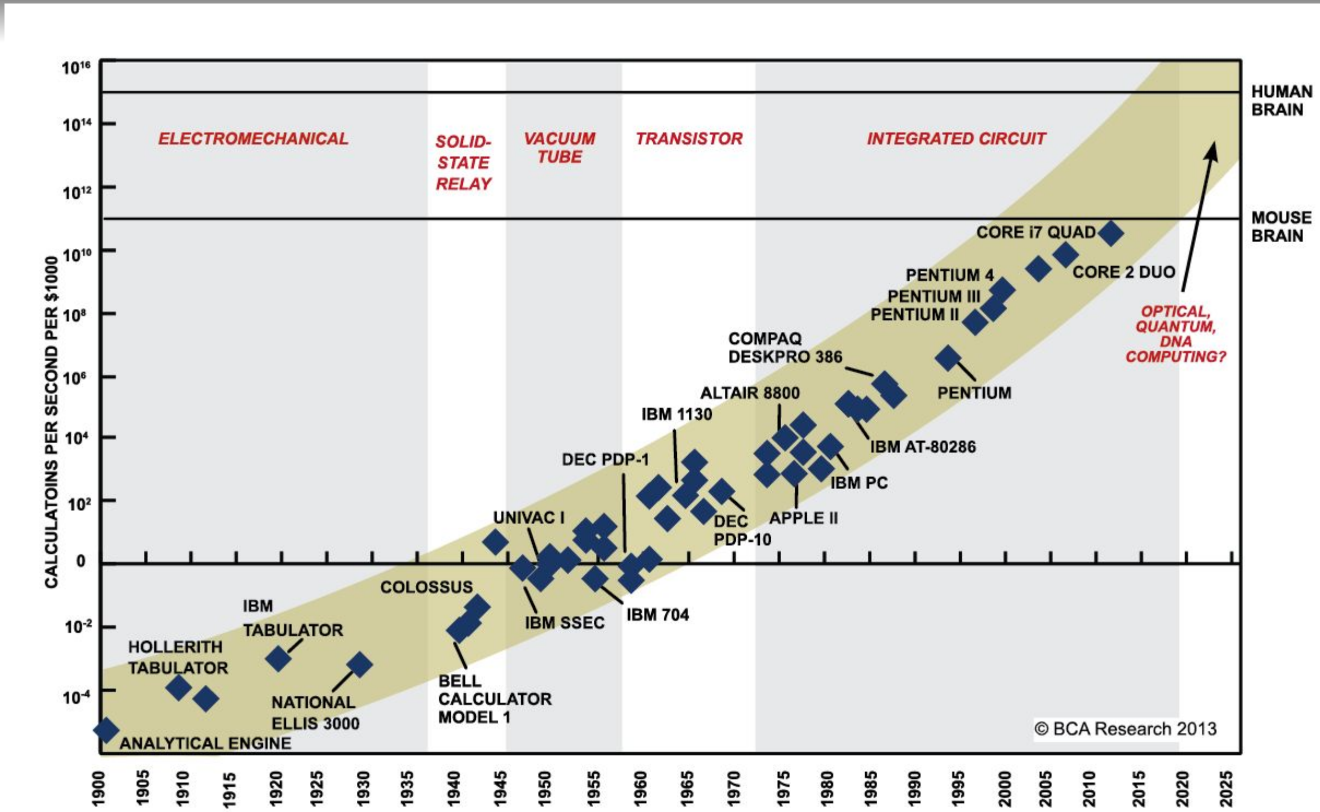
Institute for the Wireless Internet of Things at Northeastern University

EECE 5155
Wireless Sensor Networks (and the Internet of Things)

Prof. Francesco Restuccia
Email: f.restuccia@northeastern.edu

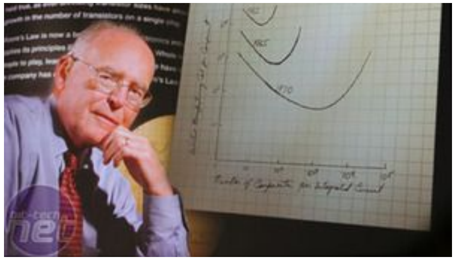
What is driving the growth of the IoT, from a **technological** perspective?

Moore’s Law: IC transistor count doubles every 18-24 mo



SOURCE: RAY KURZWEIL, "THE SINGULARITY IS NEAR: WHEN HUMANS TRANSCEND BIOLOGY", P.67, *THE VIKING PRESS*, 2006. DATAPOINTS BETWEEN 2000 AND 2012 REPRESENT BCA ESTIMATES.

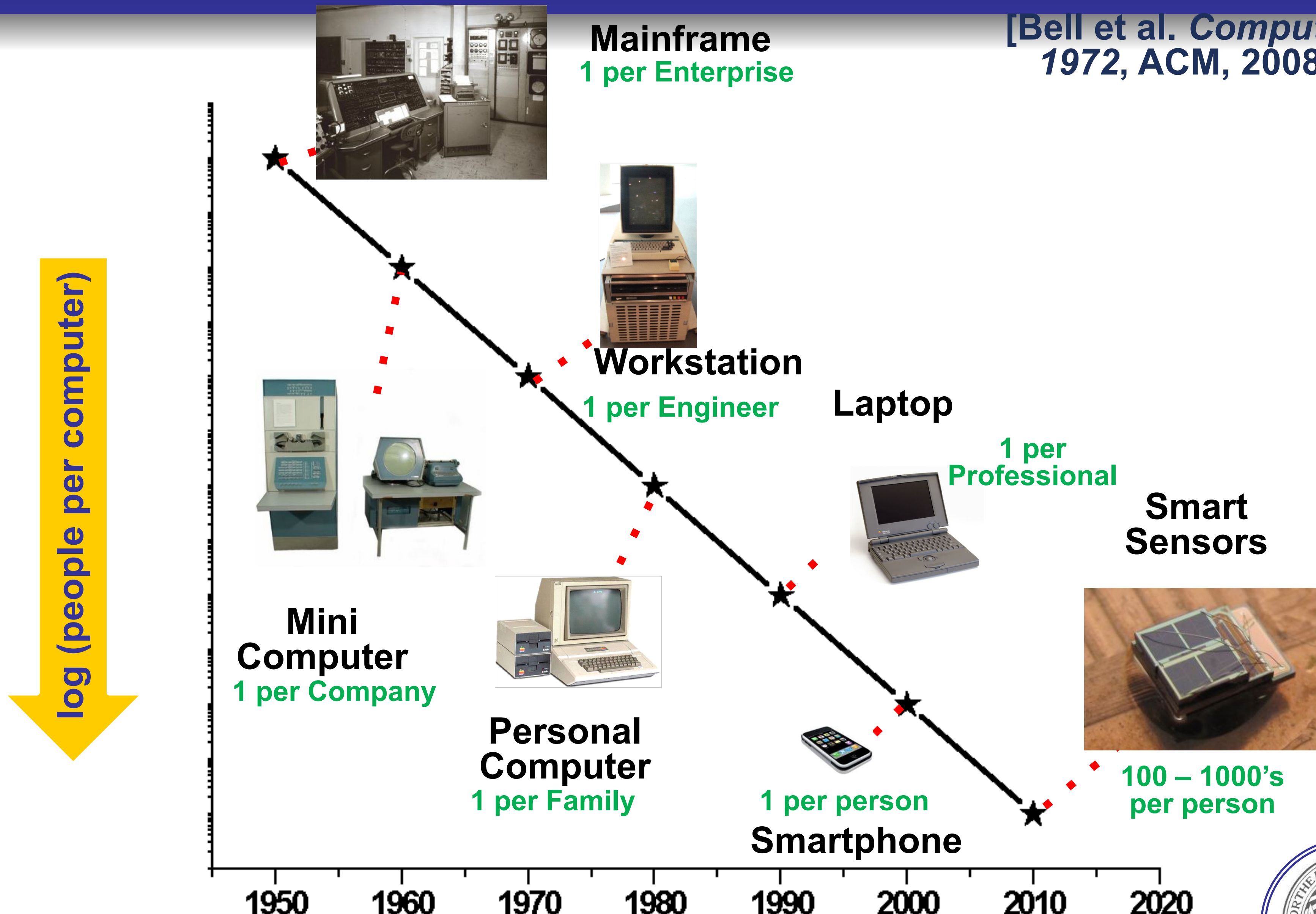
Photo Credit: Intel



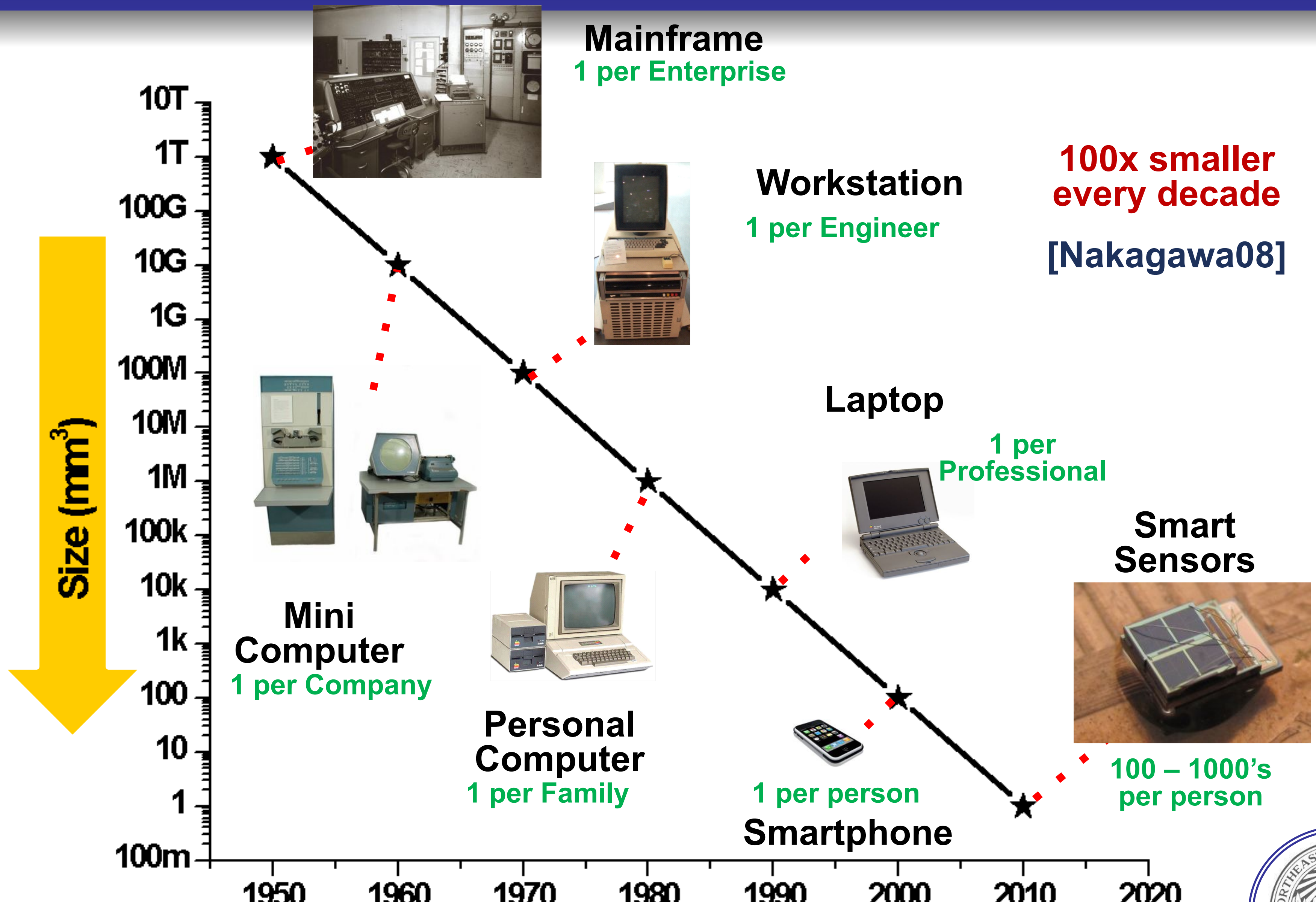
What does Moore’s Law Imply?

1 - Number of computers per person grows

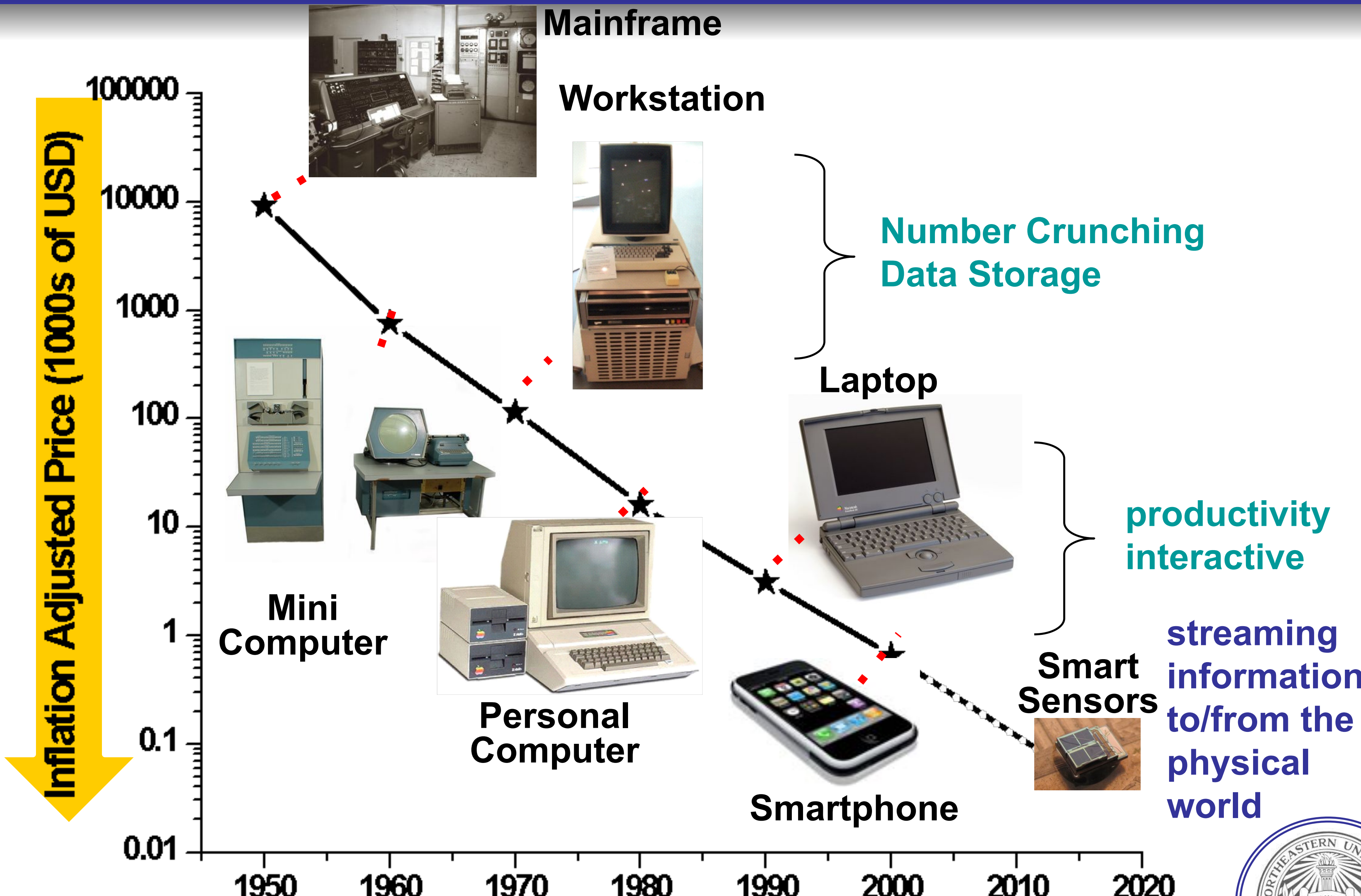
[Bell et al. *Computer, 1972*, ACM, 2008]



2 - Computer volume shrinks by 100x every decade



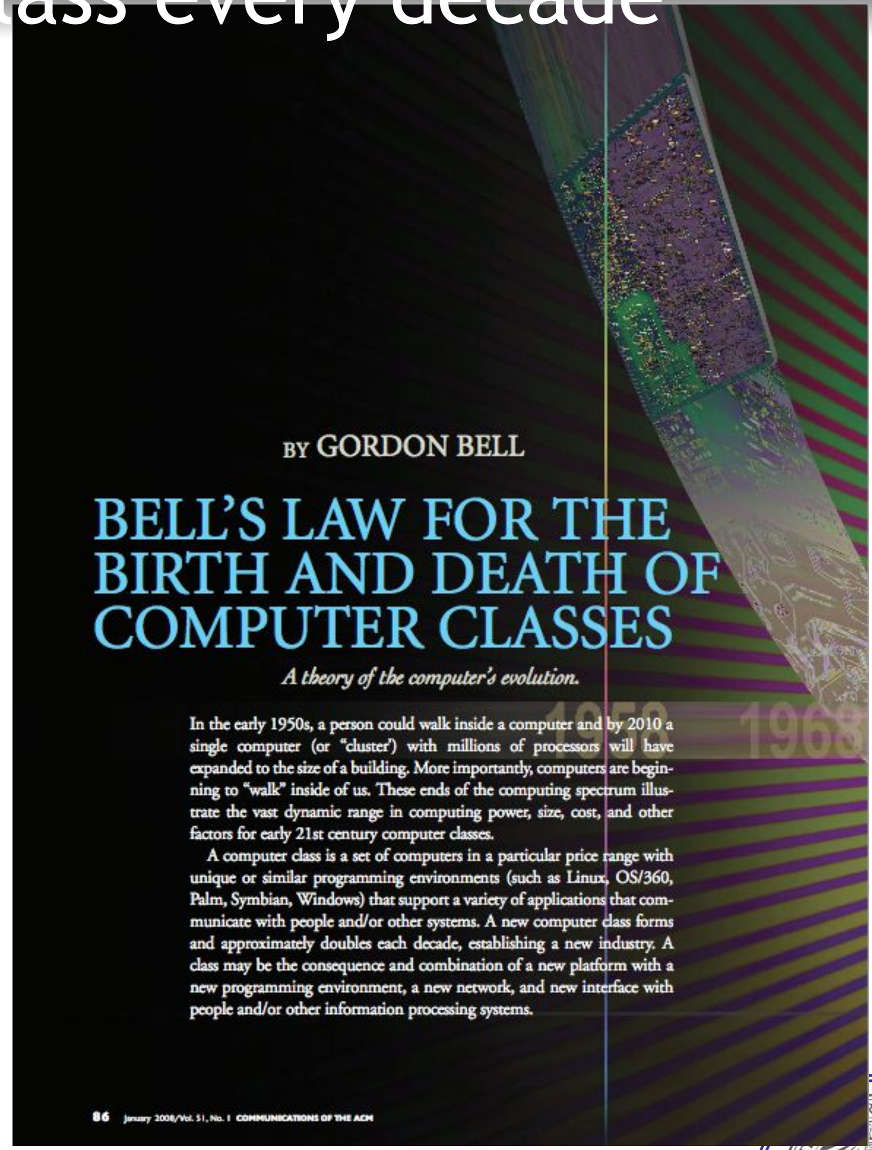
3 - Prices fall dramatically



Bell's Law:
A new computer class every decade

"Roughly every decade a new, lower priced computer class forms based on a new programming platform, network, and interface resulting in new usage and the establishment of a new industry."

- Gordon Bell [1972,2008]



BY GORDON BELL

BELL'S LAW FOR THE BIRTH AND DEATH OF COMPUTER CLASSES

A theory of the computer's evolution.

In the early 1950s, a person could walk inside a computer and by 2010 a single computer (or "cluster") with millions of processors will have expanded to the size of a building. More importantly, computers are beginning to "walk" inside of us. These ends of the computing spectrum illustrate the vast dynamic range in computing power, size, cost, and other factors for early 21st century computer classes.

A computer class is a set of computers in a particular price range with unique or similar programming environments (such as Linux, OS/360, Palm, Symbian, Windows) that support a variety of applications that communicate with people and/or other systems. A new computer class forms and approximately doubles each decade, establishing a new industry. A class may be the consequence and combination of a new platform with a new programming environment, a new network, and new interface with people and/or other information processing systems.

86 January 2008/Vol. 51, No. 1 COMMUNICATIONS OF THE ACM

What is driving Bell's Law?

- <u>Technology Scaling</u>
  - Moore's Law
    - Made transistors <u>cheap</u>
  - Dennard's Scaling
    - Made them <u>fast</u>
    - And <u>low-power</u>
  - Result
    - Holding #T's constant
      - Exponentially lower cost
      - Exponentially lower power
    - Small, cheap & low-power
      - Microcontrollers
      - Memory
      - Radios

- <u>Technology Innovations</u>
  - MEMS technology
    - Micro-fabricated sensors
  - New memories
    - New cell structures (11T)
    - New tech (FeRAM, FinFET)
  - Energy harvesting

Takeaways:

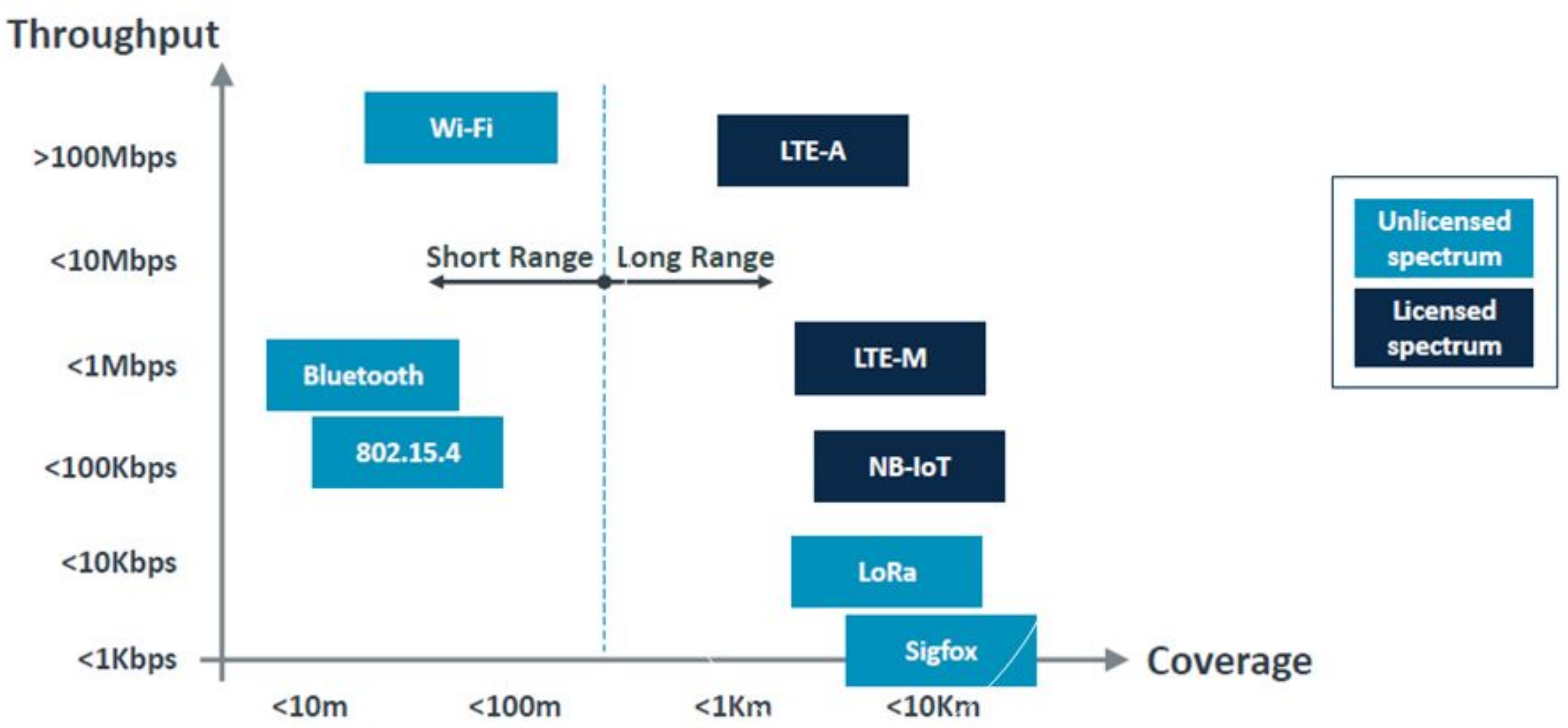
IoT devices will be everywhere

IoT is a very exciting market

LOTS of opportunities!

Technologies for IoT comms

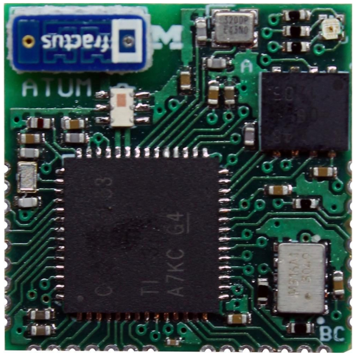
Different radio technologies



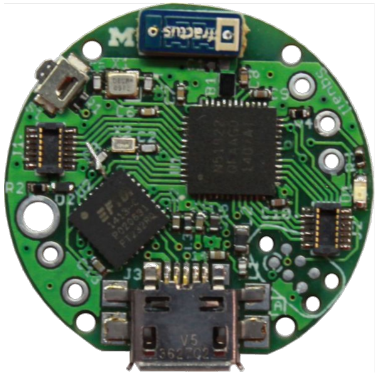
Source https://www.advantech.eu/resources/featured-article/de95b961-268f-4535-9943-ffa597b80671

Established Comms Interfaces

- IEEE 802.15.4 (a.k.a. "ZigBee" stack)
  - *"The"* technology for sensor networks!
  - Widely adopted for low-power mesh protocols
  - Middle (6LoWPAN, RPL) and upper layers
  - Can last for **years** on a pair of AA batteries



- Bluetooth Low-Energy (BLE)
  - Short-range RF technology
  - On phones and peripherals
  - Can beacon for years

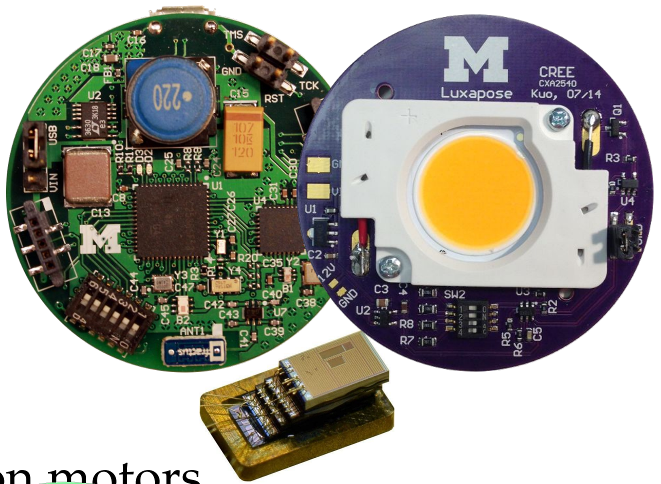


- Near-Field Communications (NFC)
  - Backscatter technology
  - Small (mobile) readers in smartphones
  - Large (stationary) readers in infrastructure

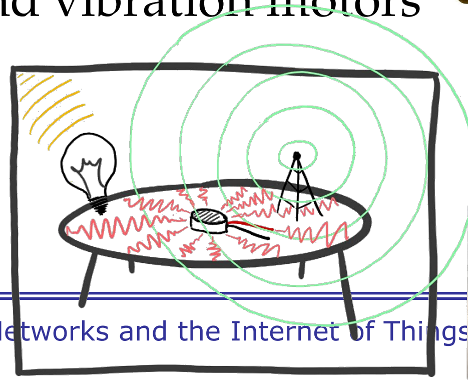
Emerging Comms Interfaces

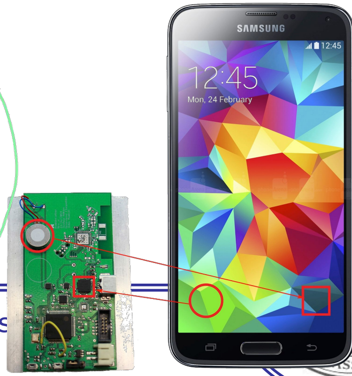
➢ Ultrasonic
  – Small, low-power, short-range devices
  – Supports very low-power wakeup
  – Can support pairwise ranging (i.e., distance) of nodes

➢ Visible Light
  – Enabled by pervasive LEDs and cameras
  – Supports indoor localization and comms
  – Leverages existing LED lighting



➢ Vibration
  – Leverages pervasive accelerometers and vibration motors





What About CPUs?

Types of Processing Unit

- Microcontroller
  - General purpose processor
  - Optimized for embedded applications
  - Flexible, can be programmed
  - Low power consumption – reduce further by going into sleep states where only parts of the controller are active

- Digital Signal Processors (DSP)
  - Instruction set and architecture optimized for **signal processing** tasks

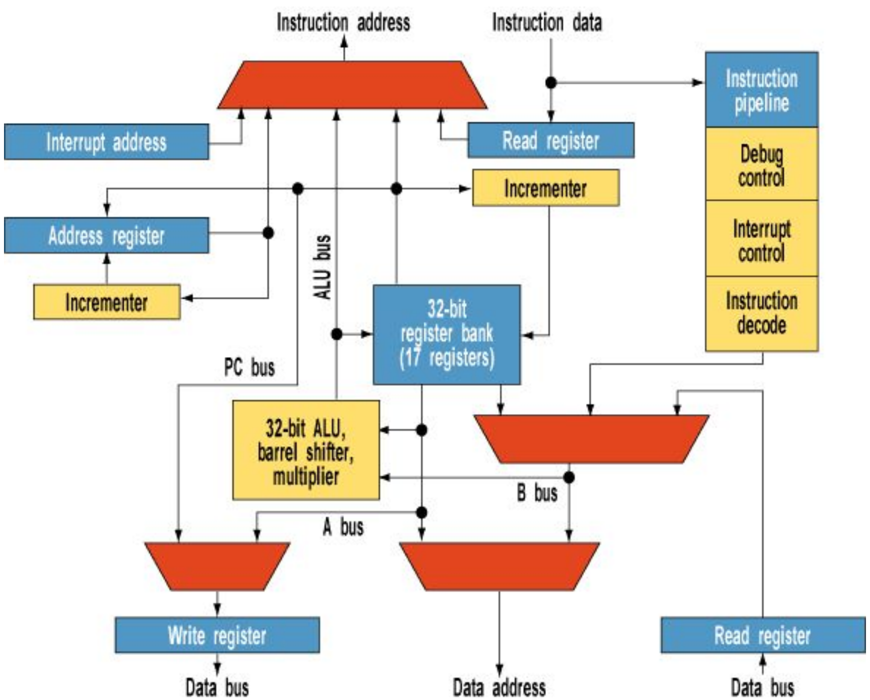
- Application Specific IC (ASIC)
  - **Not** re-programmable, hence **not flexible**
  - Only when peak performance is needed, no flexibility

Types of Processing Unit (2)

- Field-Programmable Gate Arrays (FPGA)
  - Integrated circuit designed to be configured by a customer or designer after manufacturing (field programmable)
  - FPGA configuration specified by hardware description language (HDL)
  - Array of programmable logic blocks and reconfigurable interconnects

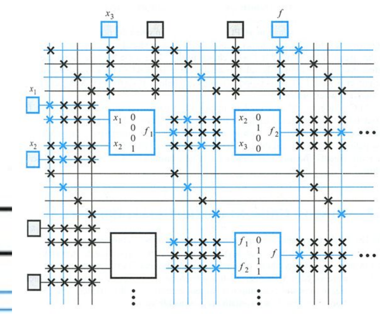
Microprocessor vs FPGA

MPU

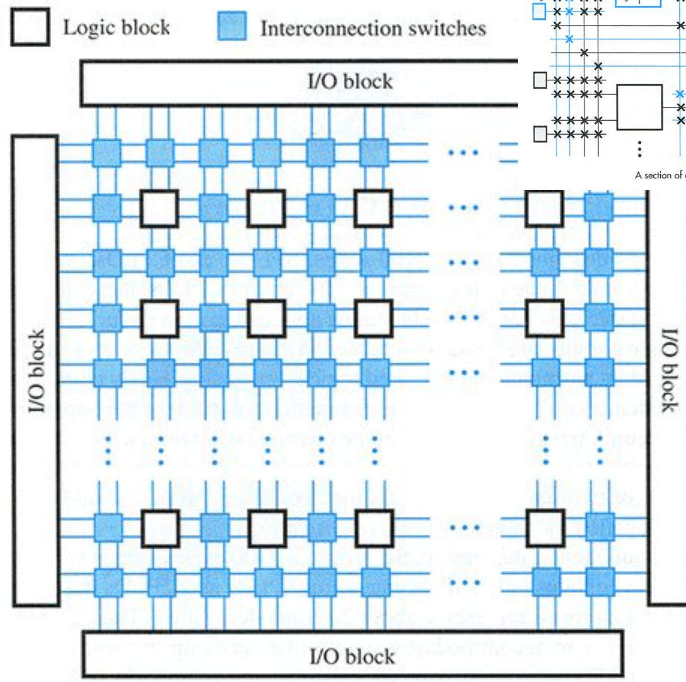


The Cortex M3's Thumbnail architecture looks like a conventional Arm processor. The differences are found in the Harvard architecture and the instruction decode that handles only Thumb and Thumb 2 instructions.

FPGA



A section of a programmed FPGA



General structure of an FPGA

Flexibility vs. Performance!

Example of Microcontrollers

- Many architectures based on microcontrollers

- Example microcontrollers (first generation sensors)
  - Texas Instruments MSP430
    - 16-bit RISC core, up to 4 MHz, versions with 2-10 kbytes RAM, several DACs, RT clock
  - Atmel ATMega
    - 8-bit controller, larger memory than MSP430, slower
  - Intel PXA255 Processor
    - 32 bit, 400MHz, Low Power Consumption <500 mA

What About RF Transceivers?

RF Transceiver Characteristics

VERY IMPORTANT FOR NETWORK PERFORMANCE EVALUATION!!!

- Front-end
  - Interface: bit, byte, packet level?
  - Supported frequency range?
    - Typically, somewhere in 433 MHz – 2.4 GHz, Industrial Scientific Medical (ISM) band
  - Multiple channels?
  - Data rates?
  - Range?

- Energy characteristics - How long is the sensor going to last?
  - Power consumption to send/receive data?
  - Time and energy consumption to change between different states?
  - Transmission power control?
  - Power efficiency (which percentage of consumed power is radiated?)

Transceiver Characteristics

- Performance parameters
  - Modulation? (ASK, FSK, …?)
  - Noise figure? NF = SNR_I/SNR_O
  - Gain? (signal amplification)
  - Receiver sensitivity? (minimum S to achieve a given E_b/N_0)
  - Blocking performance (achieved BER in presence of frequency-offset interferer)
  - Out of band emissions
  - Carrier sensing & RSSI characteristics
  - Voltage range

Transceiver States

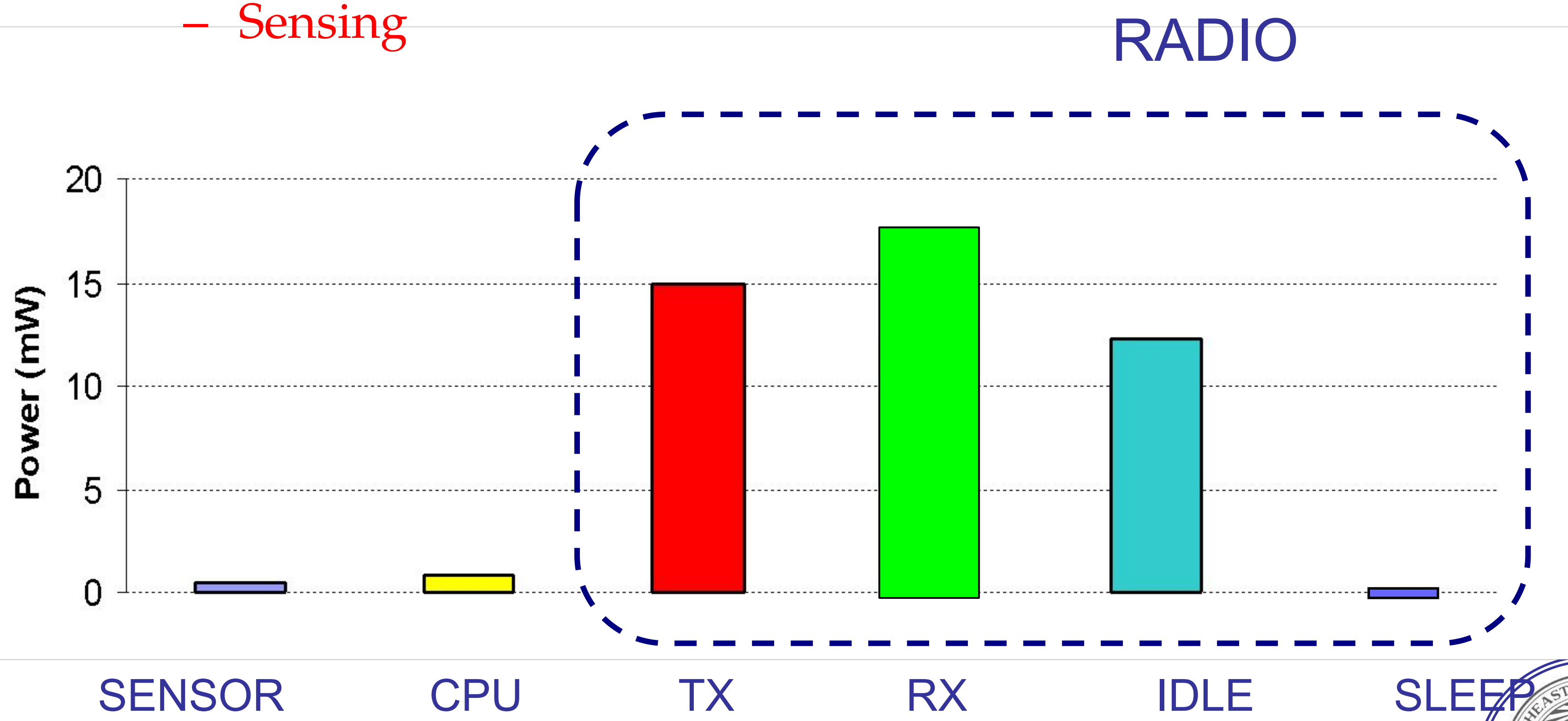
- Transceivers can be put into different operational ***states***, typically:
  - ***Transmit***
  - ***Receive***
  - ***Idle*** – ready to receive, but not doing so
    - Some functions in hardware can be switched off, reducing energy consumption a little
  - ***Sleep*** – significant parts of the transceiver are switched off
    - Not able to immediately receive something
    - ***Recovery time*** and ***startup energy*** to leave sleep state can be significant

- **Research issue:** Wake-up receivers – can be woken up via radio when in sleep state

Power Consumption

Power Consumption

- Power consumption in a sensor network can be divided into three domains
  - Communication
  - Data Processing (Computation)
  - Sensing



Power Consumption for Communication

- A sensor spends maximum energy in data communication (**transmission and reception**)
- For short range communication with low radiated power (~0 dbm), transmission and reception power costs **are approximately the same**
- Modern low power short range transceivers consume between **15 and 300 mW** of power when sending and receiving
- Transceiver circuitry has both active and start-up power consumption

Power Consumption

Power consumption for <u>data communication</u> (P_c)

$$P_c = P_{tx\_chain} + P_{rx\_chain} + P_{out\_tx}$$

Do you see any issues with this formula?

More Accurate Formula

$$P_c = N_T\,[P_{te}(T_{on} + T_{st}) + P_O(T_{on})] + N_R[P_{re}(R_{on} + R_{st})]$$

where

P_{te} is power consumed by transmitter
P_{re} is power consumed by receiver
P_O is output power of transmitter
T_{on} is transmitter "on" time
R_{on} is receiver "on" time
T_{st} is start-up time for transmitter
R_{st} is start-up time for receiver

N_T is the number of times transmitter is switched "on" per unit of time

N_R is the number of times receiver is switched "on" per unit of time

(E. Shih et al.,"Physical Layer Driven Protocols and Algorithm Design for Energy-Efficient Wireless Sensor Networks", ACM MobiCom, Rome, July 2001)

Power Consumption for Communications

- $T_{on} = L / R$

  where **L** is the packet size in bits and **R** is the data rate.
- N_T and N_R depend on MAC and applications

- REMARK: P_{re} is often **2-3 times higher** than P_{te} -> WHY?

  Typical P_{re} value ~ 180 mW; P_{te} ~ 81 mW.

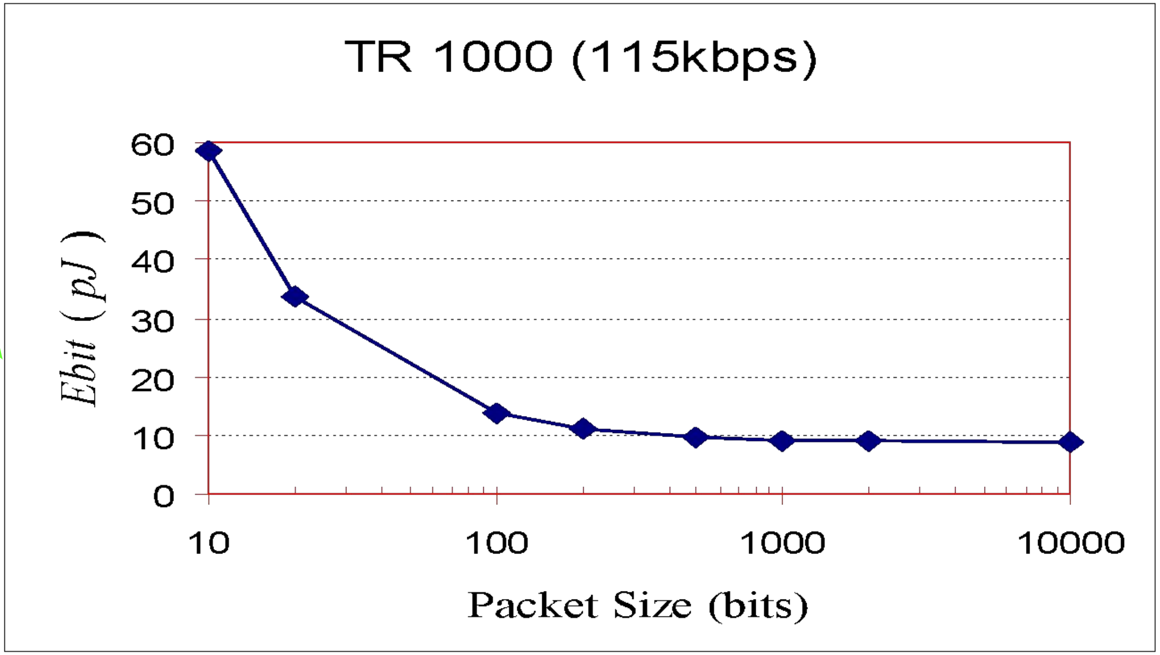
- Low power radio transceiver has typical P_{te} and P_{re} values around 20 dBm and P_O close to 0 dBm.

Start-up Power and Start-up Time

- A transceiver spends time upon waking up from sleep mode, e.g., to ramp up phase locked loops or voltage controlled oscillators
- During start-up time, **no transmission or reception of data is possible**
- Sensors often communicate in short data packets
- Start-up power starts **dominating as packet size is reduced**
- It is inefficient to turn the transceiver ON and OFF because a large amount of power is spent in turning the transceiver back ON each time

Energy vs Packet Size

Energy per bit (pJ)



As packet size is reduced the energy consumption is dominated by the startup time on the order of hundreds of microseconds during which large amounts of energy are wasted

Start-up Time and Sleep Mode

- The effect of the transceiver startup time will greatly depend on the type of MAC protocol used

- To minimize power consumption, it is desirable to have the transceiver in a sleep mode as often as possible

- However, constantly turning on and off the transceiver also consumes energy to bring it to readiness for transmission or reception

Trade-off!

What about energy spent during computation?

Power Consumption in Computation

Wang/Chandrakarasan: Energy Efficient DSPs for Wireless Sensor Networks. IEEE Signal Proc. Magazine, July 2002.

The power consumption in data processing (P_p) is

$$P_P = N * C * V^2_{dd} + V_{dd}(I_O e^{V_{dd}/n*V_T})(N/f)$$

- where N is the number of clock cycles per task
  C is the aver. capacitance switched per cycle (C ~ 0.67nF);
  V_{dd} is the supply voltage
  V_T is the thermal voltage (n~21.26; Io ~ 1.196 mA)
  f is the switching frequency
- Second term represents **leakage current**
- In general, leakage currents account for about **10%** of the total energy dissipation
- Most processor energy models only consider switching energy, but in low duty cycles leakage energy can become large (up to **50%**)

Power Consumption in Computation (2)

f is the switching frequency, upper-bounded by:

$$f <= \frac{K(V_{dd} - V_{th})^{a}}{V_{dd}} \sim K(V_{dd} - c)$$

where a, K, c *and* V_{th} are processor dependent variables.
(e.g., K=239.28 Mhz/V and c=0.5)

REMARK: For a given processor the maximum performance *f* of the processor is determined by the power supply voltage V_{dd} and vice versa

NOTE: For minimal energy dissipation, a processor should operate at the lowest voltage for a given clock frequency

Some Energy Consumption Figures

- Microcontroller
  - TI MSP 430 (@ 1 MHz, 3V):
    - Fully operational 1.2 mW
    - Deepest sleep mode 0.3 µW – only woken up by external interrupts (not even timer is running any more)
  - Atmel ATMega
    - Operational mode: 15 mW active, 6 mW idle
    - Sleep mode: 75 µW

Memory power consumption

- Crucial part: **FLASH memory**
  - Power for RAM almost negligible

- **FLASH writing/erasing is expensive**
  - Example: FLASH on Mica motes
  - Reading: 1.1 nAh per byte
  - Writing: 83.3 nAh per byte

Computation vs. Communication Energy Cost

- Tradeoff?
  - Directly comparing computation/communication energy cost not easy
  - But: we need to put them into perspective!
  - **Energy ratio of "sending one bit" vs. "computing one instruction": Anything between 220 and 2900 in the literature**
  - To communicate (send & receive) one kilobyte = **computing three million instructions!**

Let's put everything together: A simple energy model

A Simple Energy Model

- A sensor node is operating in transmit and receive modes only, with the following assumptions:

- Energy to run circuitry for one bit:

  $E_{elec} = 50 \text{ nJ/bit}$

- Energy for radio transmission:

  $e_{amp} = 100 \text{ pJ/bit/m}^2$

- Energy for sending k bits over distance D

  $E_{Tx}(k,D) = E_{elec} * k + e_{amp} * k * D^2$

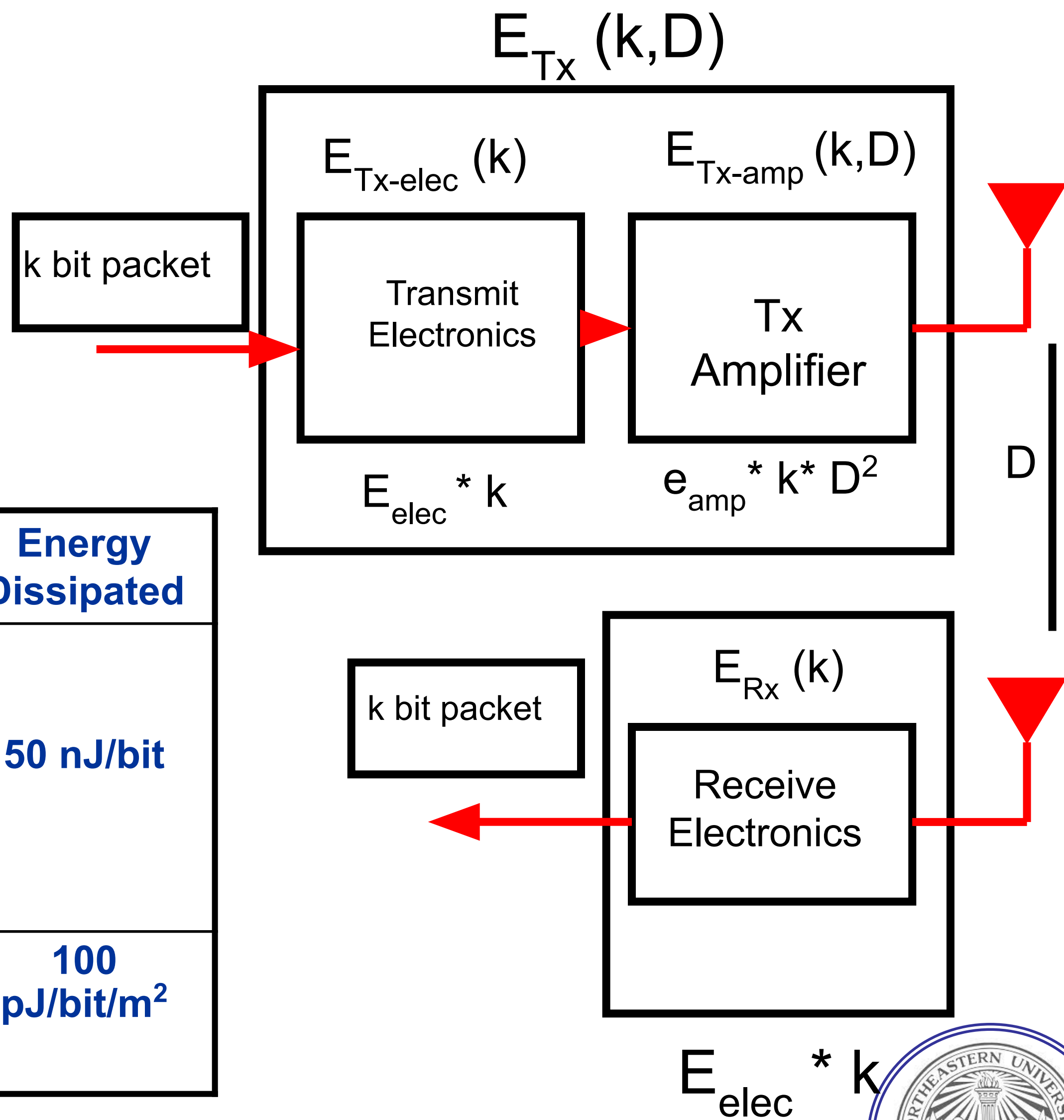
- Energy for receiving k bits:

  $E_{Rx}(k,D) = E_{elec} * k$

A Simple Energy Model

$E_{Tx}(k,D) = E_{Tx\text{-}elec}(k) + E_{Tx\text{-}amp}(k,D)$
$E_{Tx}(k,D) = E_{elec} * k + e_{amp} * k * D^2$

$E_{Rx}(k) = E_{Rx\text{-}elec}(k)$
$E_{Rx}(k) = E_{elec} * k$



Operation	**Energy Dissipated**
Transmitter Electronics ($E_{Tx\text{-}elec}$) **Receiver Electronics ($E_{Rx\text{-}elec}$)** **($E_{Tx\text{-}elec} = E_{Rx\text{-}elec} = E_{elec}$)**	**50 nJ/bit**
Transmit Amplifier {e_{amp}}	**100 pJ/bit/m^2**

Example Using the Simple Energy Model

What is the energy consumption if 1 Mbit of information is transferred from the source to the sink where the source and sink are separated by 100 meters and the broadcast radius of each node is 5 meters? Assume the neighbor nodes are overhearing each other's broadcast

100 meters / 5 meters = 20 pairs of transmitting and receiving nodes (one node transmits and one node receives)

$E_{Tx}(k,D) = E_{elec} * k + e_{amp} * k * D^2$

$E_{Tx} = 50\ \text{nJ/bit} \,.\, 10^6 + 100\ \text{pJ/bit/m}^2 \cdot 10^6 \cdot 5^2 =$
$= 0.5\text{J} + 0.0025\ \text{J} = 0.0525\ \text{J}$

$E_{Rx}(k,D) = E_{elec} * k$
$E_{Rx} = 0.05\ \text{J}$

$E_{pair} = E_{Tx} + E_{Rx} = 0.1025\text{J}$
$E_T = 20 \,.\, E_{pair} = 20.\ 0.1025\text{J} = 2.050\ \text{J}$

Comparison

Mote 

Bluetooth 

IEEE 802.11 

Energy per bit

Startup time

Idle current

Technology	Data Rate	Tx Current	Energy per bit	Idle Current	Startup time
Mote	76.8 Kbps	10 mA	430 nJ/bit	7 mA	Low
Bluetooth	1 Mbps	45 mA	149 nJ/bit	22 mA	Medium
802.11b	11 Mbps	300 mA	90 nJ/bit	160 mA	High

How can the model be improved?

Think-Share!

Questions?