

# Ασύρματα Δίκτυα Αισθητήρων

Πέππας Κωνσταντίνος

# Αρχιτεκτονική κόμβων

- The wireless sensor nodes are the central element in a wireless sensor network (WSN).
- It is through a node that sensing, processing, and communication take place.
- It stores and executes the communication protocols and the data-processing algorithms
- The quality, size, and frequency of the sensed data that can be extracted from the network are influenced by the physical resources available to the node
- Therefore, the design and implementation of a wireless sensor node is a critical step.

# Αρχιτεκτονική κόμβων

- The node consists of sensing, processing, communication, and power subsystems.
- The designer has a plethora of options in deciding how to build and put together these subsystems into a unified, programmable node.
- The processor subsystem is the central element of the node and the choice of a processor determines the tradeoff between flexibility and efficiency – in terms of both energy and performance.
- There are several processors as options: microcontrollers, digital signal processors, application-specific integrated circuits, and field programmable gate arrays

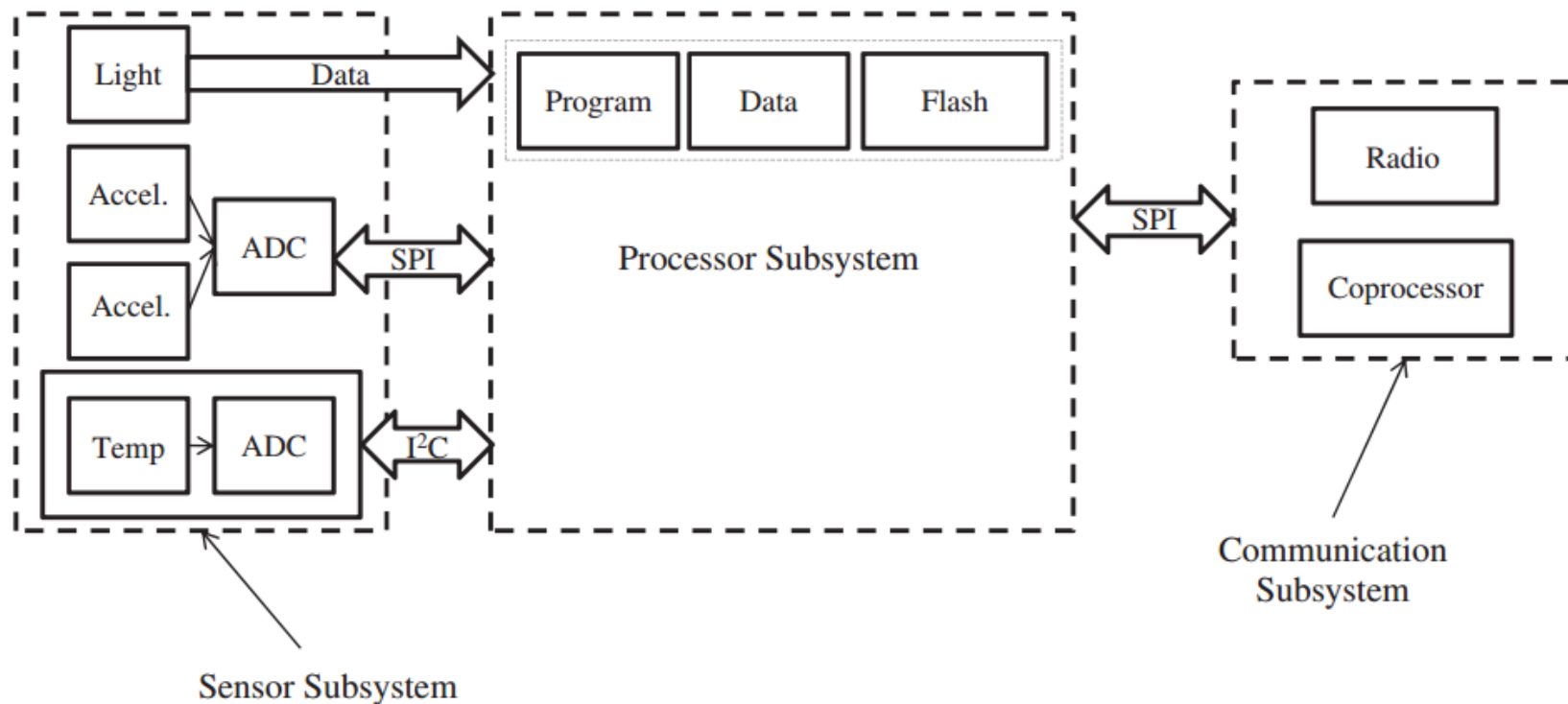
# Αρχιτεκτονική κόμβων

- There are a number of ways to connect the sensing subsystem with the processor.
- Connecting two or more analog sensors with a multichannel ADC system that integrates multiple high-speed ADCs into a single IC design is one way.
- However, these types of ADCs are known to produce crosstalk and to increase uncorrelated noise, reducing the signal-to-noise ratio (SNR) on individual channels.
- Moreover, the coupled signals can create harmonic terms, reducing total harmonic distortion (THD).
- For low-frequency signals, however, the effect is not significant.
- Some sensors have their own built-in ADC which can be directly connected with the processor through a standard chip-to-chip protocol.
- Most microcontrollers have one or more internal ADCs to interface analog devices.

# Αρχιτεκτονική κόμβων

- The communication subsystem can be interfaced with the processor subsystem in different ways.
- One way is to use the SPI serial bus. Some transceivers have their own processor board to perform low-level signal processing pertaining to the physical and the data link layer, thereby relieving the main processor from these concerns.
- The communication subsystem is the most energy intensive subsystem and its power consumption should be regulated.
- Almost all commercially available transceivers provide a controlling functionality to switch the transceiver between various active operation levels; idle and sleep state
- The power subsystem provides DC power to all the other subsystems to bias their active components such as crystal oscillators, amplifiers, registers, and counters.
- Moreover, it provides DC–DC converters so that each subsystem can obtain the right amount of bias voltage

# Αρχιτεκτονική κόμβων



**Figure 3.1** Architecture of a wireless sensor node.

# Αρχιτεκτονική κόμβων

- The sensing subsystem integrates one or more physical sensors and provides one or more analog-to-digital converters as well as the multiplexing mechanism to share them.
- The sensors interface the virtual world with the physical world.
- Nowadays, there are a plethora of sensors that measure and quantify physical attributes at a cheap price.
- A physical sensor contains a transducer, a device that converts one form of energy into another form of energy, typically into an electrical energy (voltage).
- The output of this transducer is an analog signal having a continuous magnitude as a function of time.
- Therefore, an analog-to-digital converter is required to interface a sensing subsystem with a digital processor.

# Αρχιτεκτονική κόμβων

- The analog-to-digital converter (ADC) converts the output of a sensor – which is a continuous, analog signal – into a digital signal. This process requires two steps:
- 1. The analog signal has to be quantized (i.e., converted from a continuous valued signal into a discrete valued signal; discrete both in time and magnitude).
- The most important decision at this stage is to determine the number of allowable discrete values.
- This decision in turn is influenced by two factors: (a) the frequency and magnitude of the signal; and (b) the available processing and storage resources.
- 2. The sampling frequency. In communication engineering and digital signal processing, this frequency is decided by the Nyquist rate.
- In wireless sensor networks, however, the Nyquist rate does not suffice. Oversampling is required because of noise.
- The prevailing consequence of the first step is the quantization error while the second is aliasing



# Αρχιτεκτονική κόμβων

- An ADC is specified, among other things, in terms of its resolution, which is an expression of the number of bits that can be used to encode the digital output.
- For example, an ADC with a resolution of 24 bits can represent 16,777,216 distinct discrete values.
- The resolution of an ADC can also be expressed in volts – since the output of most sensors is analog voltage.
- The voltage resolution of an ADC is equal to its overall voltage measurement range divided by the number of discrete intervals.
- In other words:

$$Q = \frac{E_{pp}}{2^M}$$

# Αρχιτεκτονική κόμβων

- $Q$  is the resolution in volts per step (volts per output code);  $E_{pp}$  is the peak-to-peak analog voltage;  $M$  is the ADC's resolution in bits.
- Here  $Q$  suggests that the interval between the discrete steps (values) is uniform. But in reality this is not so.
- In most ADCs, the least significant bit changes as a function of 0.5 times  $Q$  and the most significant bit changes as a function 1.5 times  $Q$ .
- Those bits in the middle have a resolution of  $Q$  voltage.
- In selecting an ADC, knowledge of the process or activity being monitored is important. Consider an industrial process whose thermal property ranges from  $-20$  to  $+80^\circ\text{C}$ .
- The choice of the physical sensor as well as the ADC depends on the type of thermal change that is of interest.
- If, for example, a change of  $0.5^\circ\text{C}$  is required, an ADC with a resolution of 8 bits is sufficient. If, on the other hand, a change of  $0.0625^\circ\text{C}$  is required, then the ADC should have a resolution of 11 bits.

# Αρχιτεκτονική κόμβων

- The processor subsystem brings together all the other subsystems and some additional peripherals.
- Its main purpose is to process (execute) instructions pertaining to sensing, communication, and self-organization.
- It consists of a processor chip, a nonvolatile memory (usually an internal flash memory) for storing program instructions, an active memory for temporarily storing the sensed data, and an internal clock, among other things.
- Whereas a wide range of off-the-shelf processors are available for building a wireless sensor node, one has to make a careful choice, as it affects the cost, flexibility, performance, and energy consumption of the node.
- If the sensing task is well defined from the outset and does not change over time, a designer may choose either a field programmable gate array or a digital signal processor.
- These processors are very efficient in terms of their energy consumption; and for most simple sensing tasks, they are quite adequate.
- However, as these are not general-purpose processors, the design and implementation process can be complex and costly

# Αρχιτεκτονική κόμβων

- $Q$  is the resolution in volts per step (volts per output code);  $E_{pp}$  is the peak-to-peak analog voltage;  $M$  is the ADC's resolution in bits.
- Here  $Q$  suggests that the interval between the discrete steps (values) is uniform. But in reality this is not so.
- In most ADCs, the least significant bit changes as a function of 0.5 times  $Q$  and the most significant bit changes as a function 1.5 times  $Q$ .
- Those bits in the middle have a resolution of  $Q$  voltage.
- In selecting an ADC, knowledge of the process or activity being monitored is important. Consider an industrial process whose thermal property ranges from  $-20$  to  $+80^\circ\text{C}$ .
- The choice of the physical sensor as well as the ADC depends on the type of thermal change that is of interest.
- If, for example, a change of  $0.5^\circ\text{C}$  is required, an ADC with a resolution of 8 bits is sufficient. If, on the other hand, a change of  $0.0625^\circ\text{C}$  is required, then the ADC should have a resolution of 11 bits.

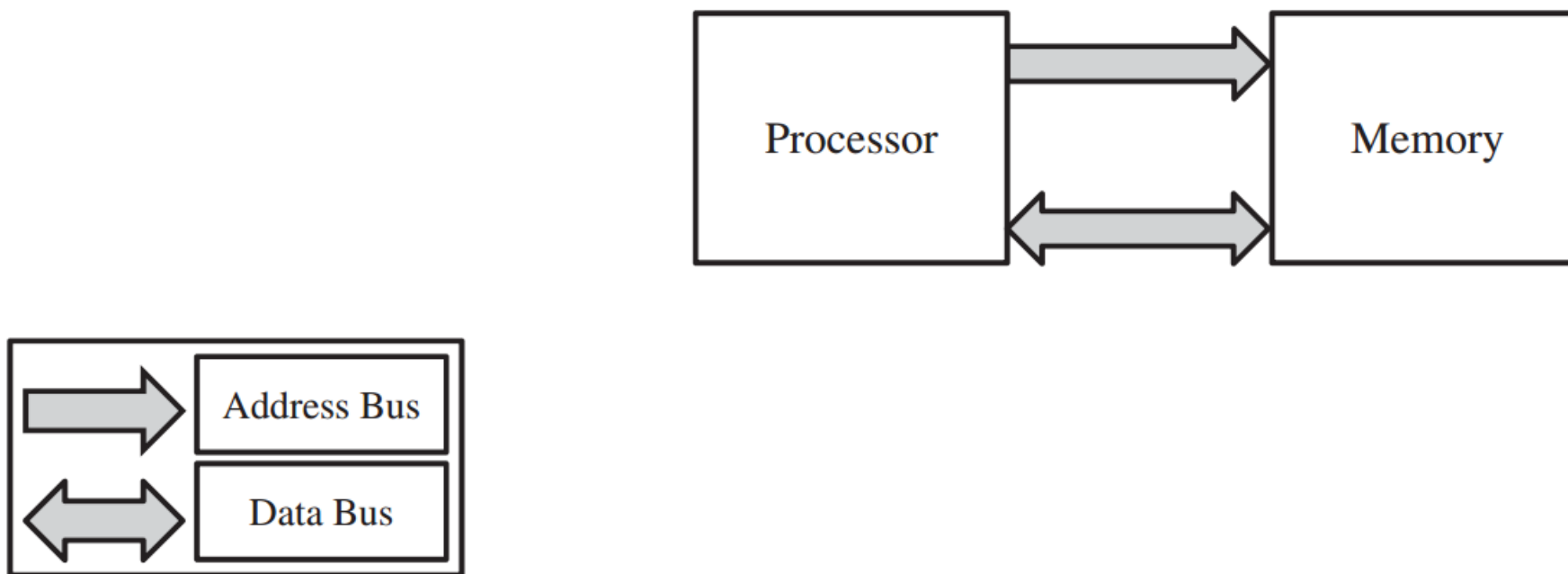
# Αρχιτεκτονική κόμβων

- In many practical cases, however, the sensing goal changes or a modification may be required.
- Moreover, the software that runs on the wireless sensor node may require occasional updates or remote debugging.
- Such tasks require a considerable amount of computation and processing space at runtime. In which case, special-purpose, energy-efficient processors are not suitable.
- Most existing sensor nodes at present use microcontrollers.
- There are some justifications besides those just mentioned.
- WSNs are emerging technologies; and the research community is still active with research for developing energy-efficient communication protocols and signal-processing algorithms.
- As this requires dynamic code installation and update, the microcontroller is the best option

# Αρχιτεκτονική κόμβων

- A major concern in resource-constrained processors is the efficient execution of algorithms, since this requires the transferring of information from and to memory.
- This includes program instructions and the data to be processed or manipulated.
- For example, in WSNs, the data stem from the physical sensors and the program instructions relate to communication, self-organization, data compression, and aggregation algorithms.
- The processor subsystem can be designed by employing one of the three basic computer architectures: Von Neumann, Harvard, and Super-Harvard (SHARC).
- The Von Neumann architecture provides a single memory space that is used by program instructions and data.
- It provides a single bus to transfer data between the processor and the memory.
- This architecture has a relatively slow processing speed because each data transfer requires a separate clock. Next Figure illustrates a simplified view of the Von Neumann architecture.

# Αρχιτεκτονική κόμβων



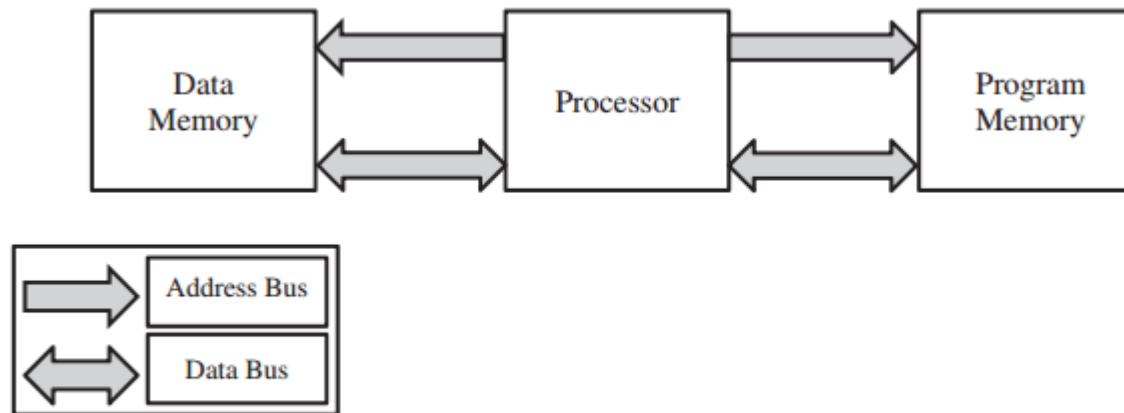
**Figure 3.2** The Von Neumann architecture.

# Αρχιτεκτονική κόμβων

- The Harvard architecture modifies the Von Neumann architecture by providing separate memory spaces for storing program instructions and data.
- Each memory space is interfaced with the processor with a separate data bus.
- In this way, program instructions and data can be accessed at the same time.
- Additional to this feature, the architecture supports a special single instruction, multiple data (SIMD) operation, a special arithmetic operation and a bit reverse addressing.
- It can easily support multitasking operating systems, but it has no virtual memory or memory protection. Next displays the Harvard architecture



# Αρχιτεκτονική κόμβων



**Figure 3.3** A simplified view of the Harvard architecture.

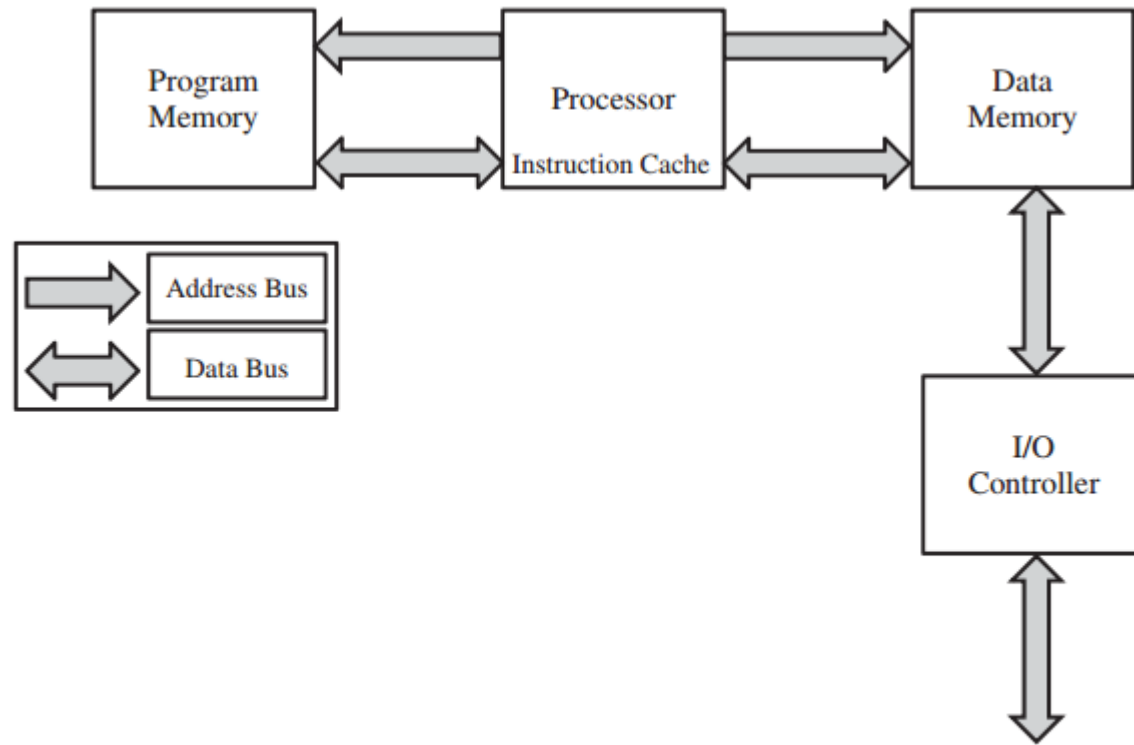
# Αρχιτεκτονική κόμβων

- The next generation of processor architecture is the Super-Harvard architecture, best known as SHARC. It is an extension of the Harvard system;
- it adds two essential components to its predecessor and provides alternatives for accessing I/O devices from within the processor subsystem.
- One of the components is an internal instruction cache that enhances the performance of the processor unit.
- It can be used to temporarily store frequently used instructions, thereby reducing the need for repeatedly fetching them from the program memory.
- Moreover, the architecture enables an underutilized program memory to be used as a temporary relocation place for data

# Αρχιτεκτονική κόμβων

- In SHARC, external I/O devices can directly be connected with the memory unit through an I/O controller.
- The configuration enables a direct data streaming from an external hardware into the data memory, without the need to involve the microcontroller. This is known as Direct Memory Access (DMA).
- DMA is desirable for two reasons:
  - (1) the costly CPU cycles can be invested in a different task; and
  - (2) it makes program memory bus and data memory bus accessible from outside the chip, providing an additional interface to off-chip memory and peripherals.
- Next Figure illustrates an overview of the SHARC architecture.

# Αρχιτεκτονική κόμβων



**Figure 3.4** An overview of the Super-Harvard architecture.

# Αρχιτεκτονική κόμβων

- A microcontroller is a computer on a single integrated circuit, consisting of a comparatively simple central processing unit and additional components such as high-speed buses, a memory unit, a watchdog timer, and an external clock.
- Microcontrollers are integrated in many products and embedded devices.
- Today such simple systems as elevators, ventilators, office machines, household appliances, power tools, and toys ubiquitously employ microcontrollers

# Αρχιτεκτονική κόμβων

- Typically, a microcontroller integrates the following components:
  - a CPU core that ranges from small and simple 4-bit to complex 32- or 64-bit processors;
  - a volatile memory (RAM) for data storage;
  - a ROM, EPROM, EEPROM, or flash memory for storing relatively simple instruction program code;
  - parallel I/O interfaces;
  - discrete input and output bits, allowing control or detection of the logic state of an individual package pin;
  - a clock generator – which is often an oscillator with a quartz timing crystal;
  - one or more internal analog-to-digital converters; and
  - serial communications interfaces such as Serial Peripheral Interface and Controller Area Network for interconnecting system peripherals such as event counters, a timer, and a watchdog.

# Αρχιτεκτονική κόμβων

- A microcontroller can be chosen over other types of small-scale processors because of the programming flexibility it offers.
- Its compact construction, small size, low power consumption, and low cost make it suitable for building computationally less intensive, standalone applications.
- Most of the commercially available microcontrollers can be programmed with assembly language and the C programming language.
- The use of higher-level programming languages increases the programming speed and eases debugging.
- There are development environments that offer an abstraction of all the functionalities of a microcontroller. This enables application developers to program microcontrollers without the need to have a low-level knowledge of the hardware.
- However, microcontrollers are not as powerful and as efficient as some custom-made processors such as digital signal processors (DSPs) and field programmable gate arrays (FPGAs).
- Moreover, for applications which demand simple sensing tasks but large-scale deployments (such as in precision agriculture and active volcano monitoring), one may prefer to use architecturally simple but energy- and cost-efficient processors such as application specific integrated circuits.

# Αρχιτεκτονική κόμβων

- A comprehensive understanding of DSPs requires a knowledge of digital signal processing.
- Broadly speaking, digital signal processing deals with processing discrete signals with digital filters. These filters minimize the effect of noise on a signal or selectively enhance or modify the spectral characteristics of a signal.
- While analog signal processing requires complex hardware components, digital signal processing, on the contrary, requires mainly simple adders, multipliers, and delay components.
- A DSP is a specialized microprocessor designed to carry out complex mathematical operations at an extremely high efficiency, processing hundreds of millions of samples every second and providing real-time performance.
- Most commercially available DSPs are designed with the Harvard architecture.



# Αρχιτεκτονική κόμβων

- Powerful and complex digital filters can be realized with commonplace DSPs.
- These filters perform remarkably well in signal detection and estimation, both of which require significant numerical computations.
- This is particularly interesting for multimedia WSN applications in which in-network audio and video signal processing may be required to compress or aggregate large size data.
- DSPs are also useful for applications that require the deployment of nodes in harsh physical settings where signal transmission may suffer corruption due to noise and interference.
- Having said this, a WSN carries out other tasks (tasks pertaining to network management, self-organization, multi-hop communication, topology control, etc.) in addition to numerical computations.
- These tasks require protocols which are not necessarily characterized as numerical operations.
- Furthermore, the protocols may require periodical upgrade or modifications, which means flexibility in network reprogramming is vital

# Αρχιτεκτονική κόμβων

- An application-specific integrated circuit (ASIC) is an integrated circuit (IC) that can be customized for a specific application.
- There are two types of design approaches: fully customized and half-customized. To understand the difference between these two, it is useful to understand the basic building blocks of an ASIC.
- The ASIC architecture consists of cells and metal interconnects.
- A cell is an abstraction of a logical functionality that is physically implemented by active components (transistors).
- When several of these cells are interconnected by the metal interconnects, they make up an application-specific integrated circuit.
- The manufacturing of cells has reached such a maturity that there is a standard library of cells consisting of a collection of low-level logic functions, including basic gates (AND, OR, and INVERT), multiplexers, adders, and flip-flops.

# Αρχιτεκτονική κόμβων

- As the standard cells have identical size, they can be arranged in rows to ease the process of automated digital layout.
- Using predefined cells from a cell library makes the ASIC design process much easier.
- In a fully customized IC, some (possibly all) logic cells, circuits, or layouts are custom made.
- The aim is to optimize cell performance (for example, execution speed) and to include features that are not defined or supported by the standard cell library.
- Fully customized ASICs are expensive and their design time is lengthy.
- On the other hand, a half-customized ASIC is built with logic cells that are available in the standard library.
- In both cases, the final logic structure is configured by the end user. This reduces time to market and financial risks by eliminating the need to cycle through an integrated circuit production.
- An ASIC is a cost-efficient solution, since the interconnections as well as the logic structure can be specified according to the user's needs. This also offers great flexibility and reusability

# Αρχιτεκτονική κόμβων

- Unlike a microcontroller, an ASIC can easily be designed and optimized to meet a specific customer demand.
- Even with a half-customized design, it is possible to design multiple microprocessor cores and embedded software in a single cell.
- Moreover, even though a fully customized ASIC is costly, with a hybrid approach (full custom and standard cell design), developers can achieve control over size and execution speed.
- Thus, it is possible to design with optimum performance and cost. Typical disadvantages include difficulties in designing, the lack of reconfigurability, and the usually high development costs.

# Αρχιτεκτονική κόμβων

- Perhaps the most suitable role of ASICs in WSNs is not to replace microcontrollers or DSPs but to complement them.
- As briefly mentioned some of the subsystems may integrate customized processors to handle rudimentary and low-level tasks and to decouple these tasks from the main processing subsystem.
- For example, some communication subsystems are shipped with an embedded processor core to enhance the quality of received signals, cancel noise, and perform cyclic redundancy checks.
- These types of special-purpose processors can be efficiently realized by making use of ASICs.

# Αρχιτεκτονική κόμβων

- The distinction between ASICs and FPGAs is not always clear.
- In fact, it is not unusual for companies that manufacture programmable ASICs to call their products FPGAs.
- While the basic anatomy of both architectures is essentially the same, FPGAs are more complex in design and more flexible to program.
- As the emphasis is on the (re)programming and reconfigurability aspect, typical features of a FPGA are summarized as follows:
  - in a FPGA, none of the mask layers is customized;
  - a FPGA includes some programmable logic components, or *logic blocks* – these are: a 4-input lookup table (LUT), a flip-flop, and an output block;
  - there is a well-defined and formal method for programming the basic logic cells and the interconnect;
  - there is a matrix of programmable interconnects surrounding the basic logic cells producing a configuration instance; and
  - there are programmable I/O cells that surround the core.

# Αρχιτεκτονική κόμβων

- FPGAs are programmed electrically, by modifying a packaged part.
- This process may take from a few milliseconds to a few minutes, depending on the programming technology and the size of the part.
- The programming is done with the support of circuit diagrams and hardware description languages, such as VHDL and Verilog.
- FPGAs have a higher bandwidth compared to DSPs; they are more flexible in their application and can support parallel processing.
- While DSPs and microcontrollers can incorporate an internal ADC, a FPGA does not.
- Similar to a DSP, a FPGA has the capability to work with floating point representation.
- Additionally, a FPGA exposes its processing speed to application developers, thereby giving them a greater flexibility of control.
- On the other hand, FPGAs are complex; and the design and realization process is costly.

# Αρχιτεκτονική κόμβων

- Working with a microcontroller is preferred if the design goal is to achieve flexibility.
- Working with all the others is preferred if power consumption and computational efficiency is desired.
- Whereas microcontrollers have limited memory, steady progress is being made to increase memory size.
- Recently, more and more microcontrollers are becoming available on the market with attractive features; for example, the TI MSP430F2618 and MSP430F5437 offer an active memory (RAM) of 8 kilobytes (KB) and 16 KB, as well as 116 KB and 256 KB flash memory respectively.
- Both consume less power and perform better than earlier models. The Atmel ATMega1281 and its next version ATMega2561 also have good architecture and better memory and performance.
- Both of them have 8 KB active memory and 128 KB and 256 KB flash memory, respectively.
- The Jennic architecture – JN5121 and JN5139 – integrates a microprocessor and a radio subsystem into a single package to enhance processing speed. It has 96 KB and 192 KB RAM, respectively; and 128 KB flash memory.



# Αρχιτεκτονική κόμβων

- In comparison, DSPs are expensive, large in size, and less flexible.
- For example the PIC 16F873 and Sx28AC microcontrollers have 5 million instructions per second (MIPs) and 75 MIPS processing power;
- 24 and 20 general purpose I/O; 4 KB and 2 KB RAM, respectively; and 28-pin each.
- The devices cost around \$5.81 and \$4.05, respectively. Compared to that, the DSP56364 has 100 MIPS processing power, 16 general-purpose I/O, 1 KB RAM, 100 or 112-pin and costs \$11.00.
- Moreover, DSPs are best for signal processing, with specific algorithms.

# Αρχιτεκτονική κόμβων

- FPGAs are faster than both microcontrollers and digital signal processors and support parallel computing.
- In wireless sensor networks, since sensing, processing, and communication should take place at the same time, FPGAs can be useful.
- However, their production cost and the difficulty with programming make them less desirable.
- ASICs have higher bandwidths; they are the smallest in size, perform much better, and consume less power than any of the other processing types.
- Their main disadvantage is the high cost of production owing to the complex design process, usually with lower production quantity and a reduced reusability.
- Performance can be improved with the application of multicore systems where several applications could run in parallel.
- This enables the integration of ASICs into the other subsystems, so that when the main processor subsystem is idle and therefore should be turned off, elementary and rudimentary tasks can be carried out by the more efficient ASICs.

# Αρχιτεκτονική κόμβων

- As the selection of the right type of processor is vital to the performance as well as the energy consumption of a wireless sensor node, the way the subcomponents are interconnected with the processor subsystem is also vital.
- Fast and energy-efficient data transfer between the subsystems of a wireless sensor node is critical to the overall efficiency of the network it sets up.
- However, the practical size of the node puts a restriction on system buses.
- Whereas communication via a parallel bus is faster than a serial bus, a parallel bus needs more space.
- Moreover, it requires a dedicated line for every bit that should be transmitted simultaneously while the serial bus requires a single data line only.
- Owing to the size of the node, parallel buses are never supported in node design.
- The choice, therefore, is often between serial interfaces such as the serial peripheral interface (SPI), the general purpose input/output (GPIO), the secure data input/output (SDIO), the inter-integrated circuit (I<sup>2</sup>C), and the Universal Serial Bus (USB).
- Among these, the most commonly used buses are the SPI and the I<sup>2</sup>C.

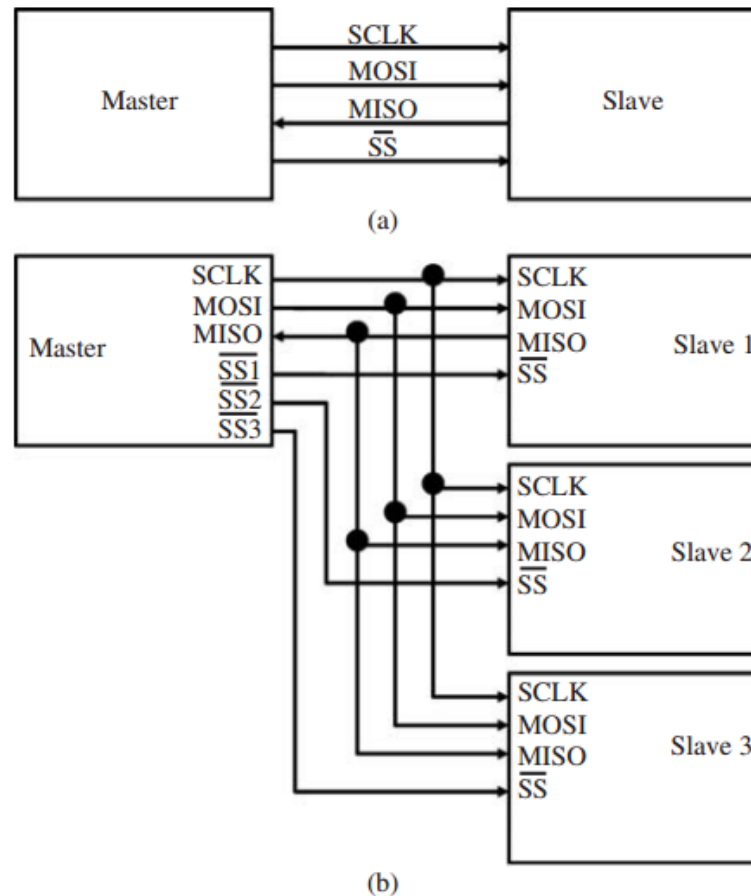
# Αρχιτεκτονική κόμβων

- The Serial Peripheral Interface (SPI – pronounced as “spy”) is a high-speed, full-duplex synchronous serial bus.
- It was developed at Motorola in the mid-1980s.
- It does not have an official standard as such, but manufacturers building devices that use the SPI should conform to the implementation specification of other manufacturers in order to support correct communication
- (for example, devices should agree on whether to transmit the most significant bit (MSB) or the least significant bit (LSB) first)
- The SPI bus defines four pins: (Master-Out/Slave-In) MOSI, (Master-In/Slave-Out) MISO, (Serial Clock) SCLK, and (Chip Select) CS.
- Some manufacturers refer to MOSI as SIMO and to MISO as SOMI, but the semantics is the same.

# Αρχιτεκτονική κόμβων

- Likewise, CS is sometimes referred to as (Slave Select) SS.
- As the name suggests, MOSI is used to transmit data from the master to the slave when a device is configured as a master.
- In case it is configured as a slave, this port is used to receive data from the corresponding master.
- The semantics are reversed for the MISO port.
- SCLK is used by the master to send the clock signal that is needed to synchronize transmission; and by the slave to read this signal.
- Every communication is initiated by the master.
- A master device signals a slave with which it wants to communicate via the CS port.
- Since SPI is a single master bus, the microcontroller is by default the master in a wireless sensor node.
- Thus, components cannot communicate directly with each other but only via the microcontroller – for example, with this configuration, an ADC cannot send a sampled data directly to a RAM.
- Figure 3.5 illustrates two types of configurations. In (a), a single master communicates with a single slave device, while in (b), a master is connected with multiple slave devices.

# Αρχιτεκτονική κόμβων



**Figure 3.5** Connecting devices with the SPI bus. (a) A single master connecting with a single slave. (b) A single master connecting with multiple slaves.

# Αρχιτεκτονική κόμβων

- Both master and slave devices hold shift registers.
- In most cases these are 8-bit registers, but heterogeneous sizes are also allowed.
- Both registers are connected in a ring-forming 16-bit shift register.
- This is the common mode of connection.
- Assuming that the MSB is transferred first, during a transmission cycle, the MSB that is sent by the master is inserted to the slave's LSB register while, in the same cycle, the slave's MSB is shifted to the master's LSB.
- After all bytes have been sent, the slave's register contains the master's word, while the master holds the slave's word
- Since master and slave form a commonly used shift register, every device in every transmission must read and send data.
- For devices that do not provide feedback (for example, LC displays do not offer status or bug messages) or do not require input data (some devices may not accept any commands at all), this means adding pseudo bytes into the shift register.

# Αρχιτεκτονική κόμβων

- SPI supports a synchronous communication protocol.
- Consequently, the master and the slave must agree on the timing.
- To do so, the master sets the clock according to the slave's maximum clock speed – the baud generator of the master reads the slave's clock and calculates the master's clock by dividing the read speed with an internally defined value.
- Furthermore, master and slave should agree on two additional parameters, namely, clock polarity (CPOL) and clock phase (CPHA).
- CPOL defines whether a clock is used in high or low-active mode.
- CPHA determines the times when the data in the registers are allowed to change and when the written data can be read.
- There are four different combinations (shown in Table 3.2) which are all incompatible with each other.



# Αρχιτεκτονική κόμβων

**Table 3.2** Common SPI modes

SPI mode	CPOL	CPHA	Description
0	0	0	SCLK is low-active. Sampling is allowed on odd clock edges. Data changes on even clock edges.
1	0	1	SCLK is low-active. Sampling is allowed on even clock edges. Data changes on odd clock edges.
2	1	0	SCLK is high-active. Sampling is allowed on odd clock edges. Data changes on even clock edges.
3	1	1	SCLK is high-active. Sampling is allowed on even clock edges. Data changes on odd clock edges.

# Αρχιτεκτονική κόμβων

- The inter-integrated circuit (I2C) is a multi-master half-duplex synchronous serial bus.
- It was developed by Philips Semiconductors, which is also the owner of the official standard
- I2C uses only two bidirectional lines (unlike SPI, which uses four).

The aim of I2C is to minimize costs for connecting devices within a system by accommodating lower transmission speeds.

- I2C defines two speed modes: a Fast-mode, with a bit rate of up to 400 kbps and a High-speed-mode (referred to as Hs-mode) that supports a transmission rate of up to 3.4 Mbps.
- The 100 kbps rate (Standard-mode) was defined in earlier versions.
- Nevertheless, Fast-mode and Hs-mode components are downwards compatible to ensure communication with older components.

# Αρχιτεκτονική κόμβων

- As the standard does not specify a CS or SS port, every device type that uses I2C must have a unique address that will be used to communicate with a device.
- In earlier versions, a 7-bit address was used, allowing 112 devices to be uniquely addressed (4 bits are reserved).
- This address space turned out to be insufficient due to an increasing number of devices. Currently I2C uses 10-bit addressing.
- In the old protocol, a master device flags the start condition(s) and transmits the slave's 7-bit address.
- Then the master expresses *read or write* interest.
- At this time, the slave sends an acknowledgment (ACK). Afterwards, the data transmitter sends a 1-byte (8-bits) data, which is then acknowledged by the receiver.
- If there is still data to be sent, the transmitter keeps on sending and the receiver keeps on acknowledging in return.
- Finally, the master raises the stop flag (stop condition) to signify the end of a communication.

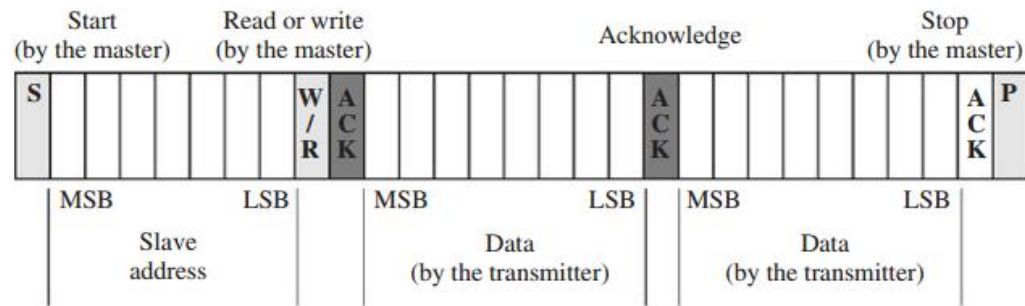
# Αρχιτεκτονική κόμβων

- In the new protocol, after the start condition (S), a leading 11110 introduces the 10-bit addressing scheme.
- The last two address bits of the first byte concatenated with the eight bits of the second byte form the whole 10-bit address.
- Devices that only use 7-bit addressing simply ignore messages with the leading 11110.
- As already mentioned, I2C provides two lines; these are Serial Clock (SCL) and Serial Data Analyzer (SDA).
- Hs-mode devices have additional ports called SDAH and SCLH.
- Since each master generates its own clock signal, communicating devices must synchronize their clock speeds.

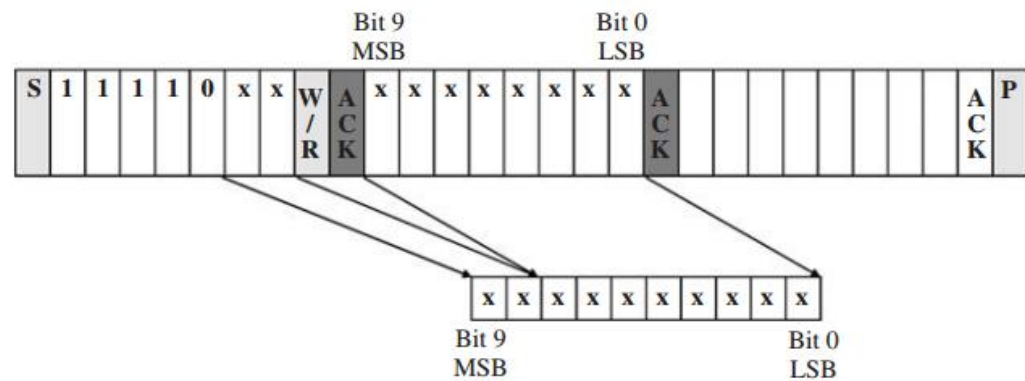
# Αρχιτεκτονική κόμβων

- In case they do not, a slower slave device could wrongly detect its address on the SDA line while a faster master device is sending data to a third device.
- Besides clock synchronization, I2C requires arbitration between master devices wanting to send or receive data at the same time.
- I2C does not explicitly define any fair arbitration algorithm. Rather the master that holds the SDA line low for the longest time wins the medium.
- Additionally, I2C enables a device to read data at a byte level for fast communication.
- This, however, may raise the need for more time to store the received bytes, in which case the device can hold the SCL low until it completes reading or sending the next byte.
- This type of clock synchronization is called handshaking.

# Αρχιτεκτονική κόμβων



(a)



(b)

Figure 3.7 Communication protocols in the I<sup>2</sup>C serial bus.

# Αρχιτεκτονική κόμβων

**Table 3.3** Comparison of SPI and I<sup>2</sup>C

SPI	I <sup>2</sup> C
<ul style="list-style-type: none"><li>• Four lines enable full-duplex transmission.</li><li>• No addressing is required due to CS. This reduces overhead and increases throughput. However, one needs additional hardware configurations to connect more than one slave.</li><li>• Allowing only one master avoids conflicts.</li><li>• Hardware requirement support increases with an increasing number of connected devices and therefore raises costs.</li><li>• The master's clock is configured according to the slave's speed. This frees the slaves from requiring clocking a device. However, speed adaptation slows down the master.</li><li>• Speed depends on the maximum speed of the slowest device.</li><li>• Heterogeneous register size allows flexibility in the devices that are supported.</li><li>• Combined registers imply every transmission should be read.</li><li>• The absence of an official standard leads to application specific implementations.</li></ul>	<ul style="list-style-type: none"><li>• Two lines reduce space and simplify circuit layout. Lowers costs.</li><li>• Addressing enables multimaster mode, which in turn enables more than one device to initiate communication.</li><li>• Multimaster mode is prone to conflicts when two or more master devices communicate simultaneously. Arbitration is required.</li><li>• Hardware requirement is independent of the number of devices using the bus.</li><li>• Slower devices may stretch the clock thereby increasing latency and keeping other devices waiting for accessing the bus.</li><li>• Speed is limited to 3.4 MHz and all devices need to support the highest speed that is used in the system, otherwise a slower device may wrongly detect its device address.</li><li>• Homogeneous register size reduces overhead, since no additional control bits are required to be transmitted.</li><li>• Devices that do not read or provide data are not forced to provide potentially useless bytes.</li><li>• Official standard eases integration of devices since developers can rely on a certain implementation.</li></ul>

# Αρχιτεκτονική κόμβων

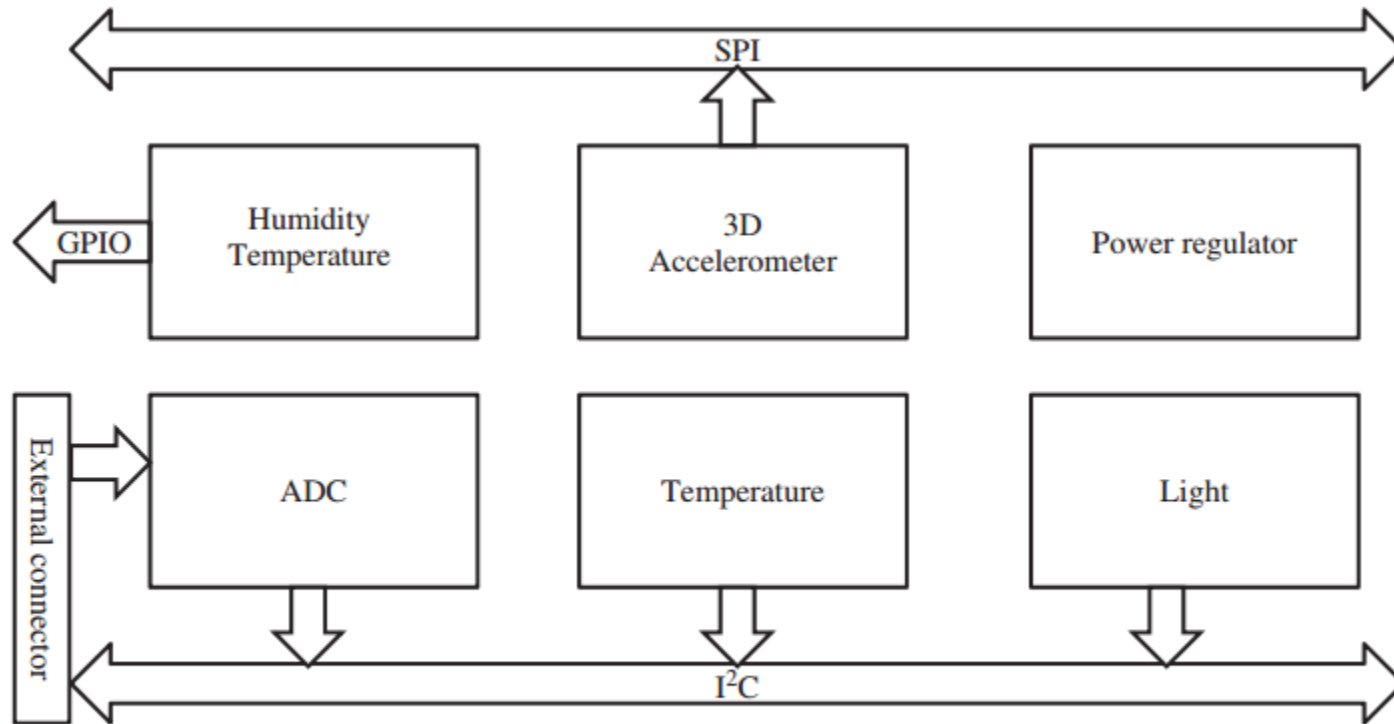
- Buses are essential highways to transfer data between the processor subsystem and the other subsystems.
- Due to the concern for size, only serial buses can be used by a wireless sensor node. These buses demand high clock speeds to gain the same throughput that can be achieved with parallel buses.
- However, they can also be bottlenecks; this is particularly the case with the Von Neumann architecture, since the same bus is used for data as well as instructions.
- They also do not scale well with processor speed.
- For example I2C in its latest version is limited to 3.4 MHz while the clock speed of one of the most commonly used microcontroller family, the TI MSP430x1xx series, has a clock frequency of 8 MHz.
- Delays due to contention for bus access become critical if some of the devices act unfairly and keep the bus occupied.
- For example, I2C allows slave devices to stretch clock signals if it is deemed appropriate to “packet” communication and give priority to components that need to exchange time-critical data.



# Αρχιτεκτονική κόμβων

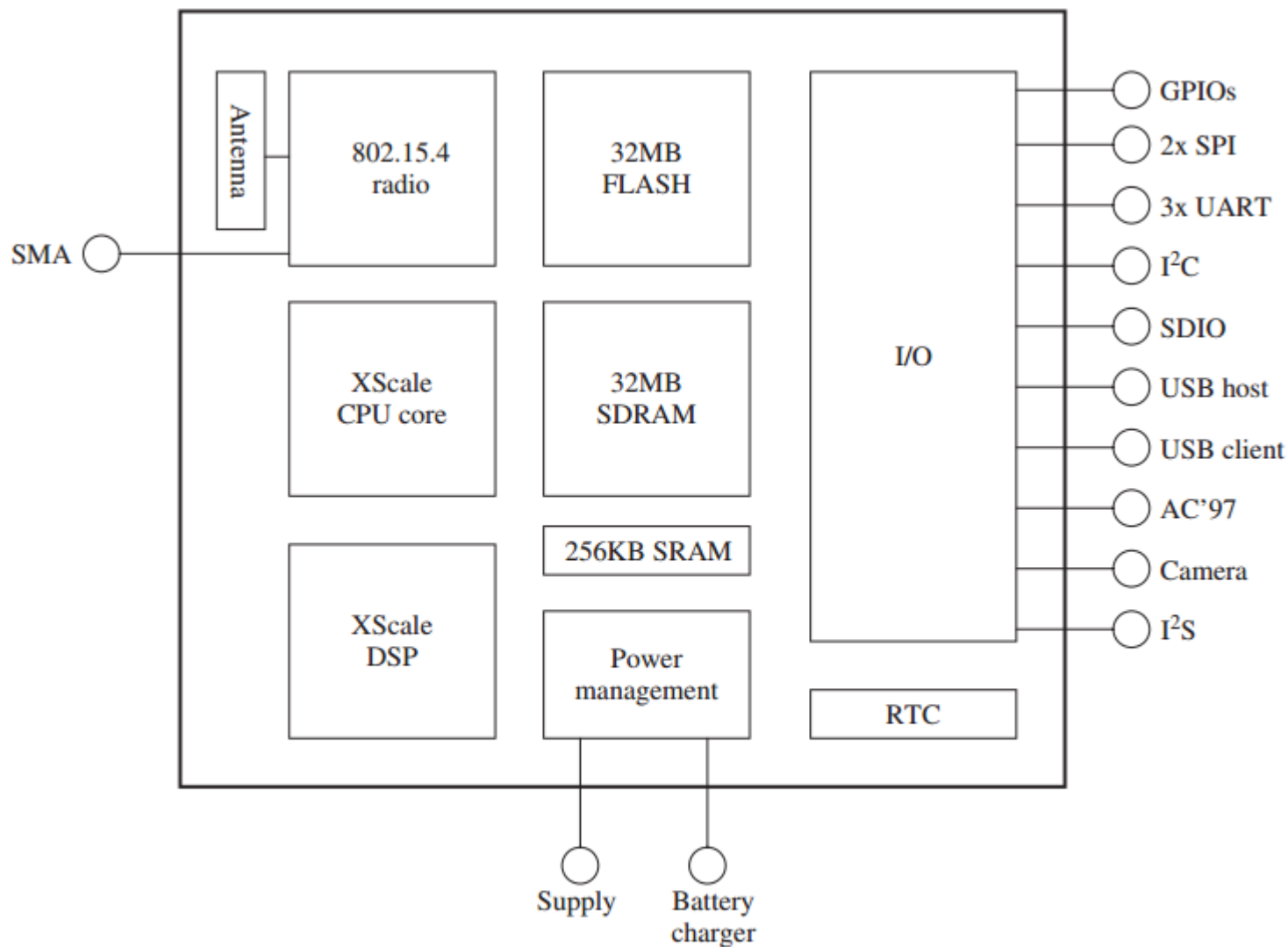
- The IMote sensor node architecture (Figure 3.8) is a multipurpose architecture that consists of a power management subsystem, a processor subsystem, a sensing subsystem, a communication subsystem, and an interfacing subsystem.
- The sensing subsystem (Figure 3.9) provides an extensible platform to connect multiple sensor boards.
- One realization of the sensor board contains a 12-bit, 4-channel ADC; a high resolution temperature/humidity sensor; a low-resolution digital temperature sensor; and a light sensor.
- These devices are interfaced to the processing subsystem through the SPI and I2C buses.
- As can be seen in the figure, the I2C bus is chosen to connect low data rate sources whereas the SPI bus is used to interface high data rate sources.
- The processing subsystem provides a main processor (microprocessor) and a digital signal processor (DSP).
- The main processor has the ability to operate at a low-voltage (0.85 V) and a low-frequency (13 MHz) mode, thus enabling low power operation.
- Likewise, the frequency can be scaled to 104 MHz at the lowest voltage level and can be increased up to 416 MHz with Dynamic Voltage Scaling (DVS).
- Moreover, it has many low power modes, including sleep and deep sleep modes. The coprocessor is intended to accelerate multimedia operations, which are computation intensive.

# Αρχιτεκτονική κόμβων



**Figure 3.9** The sensing subsystem of the IMote architecture.

# Αρχιτεκτονική κόμβων



# Αρχιτεκτονική κόμβων

- The XYZ architecture consists of four subsystems. Figure 3.10 shows the schematic diagram of the node architecture.
- The processor subsystem is based on the ARM7TDMI core microcontroller, which is capable of operating at a maximum frequency of 58 MHz.
- The microcontroller can operate at two different modes, depending on the application requirement: at 32 bits and 16 bits.
- The processing subsystem provides an on-chip memory of 4 KB boot ROM and a 32 KB RAM, which can be extended by up to 512 KB of flash memory.
- The peripheral components that interface the processing subsystem with the remaining subsystem include an embedded DMA controller, four 10-bit ADC inputs, serial ports (RS232, SPI, I2C, SIO), and 42 multiplexed general-purpose I/O pins.

# Αρχιτεκτονική κόμβων

- Most of the multiplexed GPIO pins are available on two 30-pin headers together with the DC voltage provided by the power subsystem or directly by an on-board voltage regulator.
- The communication subsystem is based on the Chipcon CC2420 radio, which is connected to the processing subsystem through a SPI interface.
- The CC2420 is a 2.4 GHz IEEE 802.15.4 compliant single-chip RF transceiver.
- The processor subsystem controls the communication subsystem by either turning it off or putting it in sleep mode.
- The communication through the SPI interface enables the radio to wake up a sleeping processor when an RF message has been successfully received.

# Αρχιτεκτονική κόμβων

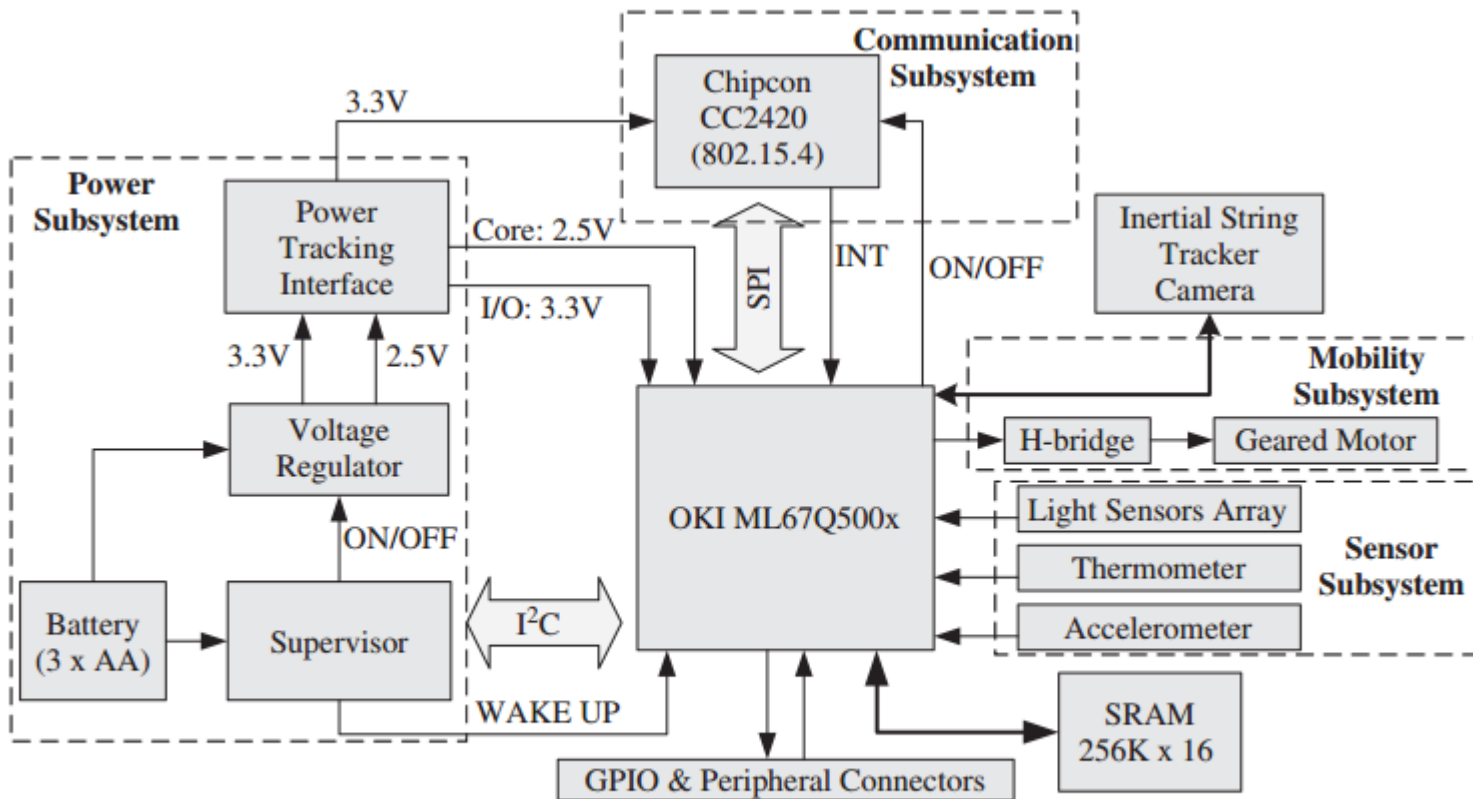


Figure 3.10 The XYZ node architecture (Lymberopoulos and Savvides 2005).

# Αρχιτεκτονική κόμβων

- The Hogthrob node architecture (Bonnet *et al.* 2006) is designed for a specific application, namely, to monitor the activities of sows in a large-scale pig production.
- The basic assumption behind the main sensing task is that there is a direct correlation between the movement of a sow and the onset of estrus.
- Therefore, a network that is established by nodes worn by sows monitors movements to capture this vital state, so that appropriate care can be given for pregnant sows.
- For example, in Denmark a law is already in place that requires pregnant sows to move freely in a large pen. Apart from this, other vital conditions, such as illness (by detecting cough or limping), are also monitored by the sensor network.

# Αρχιτεκτονική κόμβων

- The node architecture consists of the usual subsystems.
- Unlike many existing architectures, the Hogthrob node's processing subsystem consists of two processors, a microcontroller, and a field programmable gate array (FPGA).
- The microcontroller performs less complex, less energy-intensive tasks, such as controlling the communication subsystem and other peripherals.
- It also initializes the FPGA and functions as an external timer and an ADC converter to it.
- The FPGA executes the sow monitoring application and coordinates the functions of the sensor node.
- Figure 3.11 displays a partial view of the node architecture and the various interfacing buses.
- There are a number of interfaces supported by the processing subsystem, including the I2C interface for the sensing subsystem, the SPI for the communication subsystem, the JTAG interface for in-system programmability and debugging, and the serial (RS-232) interface for interaction with a PC.



# Αρχιτεκτονική κόμβων

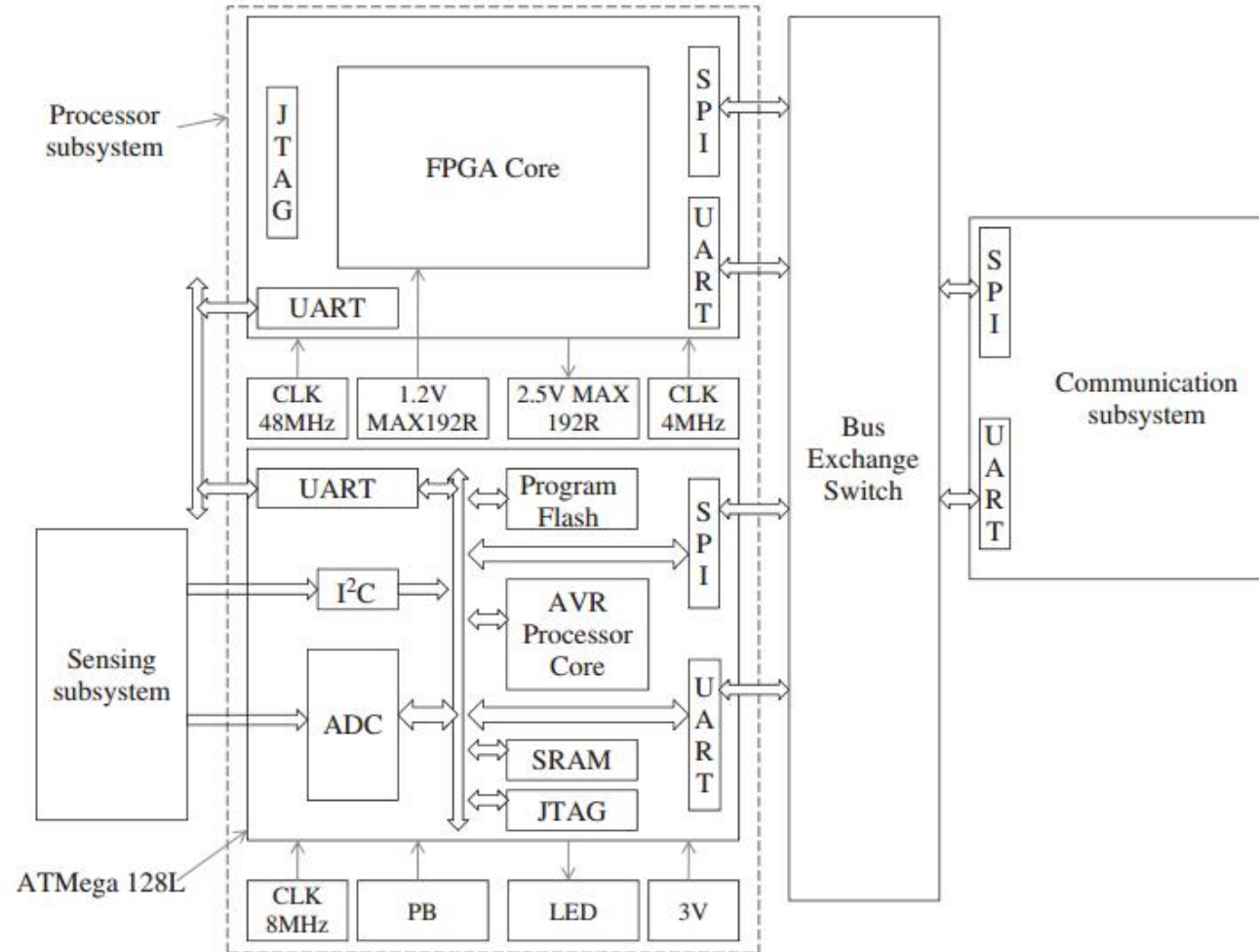


Figure 3.11 A partial view of the Hogthrob node architecture (Bonnet *et al.* 2006).