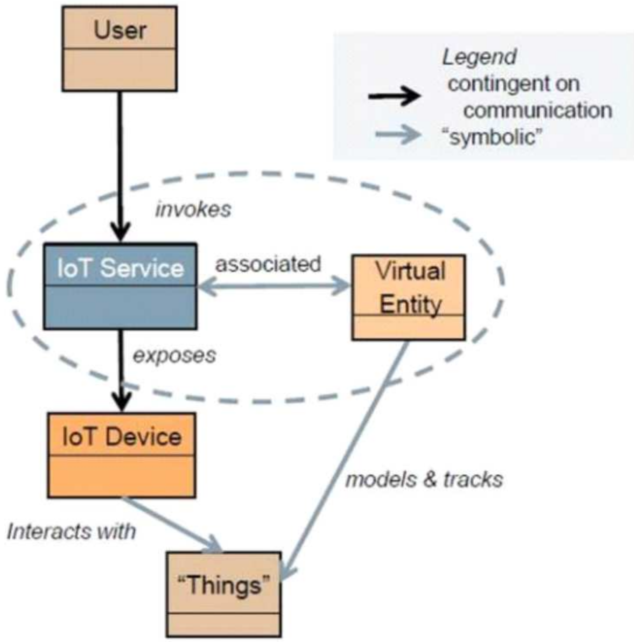
EIOT

Energy

Aleksander Pruszkowski

Instytut Telekomunikacji Politechniki Warszawskiej



This lecture

- Electrical energy: background
- Sources of energy for IoT nodes
- Energy consumption levels in IoT nodes
- How to reduce energy consumption in IoT nodes

Electrical energy: background

Electrical energy: background

- Electrical energy is energy that electrical current delivers to electrical load doing work …

$$E_{electrical} = I * U * t$$

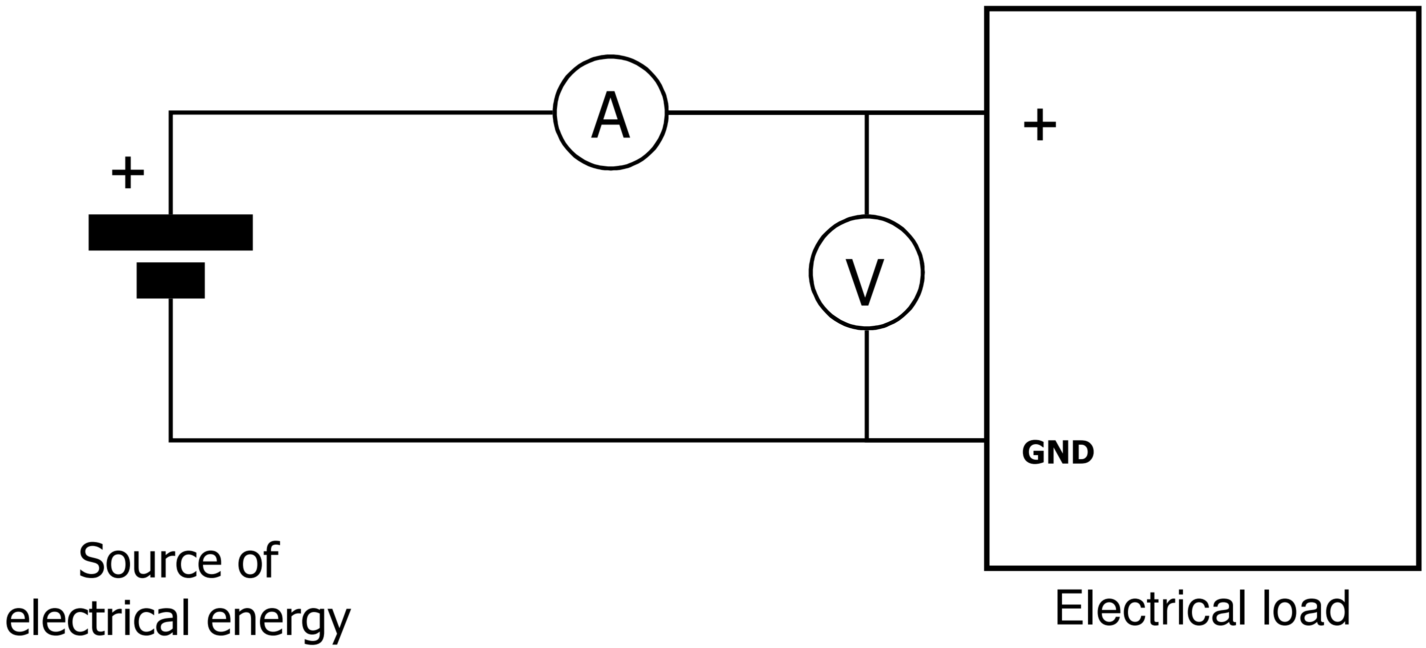
where

- I – current flowing through the electrical load [A]
- U – voltage at the load [V]
- t – time (how long the current flows) [s]

- Thus units of $E_{electrical}$ are (all are allowed)
  - A*V*s or
  - W*s or
  - (more often) Wh or
  - kWh

Electrical energy: background

- How to measure electrical energy: „textbook approach”

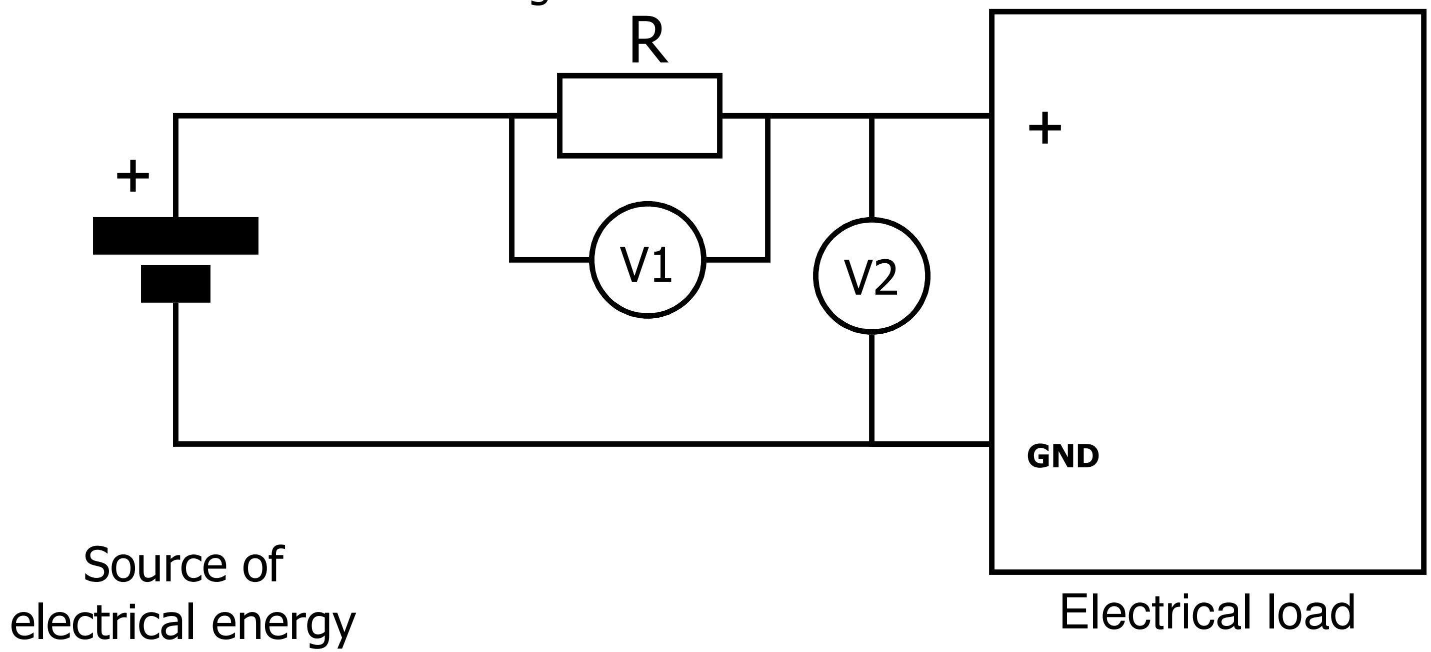


Electrical energy: background

- How to measure electrical energy: practical approach
  - find the current based on voltage drop

$$E_{electrical} = I * V2 * t = (V1 / R) * V2 * t$$

  - cons:
    - voltage drop across R amounts to losing energy
    - the voltage meter V1 is connected to non-ground terminals, which makes the entire meter hard to integrate

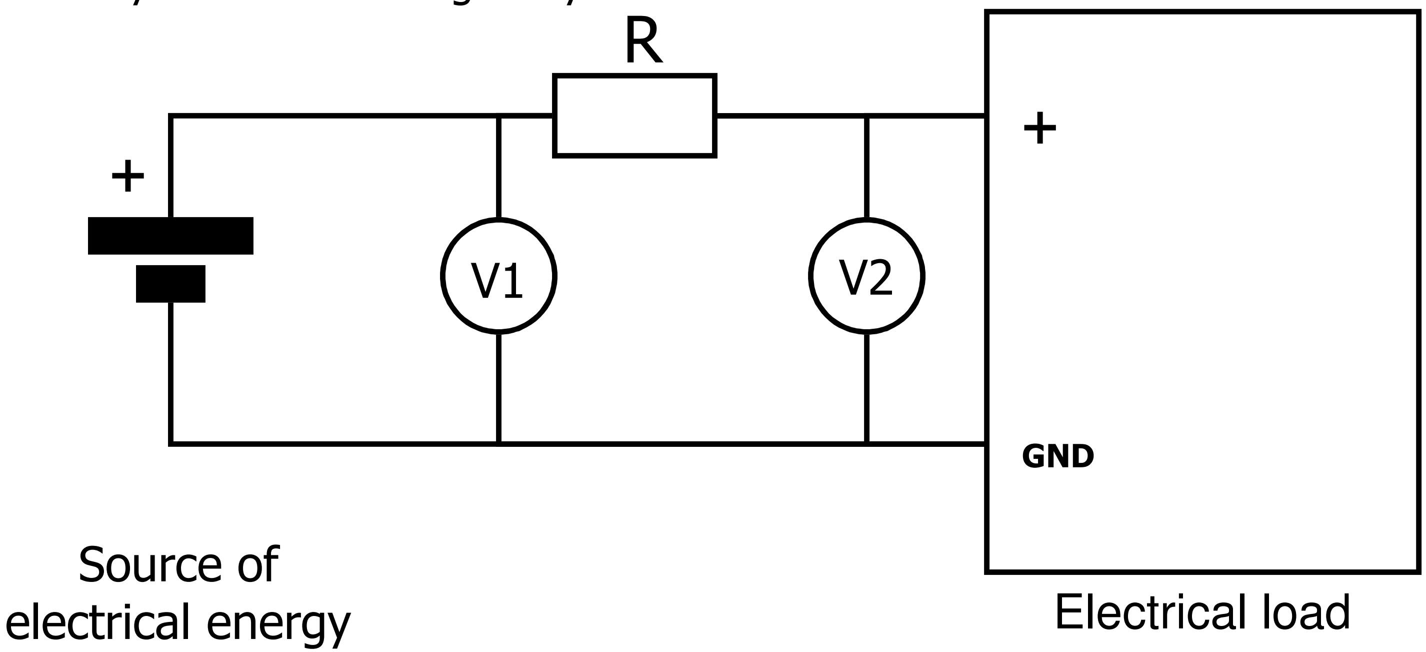


Electrical energy: background

- How to measure electrical energy: practical approach modified
  - find the current based on voltage drop

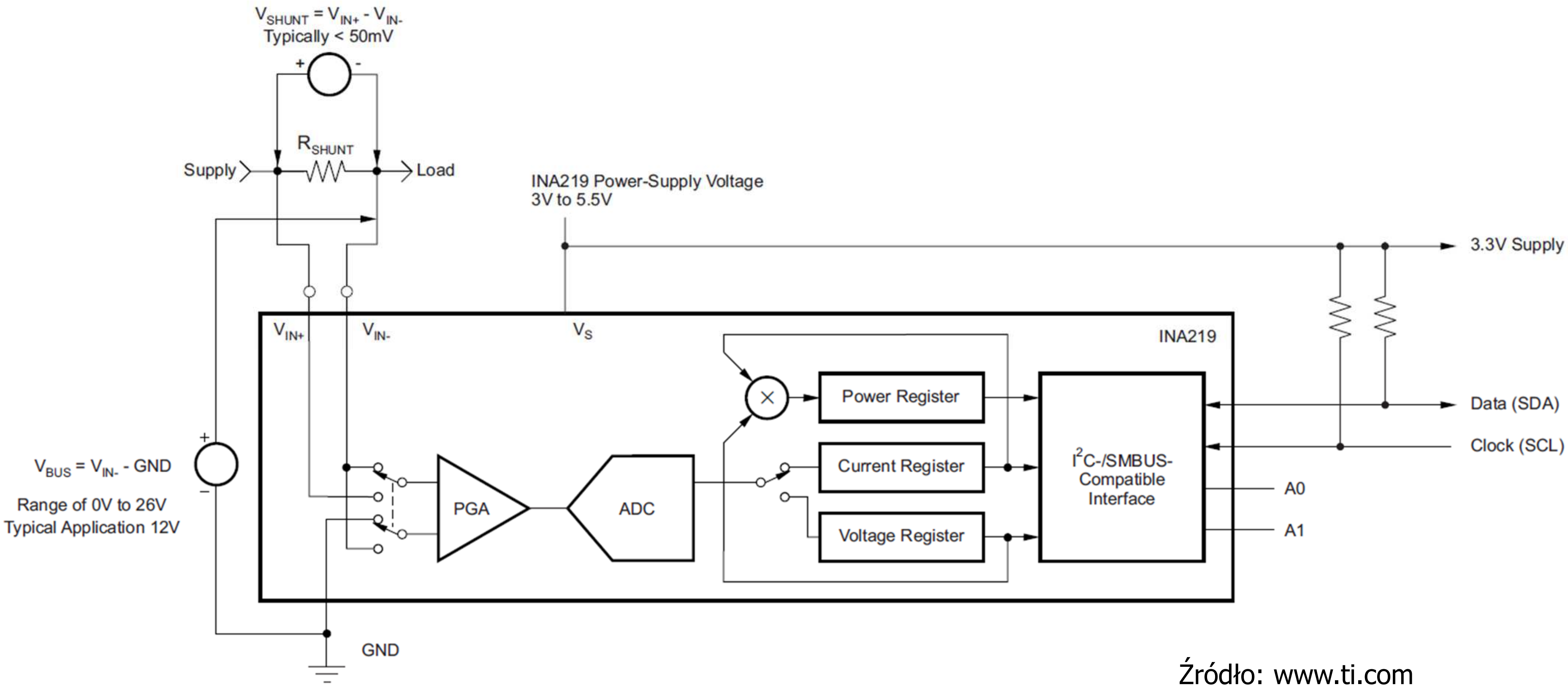
$$E_{electrical} = (V1-V2) / R * V2 * t$$

  - cons:
    - voltage drop across R amounts to losing energy
    - the voltage difference V1-V2 is hard to measure without an error, which cannot be easily eliminated in digital systems



Electrical energy: background

- How to measure electrical energy: practical approach modified
  - find the current based on voltage drop
  - example: **INA219** IC („power monitor”)
    - ADC: 12bit, 1LSB, conversion time <586us



Źródło: www.ti.com

Electrical energy: background

- Handling a power monitor in software
  - example: INA219 (Adafruit library for Arduino, R_{SHUNT}=0.1Ω)

```
#include <Wire.h>
#include <Adafruit_INA219.h>

Adafruit_INA219 ina219;

ina219.begin();          //init. for range up to: 32V/2A

shuntvoltage=ina219.getShuntVoltage_mV();
busvoltage=ina219.getBusVoltage_V();
current_mA=ina219.getCurrent_mA();
power_mW=ina219.getPower_mW();
```

Sources of energy for IoT nodes

Sources of energy: battery cells

- Electricity-related features
  - nominal voltage
    - unit: volt [V]
    - the voltage that is measured, for a given current, at the cell's terminals
      - in reality, voltage drops, as the battery is discharged
  - capacity
    - unit: ampere hour [Ah] and related ones: [As], [mAh]
      - 1Ah is electric charge of 3600 coulombs [C]
    - specifies how long you can draw energy from the cell
      - often one uses the term „10h current", e.g., for the capacity of 330mAh, you can draw, for 10 hours, the current of 33mA
  - internal resistance
    - unit: ohm [Ω]
    - resistance „in series"
      - affects how much current you can draw for a short period of time
    - resistance „in parallel"
      - affects the current flowing without any load (self-discharge)

Sources of energy: battery cells

- Physical features
  - reaction to low temperatures
    - in reaction to low temperatures (even those close to 0^{o}C), cells tend to reduce their current capacity
      - they appear as if they discharged faster
  - connectors, size, weight
    - standardizing casing and the position of terminals
      - examples: AA, AAA or LR6, LR3
    - technology applied to produce a cell determines how much electrical charge one can „package” in a given volume

Sources of energy: battery cells

- Chemical features
  - cells can be classified based on chemical compounds used to make them
    - zinc-carbon, nickel-zinc, zinc-chloride
    - lithium, lithium-ion, lithium-polymer
    - alkaline
  - ecology
    - being harmful to the environment
      - e.g., cadmium-free are less harmful
    - cost of utilization
      - low battery capacity means that you need to replace the battery more often, and that increases the negative impact on the environment
      - some technologies make it easier to utilize (consider the toxicity of lead used to manufacture lead-acid accumulators)
  - On the Earth, the elements needed to build cells (eg Lithium) are become running out

Sources of energy

- Classification by way of using
  - primary cells (non-rechargeable)
    - charged at production time
    - problems with utilization
  - elements that can store energy produced during „runtime”, e.g., supercapacitors
    - often used in energy harvesting systems, like solar cells, thermal energy sources, wind turbines, etc.
  - secondary cells (rechargeable)
    - can be charged multiple times

Sources of energy: accumulators

- Accumulators can be charged multiple times
  - in the process of charging, electrical energy is converted to chemical energy
  - in discharging, chemical energy is converted to electrical energy
  - charging/discharging cycle is possible multiple times, which makes accumulators different from primary cells
  - process of charging is designed so as to be easy under virtually any conditions
    - charging an accumulator should take less time than using it
    - currently, the replacement of electrolyte or electrodes is considered difficult, and, in new designs, the user does need to do these things

Sources of energy: accumulators

- Accumulator technologies
  - Lead-acid (1,2V)
    - electrodes made of lead (Pb), electrolyte: sulfuric acid ($H_2 SO_4$)
    - currently, electrolyte is mostly gel (used, e.g., in UPS's)
      - no dangerous liquid electrolyte
  - NiCd (1,2V) [nickel oxide hydroxide]
    - electrodes made of nickel NiO(OH) and cadmium (Cd)
  - NiMh (1,2V) [nickel metal hydride]
    - very similar to those made of NiCd, but exhibit less significant memory effect, and are more environment-friendly
  - Li-Ion (3,6V) [lithium-ion]
    - positive electrode made of carbon, negative electrode made of metal oxide
    - electrolyte: lithium salt dissolved in organic solvent
  - Li-Poly (3,6V) [lithium-ion polymer]
    - lithium-polymer, similar to Li-Ion, electrolyte: organic polymer (Polyacrylonitrile)

Sources of energy: accumulators

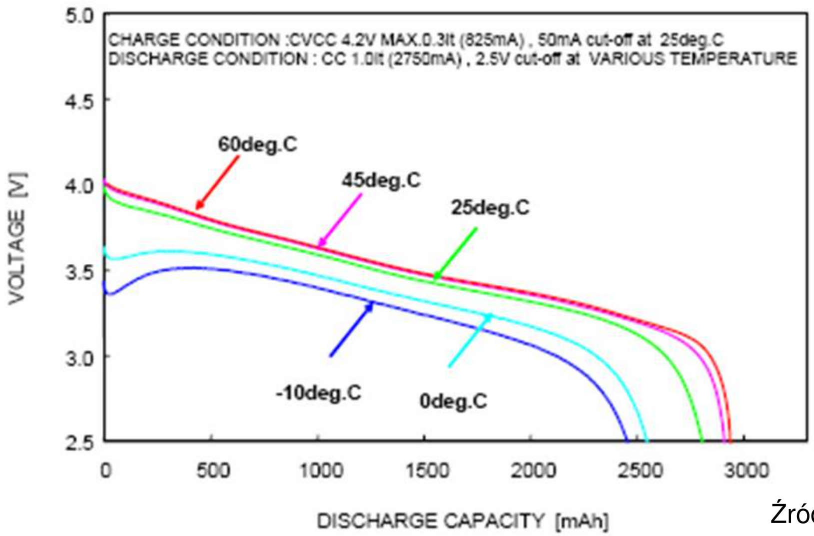
- What can go wrong during usage
  - mechanical damage
    - may be due to incorrect usage (e.g., a wrong position),
    - design may be faulty (e.g., electrolyte may evaporate via leaks)
  - memory effect
    - inability to fully charge (when you try charge an accumulator, say NiCd, that is not fully discharged)
  - capacity reduction
    - when you charge (and discharge) multiple times, you may not be able to charge to the nominal capacity
  - over-discharge
    - affects Li-Ion accumulators and similar ones; these may require charging and discharging controllers (e.g., cutting off the load at a high level of discharging)

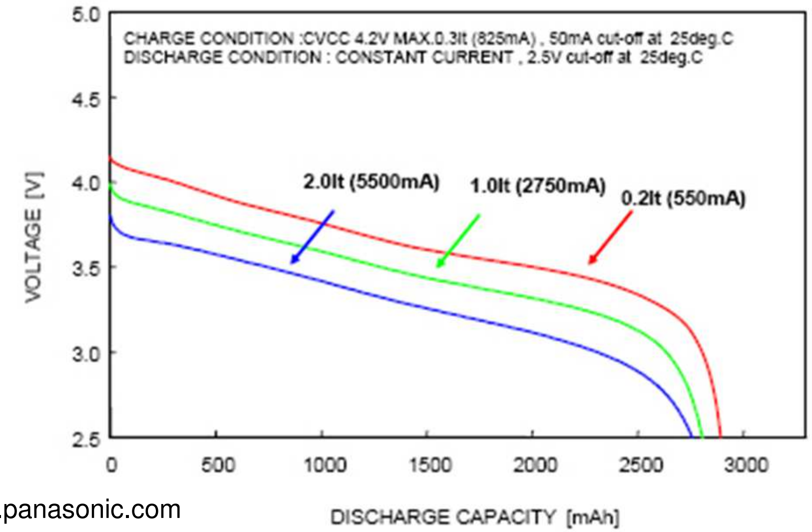
Sources of energy: accumulators

- Charging methods
  - regular: during period of 10h using a ten-hour current (1/10 of capacity)
  - fast ("intelligent"): charging with pulse current, with monitoring the accumulator's voltage, current, and temperature (!)
    - exceeding the temperature of the cell may cause self-ignition

Sources of energy: accumulators

- Accumulator's condition vs. device's working time
  - may try to estimate the working time based on the accumulator's voltage
    - need to know the accumulator's discharging characteristics
    - doesn't work well when discharging characteristics is flat (e.g., Li-Ion)
  - may try to estimate the working time based on the charge in the accumulator
    - better results, but requires continuous monitoring of charging and discharging currents
    - special purpose circuits supporting this
      - INA219 : monitors the current and its direction
      - DS2438: calculates the amount of the inflowing and outflowing charge





Źródło industrial.panasonic.com

Energy consumption levels in IoT nodes

Energy consumption levels in IoT nodes

- CPUs/MCUs: ATMega328p
  - active mode: CPU executes programs
    - entire Arduino UNO with 16MHz clock draws <49mA [www.peterbeard.co]
    - alone AVR ATmega328p with 8MHz clock draws <9mA
  - sleep modes
    - you go to a sleep mode by executing the assembly language instruction SLEEP
    - before that you need to select the sleep mode
      - SMCR (Sleep Mode Control Register) register: bits SM2/1/0 select the mode, bit SE makes it possible to go to a sleep mode
  - in C you may use avr-libc functions; include ”avr/sleep.h”

```
...
set_sleep_mode(SLEEP_MODE_POWER_DOWN);
sleep_mode();
...
```

Bit	7	6	5	4	3	2	1	0
					SM2	SM1	SM0	SE
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Źródło: www.atmel.com

Energy consumption levels in IoT nodes

- CPUs/MCUs: ATMega328p
  - available sleep modes
    - Idle (similar to „ADC Noise Reduction Mode”) [SMCR: 000 and 001]
      - CPU stopped
      - all peripherals and oscillators work as usual
      - leaving the mode (wake-up) is possible via RESET and interrupts (including Watchdog)
      - MCU draws: <2.7mA (5V, 8MHz)
    - Power-Down [SMCR: 010]
      - CPU, timers/counters and external oscillator stopped
      - interrupt detection circuits and TWI work as usual
      - leaving the mode is possible via RESET, interrupts (including Watchdog); FUSE bits (SUT i CKSEL) determine the start-up time (how long the level of a level triggered interrupt must be held)
      - MCU draws: ~15uA (3.3V), if WDT is not active: ~2uA (3.3V)

Energy consumption levels in IoT nodes

- CPUs/MCUs: ATMega328p
  - available sleep modes (cont.)
    - Power-Save [SMCR: 011]
      - just like Power-Down, except that Timer/Counter2 may be working, and it may wake up the MCU after some specified time
      - MCU draws ~0.9mA (3.3V, frequency for Timer/Counter2 TOSC =32786Hz)
    - Standby [SMCR: 110] and External Standby [SMCR: 111]
      - just like Power-Down, except that the external oscillator does work (to enable faster wake-up, which in this mode takes six clock periods only – but normally it consumes 0.5ms)

Energy consumption levels in IoT nodes

- Handling sleep modes in software
  - LowPower/Rocketscream library for Arduino (ATMega328p)
  - The LowPower library takes care of many operations related to limiting energy consumption
    - entering idle mode for 0.5 s., with all the peripherals stopped and off

```
#include "LowPower.h”
LowPower.idle(SLEEP_500MS, ADC_OFF, TIMER2_OFF, TIMER1_OFF,
              TIMER0_OFF, SPI_OFF, USART0_OFF, TWI_OFF);
```

    - entering the power down mode for 8 s. (limit for WDT), with ADC and BOD stopped

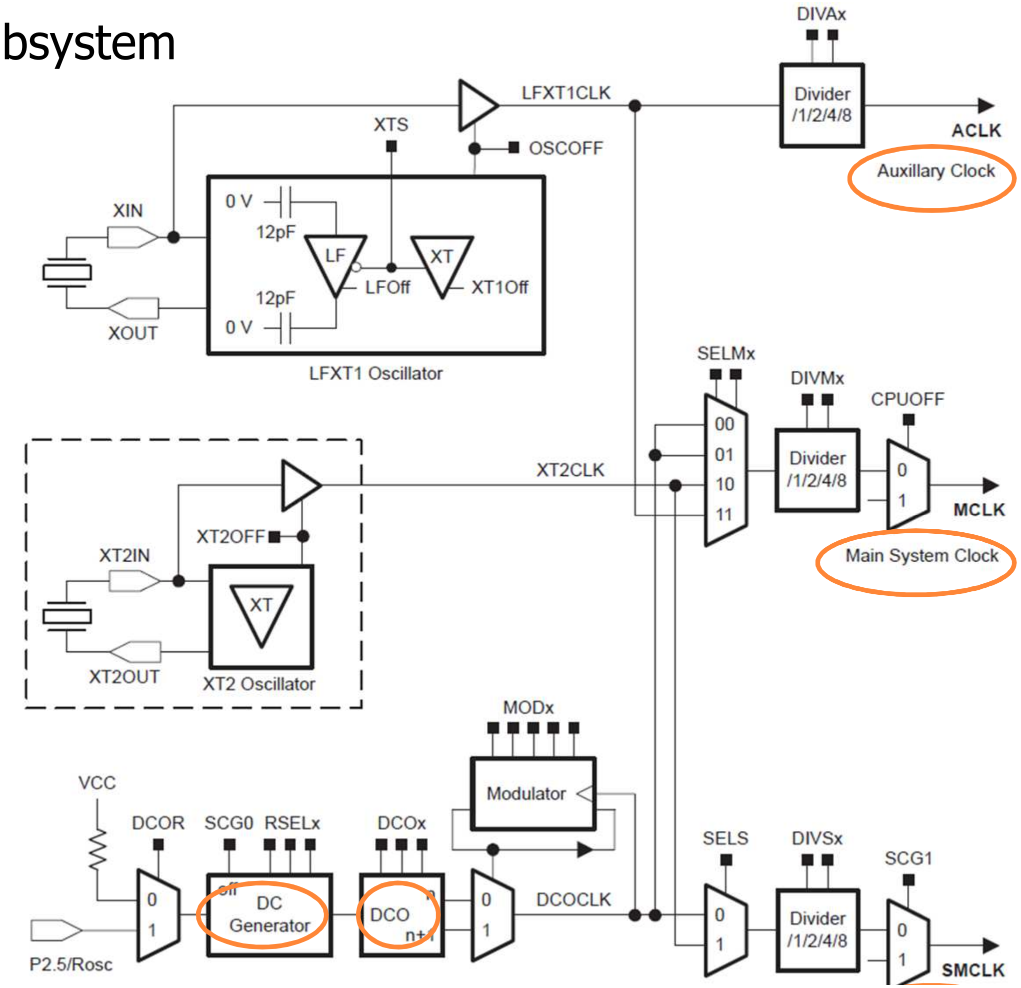
```
#include "LowPower.h”
LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
```

    - entering the power down mode (forever), but this time only RESET and INT0/1 interrupts can wake the MCU up

```
#include "LowPower.h”
LowPower.powerDown(SLEEP_FOREVER, ADC_OFF, BOD_OFF);
```

Energy consumption levels in IoT nodes

- CPUs/MCUs: MSP430
  - Includes an extensive clock subsystem
    - Main System Clock (MCLK)
    - Sub System Clock (SMCLK)
    - Auxiliary Clock (ACLK)
    - Digitally controlled oscillator (DCO)
    - DC Generator (an external clock source)



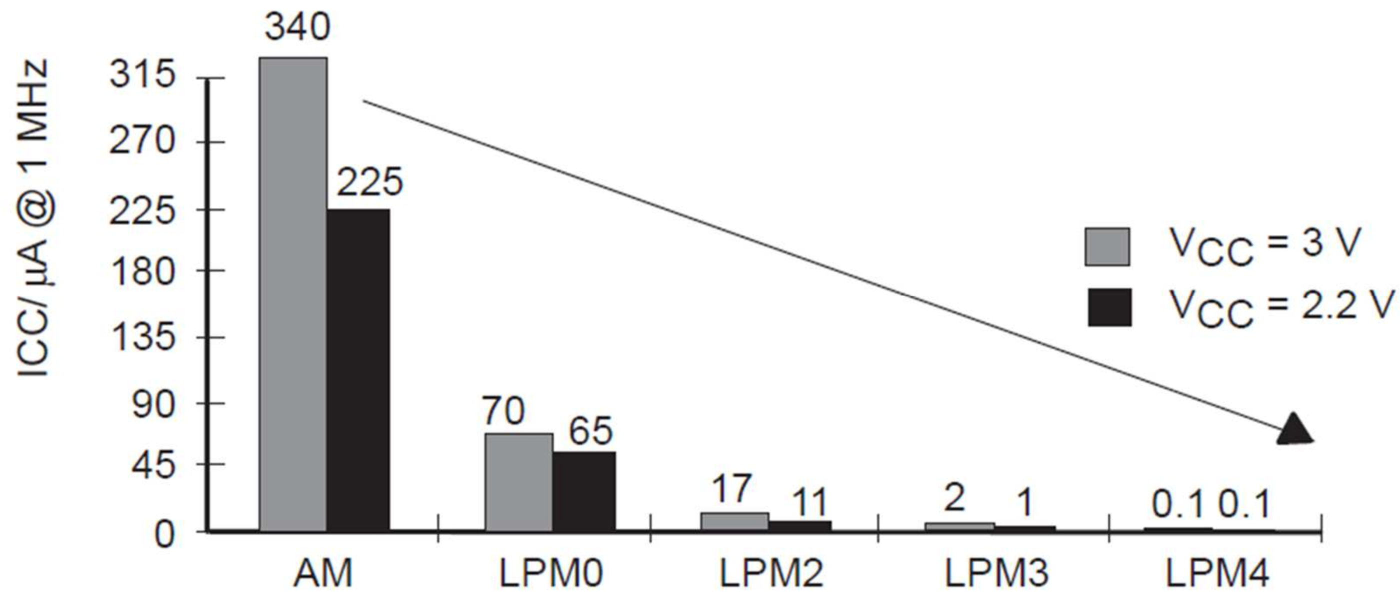
Źródło: www.ti.com

Energy consumption levels in IoT nodes

- CPUs/MCUs: MSP430
  - Active Mode
    - CPU and all the clocks active
  - LPM0
    - CPU, MCLK (Main System Clock) are disabled
    - SMCLK(Sub System Clock), ACLK(Auxiliary Clock) are active
  - LPM1
    - CPU, MCLK, DCO oscillator (Digitally controlled oscillator) are disabled, DC generator is disabled
    - SMCLK , ACLK are active
  - LPM2
    - CPU, MCLK, SMCLK, DCO oscillator are disabled
    - DC generator remains enabled, ACLK is active
  - LPM3
    - CPU, MCLK, SMCLK, DCO oscillator are disabled, DC generator is disabled
    - ACLK is active
  - LPM4
    - CPU and all clocks are disabled

Energy consumption levels in IoT nodes

- CPUs/MCUs: MSP430
  - Modes



  - Energy IDE (counterpart of Arduino IDE/API)
    - sleep()/sleepSeconds() causes the MCU to enter LPM3, reduces energy consumption from 3mW to 100uW
    - suspend() causes the MCU to enter LPM4, only an external interrupt can wake the device up
      - to successfully wake up the device you need to invoke wakeup() in the interrupt service routine

Źródło: www.ti.com

Energy consumption levels in IoT nodes

- Ethernet networking modules: **ENC28J60**
  - Power supply voltage: 3.1...3.6V
  - Power supply current
    - active
      - when transmitting : 180mA
      - when listening: 120mA
    - sleep mode: 2mA
- Ethernet networking modules: **W5100**
  - Power supply voltage: 3.1...3.6V
  - Power supply current
    - active
      - when transmitting: 183mA
      - in other modes: no data
  - Power supply current depends on the Ethernet version: is it 10Base-T or 100Base-T

Energy consumption levels in IoT nodes

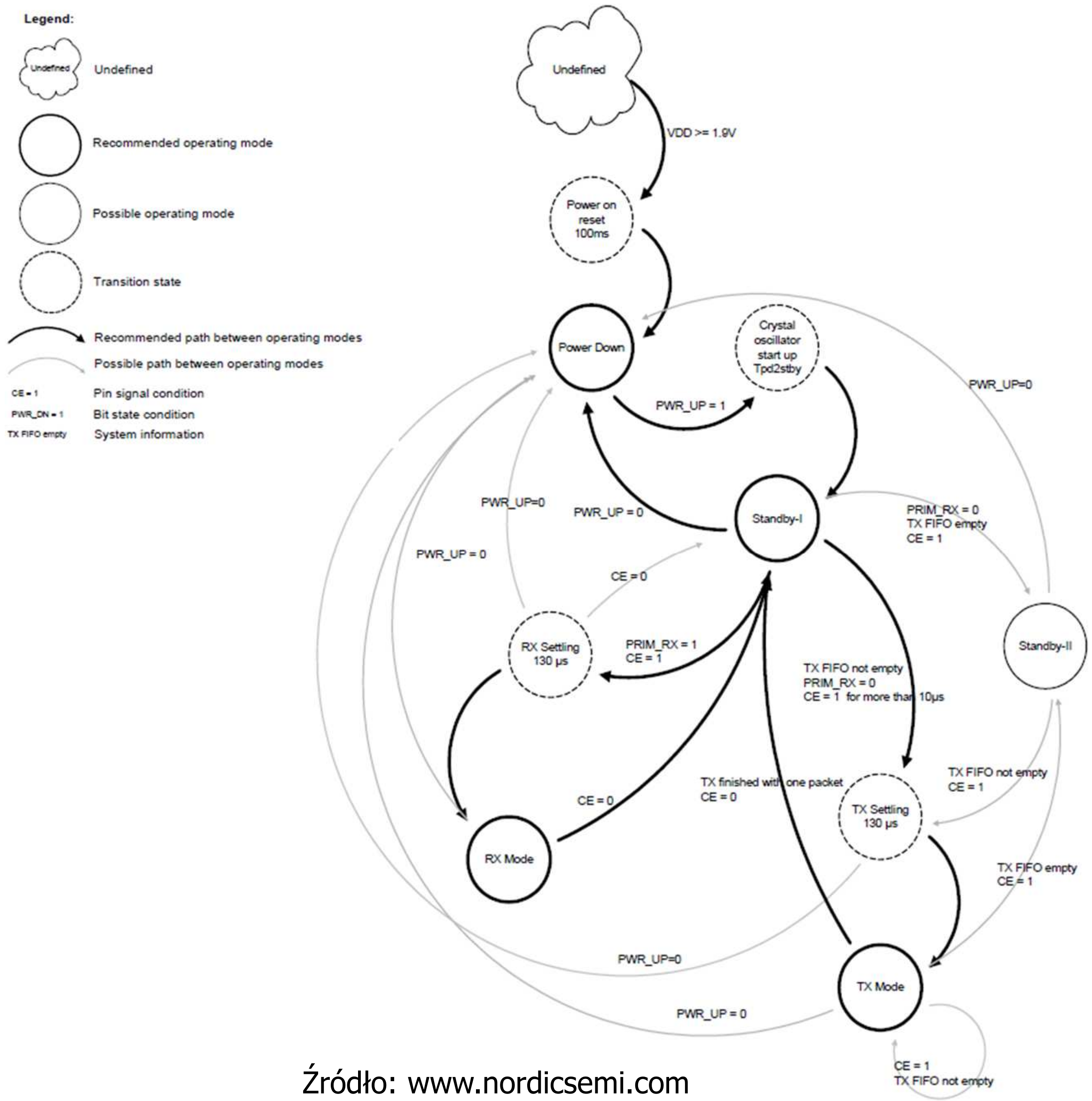
- Wireless transceivers: **CC2420 (802.15.4)**
  - Power supply voltage: 2.1...3.6V
    - can work with voltage as low as 1.6V with an external power supply (it features a built-in DC/DC converter, which needs to be powered)
  - Power supply current
    - active
      - receiving and listening: ~18.8mA
      - transmitting (depends on output power, which affects the range): from ~8.5mA(-25dBm) to ~17.4mA(0dBm)
    - low power modes
      - oscillator and voltage regulator off: ~0.02uA(!)
      - idle mode (ready to work, but without transmitting or listening): ~426uA

Energy consumption levels in IoT nodes

- Wireless transceivers: **nRF24L01+ (ISM, non standard)**
  - Power supply voltage: 1.9...3.6V
  - Power supply current
    - active mode
      - listening: 8.9mA (listening is required for services like acknowledgements)
      - receiving: 12.6mA (250Kbps)...13.5mA (2Mbps)
      - transmitting (depends on output power, which affects the range): from 7mA (-18dBm) to 11.3mA(0dBm)
    - deep sleep mode
      - can configure to consume only 900nA(!)
      - in standby modes (standby-I i standby-II): 26...320uA
      - while starting the oscillator: 400uA (can last up to 1.5ms)

Energy consumption levels in IoT nodes

- Wireless transceivers: **nRF24L01+**
  - State diagram



Źródło: www.nordicsemi.com

Energy consumption levels in IoT nodes

- Wireless transceivers: **RFM95 (LoRa)**
  - Power supply voltage: 1.8...3.7V
  - Power supply current
    - active mode
      - receiving and listening:from 10.8mA (LNA amplifier disabled) to 11.5mA (LNA amplifier enabled)
      - transmitting (depends on output power, which affects the range) : from 20mA (+7Bm) to 120mA (+20dBm)!
    - sleeping mode
      - 1.5uA (RC oscillator active)
      - 1.8mA (main oscillator active)
      - 5.8mA (synthesizer mode)

Energy consumption levels in IoT nodes

- Wireless transceivers: **CC3000 (Wifi)**
  - Power supply voltage: 3.6V
  - Power supply current
    - active mode
      - receiving and listening: <103mA
      - transmitting: (depends on output power, which affects the range): from 207mA (Po=14dBm, 54Mbps) to 275mA (Po=18dBm, 11Mbps)
    - sleeping mode: 5uA

How to reduce energy consumption IoT nodes

How to reduce energy consumption IoT nodes

- Major directions
  - lowering the clock frequency
    - at design time – choose the frequency that corresponds to your system's actual need for processing power
    - at runtime – adjust the frequency according to the processing power that the currently executing code requires
      - e.g., you may adjust the MCU clock frequency with a configurable PLL circuit
  - lowering the power supply voltage
    - lowering the voltage for entire sub-systems
      - the problem of compatibility with devices powered at different voltages
      - too little difference between the voltages for the logic levels „0" i „1" may lead to errors – noise may cause wrong interpretation of logic states
    - lowering the voltage for the CPU core only, without changing the voltages used to communicate with other modules (to preserve logic level compatibility)
      - a complex solution

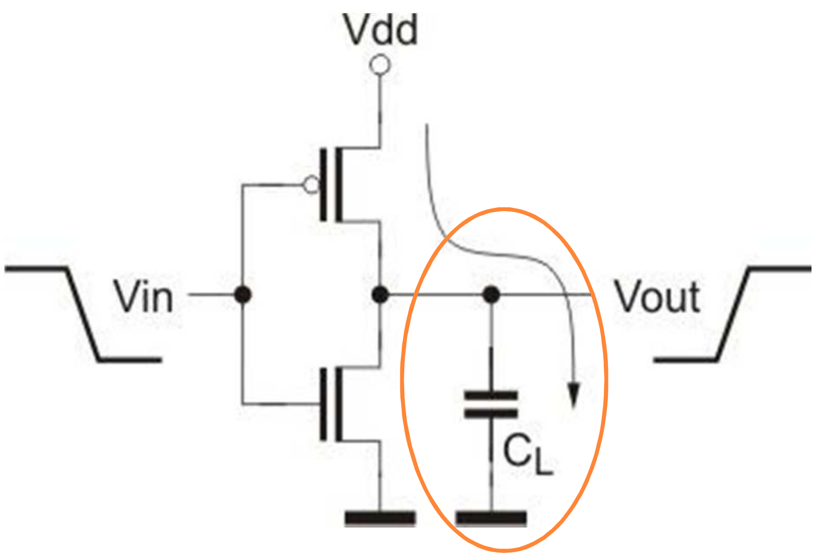
How to reduce energy consumption IoT nodes

- Selected problems exemplified by the CMOS technology
  - how is energy lost: steady states
    - leakage current
      - parasitic diodes, present in silicon, allow leakage current to flow
    - if the Gate-Source voltage gets close to the threshold voltage (V_t) sub-threshold current becomes dominant
      - the trend to reduce the size of transistors leads to a reduction of the threshold voltage

How to reduce energy consumption IoT nodes

- elected problems exemplified by the CMOS technology (cont.)
  - how is energy lost: dynamic phenomena
    - gate switching frequency
      - asymmetry of build the two transistors (P-MOS i N-MOS) output of the CMOS gate causes short-circuit current during transition between „0” and „1”, as well as „1” and „0”
      - losses due to charging the load capacitor C_L , with power supply voltage V_{dd} and frequency F_{CLK} (the load capacitance C_L could be the capacitance of the gate of the connected MOS transistor)

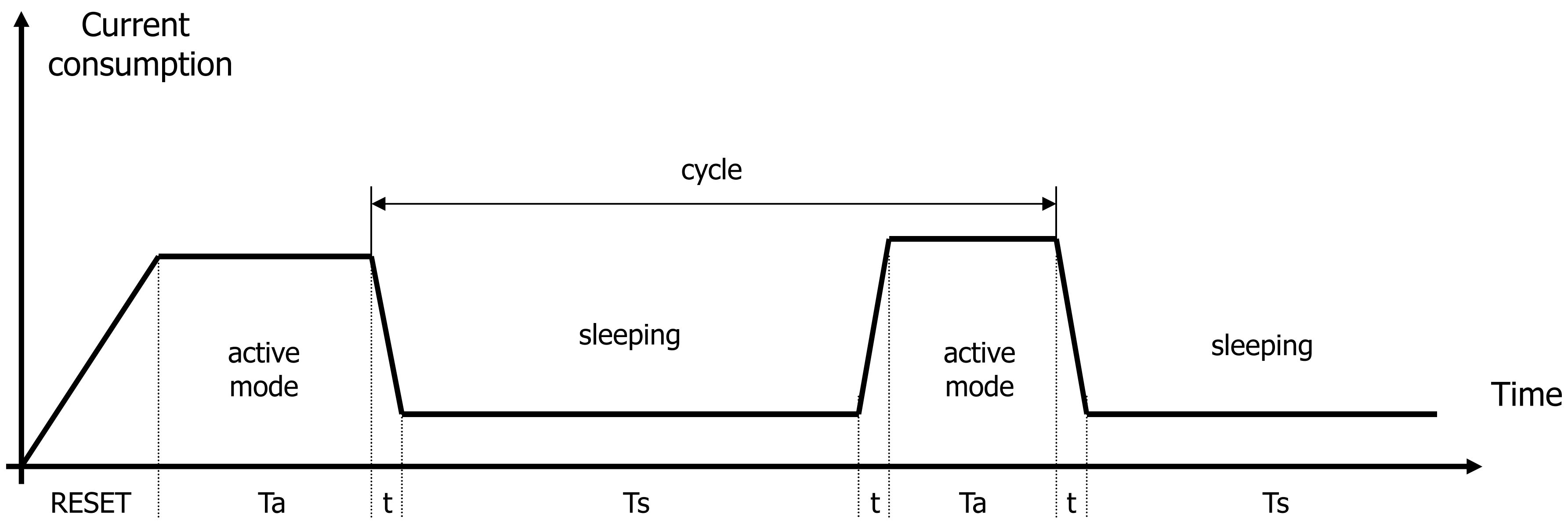
$$P = f(C_L * V_{dd}^2 * F_{CLK})$$



Źródło: elektronikab2b.pl

How to reduce energy consumption IoT nodes

- Method: periodic sleeping
  - the programmer makes the MCU/system periodically switch between the active mode and sleeping



How to reduce energy consumption IoT nodes

- Method: periodic sleeping (cont.)
  - power supply current for active mode and power-down* mode for selected CPUs/MCUs

	Active mode	Power-down
AT89C51	20mA	40..100uA
ATmega8L	6...15mA	30uA
ATmega128L	5..20mA	40uA
PIC18Lxxxx	0.03...50mA	1..2uA
MSP430F149	7...560uA	0.5...70uA
AT91RM9200	15..48mA	1.7mA

* not taking into account the impact of temperature, power supply voltage, clock frequency, and with WDT active (in CPUs/MCUs with built-in WDT)

How to reduce energy consumption IoT nodes

- Method: periodic sleeping (cont.)
  - power supply current for selected sensors and actuators*

	Active mode	Sleeping mode
Passive elements		
Thermistor (YSI 44006)	16.5...323uA	-
Photoresistor (Clairex CL94L)	0.3...275uA	-
Active elements		
Intelligent photodiode (TSL2550)	0.35mA	10uA
Humidity meter (SHT11)	0.55mA	0.3uA
Acceleration meter (ADXL202)	0.6mA	-
Pressure meter (MS5534A)	1mA	2.5uA
GPS (Leadtek9546)	60mA	-

* These are rough data, and they do not include the power consumption of the CPU/MCU controlling the sensors/actuators.

How to reduce energy consumption IoT nodes

- How long can a periodically sleeping system work
  - Notation:
    - T_a – time when MCU is in active mode [s]
    - I_a – current consumed while in active mode [A]
    - T_s – time when MCU is in deep sleep mode [s]
    - I_s – current consumed while in deep sleep mode [A]
    - Ratio (Duty cycle*) $[{}^{1}/{}_{1}]$

$$N = T_s / T_a$$

    - Battery capacity [Ah]

$$Q = I * T$$

*) - in papers that is related function

How to reduce energy consumption IoT nodes

- How long can a periodically sleeping system work (cont.)
  - Below, the time spent in-between states is neglected

$$Q = I_a * T_a + I_s * T_s$$

  - so

$$Q = I_a * T/(1+N) + I_s * T * N/(1+N)$$

  - which yields

$$Q = T * (I_a + I_s * N)/(1+N)$$

  - ultimately, the working time is given by

$$T = Q * \frac{1+N}{(I_a + I_s * N)}$$

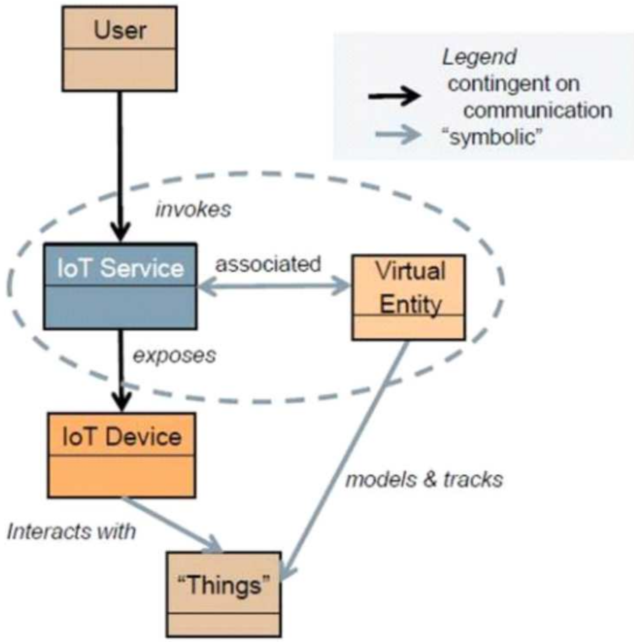
How to reduce energy consumption IoT nodes

- How long can a periodically sleeping system work (cont.)
  - estimated MCU working time with selected batteries (unit: 24h)
    - battery capacities taken from respective data sheets
    - remember about self-discharge of batteries(!)

	CR2032 (225mAh)				LR3-Alkaline (1.25Ah)				LR6-NiMh (2.5Ah)			
N	0	100	1K	10k	0	100	1K	10k	0	100	1K	10k
AT89C51	**0.5**	**31**	**78**	**92**	**2.6**	**175**	**434**	**510**	**5.2**	**350**	**868**	**1021**
ATMega8L	**0.6**	**52**	**208**	**297**	**3.5**	**292**	**1158**	**1653**	**7**	**584**	**2317**	**3307**
ATMega128L	**0.4**	**39**	**156**	**223**	**2.6**	**219**	**868**	**1240**	**5.2**	**438**	**1737**	**2480**
PIC18Lxxxx	**0.2**	**18**	**180**	**1400**	**1.0**	**104**	**1002**	**7441**	**2.0**	**210**	**2005**	**14882**
MSP430F149	**16.7**	**125**	**133**	**133**	**93**	**696**	**738**	**743**	**186**	**1392**	**1478**	**1487**
AT91RM9200*	**0.6**	**5.1**	**5.4**	**5.5**	**3.5**	**28**	**30**	**31**	**7**	**56.8**	**60.8**	**61.2**

* all built-in peripherals disabled

Thank you!


User
Legend
contingent on communication
"symbolic"
invokes
IoT Service
associated
Virtual Entity
exposes
IoT Device
models & tracks
Interacts with
"Things"