SENSORS, ACTUATORS AND SENSOR NETWORKS

Sensors, Actuators and Sensor Networks and Real-Time and Distributed Systems - Fundamental principles and applications of sensors, actuators.

Smart sensors and micro sensor/micro actuator array devices.

Introduction to signal processing and sensor/actuator networks, deployment and architecture, wireless communication, multiple access control layer, data gathering, routing and querying, collaborating signal processing - Time dependent systems, clock synchronization, real-time communication protocols, specification of requirements, task scheduling. Validation of timelines, real-time configuration management. Middleware architecture for distributed real-time and secure services.

Sensors, Fundamental principles and applications of Sensors

Sensors

The sensor is the bond of a sensor network node. Examples of sensors include temperature sensors, accelerometers, infrared detectors, proximity sensors, and motion detectors.

Figure 3, shows the sensors used in a self-driving (autonomous) car.

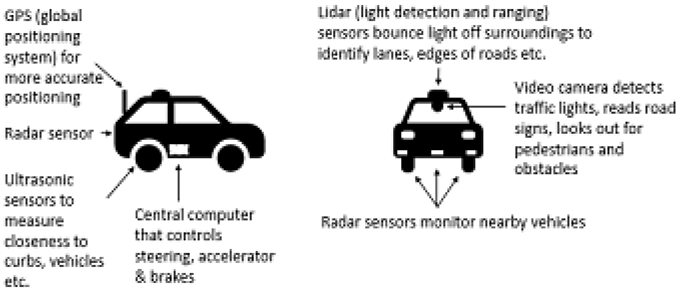

GPS (global positioning system) for more accurate positioning
Radar sensor
Ultrasonic sensors to measure closeness to curbs, vehicles etc.
Central computer that controls steering, accelerator & brakes
Lidar (light detection and ranging) sensors bounce light off surroundings to identify lanes, edges of roads etc.
Video camera detects traffic lights, reads road signs, looks out for pedestrians and obstacles
Radar sensors monitor nearby vehicles

Figure 3: Sensors used in a self-driving car

Fundamental principles of sensors

There are two types of sensing measuring principles available,

Static measurement: measures deflection induced by applied force or surface stress.

Dynamic measurement: measures change in resonant frequency.

Static measurement

For sensor systems where analyte binding induces surface stress, flexible cantilevers functionalized on one side are used to transduce that stress into a measurable deflection. The operating principles and physics behind deflection-based MEMS and NEMS cantilever sensors will be explored.

Principles

Change of stress in a thin film on the surface of a cantilever is illustrated in Fig. 1. On the left, the cantilever bends downward and expands until the stress created in the cantilever balances the expanding film on top. The stress on the top side of the supporting cantilever is tensile, because it is pulled by the film. In the case of a contracting film, the cantilever bends upwards, creating compressive stress on the top side.

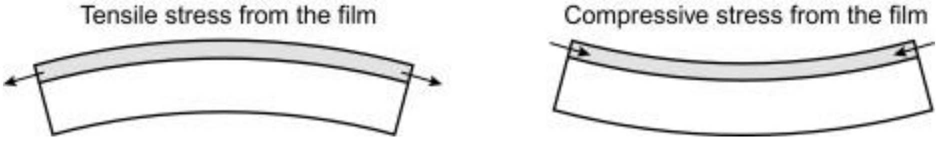

Tensile stress from the film
Compressive stress from the film

Fig 1 Expansion and contraction.

Dynamic measurement

Principles

Most of the dynamic sensing methods are based on measurement of resonant frequencies. When a cantilever is initially actuated, it will oscillate at a particular frequency, known as its natural frequency. When analyte molecules are adsorbed to the cantilever, the additional mass induces a change in the resonant frequency. Here, we remind the readers of the *responsivity* of a mechanically resonating system:

$$\mathrm{Rsp} = -f/2m_{\mathrm{eff}} = \Delta f/\Delta m$$

where f and m_{eff} are the resonant frequency and the effective mass of the system, respectively, and Δm and Δf are changes in the mass and the frequency, respectively. *Sensitivity* is defined as the minimum input level required to produce an output that overcomes the threshold, which is typically chosen just above the noise level.

Applications of Sensors

Sensors find usage in various industries like Automotive, Manufacturing, Aviation, Marine, Medical, Telecom, Chemical, and Computer Hardware. Let's examine some of the applications of sensors in these Industries.

1. Automotive

Here are some of the automotive applications of sensors given below:

- Braking and Traction control: Antilock Braking System (ABS) Sensors connected to the wheel, measures the speed of the wheel and braking pressure and keeps sending them to ABS controlling When the driver applies the sudden brake, ABS system, with breaking pressure and speed data received from the sensors, releases the braking pressure to avoid skidding/locking of wheels. It is one of the critical safety aspects of vehicles.
- Air Bags – Anti Cushion Restraint System (ACRS): Crush sensors and accelerometers placed in the vehicle measures the force and sends it to During accidents on sensing the force exceeds the limit, ACRS will activate the Airbag and save the life of passengers.
- Avoiding Collisions: Proximity sensors in the front, back, and sides of the vehicle forewarn the driver of a possible Infrared, Video assistance, Ultrasonic technologies assist drivers while parking their vehicles.

- Comfort and Convenience: Many sensors provide inputs and warnings to drivers on Vehicle Speed, Engine Speed, Fuel level, Tire pressure, Door/deck, light bulbs for driving comfort and convenience.
- Engine Data: Sensors provides so much data on Engine performance, such as Ignition, b. Combustion, c. Exhaust gas oxygen, d. Fuel mix, e. Exhaust gas recycling, f. Transmission control etc.,
- Other Applications

2. Manufacturing

Here are some of the manufacturing applications of sensors given below:

- Predictive maintenance of the machinery, Assembly equipment using the data collected from sensors in the machines.
- Optimal utilization of Machines by continuously monitoring the performances and effectively rejigging the operations with the data collected from sensors.
- Fine-tuning the Quality systems and enhance the quality standards using the data collected from sensors. Design notifications and alerts in case of a deterioration of quality and process standards.
- Agility in reacting to market demands.

3. Aviation

Sensors deployed in the aviation industry measures the data during navigation of aircraft, monitoring various systems, and controlling instruments. These data are utilized inefficient flight operations, improve aircraft performance and design improvements.

Some of the instrumentation sensors are tachometers, gauges to measure engine pressure and oil& fuel quantity, Altimeters, airspeed meters, etc. Sensors help measure the testing of the ground conditions, vibration and environment factors and provide useful inputs to the pilot to manage the general operation and emergency conditions.

4. Medical & Healthcare

Signals generated by Sensors in Medical equipment, surgical instruments and devices are used for diagnosis, treatment and control functions by Doctors.

Some of the applications are:

1. Blood pressure monitoring (self).
2. Continuous glucose monitoring by Individuals.
3. Automatic measurement of vitals of the patient and sending it to the patient’s doctor.
4. More home care facilities and ambulatory treatments.
5. Automatic detection of visitors spreading the disease to patients in hospitals.
6. Decentralized laboratories.
7. Robotics in Operation Theater.

5. *Marine*

Sensors in ship measures fuel tank levels, liquid cargo levels, tank pressure/temperature. Pitch, roll, speed and other vessel moments are also measured and monitored with sensors' help. There are a lot of sensors in Engines measuring typical attributes of internal combustion parameters.

Actuators, Fundamental principles and applications of actuators

Actuators

An actuator is **a machine component that is used for moving and controlling a system or mechanism**. To perform its operation, An actuator needs a control signal and a power source. They are widely used in valves, gates, conveyors, automatic control systems, etc.1

Actuators are an integral part of instrumentation systems that power the machines that exist today. These devices make it possible for the machines to interact with their surroundings the way they do. In this article, we take a closer look at the omnipresent technology.

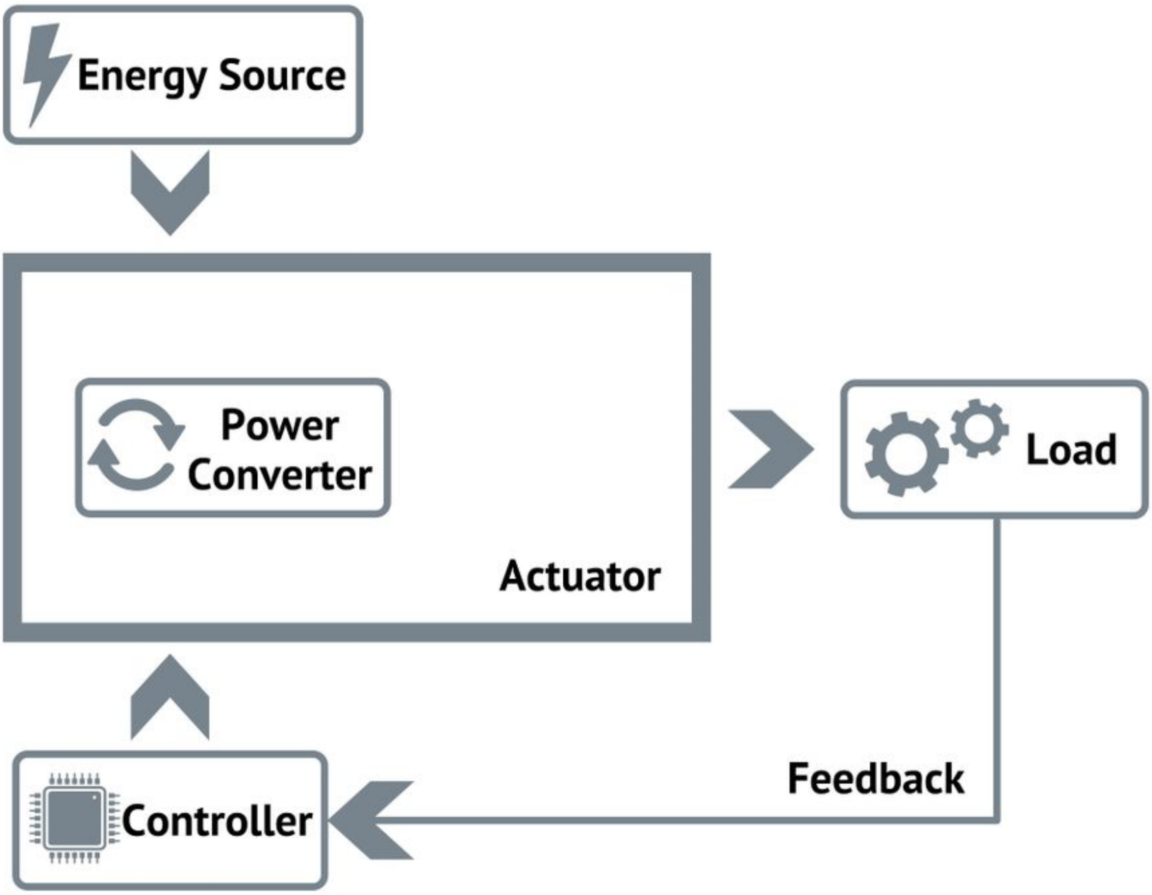
An actuator is a machine, or rather a part of a machine used to convert externally available energy into motion based on the control signals. [1] Much like how hands and legs enable humans to move around and perform actions, actuators let machines perform various mechanical movements. The topic for discussion for this article is actuators. We will explain what is an actuator, how actuators work, and what are the different types of actuators used in industrial and domestic applications.

From the perspective of systems engineering, functions of any engineering product can be classified into three distinct categories; the collection of input, processing and producing an output.

For electromechanical systems, the input is detected and measured by a device called a sensor. The task of a sensor is to sample the signals available to it and convert them into a form understandable by the system. The system then processes the information and decides how to respond. But how exactly does a system respond?

The answer is, with the help of an Actuator. Typically, an actuator consists of:

- **Energy source:** Energy sources provide actuators with the ability to do work. Actuators draw electrical or mechanical energy from external sources for carrying out their operation. The energy available to the actuator can be regulated or unregulated depending on the system that it is a part of.
- **Power converter:** If the energy source attached to the actuators is unregulated, it requires some additional apparatus to regulate it and convert it into a form suitable for the actuation action. Hydraulic valves or solid-state power electronic converters are examples of converters used in industrial actuators.



- **Controller:** In addition to enabling the operation of the power converter, a control unit is responsible for generating actuating signals. In some systems, it provides the user with an interface to provide inputs or check the system's status.
- **Load:** The mechanical system attached to the actuator that uses the motion of the actuator is called the load. Characteristics like Force/Torque and Speed are carefully tuned before interfacing an actuator with the load.

Applications for Actuators

1. Material handling. This is a universal need for every type of manufacturing operation.
2. Robotics. The automotive industry and any number of others are now using robotics to improve production quality and accuracy and control production costs. Electric linear actuators meet the sophisticated needs of robotics. They can control and repeat extremely precise movements, control rate of acceleration and deceleration, and control

the amount of force applied. And they can combine all these movements on multiple axes simultaneously.

3. Food and beverage manufacturing. Cleanliness is critical in these industries, and electric linear actuators are both clean and quiet. In addition, food and beverage, medical device, semiconductor, and some other applications also require stringent washdown protocols. Electric actuators are corrosion-resistant and have a smooth design that offers few crevices where bacteria or dirt might accumulate.
4. Window automation. Manufacturing facilities and other large-scale indoor operations are constructed with heavy-duty ventilation systems, but in some cases, natural ventilation is also desirable, especially to help control indoor temperature. Electric linear actuators make it easy to remotely open and close heavy and/or high windows.
5. Agricultural machinery. Although heavy equipment and attachments are often powered with hydraulics, machines that directly contact food or which require finessed movements can be fitted with electrical actuators instead. Examples include combines that thresh and convey grains, spreaders with adjustable nozzles, and even tractors.
6. Solar panel operation. For optimal operation, solar panels must tilt to directly face the sun as it moves across the sky. Electric actuators enable commercial installations and utilities to efficiently and consistently control large solar farms.
7. Cutting equipment. For example, factories that manufacture carpet or printing facilities may use electric actuators to raise and lower cutting blades. Again, cleanliness can be important in these environments even if it isn't a food safety issue.
8. Valve operation. Many types of processing plants incorporate valves to control the flow of raw materials and finished products throughout the facility.
9. Non-Industrial Applications. We're talking about how electric linear actuators are used in industrial applications, but they are also used increasingly in residential or office settings where hydraulics and pneumatics are not an option. They are tidy, clean, and simple. Electric actuators now offer easy remote operation of windows and window coverings, for example, strictly as a convenience feature or to assist disabled individuals.

Sensor Networks:

A **sensor network** comprises a group of small, powered devices, and a wireless or wired networked infrastructure. They record conditions in any number of environments including industrial facilities, farms, and hospitals. The sensor network connects to the internet or computer networks to transfer data for analysis and use.

Sensor network nodes cooperatively sense and control the environment. They enable interaction between persons or computers and the surrounding environment.

Figure 1, illustrates the widespread use of sensors connected via multiple sensor networks.

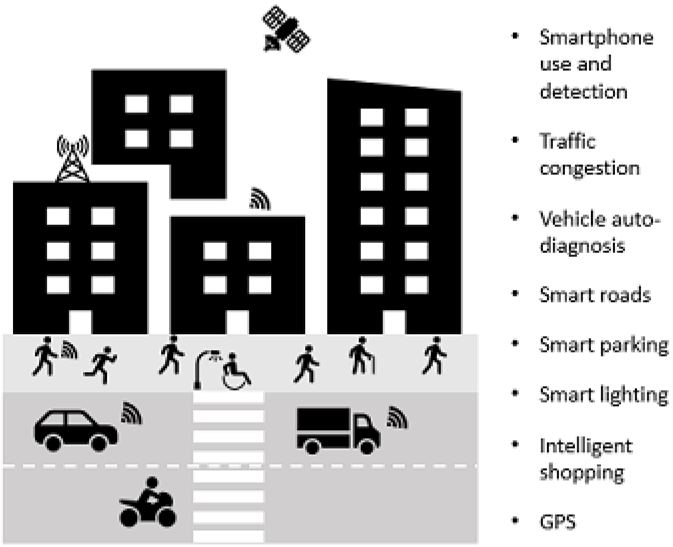


Figure 1: Smart City - Sensors and Sensor Networks

Wired vs. Wireless Sensor Networks

Sensor networks can be wired or wireless. **Wired sensor networks** use ethernet cables to connect sensors. **Wireless sensor networks (WSNs)** use technologies such as Bluetooth, cellular, wifi or near field communication (NFC) to connect sensors.

WSNs are easier to deploy and maintain and offer better flexibility of devices. With the rapid development of sensors and wireless technologies, WSNs have become a key technology of the IoT. WSNs don't need the physical network infrastructure to be modified.

Operation of a Sensor Network

Sensor networks typically include sensor nodes, actuator nodes, gateways, and clients. Sensor nodes group inside the sensor field and form networks of different topologies. The following process describes how sensor networks operate:

- A sensor node monitors the data collected by the sensor and transmits this to other sensor nodes.
- During the transmission process, data may be handled by multiple nodes as it reaches a gateway node.
- The data is then transferred to the management node.
- The management node is managed by the user and determines the monitoring required and collects the monitored data.

Figure 2, shows the components of a sensor network.

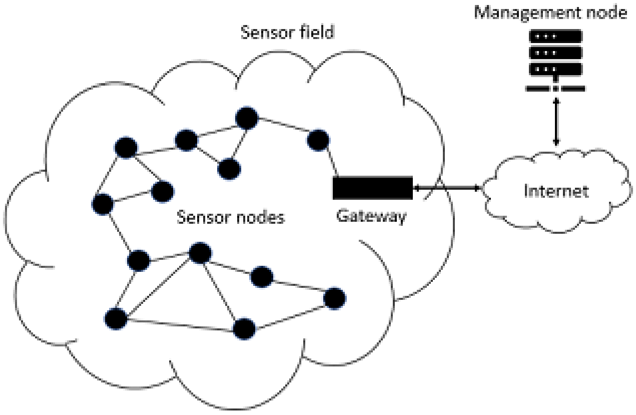


Figure 2: Sensor network components

Sensor Nodes

There are many nodes in a sensor network. These nodes are the detection stations. There is a sensor/transducer, microcontroller, transceiver, and power source:

- A sensor senses the physical condition, and if there is any change, it generates electrical signals.
- The signals go to the microcontroller for processing.
- A central processor sends commands to the transceiver and data is transmitted to a computer.

A **real-time system** is any information processing system which has to respond to externally generated input stimuli within a finite and specified period – the correctness depends not only on the logical result but also the time it was delivered

Distributed System

Distributed system a system in which components are distributed across multiple locations and computer-network. Today's information systems are no longer monoUdl.ic, m.alnf.rame computer-based system . Instead, they are built on some combination of networks to form distributed systems. A distributed system is one in which the components of an information system are distributed to multiple locations in a computer network. Accordingly, the processing workload required to support these components Is also distributed across multiple computers on the network. The opposite of distributed systems are centralized systems. In centralized systems, a central, multi user computer (usually a mainframe) hosts all components of an information system. The users interact with this host computer via terminals (or, today, a PC emulating a terminal), but virtually all of the actual processing and work Is done on the host computer. Distributed systems are Inherently more complicated and more difficult to implement than centralized solutions. So why is the trend toward distributed systems?

- Modem businesses are already distributed, and, thus, they need distributed system solutions.

- Distributed computing moves information and services closer ro the customers that need them.
- Distributed computing consolidates the incredible power resulting from the proliferation of personal computers across an enterprise (and society in general). Many of these personal computers are only used to a fraction of their processing potential when used as stand-alone PCs.
- In general, distributed system solutions are more user.frlendly because they use the PC as the user interface processor.

- Personal computers and network servers are much Jess expensive than main-frames. (But admittedly)', the total cost of ownership is at least as expensive once the networking complexities are added ln.)

There is a price to be paid for distributed systems. Network data traffic can cause congestion that actually slows performance. Data security and Integrity can also be more easily compromised In a distributed solution. Still there Is no arguing the trend toward distributed systems architecture. While many centralized, legacy applications still exist, they are gradually being transformed Into distributed Information systems. Conceptually, any information system application can be mapped to the layers:

- The presentation layer is the actual user interface- the presentation of inputs and outputs ro the user.
- The presentation logic layer is any processing that must be done to generate the presentation. Examples include editing input data and formatting output data.
- The application logic layer Includes all the logic and processing required to support the actual business application and rules. Examples include credit checking, calculations, data analysis.
- The data manipulation layer includes all the commands and logic required to store and retrieve data to and from the database.
- The data layer is the actual stored data in a database.

Smart sensors and micro sensor/micro actuator array devices.

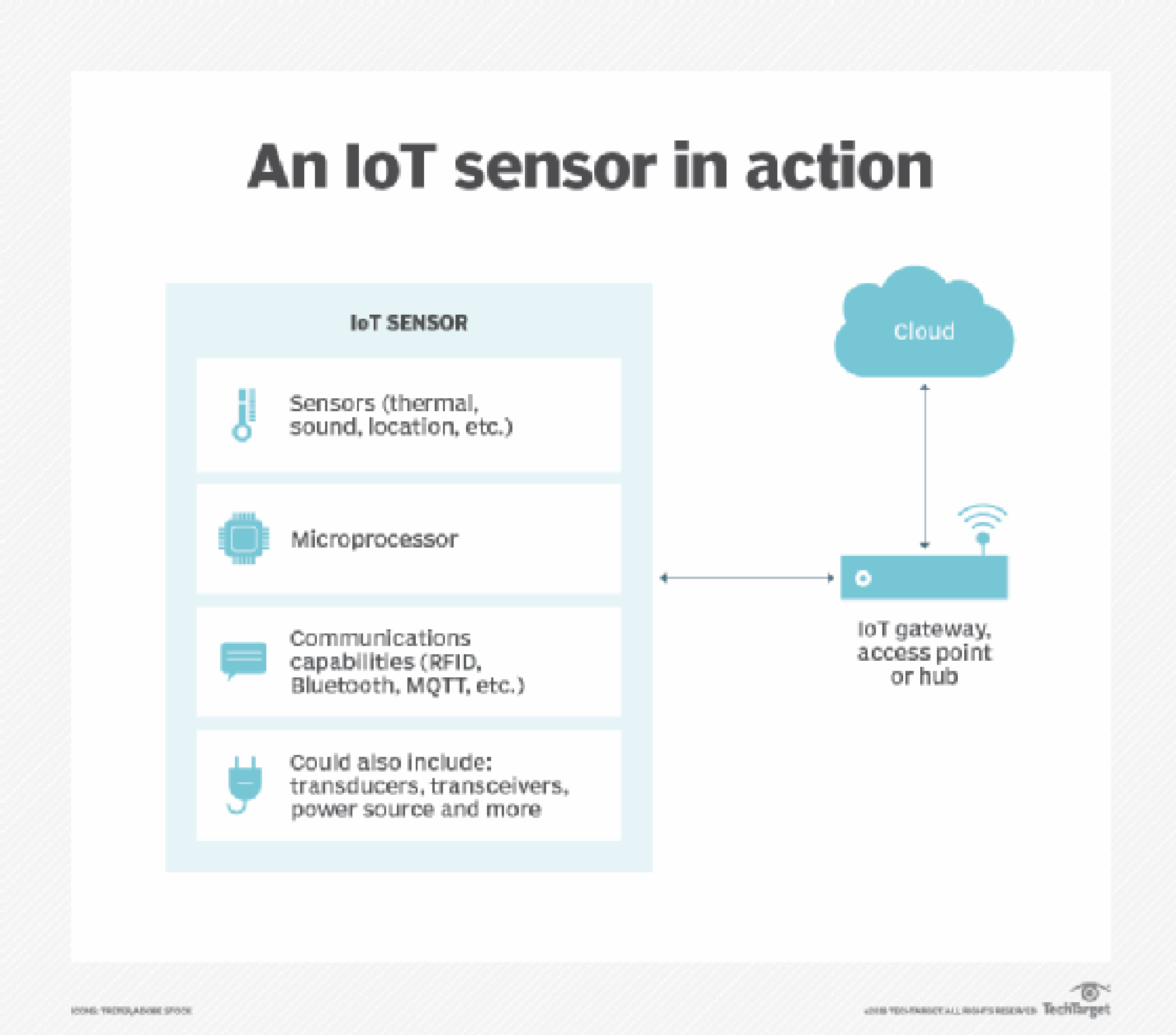
Smart sensors

A smart sensor is a device that takes input from the physical environment and uses built-in compute resources to perform predefined functions upon detection of specific input and then process data before passing it on.

Smart sensors enable more accurate and automated collection of environmental data with less erroneous noise amongst the accurately recorded information. These devices are used for monitoring and control mechanisms in a wide variety of environments including smart grids, battlefield reconnaissance, exploration and many science applications.

The smart sensor is also a crucial and integral element in the internet of things (IoT), the increasingly prevalent environment in which almost anything imaginable can be outfitted with a unique identifier and the ability to transmit data over the internet or a similar network. One implementation of smart sensors is as components of a wireless sensor and actuator network (WSAN) whose nodes can number in the thousands, each of which is connected with one or more other sensors and sensor hubs, as well as individual actuators.

Compute resources are typically provided by low-power mobile microprocessors. At a minimum, a smart sensor is made of a sensor, a microprocessor and communication technology of some kind. The compute resources must be an integral part of the physical design -- a sensor that just sends its data along for remote processing isn't considered a smart sensor.


An IoT sensor in action
IoT SENSOR
Sensors (thermal, sound, location, etc.)
Microprocessor
Communications capabilities (RFID, Bluetooth, MQTT, etc.)
Could also include: transducers, transceivers, power source and more
Cloud
IoT gateway, access point or hub
TechTarget

A smart sensor has three components: a sensor that captures data, a microprocessor that computes on the output of the sensor via programming and communications capabilities.

A smart sensor might also include several other components besides the primary sensor. These components can include transducers, amplifiers, excitation control, analog filters and compensation. A smart sensor also incorporates software-defined elements that provide functions such as data conversion, digital processing and communication to external devices.

How do smart sensors work?

A smart sensor ties a raw base sensor to integrated computing resources that enable the sensor's input to be processed.

The base sensor is the component that provides the sensing capability. It might be designed to sense heat, light or pressure. Often, the base sensor will produce an analog signal that must be processed before it can be used. This is where an intelligent sensor's integrated technology comes into play. The onboard microprocessor filters out signal noise and converts the sensor's signal into a usable, digital format.

Smart sensors also contain integrated communications capabilities that enable them to be connected to a private network or to the internet. This enables communication to external devices.

What are smart sensors used for?

There are countless use cases for smart sensors. They are very commonly used in industrial environments and are the driving force behind Industry 4.0.

Factories often use smart temperature sensors to make sure machines aren't overheating, and vibration sensors to make sure machines aren't at risk of vibrating loose. Smart sensors also enable process control, such as monitoring a process, like manufacturing an item, and making any adjustments that might be required to meet quality or production goals. This was once a manual process, but smart sensors can be used to automate process control.

Smart sensors also play a key role in modern security systems. Thermal imaging sensors can be used to detect an intruder's body heat. Similarly, devices such as smart locks, motion sensors, and window and door sensors are commonly connected to a common network. This enables the security sensors to work together to paint a comprehensive picture of the current security status.

Micro sensors - Introduction

Presently, there is a trend to make sensors smaller and smaller. Initial stages show an evolution from a single sensor element to an intelligent sensor system with extremely small dimensions by MST. The so-called smart (or integrated) sensing devices can be developed by integrating sensor components with those for signal processing. This integration also decreases the noise that is often created by the transmission of signals to an external data processing unit. Thus it will be possible to measure and evaluate for a certain task all interesting parameters at one place and at one time. An important step toward the further development of micro sensors is the conception and design of intelligent electronic signal processors. This will lead to advanced distributed sensor systems in which noisy sensor signals, resulting from cross-talk or insufficient selectivity, can be successfully evaluated. The signal processing system of humans is very advanced; sensor signals are received over the nervous system and transferred to the brain which reliably evaluates them by a natural parallel computing system

Force and Pressure Micro sensors

Due to their simple construction and wide applicability, mechanical sensors play the most important part in MST. Pressure micro sensors were the first ones developed and used by industry. Miniaturized pressure sensors must be inexpensive and have a high resolution, accuracy, linearity and stability. Presently, silicon-based pressure sensors are most often used; they can easily be integrated with their signal processing electronics on one chip. Their advantages include low production costs, high sensitivity and low hysteresis.

Pressure is most often measured via a thin membrane which deflects when pressure is applied. Either the deflection of the membrane or its change in resonance frequency is measured; both of these values are proportional to the pressure applied. These mechanical changes are transformed into electric signals. Membranes can be manufactured by bulk micromachining of a (100) silicon substrate, whereby the membrane is produced with one of the etch stop techniques. Pressure sensors usually employ capacitive or piezo resistive measuring principles.

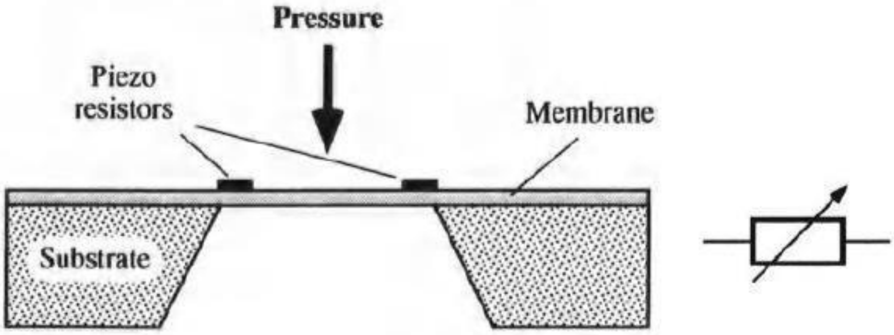


Fig. 1. Piezo resistive pressure sensor

Fig.1. shows the design of a piezo resistive pressure sensor. The piezo resistors are integrated in the membrane; they change their resistance proportionally to the applied pressure. The resistance change indicates how far the membrane is deflected and is measured with a Wheatstone bridge. The deflection value is proportional to the pressure.

Capacitive pressure sensor

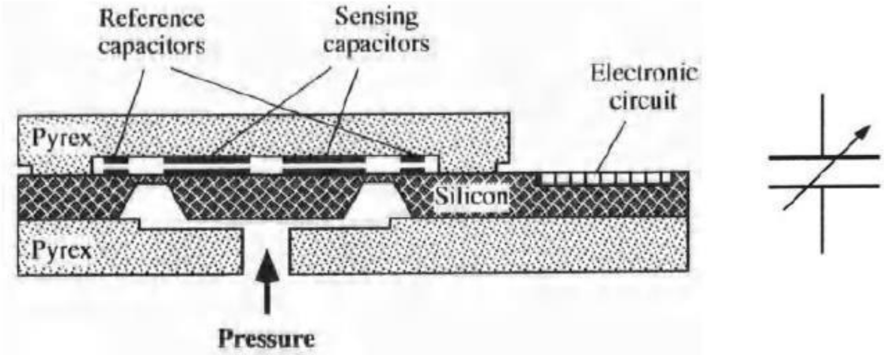


Fig.2. Capacitive pressure sensors

Capacitive sensors make use of the change of the capacitance between two metal plates. The membrane deflects when pressure is applied, which causes the distance between the two electrodes to be changed. Through this the capacitance increases or decreases. From the amount of membrane deflection the capacitance change is measured and the pressure value can be calculated.

A capacitive sensor is shown in Fig.2. The electrodes are made up of a planar comb structure. Here, the applied force is exerted parallel to the sensor surface. In force sensors which use membranes, the force is usually applied perpendicular to the sensor surface. Here, nonlinearity and cross-sensitivity may cause problems. In the device described here, the sensor element mainly consists of two parts: first, a movable elastic structure which transforms a force into a displacement, and second, a transformation unit consisting of the electrodes which transform the displacement into a measurable change of capacitance. Capacitive force sensor made of silicon.

The displacement is restored by an elastic suspension beam. The capacitors consist of two electrically insulated thin electrodes with a very narrow gap between them (approximately 10um). They are placed on both sides of the sensor chip, making the capacitance on one side increase and decrease on the other side. By the separate measurement of the capacitance changes on both sides a high linearity and sensitivity is obtained. The sensor unit is made by anisotropically etching (110) silicon and then fastening it to a pyrex substrate through anodic bonding. The prototype of the capacitive micro sensor had a nominal capacitance of1 pF. Measurements in this range can easily be handled by commercially available microelectronic measuring devices. It was possible to measure very small forces with a resolution of 20 nm (0.01-10 N). The same structure can be used as a positioning unit for nanorobots.

A force sensing resistor

A new measurement principle was realized by using a so-called force sensing resistor. The device is fundamentally different from capacitive, piezo resistive and resonant sensors, since here the resistance is inversely proportional to the pressure. The sensor consists of a polymer foil to which planar electrodes are fastened, on top of this a semiconductor polymer film is placed, Fig. 6.10. If a voltage is applied to the electrodes and there is no force, the resistance is at least 1 MOhm. When a force is applied, the resistance decreases due to current that flows across the shunting polymer foil.

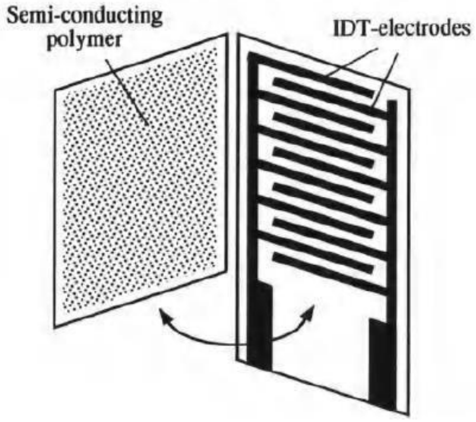


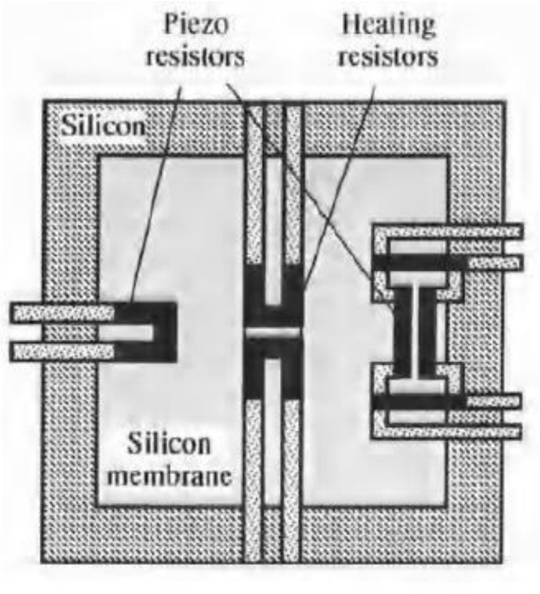
Fig. 3. A force sensing resistor

The dynamic range of the sensor can be influenced by producing a finer electrode structure. This, however, is accompanied by increasing production costs due to a lower yield rate. The sensitivity can be increased by varying the foil thickness. The device can be operated at temperatures of up to 400°Cand is very durable, e.g. over 10 million repeated measurements were made with a 5% deviation. The measurement range is between 10 g and 10 kg. A major disadvantage is the hysteresis, which appears during pressure changes. Despite of this fact, the device can be usefully employed for many dynamic measuring applications. It is inexpensive, compact, robust and resistant to external influences.

Ultrasound distance sensors

Ultrasound distance sensors are well suited as position sensors for micro robots, since they do not depend on the optical properties of the object being detected and they are robust and can obtain reproducible results. Ultrasound distance sensors use the pulse-echo principle. Here a pulse sequence is emitted with the help of an ultrasound transducer which is usually made from piezo ceramics. The signals reflected by objects as echos are received by a sensor and evaluated using the propagation time of the sound signal.

Since the transducer needs some time for recovery after transmission; a "blind spot" appears when the detector is too close to an object, which means that an object might not be detected. The results obtained with a new concept of an ultrasound micro transformer were reported in (M2S293].



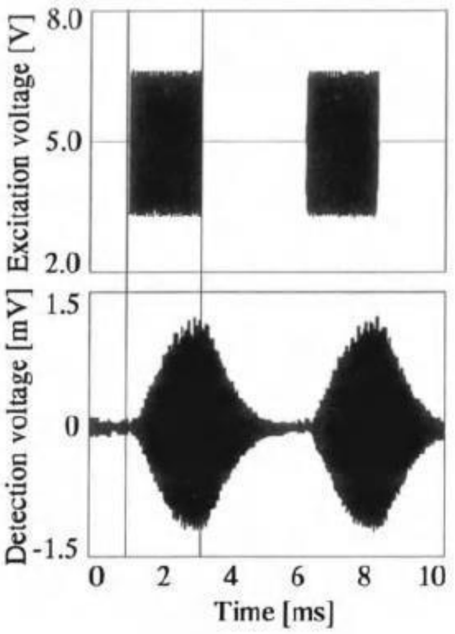


Fig. 4.Ultrasound distance micro sensor

In this device, two identical independent ultrasound membranes were integrated next to each other on a silicon substrate; one served as a transmitter and the other one as a receiver. The schematic design of a single sensor membrane and the measurement principle are shown.

The transmitter membrane is brought to resonance electro thermally with integrated heating resistors. The acoustic pressure response is then detected by piezo resistors, integrated in the form of a Wheatstone bridge in the receiver membrane. The sensitivity of this prototype was about 3 uV/mPa at abridge voltage of 5V.

Capacitive rotational speed sensor

In many technical systems like navigation and landing gear controllers, compact and inexpensive angular speed sensors are required. Conventional sensors using piezoelectric resonators or optical glass fibers are very sensitive, but are usually expensive. The following described silicon sensor was produced using a batch fabrication method [Hash94]. The operating principle of the resonating sensor is presented in Fig.5.

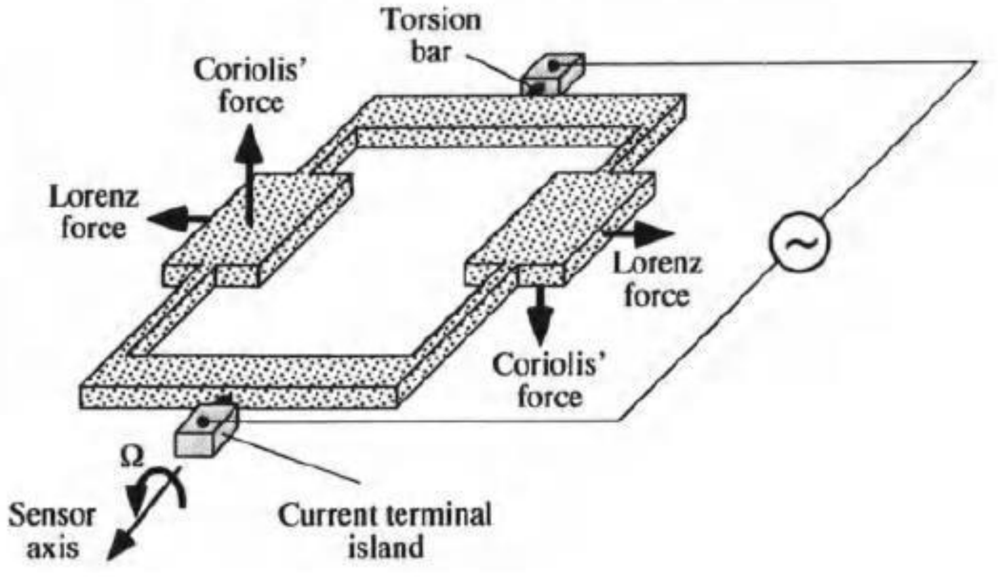


Fig.5.Operating principle of a rotational speed micro sensor

A 200umthick tuning fork arrangement made of (110) silicon is used as the resonator. It is positioned by two torsion bars which also serve as electrical terminals. When the resonator is introduced into a magnetic field and alternating current is applied it starts to oscillate due to Lorenz forces. If the sensor rotates at omega degrees about its longitudinal axis, the Coriolis forces induce a rotational movement in the opposite direction about this axis; this movement is proportional to the swing angle omega. The amplitude of the swing is detected by the capacitance change between the fork prongs (movable electrodes)and the fixed detection electrodes, not shown in Figure 6.17. The latter are integrated into a glass casing consisting of two pyrex glass layers, each250umthick. A sensor prototype with a base area of 2 cm x 2 cm was built; it had a sensitivity of 0.5 mVsec/deg at an exciting frequency of 470Hz

Acceleration Micro sensors

Cantilever principle

Miniaturized acceleration sensors will mostly find their place in the automotive industry. They are also of interest to the air and space industries and for many other applications. Acceleration micro sensors will help to improve the comfort, safety and driving quality of automobiles. However, in order for them to become a product of general interest, their production costs must be drastically lowered. As with pressure (Fig. 6 and Fig. 7), acceleration is usually detected by piezo resistive or capacitive methods. Mostly an elastic cantilever is used to which a mass is attached. When the sensor is accelerated the mass displaces the cantilever and the displacement is picked up by a sensor. Such a sensor is shown in Fig. 6. It uses the capacitive measuring method to record deflection. From the deflection the acceleration can be calculated.

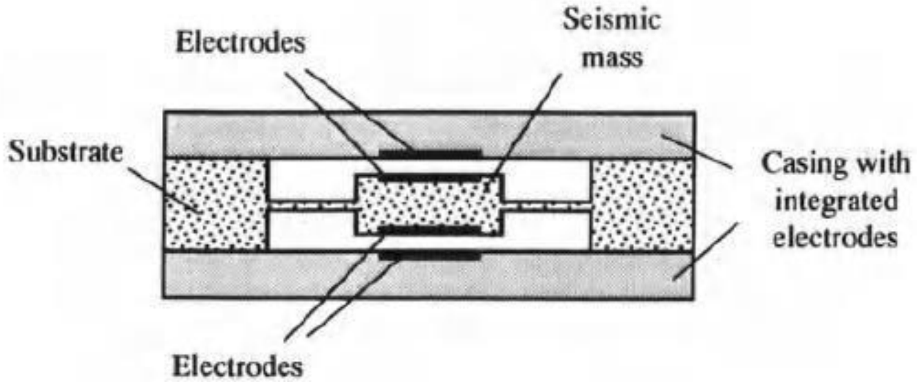


Fig.6. Capacitive measurement of accelerations

Piezo resistive principle

To effectively measure acceleration with this principle, piezo resistors are placed at points of the cantilever where the largest deformation takes place. The stability and accuracy of the sensor improves with increasing number of piezo elements. If a mass moves due to acceleration, it deforms the piezo resistors, thereby changing their resistance, in fig.7. The acceleration is determined from the resistance change.

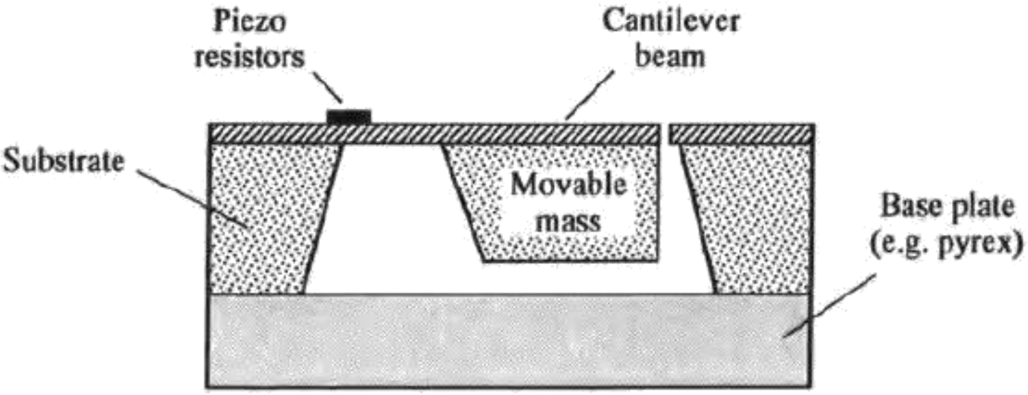


Fig.7 Piezo resistive principle for acceleration measurement

By increasing the movable mass the sensitivity of the sensor will be improved. The mass's center of gravity should be as close to the end of the cantilever as possible. Piezo resistive acceleration micro sensors are usually produced using the silicon technology. This allows the microelectronic processing unit to be integrated onto the sensor chip, making the system compact and robust.

Capacitive cantilever micro sensor

An acceleration sensor produced by the surface micromachining technique was described in [Fricke93]. A sketch of this sensor is shown in Figure .The sensor consists of one or more cantilevers acting as one electrode; they are suspended freely over an opposite electrode and a contact strip. There is only a small gap between the cantilever and the electrode to maximize the electrostatic forces and to keep the mechanical stresses as small as possible.

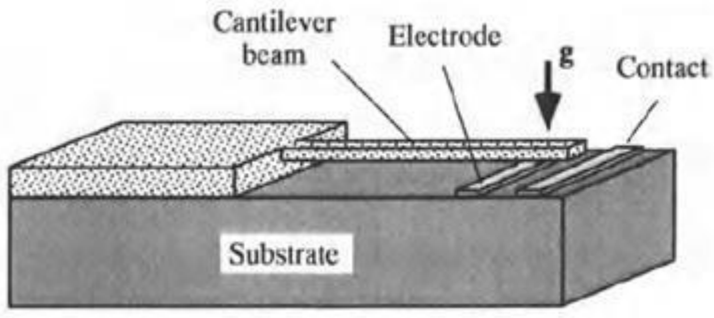


Fig. 8. Capacitive cantilever acceleration micro sensor

As opposed to conventional capacitive sensors, a so-called threshold voltage is applied to offset the forces caused by the acceleration; it will give an indication of the current acceleration. With this device, a saw tooth voltage is applied in defined steps across the cantilever and the electrode, which gradually increases the electrostatic force acting on the cantilever. When the critical voltage is reached, the system becomes unstable and the cantilever bends towards the contacts and finally touches them. The voltage falls to zero, and the saw tooth voltage is applied again. The actual value of the threshold voltage to be applied depends on the magnitude of the acceleration.

Chemical Sensors

Chemical sensors detect the presence or concentration of a chemical substance in a solution. They may be used for qualitative and quantitative measurements. In medical diagnostics, nutritional science, environmental protection and the automobile industry, many different chemical quantities are to be measured.

About 60% of all chemical sensors are gas sensors. The rest is used to detect substances and concentrations in liquids. An important application potential of chemical sensors is in environmental protection, medical applications and process engineering. Many industrial countries will soon be adopting very strict environmental standards and laws that will rapidly increase the demand for gas and liquid sensors. Present research is concentrated on the integration of these sensors in measurement systems.

The potentiometer principle in connection with field effect transistors (FET), acoustic sensors using the change of mass principle and optical sensors are most often applied. Many gas and liquid sensors are based on these principles and have similar structures. It is usually very important for chemical sensors to have a low cross-sensitivity, i.e. the measured values are not influenced significantly by other substances in the solution being analyzed. For measuring chemical substances, a sensitive layer or a specific area of the sensor is used to contact the chemical substance.

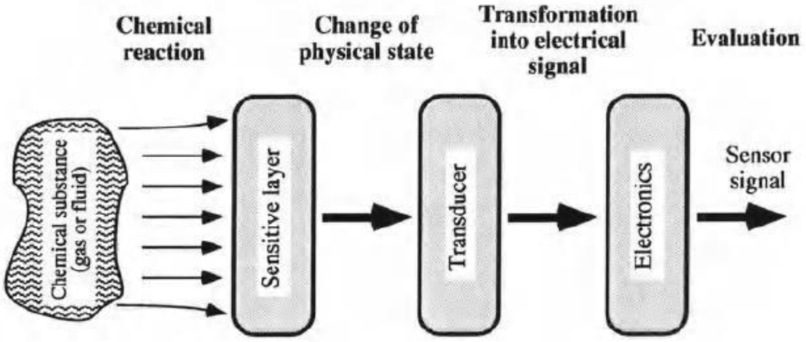


Fig.9. General Structure of chemical sensors

During measurements, a chemical reaction occurs on this sensitive layer/area and a transducer, of which the physical, optical, acoustical or dielectric properties are changed, transforms the recorded phenomenon into an electric signal. This signal is then amplified and evaluated by a microelectronic component. The general structure of a chemical sensor system is shown in Fig.9.

Field effect transistor sensor principle

Ion-sensitive field effect transistors are used to measure the concentration of ions of various elements such as hydrogen, sodium, potassium or calcium. The structure of an ion sensitive FET and its measurement principle can be seen in Figure.

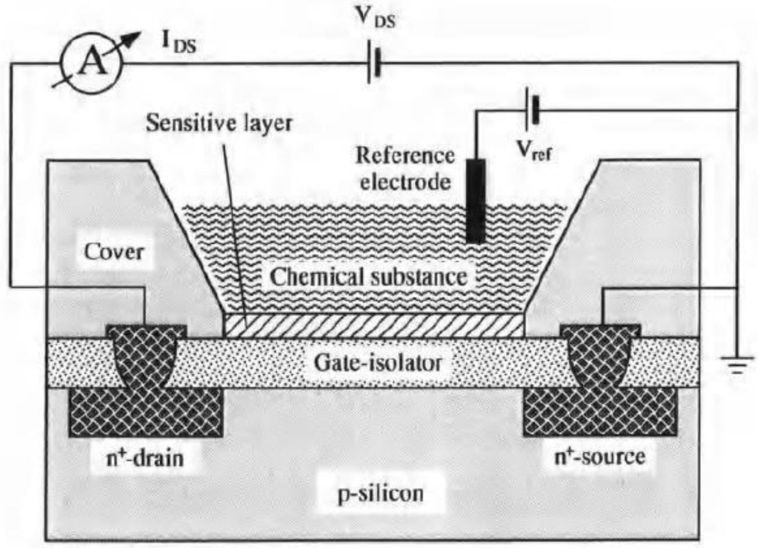


Fig.10. Chemical sensor

Initially, when no chemical substance is in contact with the ion-sensitive layer deposited on the gate area of the transistor, the gate potential Vas is equal to Vref· When the substance to be measured contacts the ion-sensitive layer, the gate potential Vas changes; this voltage change is caused by the ions in the chemical substance. Thereby, the current los between the source and drain changes as well. The gate potential Vas is then corrected by adjusting the voltage Vrer until the original transistor current los flows again; the voltage Vos is held constant. The value delta V ref is proportional to the ion concentration of the analyzed substance to be measured, Fig. 10. The area of the sensitive layer can be as small as a few um^2, allowing very small amounts of substance to be measured.

Biosensors

However, in a biosensor the biologically sensitive elements such as enzymes, receptors and antibodies are integrated with the sensor. The interaction between the protein molecules of the bio element and the molecules of the substance causes a modulation of a physical or chemical parameter. This modulation is converted into an electrical signal by a suitable transducer. The signal represents the concentration to be measured. In many molecular interactions, gases are either released or consumed, e.g. the change of oxygen can be registered by a chemical 0_2 sensor. Very selective and sensitive measurements are possible with biological receptors.

There are many applications for miniaturized biosensors. In biological and nutritional research, these sensors are extremely important to analyze trace elements, especially when toxic substances like heavy metals or allergens have to be found. Considering that there are more than 5 million different inorganic and organic compounds known today, and that 100,000 different substances can be identified, it is getting clear that there is an enormous need for such small, inexpensive and reliable biosensors.

Also in medicine, where a variety of substances are to be monitored during surgery or during an *in-situ* investigation, an increasing number of small biosensors will be used to record vital patient data for a correct and quick diagnosis. Biosensors are divided into two groups. They are metabolism sensors and immuno-sensors. A metabolism sensor uses bio sensitive enzymes as biocatalysts to detect molecules in a substance and to catalyze a chemical reaction. The analyzed substance is chemically transformed and the course of a reaction can be detected and evaluated by a chemical sensor indicating the concentration of the substance in a solution.

This mechanism is illustrated in Figure with an example of an enzyme based measurement of phosphate in waste water treatment. The enzyme nucleoside phosphorylase (NP) is used to determine the phosphate content. This enzyme detects the phosphate and triggers a chemical reaction when inosine is added. One product of this reaction is hypoxanthine (HX). This substance then takes part in another chemical reaction and is transformed into xanthine oxidase (XO) after consuming oxygen. The amount of oxygen consumed can be registered with a chemical 0 2 sensor and the phosphate concentration can be determined from this.

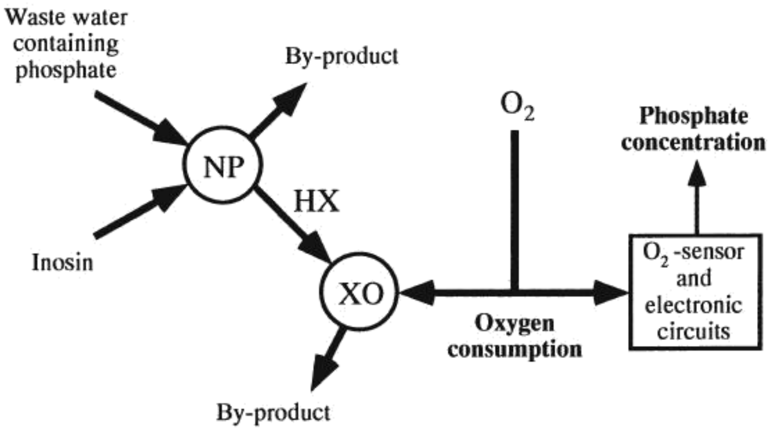


Fig.11 Phosphate measurement with metabolism sensor

To detect chemically inactive molecules in a substance, immuno-sensors are used; their bio sensitive elements are antibodies. The detection method for an antigen molecule is known as the lock and key principle. When it interacts with the analyzed substance, immobilized antibody molecules ("lock") on the sensor surface bond with an antigen molecule "key" in the substance. No other molecule can bond with these antibodies.

The bonding process can either be directly registered over a transducer or indirectly through antigen markers, e.g. using molecules of another substance; depending on the type of sensor. From this measurement the concentration can then be determined. The sensor in Fig.12 detects the concentration of the antigens directly with an interferometric method. The light intensity changes are here due to the bonding process. An attempt is being made to integrate biosensors into Microsystems to take advantage of the many functions they offer. The integration of biosensors with micro pumps and micro valves would make it possible to manufacture very small measuring systems that only need small samples and can measure quickly.

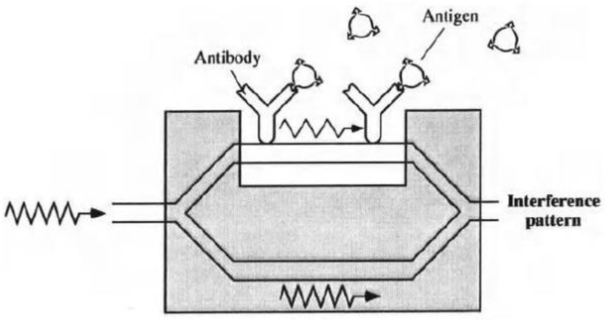


Fig. 12. Immuno sensing using an optical transducer

One difficulty encountered, however, is that a system integration of biosensors may produce only short-lived sensors because the proteins are not very stable for a very long time. Another problem to date is the immobilization of the proteins. Now, some practical developments in the area of miniaturized biosensors will be introduced.

Temperature Sensors

Temperature sensors play an important role in different types of monitoring systems, especially in process, medicine and environmental protection technologies. They are also indispensable for controlling beatings or air conditioning systems and many household appliances. A temperature value may also be a parameter of indirect measurements of other parameters in gas or flow sensors or it may be used for error compensation of temperature-dependent sensors and actuators. There is a wide range of conventional temperature sensors available, like the thermo element, thermo resistor, thermo diode, etc. The next examples describe the development of miniaturized temperature sensors.

Fiber optical thermometer

Several glass fiber thermometers were described in Fig.13 shows a simplified temperature sensor. The sensor contains a light source, a glass fiber, serving both as waveguide and temperature sensor, and a photodiode. The multi-modal glass fiber is made of materials that have different temperature coefficients in the core and the mantle (quartz-silicon system).The light is introduced by an LED into the glass fiber and is propagated through the sensitive fiber area. When the temperature varies in the sensor surrounding, the local index of refraction in the fiber changes, which results in an optical light attenuation.

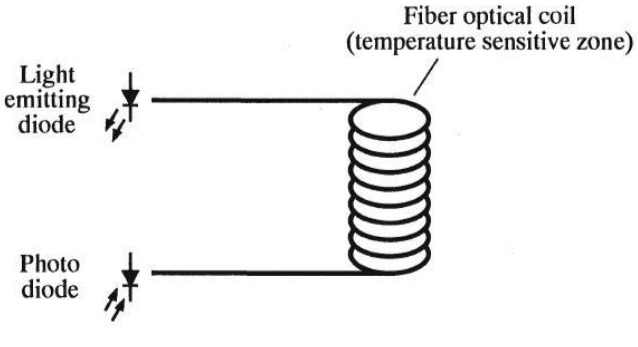


Fig.13. Fiber optic thermometer

This, in turn, leads to a change in the intensity of light leaving the thermometer; the change of light is measured by the photodiode. This measurement can be used to calculate the temperature. A prototype of this sensor has been used for a long time. It can measure temperatures of up to 90°C with an accuracy of 0.1 oc. The thermometer is insensitive to electromagnetic noise, it costs about seven dollars.

Flow Sensors

There is a need for miniaturized sensors to measure very small liquid and gas flows, since in many applications, like in medical instruments and automobiles, micro fluidic components are becoming an indispensable part of a system. Most of these sensors operate on the principle of thermal energy loss, which occurs when the heating element is located in a flowing substance (thermal dilution). Also, transit time measurements of a trace element injected into the flow can be used to determine a velocity. Another measurement principle uses the forces or torques exerted on an object which is placed in the fluid flow.

Two-mode flow sensor

A flow sensor was shown which can be operated in two modes. A sketch of this sensor is depicted in Fig14. In one mode, the sensor uses the elapsed time of the locally heated flow medium. A 5 Hz signal was applied to the heater and the time was measured until the temperature rise is recorded downstream by the sensor. In the second mode (thermal dilution),the heater was supplied with constant energy and the temperature difference between the upstream and downstream sensors was measured. The highest sensitivity registered was in the range of 0.05-0.2 ml/min. One disadvantage of this method would be a possible change in the liquid property due to the heat impulses.

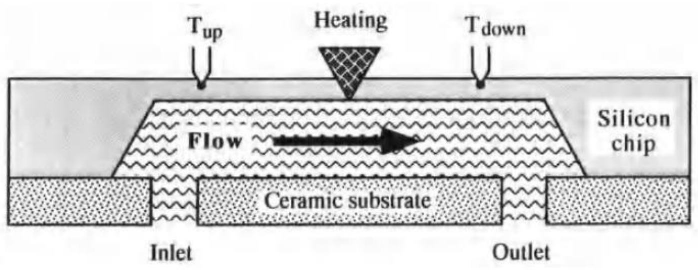


Fig.14. Flow sensor

The device was made to measure both liquid and gas flows. It has two temperature sensors, one in front of the heating element and one behind it. Here, the thermal energy loss between the two temperature sensors was measured. All components of the flow sensor are integrated into one silicon chip. The flow channel is 7 mm long, 1.3 mm wide and 350 um high; it was made by the bulk micromachining technique. Both the inlet and outlet opening have a diameter of 0.7 mm. The silicon membrane is 40um thick.

Thermal flow sensor

A sensor which uses a thermal measurement principle for recording gas flows was developed and shown. A schematic diagram of the sensor is shown in Fig.15. The sensor has a circular silicon disc into which a heating resistor is implanted. The center of the chip is in contact with the gas and serves as a heating element. There is a ring-like silicon dioxide layer around the disc, which guarantees good thermal insulation between the hot and cold sensor parts. The chip cover was made of a 3um thick polyimide film. The flow speed is determined from the thermal interaction between the hot silicon and the gas. Two diodes are implanted into the silicon chip using a CMOS process (not shown in the diagram). They are the thermometers to measure the temperature difference of the gas at the heating element and at a fixed point downstream.

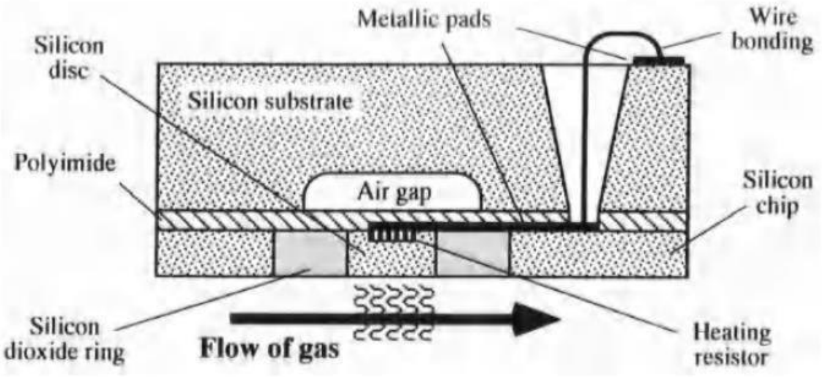


Fig.15. Thermal flow sensor

Several prototypes with an overall dimension of 5 mm x 5 mm were produced. The heating membranes had a diameter between 75um and 500um; the membrane and Si02 layer thicknesses were 15um and 30um, respectively. The measured results agreed with the theoretically calculated values up to a flow speed of 2.5 m/s. The sensor sensitivity was 10 cm/s. The development of new chemical, physical and biological sensors is a continuous process. Only a small sample of existing sensors could be covered here. A detailed description of current research in this area is given in[Gard94]. Numerous sensor principles had been developed in a variety of forms. The manufacturing technology of semiconductors and thin-films will probably play the largest role in the future to make the sensors and to integrate them with the control circuits on one chip.

Micro actuators:

Principles and Examples

Introduction

The MST applications suggest the use of new micro actuator systems which allow motions to be realized with micrometer accuracy. Conventional motion concepts or manufacturing methods are no longer able to fulfil the demands concerning miniaturization and all questions connected with it. Microsystems, and in particular future micro robots, require the development of new advanced actuators with very small dimensions, simple mechanical construction and high reliability.

It is rather difficult to determine exactly what the name "micro actuator" implies. In the literature, the term "micro actuators" is used for devices ranging in size from micrometers to several decimeters, proving the typical classification difficulties of this young scientific field. This book considers a micro actuator as a device of a few micrometers to a few centimetres in size having a functional principle applicable in the micro world.

In addition to the miniaturization, mechanical micro devices having components such as pumps, valves, robot grippers, linear and rotational positioning elements, simple cantilever actuators and complex artificial muscle systems, must be functional to provide a micro system with task-dependent capabilities. Micro pumps and -valves for treating liquids and gasses at the micro level can be applied in medicine, where implantable, highly accurate micro systems are needed for the dosing of medication or for chemical and biotechnological analysis where minute volumes of liquid must be transported and measured. They can also be applied for technical devices such as ink jet printers.

Micro actuators using the cantilever principle can be applied for various applications to generate minute motions. In optics such micro actuators can serve as electronically tuneable mirrors, in fluid dynamics as valves, and in micro robotics as grippers. Micro motors also have a great potential for MST applications.

Actuation Principles

Functional properties of actuators are determined above all by the underlying actuation principles. The forces and displacements that can actually be reached for a certain component size depend on the scaling behaviour, actuator design, and technical limits. Therefore, the total dimensions of an actuator determine which actuation principle is best suited for a given application. The scaling behaviour of the forces is derived from the force laws by introducing a scaling variable *r,* which describes the component size in any spatial direction. In analogy, the scaling behaviour of work and power density can be determined.

Electrostatic Principles

Electrostatic actuators make use of Coulomb's attraction force between oppositely charged bodies. **In** the simplest case, two charged plates oriented in parallel are available. Apart from the force component *F z* that acts vertically between the plates, lateral-plate offset additionally causes a force *F x* acting in the lateral direction.

Magnetic Principles

The magnetic principles of force and movement generation can be classified in electro dynamic, reluctance, and electromagnetic principles. Electro dynamic micro actuators make use of the Lorentz force on a conductor passed by a current in a magnetic field. So far, such actuators have hardly been noticed due to their difficult-to-achieve three-dimensional geometry. Contrary to this, the reluctance principle is the most frequently applied magnetic actuation principle in micro technology. Micro technical implementation requires an acceptable effort, since only a single field source and no hard magnetic materials are needed.

Fluidic Principles

Fluidically driven micro pistons allow high testing forces and large displacements to be reached in, e.g., micro tensile testing machines. The force-displacement characteristics exhibit a constant behaviour. The principle of pressure-dependent membrane deflection is applied in a number of micro valve versions. Other fluidic arrangements are based on fluid dynamics principles or electro rheological and magneto rheological principles.

Inverse Piezo effect

Piezoelectric actuators make use of the coupling of mechanical deformation and electric polarization in ferroelectric crystals, in crystals with a triad axis, or in certain polymers. A major class of materials is made up of Pb(ZrxTi1_x)03 (PZT) crystals due to their high coupling factors of about 0.7 and piezoelectric coefficients *du* of about 0.5 *nm/V.*

Shape Memory Effects

The term "shape memory" describes the unusual ability to remember shape, which can be initiated in certain materials either thermally or mechanically. Even after heavy deformation, materials with shape memory are able to recover a previously memorized shape. This phenomenon was discovered in brass alloys as early as the late thirties. However, the real importance of the effect has become obvious only since its discovery in a NiTi alloy, where it is particularly pronounced.

The most important materials of commercial significance can be classified either as metal alloys or as polymers. Furthermore, there are ceramics and biological systems in which shape memory properties are observed as well. An example to be mentioned in this respect is bacteriophages, which use a shape memory mechanism when entering host cells. SMAs are currently the focus of interest, as they have proved to function in a number of applications and show an unforeseeable potential for future applications.

In contrast to conventional structural materials, in which shape changes are made up of elastic, plastic or thermal contributions, SMAs show three additional types of shape changes that are associated with shape memory characteristics. These effects are illustrated in Fig. by the example of a helical spring:

(a) One-way effect: After removal of the load *F,* the helical spring shows permanent deformation. This seemingly plastic deformation recedes completely upon heating.

(b) Two-way effect: In addition to the one-way effect, there is also a defined shape change upon cooling.

(c) Pseudo elasticity: Mechanical loading, *F,* expands the helical spring to a large extent. When the load is removed, the spring still returns to its initial shape.

One-Way Effect

In the martensitic state (Fig.1, *T<Mf),* the material exhibits a very low elastic limit. Elastic straining is followed by a pseudo plastic strain range in which the component can be strained reversibly by up to several percent. Further strain behaviour is characterized by plastic deformation up to fracture. After relaxation in the pseudo-plastic range, an apparent deformation, *CIW,* is retained. By heating the deformed component above the austenite finish temperature, *Af* , a complete shape recovery is possible. The maximum reversible strain, for instance of NiTi single crystals in the direction, is 10.7%. As this effect only occurs upon heating, i.e., only in one direction, it is referred to as a one-way effect. Renewed cooling and heating no longer changes the shape. The original memorized shape is imprinted upon the component prior to loading by heat treatment. If the shape recovery during reverse transformation to austenite is hindered, high forces occur that can be used to perform work. This is the basis of SMA actuators.

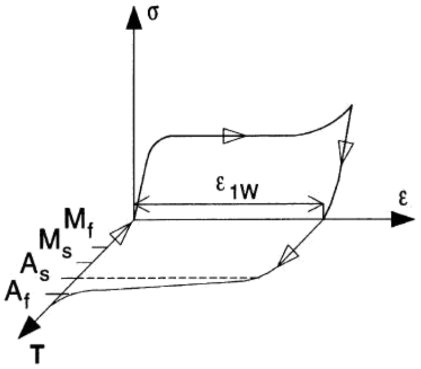


Fig.1 Stress- Strain – temperature characteristic for the one-way effect **Two-Way Effect**

The two-way effect is associated with a shape change upon heating and cooling without requiring any external load. This gives rise to a strain characteristic located within the s-T plane, Fig. 2. The shape change can be repeated without renewed deformation. However, in principle, the shape change attained is less pronounced than with the one-way effect. The shape change upon cooling is achieved by imprinting ordered internal stress fields on the material. The underlying mechanism is based on the formation of preferred martensite variants . One special case of the two-way effect is the all-round effect caused by the formation of ThNi4 precipitates with a preferred orientation.

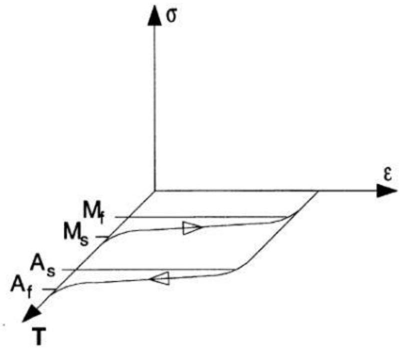


Fig.2 Strain – temperature characteristic for the two-way effect

Pseudo elasticity

In the austenitic state *(T>Ar),* the material exhibits pseudo elastic behaviour, Fig. 3. In contrast to previous effects, no temperature change is required in this case. The strain characteristic is, therefore, located in the cr-s plane. Above the elastic limit, there is a plateau in which a highly nonlinear deformation occurs up to a virtual yield limit, Spe . Above Spe , there is plastic deformation up to fracture. If the component is loaded only as far as Spe , unloading passes through the lower hysteresis loop with the strain disappearing completely. In polycrystalline NiTi, the maximum reversible strain can be 7 to 8% , and in some Cu-

based SMAs, up to 18% . The plateau region is caused by stress-induced transformation of austenite into martensite.

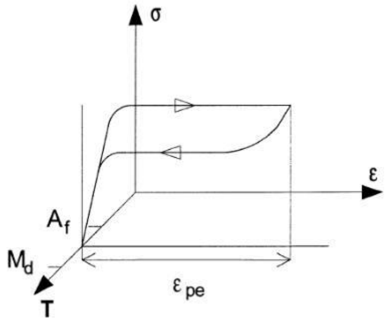


Fig.3 Stress- Strain characteristic for pseudo elasticity

For the stress-induced formation of martensite there is an upper temperature limit, *Md*, above which competing irreversible processes, such as the formation of dislocations and slipping, are favoured thermodynamically. The temperature window, *Md>T>Ar*, in which pseudo elastic behaviour occurs can be set by various thermo mechanical processes [146]. Above *Md*, SMAs behave like conventional materials with elastic strain characteristics and subsequent plasticity up to fracture.

Electrostatic Micro actuators

Electrostatic micro shutter

Electrostatic actuators using membranes will be presented first. In metrology and in micro optics, so-called micro shutters have become of great interest. The principle of this shutter is based on the electrostatic deflection of a movable electrode (micro shutter), made of aluminium, gold or doped poly silicon. During an operation, the shutter moves against a fixed silicon electrode (substrate) which was produced by anisotropic wet etching in (110) silicon. This principle of such a shutter is shown.

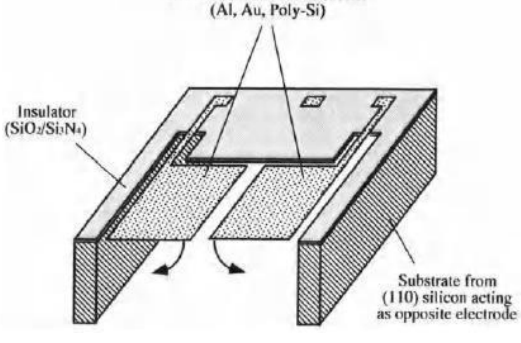


Fig.4. Electrostatic micro shutter

Electrostatic micro pump

The electrostatic membrane principle is well-suited for designing micro pumps. Fig.5 shows a sketch of an electrostatic micro membrane pump. The device consisting of 4 silicon chips was produced by the bulk micromachining technique. The two upper chips form the drive consisting of the membrane and the electrode; the latter is part of the outer frame. The identical lower chips form the inlet and outlet valves. If a voltage is applied between the membrane and the electrode, the membrane arches toward the electrode thereby generating a partial vacuum .in the chamber. This causes the inlet valve to open and liquid to be drawn into the pump chamber. By removing the voltage, the liquid is pushed through the outlet valve. Since the drive unit and the pump chamber are separated, the liquid is not affected by the electric field. This is important when the liquid contains ions, e.g. as in salt solutions or medicines.

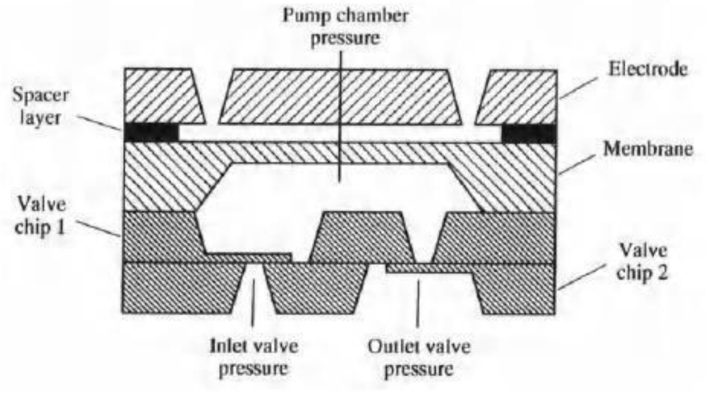


Fig.5. Electrostatic micro pump

Piezoelectric Micro actuators

Micro membrane pump

The micro membrane pump consists of two glass plates and a silicon disc which is sandwiched between the plates, Fig.6.

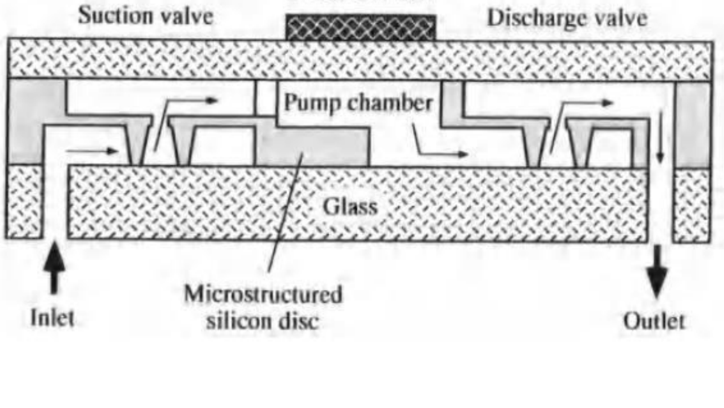


Fig.6. Micro membrane pump

The silicon disc is structured by etching and contains a pump chamber as well as suction and discharge valves. The upper glass plate serves as a pressure-sensitive membrane.

It can change the volume of the pump chamber with the help of a bonded piezo disc (the actuator When a voltage is applied, the membrane buckles downward and the liquid is forced out through the discharge valve. When the voltage is removed, the membrane returns to its original position and the pump sucks in liquid through the suction valve.

Magnetostrictive Micro actuators

Various solid state actuators have been investigated in the past few years; one example using piezo ceramic was described in the previous section. While piezo actuators have served for a variety of applications, magnetostrictive actuators are still at the threshold of industrial exploration.

The elastic wave motor

A very interesting development is the *Elastic Wave Motor* (EWM), which takes advantage of the properties of Terfenol-D. Here, electric energy is directly transformed into a continuous linear movement. A sketch of the function principle and a schematic design of the motor having a movable Terfenol-D rod and external coils (stator) is shown in Figure. In this device, the rod is placed into a guide tube, the end of which is attached to a rigid support. Several short coils are placed along the outer surface of the tube to produce the magnetic field. If the magnetic field is successively switched on and off from one end of the tube to the other in the coils, the Terfenol-D rod moves within the tube in the opposite direction, as shown in Figure. The speed, force and position of the rod are controlled by the magnetic field. The design is successfully being used in the paper industry. It controls the paper thickness during manufacture and piling by moving a blade across the entire paper width.

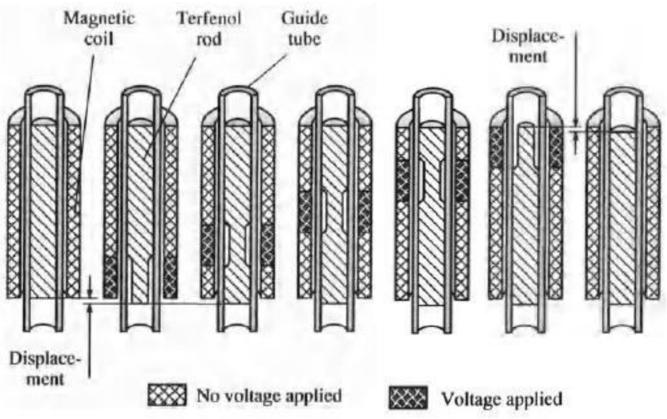


Fig.7. Magnetostrictive Micro actuators

Electromagnetic Micro actuators

Electromagnetically driven micro actuators are gaining in significance as manufacturers are improving the three-dimensional production methods for a variety of materials. With

electromagnetic actuators, electric energy is transformed into mechanical energy like forces or torques.

Linear micromotor

Numerous research projects are concerned with the development of electromagnetic linear actuators. Since almost all present efforts to design linear actuators are based on the silicon technology, the available structures are limited to a height of about 20um, which means that the forces that can be produced are very weak. There are few devices using planar coils; a linear motor with a sliding rare earth magnet. The magnet slides in a channel between two silicon chips which are attached to a glass substrate. The operating principle of this motor is depicted in Fig. Planar coils located in the silicon chips are progressively energized to generate the linear motion of the magnet. There are 8 pairs of planar coils, integrated in parallel to the guiding channel of the chip, Fig. The coils opposite one another are driven sequentially with a current of the same magnitude so that a travelling perpendicular magnetic field (parallel to the magnetization of the permanent magnet) is produced. Thus, the magnet is pulled along the channel in a synchronous manner by the moving magnetic field.

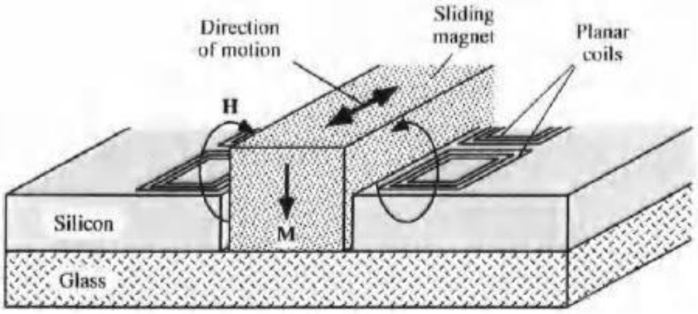


Fig.8.Linear micro motor

SMA-based Micro actuators

When shape memory alloys (SMAs) are deformed under a certain critical temperature and then heated up to above this critical temperature, they will "remember" their original form and assume it again. This effect can be used for generating motions or forces. Characteristic for actuators that use SMA are their low complexity, light weight, small size and large displacement; e.g. SMA components have been used for several years as active pipe connectors. However, the potential use of these alloys in MST has just recently been recognized.

The SMA effect was discovered in various copper alloys, in which a reversible, thermal-mechanical transformation of the atomic structure of the metal takes place at certain temperatures. When the temperature is raised or lowered, the metallurgical structure of an SMA transforms from a martensitic state (low temperatures) to the austenitic state (high temperatures), or vice versa.

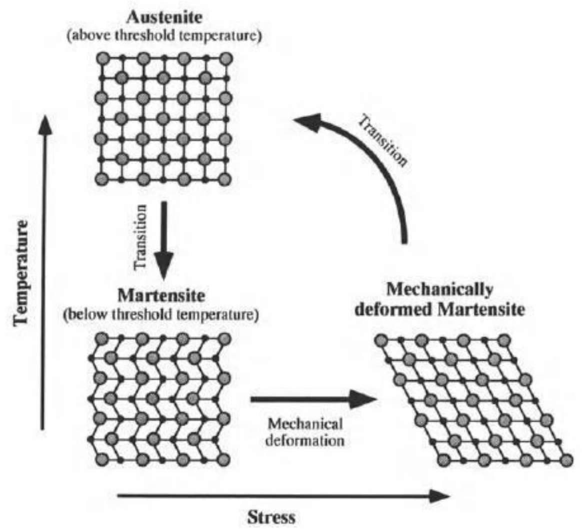


Fig.9. Schematic representation of SMA effect

In Fig,9 the basic transformation mechanism is schematically shown. Starting from a stable and rigid austenitic state, the SMA transforms into the martensitic state as the temperature sinks under the critical temperature; thereby the shape of the SMA can be

deformed by up to 8% (as for NiTi-alloys [Menz93]). In the low temperature state, the SMA keeps the desired deformed shape until it is exposed to a higher temperature. When it is warmed up above a threshold temperature, the deformed martensite is transformed back to austenite and the SMA takes on its original form (thermal shape memory). With this property, large displacements can be obtained compared to other actuator principles.

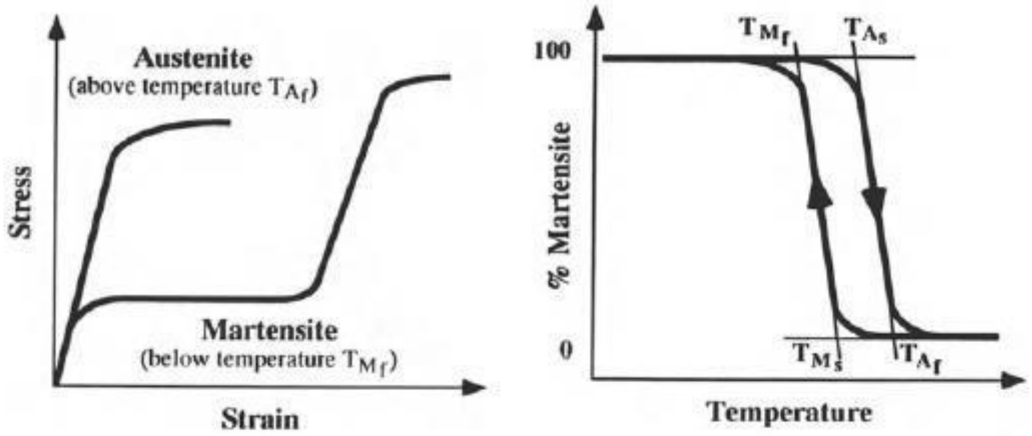


Fig.10. (a)Stress-Strain diagram and (b) the hysteresis cureve of an SMA

Micro endoscopes and catheters

Minimal-invasive surgery and new diagnostic techniques require the availability of a new class of micro and miniature instruments, like endoscopes and catheters, which are equipped with sensors and effectors. The present trend towards minimal invasive therapy requires that precise catheter systems with active guidance will be available to enable the surgeon to enter the various cavities of the human body or to direct them into a specific branch of a blood vessel

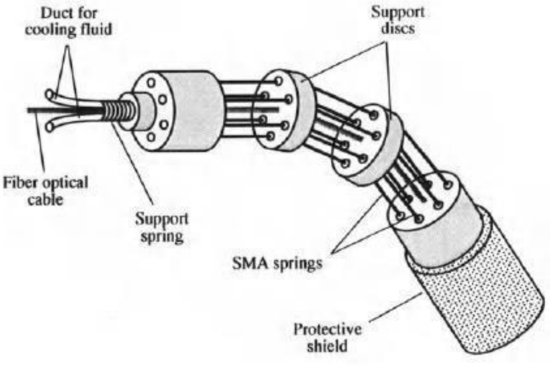


Fig.11. Active endoscope using SMA

There is an electrical connector on each end of the SMA wires to which the electric voltage can be applied. The wires contract when an electric current is applied to them, causing a temperature increase. When the power is turned off, the wires take on their original form after cooling. The direction of motion of the endoscope and its angle of the bend can be controlled by selectively applying electric voltage to each of the three wires.

Hydraulic and Pneumatic Micro actuators

Flexible rubber micro actuators

A flexible micro actuator to be used by miniaturized robots was shown.The actuator is driven by hydraulic or pneumatic pressure, can be bent in every direction and is designed for use as robot hands or legs for various applications. The structure of this device is shown in Fig12. It is made of rubber reinforced with nylon fibers and has three autonomous actuator chambers. The internal pressure in every chamber can be controlled individually by flexible hoses and valves leading to them. The device can be expanded along its longitudinal axis when the pressure is increased equally in all three chambers. If the pressure is only increased in one chamber, the device bends in the opposite direction.

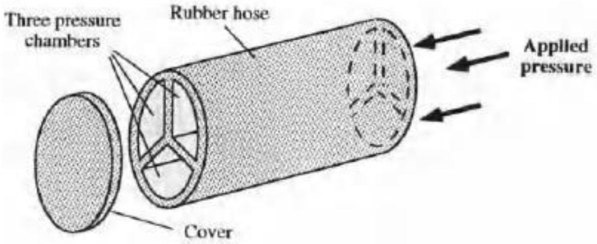


Fig.12. Principle of the pneumatically driven flexible micro actuator

Hydraulic piston micro actuator

An interesting hydraulic micro actuator system was shown.. The piston actuator and its integrated calibration system is depicted in Figure. The actuator chamber with its inlet for its operating fluid, (ex) water, was made by the LIGA process. The unit contains a force-transmitting piston which can be moved along the side walls of the chamber by a fluid. The device is covered by a glass plate (not shown in the figure). A stop groove is added to absorb excessive adhesive which may ooze out when the glass cover plate is being fixed; this is necessary to prevent the piston from sticking to the walls of the chamber.

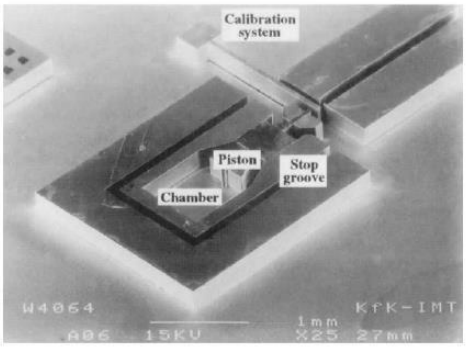


Fig.13. Hydraulic piston micro actuator

Chemical Microactuators

Chemical actuators are based on different chemical processes taking place in fluid or gaseous media. E.g. many chemical reactions produce gases which can be used to create a high pressure in a chamber.

Polymer micro pump

The uni-directional microcapsule polymer pump is shown in Fig.14. It can be used as a medicine dosing system implanted in a patient. The pump cylinder has a semi permeable membrane on its inlet side, which only allows a substance to flow in the direction indicated; a one-way valve is located on the outlet side.

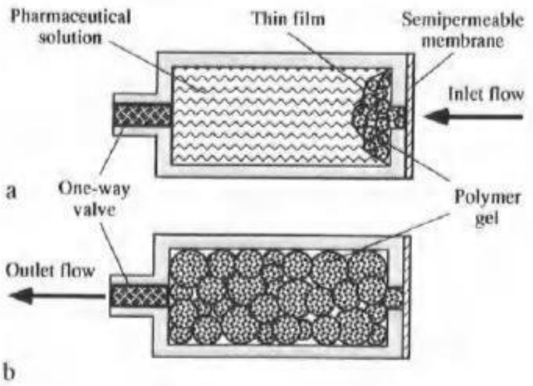


Fig.14. Polymer micro pump (a) Initial state (b) final state

The pump is separated by a thin film and filled with a pharmaceutical solution which is on the left side of the chamber. On the right side of the chamber a highly concentrated water-absorbing polyacrylamide gel is located.

The work cycle of the pump is as follows: the osmotic pressure difference across the membrane takes water from the ambient solution, i.e. from the patient's blood, and drives it through the membrane into the right side of the pump chamber, causing the polymer gel to swell. The pharmaceutical in the microcapsule is pushed out due to the volume increase of the polymer gel. When the pump space is completely occupied by the gel, the medicine is fully injected and the cycle is completed. The duration of the cycle depends on the concentration difference between the polymer gel solution and the ambient solution.

Introduction to signal processing and sensor/actuator networks, deployment and architecture, wireless communication, multiple access control layer, data gathering, routing and querying,

Signal processing is an electrical engineering subfield that focuses on analysing, modifying, and synthesizing signals such as sound, images, and scientific measurements.

Sensors are sophisticated devices used for sensing the phenomena (temp., intensity of earthquake etc.) or detecting the targets (moving vehicles, individuals etc.). So, there may be many types of sensors depending upon their applications, e.g. seismic, passive infrared intrusion (PIR), magnetic sensing device, still video cameras, acoustic microphone arrays etc. Sensor may be stationary or mobile. The individual sensor can share the information with its neighbors and also with the user via a processing unit (sink). Thus the individual sensor can be grouped to do some particular task or multiple tasks. So, it is needed to realize a network of sensors called sensor network. This sensor network may be used in military application, health, security, surveillance etc.

(i) Networking of sensors:-

Sensors are deployed in the region called "Sensor field", where we want to sense or track or detect the target. So, each sensor must have capability for sensing, as well as transmitting the data to their neighbors. This means, each sensor can be activated; it can transmit data to its neighbors and also can be able to communicate with the user through centre processing location (Sink) [1]. This is illustrated in fig 1a.

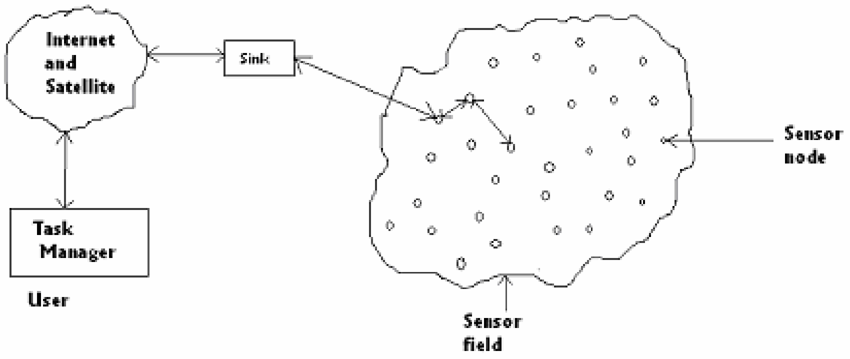
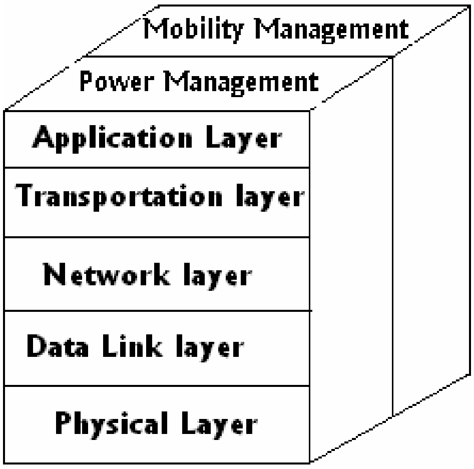

Internet
and
Satellite
Sink
Sensor
node
Task
Manager
User
Sensor
field

Fig 1a. Sensors deployed in sensor field

There may be other type of sensor network which includes less no of sensors (few tens ofsensors) for medical application or tracking an enemy vessel in territory.
But here we consider the network of a large number of micro sensors (hundreds toseveral thousands)

(ii) **Protocol Stack**:

The protocol stack used by the sensor network and sink, is similar to that of computer network [1].
Here each of the layers and protocol used in those layers are illustrated briefly.



Protocol Stack for Sensor Network

Fig 1b.

Physical layer:

This is the ground layer of the network protocol stack. The task of this layer is toselect the frequency, generate the carrier frequency, detect the signal and modulate it. So, signal propagation effects, modulation schemes for effective communication, are important things in this layer.

Binary or M-array modulation scheme can be used in this layer. Low power spreadspectrum modulation can also be used [1].

Data link layer:

This layer is above the physical layer and below the network layer. The main task of this layer is multiplexing of data streams and to detect data frame. This layer also establishes point to point or point to multipoint connection. The Medium access Protocol (MAC) used in this layer must be different.

Sensor networks must be designed in such a way that power consumption is less. Unlike cellular system, the power sources in sensor network can not be replaced or recharged. So, MAC protocol used in this layer must be different from that case of cellular system. Self Organization Medium Access Control for Sensor Network (SMACS) protocol can be used in this layer.

Self **Organization Medium Access Control** for sensor Network is illustrated in [2]. In this network, nodes in the sensor networks are assumed to have the capability of turningits 'radio' on and off and also able to tune the carrier frequency. 'Channel or Time division multiple access (TDMA) slot is assigned to a link just after its existence is discovered. Here each link will operate on a different frequency to avoid collision during transmission or receiving.

(a) In this protocol, when a node (say node 1) is 'waken up', it listen to the 'channel' fora certain time. If it does not receive any response for that time interval, it sends one message (invitation) by the end of its listening time.

(b) The other nodes which are activated in between, will receive that message. And as soon as they receive it, they also send reply to that node. Then the node 1 will receive those entire message if those 'reply' do not collide. After that, node 1 will select one node (say node 2) whose reply is received first or depending on the highest signal level of the replies it received.

(c) In third step, it will send another message to all the nodes informing which node it selects. All the nodes, except the node which is selected, turn off their' radio' for some time and after some time those nodes repeat the searching procedure.

So, finally one link between node 1 and node 2 is established. And they start receiving and transmitting each other for two time slot (One for receiving and another for transmitting). And these time slots will be repeated periodically every certain fixed time interval (T_{frame}) called super frame. This T_{frame} is fixed for each node. When any new node is found, super frame will be filled with another two time slots for that. The main characteristic of this protocol is to use 'non synchronous' assignment in time slot.

After a link is established, a node knows when to transmit and when to receive. Hence, it will turn off its radio in its idle state.

Network layer:

The network layer of the sensor network is quite different from other networks. So aspecial type of routing protocol discussed in [1] can be used here.

Energy efficient routes discussed in [1] consider the available power (PA) in the nodesand the energy reqαired ( ) to transmit data from one node to another.

Node S (Fig 2a) is the source and it sends data to the sink. There are 4 routes from sinkto source.

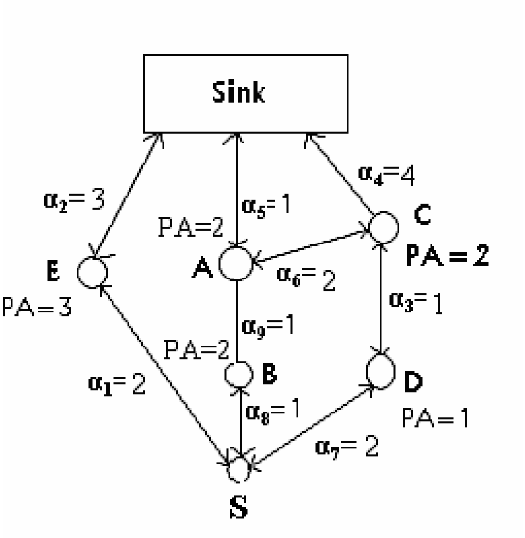


Fig 2a.

Route 1: Sink-A-B-S Total PA = 4
Route 2: Sink-E-S Total PA= 3
Route 3: Sink-C-D-S Total PA = 3
Route 4: Sink-C-A-B-S Total PA = 6

Maximum PA Route:

Route 4 has maximum PA. But it is nothing but the extended path of the *Route 1*. So, itis not power efficient.

Minimum hop (MH) Route:

Here *Route 2* has the minimum hop. So, route 1 will be selected when minimum hoproute is considered.

Maximum minimum PA node Route:

Route 2 has minimum maximum PA.

Minimum energy Route:

Route 1 is minimum energy route.

Data aggregation:

Sometimes it is effective to aggregate the data of an 'attribute' of‘a phenomena’ . In the picture shown below, node 1 and node 2 are used to sense the same phenomena and send the data to the node 4. Here, node 4 aggregates the data from node 1 and 2 and transmits it to the next node, instead of sending the data collected form node 1 and 2 separately.

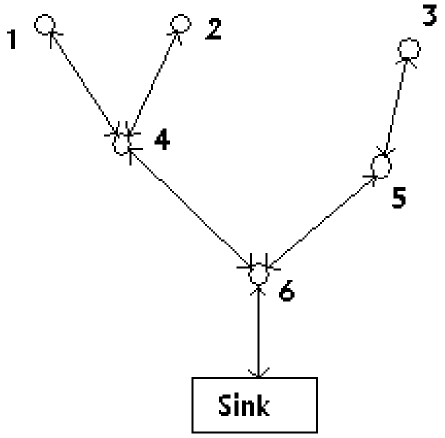


Fig 2b.(Data aggregation)

Transport Layer.

It is above the network layer and below application layer. This layer is responsible for communication with other networks through internet. TCP protocol used for computer network may be applicable here. But UDP-type protocol will be preferred during communication between sink and sensor nodes for limited memory of sensor nodes.

Application layer:

Application layer is important when communication is needed between multiple networks. Protocols used in this layer are Sensor management protocol, Task Assignment and Data Advertisement Protocol (TADAP) and Sensor Query and Data Disseminationetc [1].

Collaborative Signal Processing:

(i) What is collaborative signal processing?

By signal we understand 'something' that signifies some occurrence of events of our interest. It may be deterministic in nature or may not be. But it conveys some information. Processing means understanding that signal, or to modify (transformation, selective retention) that signal in order to extract the information it carries. So, signal “processing means” to “process the “signal” to extract information that it carries. Collaboration means co-operation or working together . Hence, collaborative signal processing means to process the signals received by a group of“ elements. In case ofsensor” network, these elements are sensors and the information is the location of the target(s) .

(ii) Why is it needed in Sensor Network?

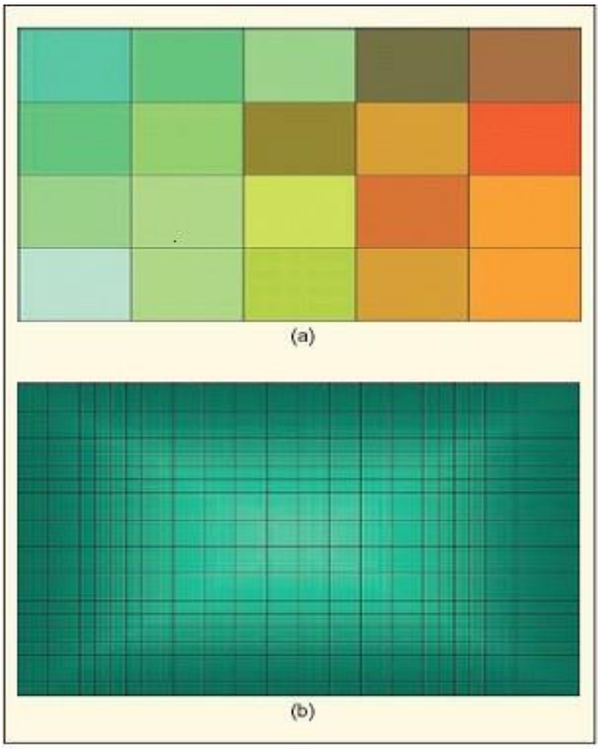
We know that “Unity is strength ”. The importance of collaborative signal processing lies in this line. In order to achieve a bigger goal, information must be shared. In case of sensor network, main critical thing is power and the 'goal' is to detect, identify and track any target. Again sensors are powered by fixed energy sources which are supplied at the time of network forming. So, 'limited power' is key factor here. Receiving, transmitting, and processing of data is to be done with that limited power.

Each sensor must process the signal for certain time period. Again it may detect only the local events. So, it must share the information with other. Hence, collaborative Signal Processing among sensor-nodes is needed to fulfill the goal.

(iii) Sampling in space and time [3]:

The signal from each object is time varying and that signal must be sensed by the sensors. Sensors are densely deployed in a region. This can be thought that nodes sample the “time varying signature field” spatially. The “total number ”of sensors in the sensor field will depend upon the changing rate of the signature field . Again each sensor will sample the signal (time sampling) at a rate depending upon the bandwidth.

Now, one may consider moving objects as“ moving peak” in the “spatial signature field”. So, they may think the entire space time region, consisting of a number of space timecells, where the 'space time signature field ' is constant. The size of this cell depends on the velocity of moving target.



(a) Uniformly sized space-time cell
(b) Non uniformly sized space-time cell

Fig 3. (taken from [3])

The figure (taken from [3]) above illustrates this. Here different shades indicate thevariation of “the signature field”.

Time dependent systems, clock synchronization

Distributed System is a collection of computers connected via the high speed communication network. In the distributed system, the hardware and software components communicate and coordinate their actions by message passing. Each node in distributed systems can share their resources with other nodes. So, there is need of proper allocation of resources to preserve the state of resources and help coordinate between the several processes. To resolve such conflicts, synchronization is used. Synchronization in distributed systems is achieved via clocks.

The physical clocks are used to adjust the time of nodes.Each node in the system can share its local time with other nodes in the system. The time is set based on UTC (Universal Time Coordination). UTC is used as a reference time clock for the nodes in the system.

The clock synchronization can be achieved by 2 ways: External and Internal Clock Synchronization.

1. **External clock synchronization** is the one in which an external reference clock is present. It is used as a reference and the nodes in the system can set and adjust their time accordingly.
2. **Internal clock synchronization** is the one in which each node shares its time with other nodes and all the nodes set and adjust their times accordingly.

There are 2 types of clock synchronization algorithms: Centralized and Distributed.

1. **Centralized** is the one in which a time server is used as a reference. The single time server propagates its time to the nodes and all the nodes adjust the time accordingly. It is dependent on single time server so if that node fails, the whole system will lose synchronization. Examples of centralized are- Berkeley Algorithm, Passive Time Server, Active Time Server etc.
2. **Distributed** is the one in which there is no centralized time server present. Instead the nodes adjust their time by using their local time and then, taking the average of the differences of time with other nodes. Distributed algorithms overcome the issue of centralized algorithms like the scalability and single point failure. Examples of Distributed algorithms are – Global Averaging Algorithm, Localized Averaging Algorithm, NTP (Network time protocol) etc.

Real Time communication protocol

Real-time communication (RTC) is a category of software protocols and communication hardware media that gives real-time guarantees, which is necessary to support real-time guarantees of real-time computing.[1] Real-time communication protocols are dependent not only on the validity and integrity of data transferred but also the timeliness of the transfer.

The following protocol involved in Real time communication protocols are

WebSocket

WebSocket is a web technology providing full-duplex communications channels over a single TCP connection. The WebSocket API is being standardized by the W3C, and the WebSocket protocol has been standardized by the IETF as RFC 6455.

XMPP (Jabber)

Extensible Messaging and Presence Protocol (XMPP) is a communications protocol for message-oriented middleware based on XML (Extensible Markup Language).

WebRTC

WebRTC (Web Real-Time Communication) is an API definition being drafted by the World Wide Web Consortium (W3C) to enable browser to browser applications for voice calling, video chat and P2P file sharing without plugins.

The Bayeux Protocol

Bayeux is a protocol for transporting asynchronous messages (primarily over HTTP), with low latency between a web server and a web client.

Server-Sent Events

Server-Sent Events (SSE) are a way for server to initiate data transfer to clients after the client connects. It is used for streaming continuous or low latency messages to the client. The browser API is called EventSource.

Wave Federation Protocol

The Wave Federation Protocol (formerly Google Wave Federation Protocol) is an open protocol, extension of the Extensible Messaging and Presence Protocol (XMPP) that is used in Apache Wave. It is designed for near real-time communication between the computer supported cooperative work wave servers.

IRC

Internet Relay Chat (IRC) is a protocol for real-time Internet text messaging (chat) or synchronous conferencing. It is mainly designed for group communication in discussion forums, called channels, but also allows one-to-one communication via private message as well as chat and data transfer, including file sharing.

Real-Time Publish-Subscribe (RTPS) Protocol

The Real-Time Publish-Subscribe (RTPS) protocol is designed for use with Internet Protocol (IP) one-to-many Multicast and connectionless best-effort transports such as IP User Datagram Protocol (UDP). It enables, among other things, best-effort and reliable publish-subscribe communications for real-time applications using standard IP networks.

Task Scheduling

Task scheduling algorithms :

The process of deciding which task will utilize the cpu time is called task scheduling. The scheduling of the task may be on the basis of their priorities. The priority assignment mechanism for the tasks can be either static or dynamic. In the case of static priority assignment the priority of task is assigned as soon as the task is created, thereafter the priorities cannot be changed. In dynamic assigning the priorities of the task can be changed during the runtime.

The various scheduling algorithms followed are:

1. First in first out (FIFO): In this algorithm simply the task that came in first in the ready to run state will go out first into the running state. The task that goes out of the running state goes to the ready to run state. Fig 11.5.a shows the movement of the task in the ready to run and running state

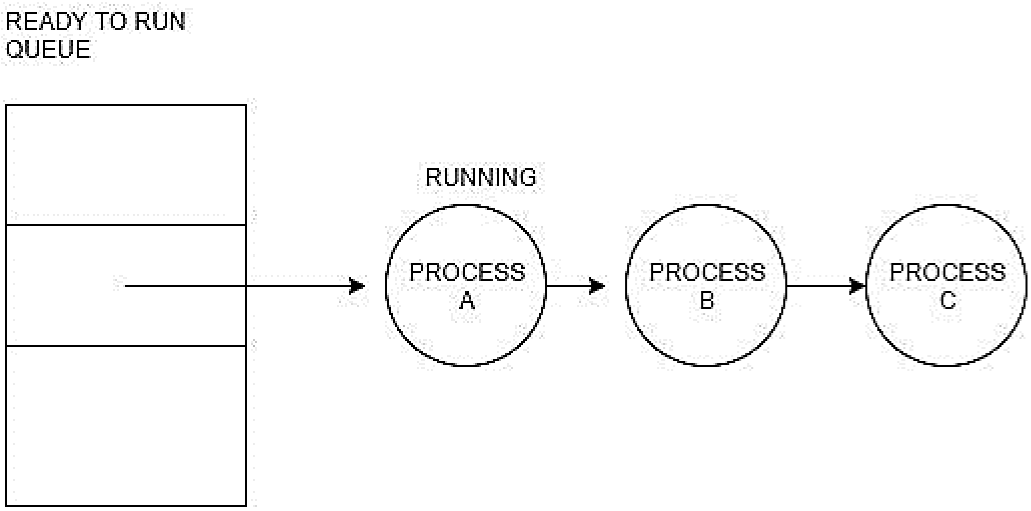
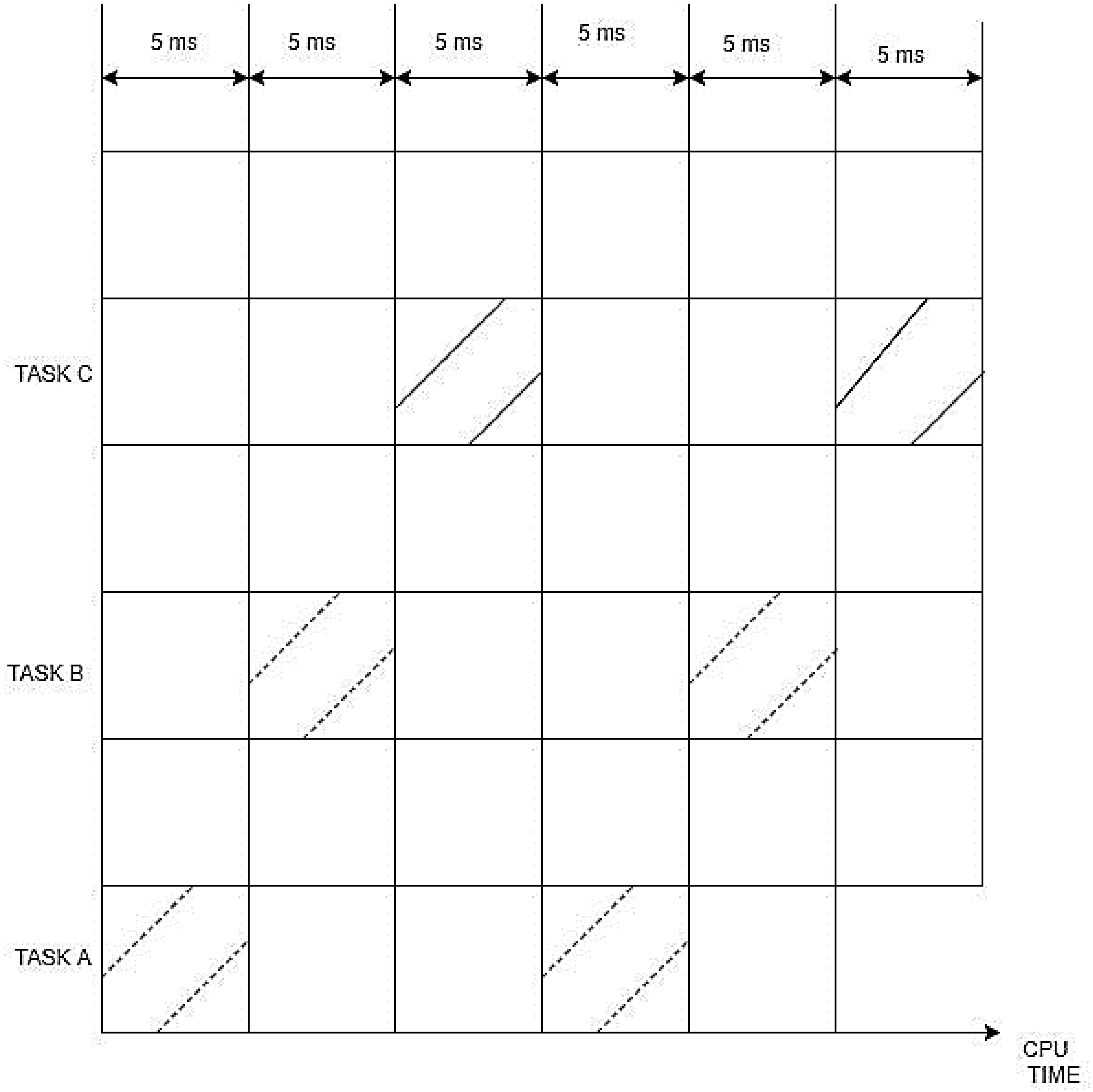


Fig 11.5.a Advantage – it is very easy to implement Disadvantage – we cannot have any priority mechanisms, in real time examples each task has a different priority and it is to be implemented, but using FIFO we cannot implement priority based scheduling.

2. Round robin scheduling: In this case each task gets their turn after all the task are given their time slots. Thus it can be said that it is a time slicing wherein the time for each time slot is same and is given tasks one by one. The CPU time utilization for three tasks according to round robin is shown in fig 11.5.b. In this example, it is assumed that the time slots are 5 milliseconds each.



The switching of tasks occurs in the following case:

1. current task has completed its work before the completion of its time slot
2. current task has no work to be done
3. current task has completed the time slice allocated to it

- advantage – very easy to implement
- disadvantage – all the tasks are considered at same level

3. Round robin scheduling with priority :

- It is modified version of round robin scheduling mechanism
- In this case the tasks are given priorities based on their significance and deadlines. Thus task with higher priority can interrupt the processor and utilize the CPU time

- If multiple tasks have same priority then round robin scheduling will be used for them. But whenever a higher priority task occurs, it will be executed first. The CPU will suspend the task it was executing and will execute higher priority task.
- For example, bar code scanner can use this scheduling algorithm. This method can be used in soft real time systems.

4. Shortest job first (SJF) scheduling:

● In this case the task with the shortest execution time is executed first. This ensures less number of tasks in the ready state. Thus it can be said that the priority is higher for a task with lesser execution time and the priority of the task is lesser for the task with higher execution time.

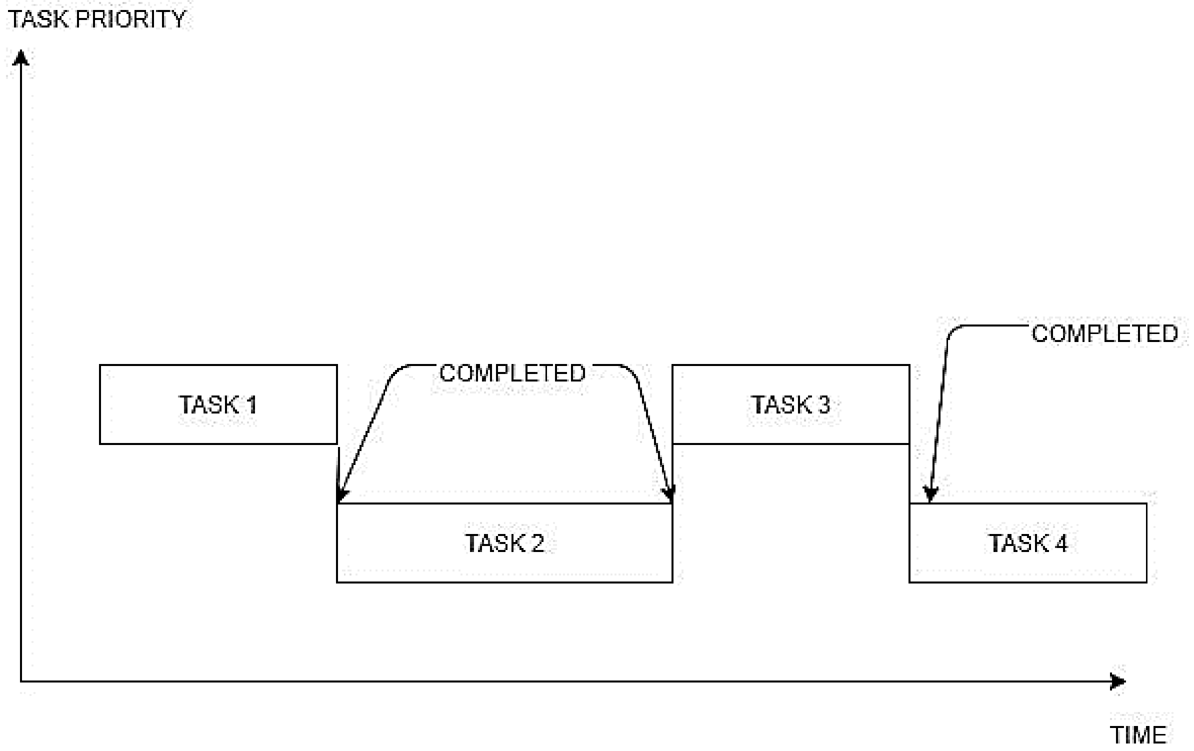
● Disadvantage – if there were too many short execution time tasks, then the task with more execution time will never be executed.

● Advantage – the implementation of this scheduling method is also simpler as just the execution time of each of the tasks is ready to run state are to be compared to decide which task will be executed by the processor.

5. Non preemptive scheduling:

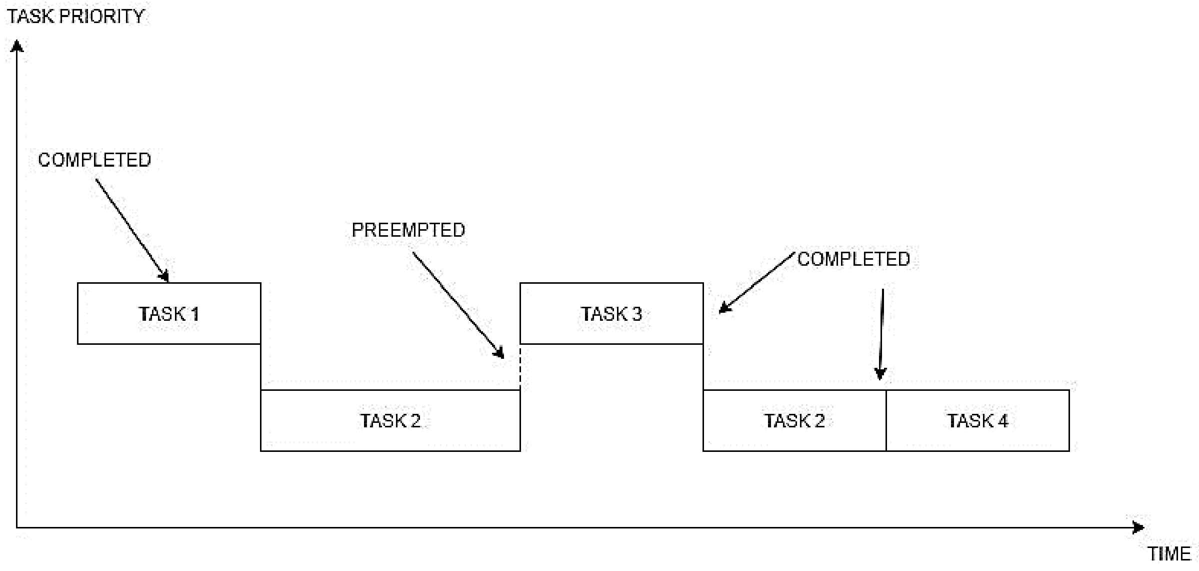
● This scheduling mechanism can be implemented in any of the previously seen scheduling mechanisms that have the concept of priority.

● As the name says in this case if a task (say task 1) is in running state and another task (say task 2) with higher priority enters into the ready to run state, the earlier task i.e. task 1 continues with the execution until its time slice and the higher priority task i.e. task 2 has to wait for its turn. The fig 11.5.c shows an example of non-pre-emptive scheduling.



6. Preemptive scheduling:

- This scheduling can be implemented on any of the scheduling mechanisms having concept of priority.
- As the name says in this case if a task (say task 1) is in running state and another task (say task 2) with higher priority enters into the ready to run state , the earlier task i.e. task 1 has to release the CPU and the later task i.e. task 2 will get the execution.
- Thus the higher priority task will get the CPU as soon as it enters into the ready to run state, and this higher priority task will enter to running state. The fig 11.5.d shows an example of preemptive scheduling.



Validation of timeline, Real-time Configuration Management

Configuration Management is the process of maintaining systems, such as computer hardware and software, in a desired state. Configuration Management (CM) is also a method of ensuring that systems perform in a manner consistent with expectations over time.

Originally developed in the US military and now widely used in many different kinds of systems, CM helps identify systems that need to be patched, updated, or reconfigured to conform to the desired state. CM is often used with IT service management as defined by the IT Infrastructure Library (ITIL). CM is often implemented with the use of configuration management tools such as those incorporated into VMware vCenter.

Why is Configuration Management important?

Configuration Management helps prevent undocumented changes from working their way into the environment. By doing so, CM can help prevent performance issues, system inconsistencies, or compliance issues that can lead to regulatory fines and penalties. Over time, these undocumented changes can lead to system downtime, instability, or failure.

Performing these tasks manually is too complex in large systems. Software configuration management can involve hundreds or thousands of components for each application, and without proper documentation IT organizations could easily lose track of which systems require attention, what steps are necessary to remediate problems, what tasks should be prioritized and whether changes have been validated and propagated throughout the system.

A Configuration management system allows the enterprise to define settings in a consistent manner, then to build and maintain them according to the established baselines. A configuration management plan should include a number of tools that:

- Enable classification and management of systems in groups
- Make centralized modifications to baseline configurations
- Push changes automatically to all affected systems to automate updates and patching
- Identify problem configurations that are underperforming or non-compliant
- Automate prioritization of actions needed to remediate issues•
- Apply remediation when needed.

Structure and Functionality of DOC Middleware

Just as networking protocol stacks can be decomposed into multiple layers, such as the physical, data-link, network, transport, session, presentation, and application layers, so too can DOC middleware be decomposed into multiple layers, such as those shown in Figure 1.

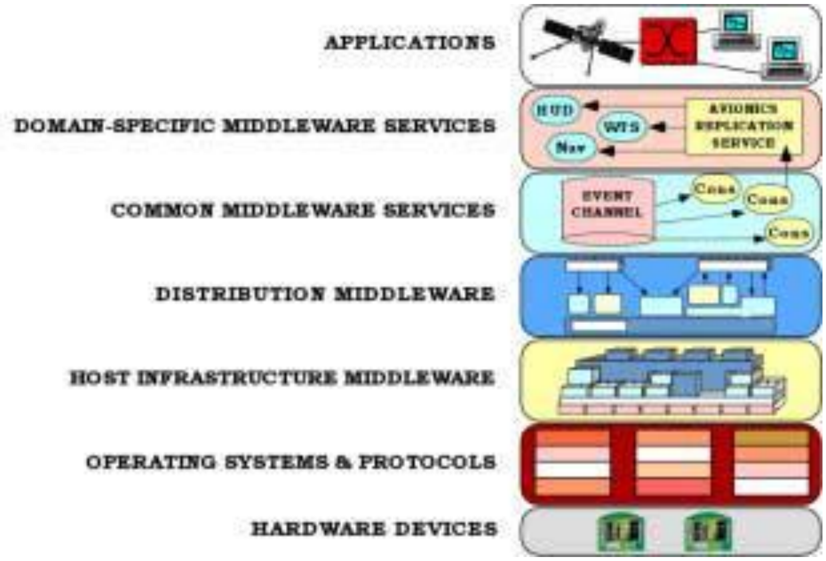

APPLICATIONS
DOMAIN-SPECIFIC MIDDLEWARE SERVICES
COMMON MIDDLEWARE SERVICES
DISTRIBUTION MIDDLEWARE
HOST INFRASTRUCTURE MIDDLEWARE
OPERATING SYSTEMS & PROTOCOLS
HARDWARE DEVICES
HUD
WTS
Nav
AVIONICS REPLICATION SERVICE
EVENT CHANNEL
Cons
Cons
Cons

Figure 1. Layers of DOC Middleware and Surrounding Context

Below, we describe each of these middleware layers and outline some of the COTS technologies in each layer that have matured and found widespread use in recent years.

Host infrastructure middleware encapsulates and enhances native OS communication and concurrency mechanisms to create reusable network programming components, such as reactors, acceptor-connectors, monitor objects, active objects, and component configurators [Sch00b, Sch01]. These components abstract away the peculiarities of individual operating systems, and help eliminate many tedious, error-prone, and non-portable aspects of developing and maintaining networked applications via low-level OS programming APIs, such as Sockets or POSIX pthreads. Widely used examples of host infrastructure middleware include:

- The Sun Java Virtual Machine (JVM) [JVM97], which provides a platform-independent way of executing code by abstracting the differences between operating systems and CPU architectures. A JVM is responsible for interpreting Java bytecode, and for translating the bytecode into an action or operating system call. It is the JVM’s responsibility to encapsulate platform details within the portable bytecode interface, so that applications are shielded from disparate operating systems and CPU architectures on which Java software runs.

- .NET [NET01] is Microsoft's platform for XML Web services, which are designed to connect information, devices, and people in a common, yet customizable way. The common language runtime (CLR) is the host infrastructure middleware foundation upon which Microsoft’s .NET services are built. The Microsoft CLR is similar to Sun’s JVM, *i.e.*, it provides an execution environment that manages running code and simplifies software development via automatic memory management mechanisms, cross-language integration, interoperability with existing code and systems, simplified deployment, and a security system.

- The ADAPTIVE Communication Environment (ACE) [Sch01] is a highly portable toolkit written in C++ that encapsulates native operating system (OS) network programming capabilities, such as connection establishment, event demultiplexing, interprocess communication, (de)marshaling, static and dynamic configuration of application components, concurrency, and synchronization. The primary difference between ACE, JVMs, and the .NET CLR is that ACE is always a compiled interface, rather than an interpreted bytecode interface, which removes another level of indirection and helps to optimize runtime performance.

Distribution middleware defines higher-level distributed programming models whose reusable APIs and components automate and extend the native OS network programming capabilities encapsulated by host infrastructure middleware. Distribution middleware enables clients to program distributed applications much like stand-alone applications, *i.e.* , by invoking operations on target objects without hard-coding dependencies on their location, programming language, OS platform, communication protocols and interconnects, and hardware. At the heart of distribution middleware are request brokers, such as:

- The OMG's Common Object Request Broker Architecture (CORBA) [Omg00], which is an open standard for distribution middleware that allows objects to interoperate across networks regardless of the language in which they were written or the platform on which they are deployed. In 1998 the OMG adopted the Real-time CORBA (RT-CORBA) specification [Sch00a], which extends CORBA with features that allow real-time applications to reserve and manage CPU, memory, and networking resources.

- Sun's Java Remote Method Invocation (RMI) [Wol96], which is distribution middleware that enables developers to create distributed Java-to-Java applications, in which the methods of remote Java objects can be invoked from other JVMs, possibly on different hosts. RMI supports more sophisticated object interactions by using object serialization to marshal and unmarshal parameters, as well as whole objects. This flexibility is made possible by Java’s virtual machine architecture and is greatly simplified by using a single language..

- Microsoft's Distributed Component Object Model (DCOM) [Box97], which is distribution middleware that enables software components to communicate over a network via remote component instantiation and method invocations. Unlike CORBA and Java RMI, which run on many operating systems, DCOM is implemented primarily on Windows platforms.

- SOAP [SOAP01] is an emerging distribution middleware technology based on a lightweight and simple XML-based protocol that allows applications to exchange structured and typed information on the Web. SOAP is designed to enable automated Web services based on a shared and open Web infrastructure. SOAP applications can be written in a wide range of programming languages, used in combination with a variety of Internet protocols and formats (such as HTTP, SMTP, and MIME), and can support a wide range of applications from messaging systems to RPC.

Common middleware services augment distribution middleware by defining higher-level domain-independent services that allow application developers to concentrate on programming business logic, without the need to write the "plumbing" code required to develop distributed applications by using lower-level middleware directly. For example, application developers no longer need to write code that handles transactional behavior, security, database connection pooling or threading, because common middleware service providers bundle these tasks into reusable components. Whereas distribution middleware focuses largely on managing end-system resources in support of an object-oriented distributed programming model, common middleware services focus on allocating, scheduling, and coordinating various resources throughout a distributed system using a component programming and scripting model. Developers can reuse these component services to manage global resources and perform common distribution tasks that would otherwise be implemented in an *ad hoc* manner within each application. The form and content of these services will continue to evolve as the requirements on the applications being constructed expand. Examples of common middleware services include:

- The OMG's CORBA Common Object Services (CORBAservices) [Omg98b], which provide domain-independent interfaces and capabilities that can be used by many DOC applications. The OMG CORBAservices specifications define a wide variety of these services, including event notification, logging, multimedia streaming, persistence, security, global time, real-time scheduling, fault tolerance, concurrency control, and transactions.

- Sun's Enterprise Java Beans (EJB) technology [Tho98], which allows developers to create n-tier distributed systems by linking a number of pre-built software services—called "beans"—without having to write much code from scratch. Since EJB is built on top of Java technology, EJB service components can only be implemented using the Java language. The CORBA Component Model (CCM) [Omg99] defines a superset of EJB capabilities that can be implemented using all the programming languages supported by CORBA.

- Microsoft's .NET Web services [NET01], which complements the lower-level middleware .NET capabilities, allows developers to package application logic into components that are accessed using standard higher-level Internet protocols above the transport layer, such as HTTP. The .NET Web services combine aspects of component-based development and Web technologies. Like components, .NET Web services provide black-box functionality that can be described and reused without concern for how a service is implemented. Unlike traditional component technologies, however, .NET Web services are not accessed using the object model–specific protocols defined by DCOM, Java RMI, or CORBA. Instead, XML Web services are accessed using Web protocols and data formats, such as the Hypertext Transfer Protocol (HTTP) and eXtensible Markup Language (XML), respectively.

Domain-specific middleware services are tailored to the requirements of particular domains, such as telecom, e-commerce, health care, process automation, or aerospace. Unlike the other three DOC middleware layers, which provide broadly reusable "horizontal" mechanisms and services, domain-specific middleware services are targeted at vertical markets. From a COTS perspective, domain- specific services are the least mature of the middleware layers today. This immaturity is due partly to the historical lack of distribution middleware and common middleware service *standards*, which are needed to provide a stable base upon which to create domain-specific services. Since they embody knowledge of a domain, however, domain-specific middleware services have the most potential to increase system quality and decrease the cycle- time and effort required to develop particular types of networked applications. Examples of domain-specific middleware services include the following:

- The OMG has convened a number of Domain Task Forces that concentrate on standardizing domain-specific middleware services. These task forces vary from the *Electronic Commerce Domain Task Force*, whose charter is to define and promote the specification of OMG distributed object technologies for the development and use of Electronic Commerce and Electronic Market systems, to the *Life Science Research Domain Task Force*, who do similar work in the area of Life Science, maturing the OMG specifications to improve the quality and utility of software and information systems used in Life Sciences Research. There are also OMG Domain Task Forces for the healthcare, telecom, command and control, and process automation domains.

- The Siemens Medical Engineering Group has developed *syngo* which is both an integrated collection of domain-specific middleware services, as well as an open and dynamically extensible application server platform for medical imaging tasks and applications, including ultrasound, mammography, radiography, flouroscopy, angiography, computer tomography, magnetic resonance, nuclear medicine, therapy systems, cardiac systems, patient monitoring systems, life support systems, and imaging- and diagnostic-workstations. The *syngo* middleware services allow healthcare facilities to integrate diagnostic imaging and other radiological, cardiological and hospital services via a blackbox application template framework based on advanced patterns for communication, concurrency, and configuration for both business logic and presentation logic supporting a common look and feel throughout the medical domain.

- The Boeing Bold Stroke [Sha98, Doe99] architecture uses COTS hardware and middleware to produce a non-proprietary, standards-based component architecture for military avionics mission computing capabilities, such as navigation, display management, sensor management and situational awareness, data link management, and weapons control. A driving objective of Bold Stroke was to support reusable product line applications, leading to a highly configurable application component model and supporting middleware services. Associated products ranging from single processor systems with $O(10^5)$ lines of source code to multi-processor systems with $O(10^6)$ lines of code have shown dramatic affordability and schedule improvements and have been flight tested successfully. The domain-specific middleware services in Bold Stroke are layered upon common middleware services (the CORBA Event Service), distribution middleware (Real-time CORBA), and host infrastructure middleware (ACE), and have been demonstrated to be highly portable for different COTS operating systems (e.g. VxWorks), interconnects (e.g. VME), and processors (e.g. PowerPC).

2.2 Benefits of DOC Middleware

Middleware in general–and DOC middleware in particular–provides essential capabilities for developing distributed applications. In this section we summarize its improvements over traditional non-middleware oriented approaches, using the challenges and opportunities described in Section 1 as a guide:

- ***Growing focus on integration rather than on programming*** – This visible shift in focus is perhaps the major accomplishment of currently deployed middleware. Middleware originated because the problems relating to integration and construction by composing parts were not being met by either

    1. Applications, which at best were customized for a single use,
    2. Networks, which were necessarily concerned with providing the communication layer, or
    3. Host operating systems, which were focused primarily on a single, self-contained unit of resources.

    In contrast, middleware has a fundamental integration focus, which stems from incorporating the perspectives of both operating systems and programming model concepts into organizing and controlling the composition of separately developed components across host boundaries. Every DOC middleware technology has within it some type of request broker functionality that initiates and manages inter-component interactions.

    Distribution middleware, such as CORBA, Java RMI, or SOAP, makes it easy and straightforward to connect separate pieces of software together, largely independent of their location, connectivity mechanism, and technology used to develop them. These capabilities allow DOC middleware to amortize software life-cycle efforts by leveraging previous development expertise and reifying implementations of key patterns into more encompassing reusable frameworks and components. As DOC middleware continues to mature and incorporates additional needed services, next-generation applications will increasingly be assembled by modeling, integrating, and scripting domain-specific and common service components, rather than by

    1. Being programmed either entirely from scratch or
    2. Requiring significant customization or augmentation to off-the-shelf component implementations.

- ***Demand for end-to-end QoS support, not just component QoS*** – This area represents the next great wave of evolution for advanced DOC middleware. There is now widespread recognition that effective development of large-scale distributed applications requires the use of COTS infrastructure and service components. Moreover, the usability of the resulting products depends heavily on the properties of the whole as derived from its parts. This type of environment requires *visible*, *predictable*, *flexible*, and *integrated* resource management strategies within and between the pieces.

    Despite the ease of connectivity provided by middleware, however, constructing integrated systems remains hard since it requires significant customization of non-functional QoS properties, such as predictable latency/jitter/throughput, scalability, dependability, and security. In their most useful forms, these properties extend end-to-end and thus have elements applicable to

    - The network substrate
    - The platform operating systems and system services
    - The programming system in which they are developed
    - The applications themselves and

- The middleware that integrates all these elements together.

Two basic premises underlying the push towards end-to-end QoS support mediated by middleware are that:

1. Different levels of service are possible and desirable under different conditions and costs and
2. The level of service in one property must be coordinated with and/or traded off against the level of service in another to achieve the intended overall results.

To manage the increasingly stringent QoS demands of next-generation applications, middleware is becoming more adaptive and reflective. *Adaptive middleware* [Loy01] is software whose functional and QoS-related properties can be modified either:

- *Statically*, *e.g.,* to reduce footprint, leverage capabilities that exist in specific platforms, enable functional subsetting, and minimize hardware/software infrastructure dependencies or
- *Dynamically*, *e.g.,* to optimize system responses to changing environments or requirements, such as changing component interconnections, power levels, CPU/network bandwidth, latency/jitter; and dependability needs.

In mission-critical systems, adaptive middleware must make such modifications dependably, *i.e.*, while meeting stringent end-to-end QoS requirements. *Reflective middleware* [Bla99] goes further to permit automated examination of the capabilities it offers, and to permit automated adjustment to optimize those capabilities. Reflective techniques make the internal organization of systems–as well as the mechanisms used in their construction–both visible and manipulable for middleware and application programs to inspect and modify at runtime. Thus, reflective middleware supports more advanced adaptive behavior and more dynamic strategies keyed to current circumstances, *i.e.*, necessary adaptations can be performed autonomously based on conditions within the system, in the system's environment, or in system QoS policies defined by end-users.

- ***The increased viability of open systems*** architectures and open-source availability – By their very nature, systems developed by composing separate components are more open than systems conceived and developed as monolithic entities. The focus on interfaces for integrating and controlling the component parts leads naturally to *standard* interfaces. In turn, this yields the potential for multiple choices for component implementations, and open engineering concepts. Standards organizations such as the OMG and The Open Group have fostered the cooperative efforts needed to bring together groups of users and vendors to define domain-specific functionality that overlays open integrating architectures, forming a basis for industry-wide use of some software components. Once a common, open structure exists, it becomes feasible for a wide variety of participants to contribute to the off-the-shelf availability of additional parts needed to construct complete systems. Since few companies today can afford significant investments in internally funded R&D, it is increasingly important for the information technology industry to leverage externally funded R&D sources, such as government investment. In this context, standards-based DOC middleware serves as a common platform to help concentrate the results of R&D efforts and ensure smooth transition conduits from research groups into production systems.

  For example, research conducted under the DARPA Quorum program [Dar99] focused heavily on CORBA open systems middleware. Quorum yielded many results that transitioned into standardized service definitions and implementations for Real-time [OMG00B, Sch98A] and Fault-tolerant [Omg98a, Cuk98] CORBA specification and productization efforts. In this case, focused government R&D efforts leveraged their results by exporting them into, and combining them with, other on going public and private activities that also used a standards-based open middleware substrate. Prior to the viability of common middleware platforms, these same results would have been buried within a custom or proprietary system, serving only as the existence proof, not as the basis for incorporating into a larger whole.

- ***Increased leverage for disruptive technologies leading to increased global competition***– Middleware supporting component integration and reuse is a key technology to help amortize software life-cycle costs by:

  1. Leveraging previous development expertise, *e.g.*, DOC middleware helps to abstract commonly reused low-level OS concurrency and networking details away into higher-level, more easily used artifacts and
  2. Focusing on efforts to improve software quality and performance, *e.g.*, DOC middleware combines various aspects of a larger solution together, *e.g.*, fault tolerance for domain-specific objects with real-time QoS properties.

  When developers needn’t worry as much about low-level details, they are freed to focus on more strategic, larger scope, application-centric specializations concerns, such as distributed resource management and end-to-end dependability. Ultimately, this higher level focus will result in software -intensive distributed system components that apply reusable middleware to get smaller, faster, cheaper, and better at a predictable pace, just as computing and networking hardware do today. And that, in turn, will enable the next-generation of better and cheaper approaches to what are now carefully crafted custom solutions, which are often inflexible and proprietary. The result will be a new technological economy where developers can leverage:

  1. Frequently used common components, which come with steady innovation cycles resulting from a multi-user basis, in conjunction with

2. Custom domain-specific capabilities, which allow appropriate mixing of multi-user low cost and custom development for competitive advantage.

- ***Potential complexity cap for next-generation complex systems*** – As today’s technology transitions run their course, the systemic reduction in long-term R&D activities runs the risk of limiting the complexity of next-generation systems that can be developed and integrated using COTS hardware and software components. The advent of open DOC middleware standards, such as CORBA and Java-based technologies, is hastening industry consolidation towards portable and interoperable sets of COTS products that are readily available for purchase or open-source acquisition. These products are still deficient and/or immature, however, in their ability to handle some of the most important attributes needed to support future systems. Key attributes include end-to-end QoS, dynamic property tradeoffs, extreme scaling (large and small), highly mobile environments, and a variety of other inherent complexities. Complicating this situation over the past decade has been the steady flow of faculty, staff, and graduate students out of universities and research labs and into startup companies and other industrial positions. While this migration helped fuel the global economic boom in the late ‘90s, it does not bode well for long-term technology innovation.

  As distributed systems grow in complexity, it may not be possible to sustain the composition and integration perspective we have achieved with current middleware platforms without continued R&D. Even worse, we may plunge ahead with an inadequate knowledge base, reverting to a myriad of high-risk independent solutions to common problems. Ultimately, premium value and competitive advantage will accrue to those who master the patterns and pattern languages

[Sch00b] necessary to integrate COTS hardware and DOC middleware to develop low cost, complex distributed systems with superior domain-specific attributes that cannot be bought off-the-shelf at any given point in time.