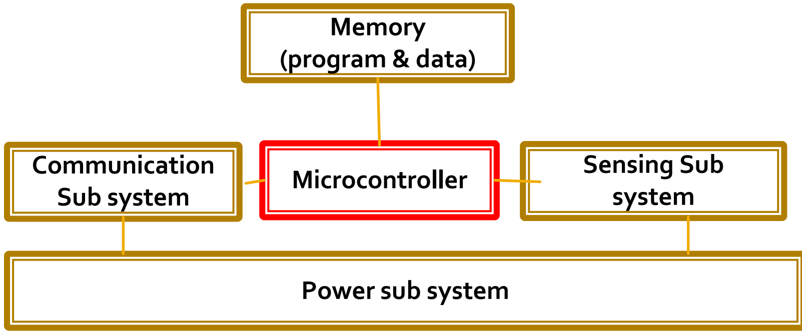
Microcontroller

- Basic Functions
  - Acts as the brain of the Smart Object
  - Processes data e.g. reading & converting raw data read from sensors
  - Controls the functionality of other components e.g. turning off communications when not in use



- Basic characteristics
  - Typically low cost
  - Limited processing power & in-built memory for storage
  - Flexibility to connect to other devices
  - Ease of programming
  - Low power consumption

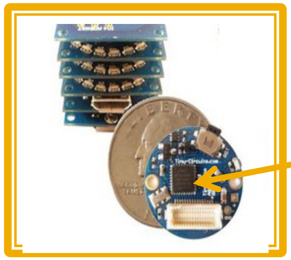
Microcontroller



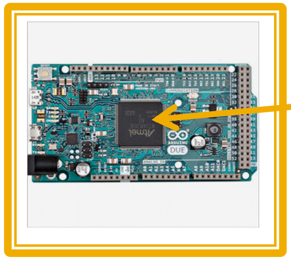
- Microcontroller has both Volatile & Non-Volatile Memory
- Non-volatile memory:
  - Can retain the stored information even when not powered
  - Typically used for secondary or long-term persistent storage.
  - Example of Non Volatile memory is Flash memory : typically used in Smart Objects for storing program code
- Volatile memory:
  - Requires power to maintain the stored information
  - Example is Random Access Memory (RAM): used for program execution & storing variables

Microcontroller : Examples

- ✓ Arduino Uno
- ✓ ATmega328 microcontroller
- ✓ 32 KB flash memory with read-while-write capabilities, 1 KB EEPROM, 2 KB SRAM



- ✓ TinyDuino
- ✓ Atmega328P processor
- ✓ 32KB Flash, 2KB RAM, 1KB EEPROM

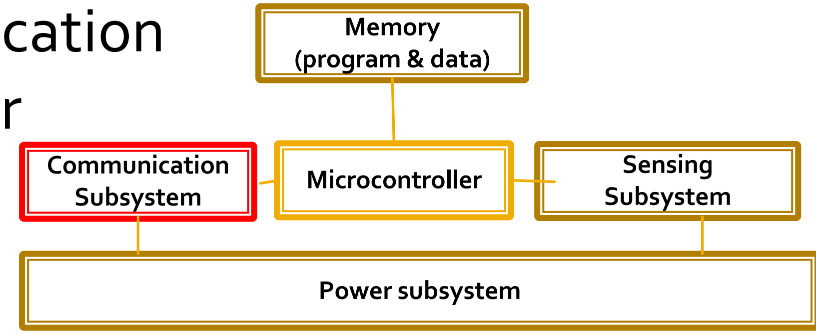


- ✓ Arduino Due
- ✓ Cortex M3 processor
- ✓ 512KB Flash, 96KB SRAM

[Libellium Video]

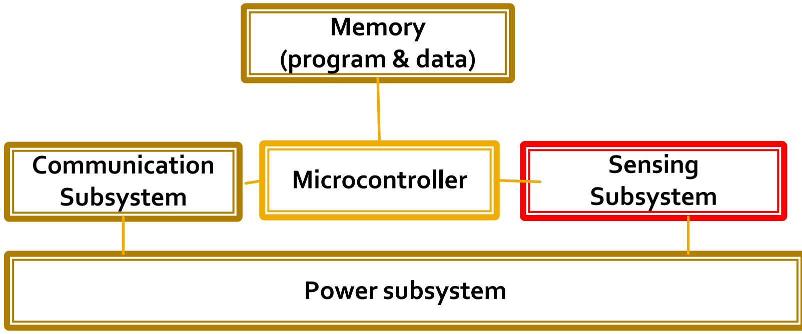
Communication Sub System

- For wireless smart object, communication module is usually a Radio Transceiver module
- Provides wireless transmission and receiving of data
- Basic characteristics
  - Communication subsystem consumes most power in a smart object - as much as 10x of microcontroller
  - Power is used to convert signals for transmission & reception
  - Listening for incoming data consumes as much power as sending. Hence, radio must be switched off to conserve power



Sensing sub system

- Smart objects interact with environment through sensing sub system

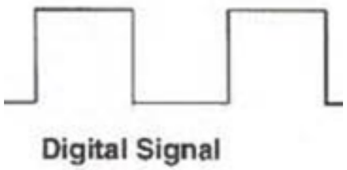


- **The sensing subsystem** measures a physical quantity from environment and converts it into a signal which can be read by the microcontroller

- The sensing subsystem comprises:
  - Sensor
  - Signal Conditioning
  - Analogue to Digital Conversion

Sensing Sub system

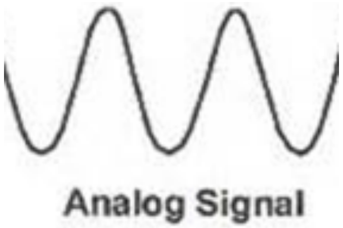
- Sensing subsystem typically interface to Digital and Analog sensors.



- Digital sensors provide a digital input (e.g. 0101001) to be read by the microcontroller

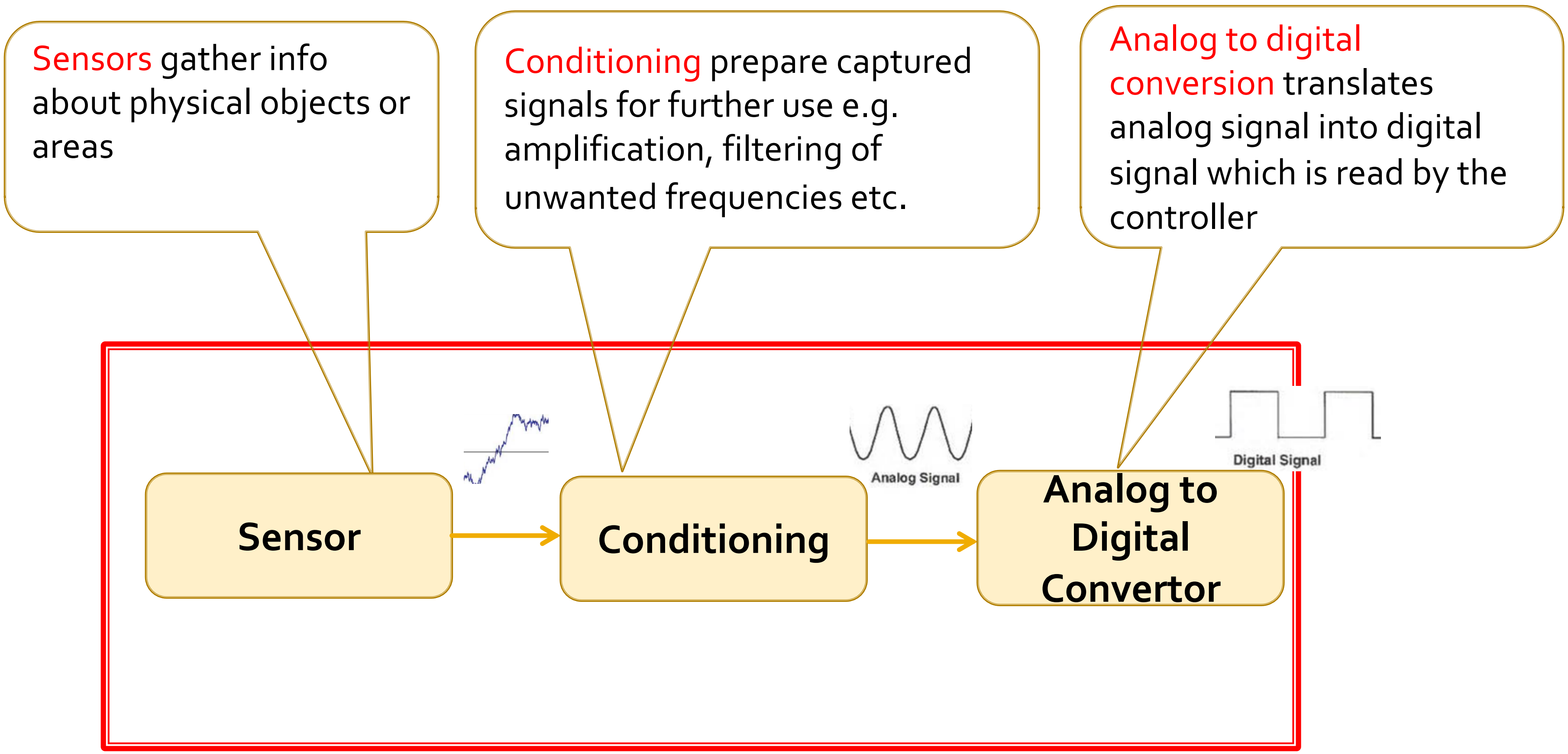
Sensing Subsystem

- Analog sensors produce a *continual* waveform



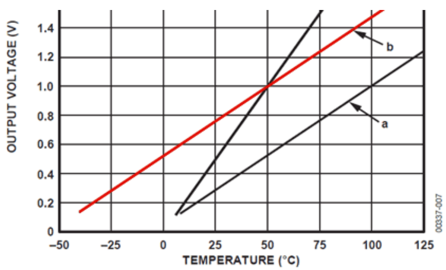
- These Analog waveforms need to be *conditioned* and *converted* to Digital signals which can be easily processed by the Microcontroller (see block diagram in next slide)

Sensing subsystem



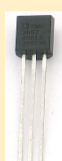
Sensing Subsystem

Example: Temperature Sensors

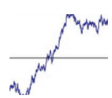


Conditioning prepare captured signals for further use e.g. amplification, filtering of unwanted frequencies etc.

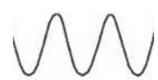
Analog to digital conversion translates analog signal into digital signal which is read by the controller



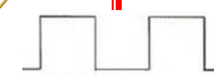
Temperature Sensor



Conditioning

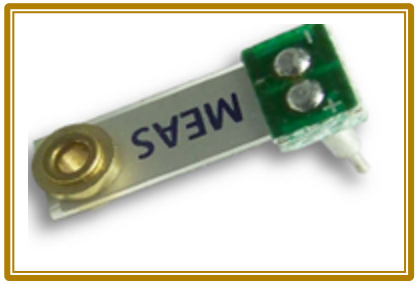


Analog to Digital Convertor

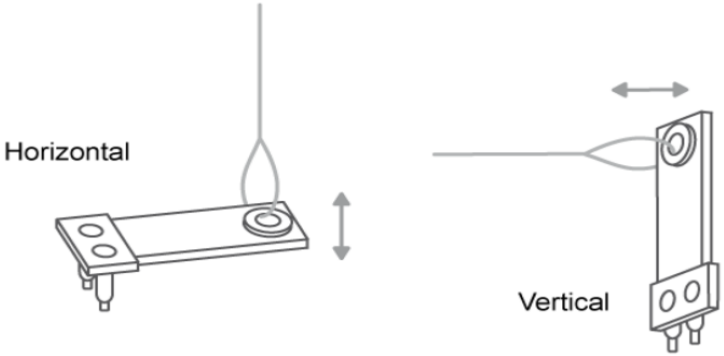


Voltage increases linearly with increase in temperature

Other Examples of Sensors

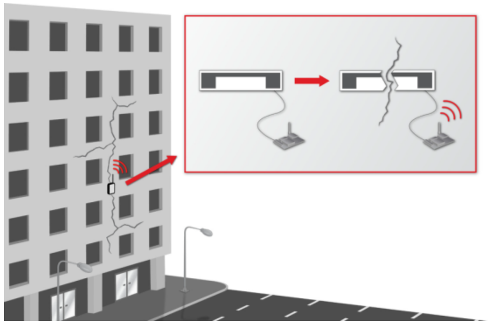


Vibration Sensor



Application of vibration sensor

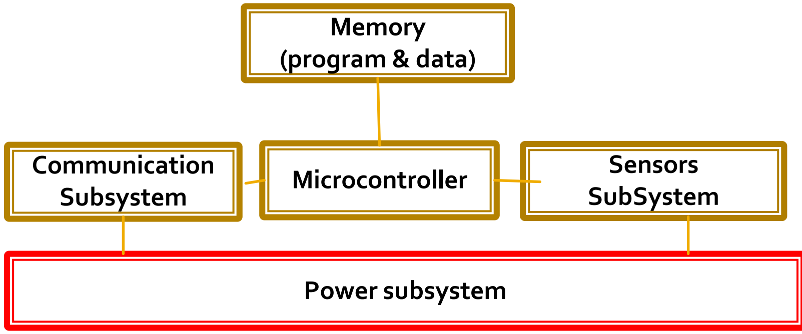
Crack Detection Sensor



Application of Crack Detection sensor

Power Subsystem

- The power subsystem stores and supply energy and converts it to an appropriate supply voltage level.
- The subsystem consists of
  - An energy storage
  - A voltage regulator
  - Optionally an energy harvesting unit



Power Subsystem

Power subsystem

Energy Storage

- Typically using batteries, both rechargeable and non-rechargeable
- Last several years battery life
- Non rechargeable Lithium batteries are most common
- Rechargeable batteries are not suited for smart objects. Why?

Voltage Regulator

- Use to supply a certain constant level of voltage to the smart object hardware

(optional)Energy harvesting unit

- Energy Harvesting : A process by which energy from environment & converted into usable electrical energy

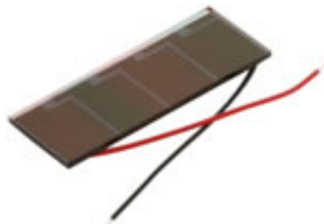
(a) Smart Objects are designed to operate without human supervision
(b) Many smart objects are deployed by thousands and located in difficult to reach places. It is impractical to recharge batteries used in smart objects.

Types of Energy Harvesting

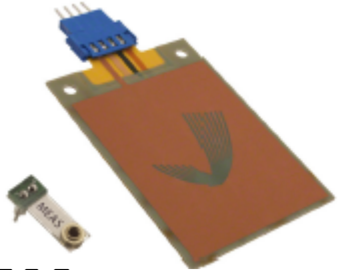
- Energy Harvesting is a process by which energy from environment & converted into usable electrical energy
- Examples:
  - Radiant energy harvesting
  - Mechanical energy harvesting

Examples of Energy Harvesting

- Radiant energy harvesting
  - A solar cell is used to converts light energy directly into electricity
  - The output power of the cell is proportional to brightness of the light landing on the cell



- Mechanical energy harvesting
  - The piezoelectric effect converts mechanical strain into electric current or voltage.
  - This strain can come from many different sources e.g. vibrations on roads, machines
  - Electrical output depends on frequency and acceleration



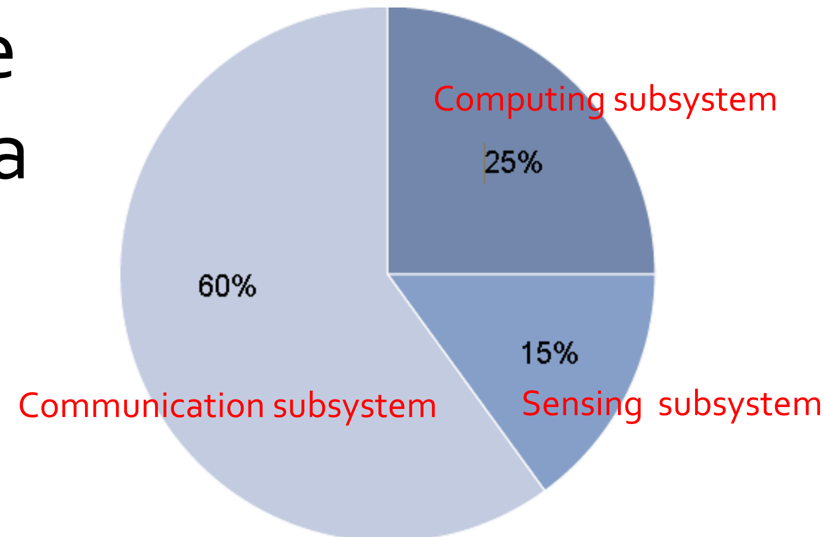
Effect in which electric charge accumulates in certain solid materials (such as crystals] in response to applied mechanical stress

Power Management Techniques

- Energy is a scarce resource in Smart Objects because:
  - **Nodes are very small in size**
    - Difficult to accommodate high-capacity power supplies
    - Size still a *constraining* factor for renewable energy and self-recharging *mechanisms*
  - **Difficult to replace batteries**
    - Smart objects are deployed by thousands and remote locations

Power Management Techniques

- Power is consumed by the following components in a Smart Object
  - Microcontroller subsystem
  - Communication subsystem
  - Sensing subsystem

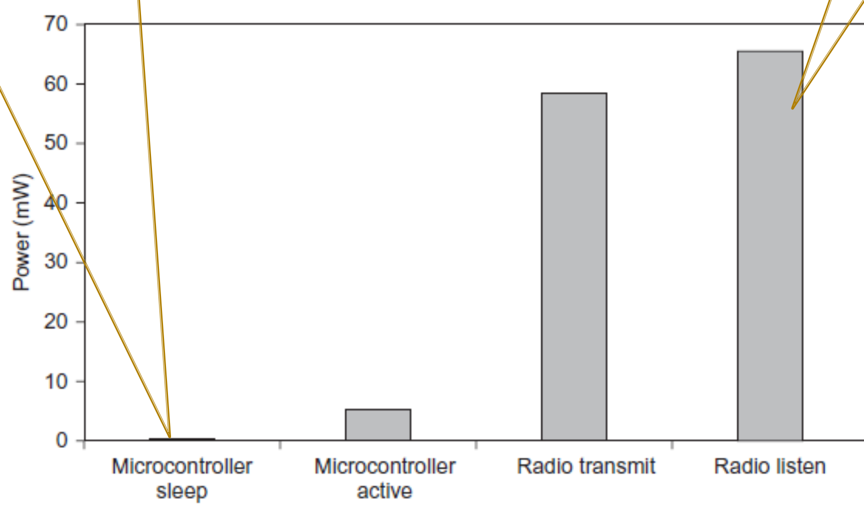


Power Management Techniques

- Where does power go to?

Power consumption of microcontroller in sleep mode is very low

- Radio transceiver consumes x10 times power as microcontroller in active mode
- Power consumption of radio in listen mode is almost as high as the power consumption of radio in transmit mode



Power consumption of the microcontroller

Communication is the major consumer of power

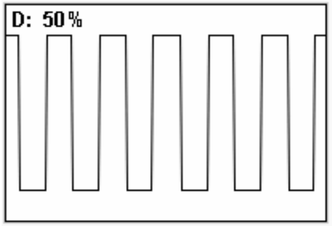
Power Management Techniques

- Power to Smart Objects can be managed by optimising performance of its various subs systems:
  - Microcontroller subsystem
  - Sensor subsystem
  - Communication subsystem

http://www.sensorsmag.com/networking-communications/power-management-critical-choices-wireless-1328?page_id=1

Power Management Techniques

- Power Management by Microcontroller subsystem
  - Optimising Code in Operating System (OS)
    - ✓ By optimising code size in the OS, programs run more efficiently and reduces energy consumption.
  - Control and regulate duty cycles
    - ✓ A Duty cycle describes percent time of a signal is active
    - ✓ The higher the duty cycle, the more power is consumed
    - ✓ Once it is determined how often data from a sensor should be read or transmitted, wake-up rate of the sensor and communication sub system can be set to prolong duration in sleep mode, thus reducing power consumption.



The duty cycle D is defined as the ratio between the pulse duration (T) and the period (P) of a rectangular waveform

Power Management Techniques

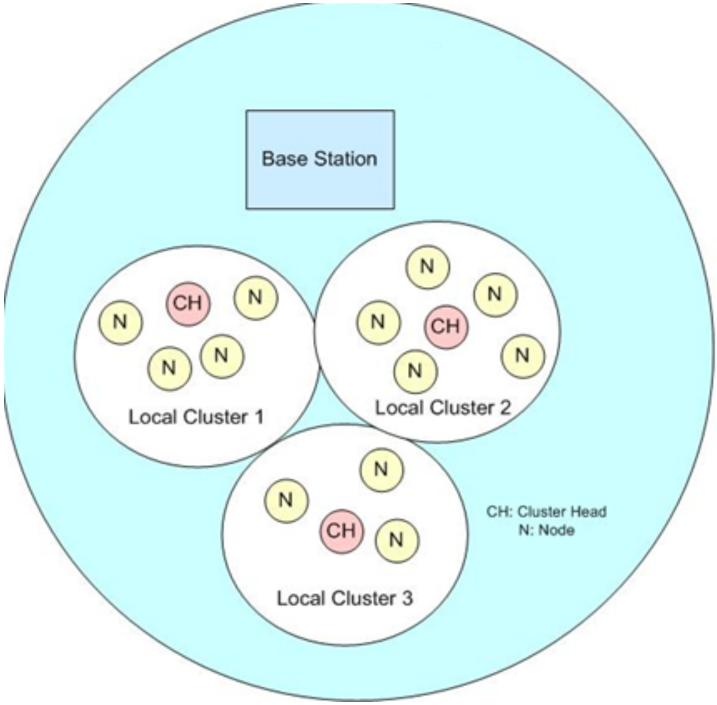
- Power Management by Sensors subsystem :
  - Proper selection of sensors
    - For example, when turning on a sensor, there is a transient state during which time it has to stabilize.
    - The faster a sensor stabilizes, faster measurements can be taken and return to sleep mode, using less power
  - Software functionality
    - Software enables power management by controlling the sensor sub system: turning on the sensor, taking a measurement, and putting the sensor back into a low-power sleep mode.

Power Management Techniques

- Power Management by Communication subsystem is achieved by putting the radio transceiver to sleep mode when communication is not required.
- There are many communication algorithms. An example is **Low Energy Adaptive Clustering Hierarchy** (LEACH)

Power Management Techniques

- How Low Energy Adaptive Clustering Hierarchy (LEACH) works
  - All Smart Object nodes organize themselves into local clusters, with one node in the local cluster acting as Cluster Head (CH).
  - All nodes communicate only to CH which conveys data to the base station
  - As CH has to spend lot of energy ,after certain time, randomized rotation of CH is done
  - Member nodes communicate with CH only during that duration and sleep rest of rest of the time.



Case Study : Libellium Waspmote Hardware

- [Libellium Video]

End