Outline

• Power management in field instruments
  – Power consumption
  – Power generation
  – Power storage

• Sensor systems for environmental monitoring:
  – Motivation
  – The Sensorscope project
Power Management in Field Instruments
Motivation

• Energy is one of the key bottlenecks for autonomous/unattended operation of embedded systems
• Field instruments as extreme examples
  – Power consumption
  – Power generation
  – Power storage
Power Consumption
Power

\[ P = U \cdot I \]

Examples:

- MICAz: 2 * 1,5V battery, 25 mA power consumption → 2*1,5V*0,025A=80mW (standby: 80 \( \mu \)W)

- Campell Scientific: 3D ultrasonic anemometer: 1,2W or 2,4W

- SHT1x temperature and humidity sensor: 2\( \mu \)W – 3mW

2.4 Power Requirements

VOLTAGE SUPPLY: 10 to 16 Vdc

POWER:
- 2.4 W @ 60 Hz measurement frequency
- 1.2 W @ 20 Hz measurement frequency

Electrical and General Items

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Condition</th>
<th>min</th>
<th>typ</th>
<th>max</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Voltage</td>
<td></td>
<td>2.4</td>
<td>3.3</td>
<td>5.5</td>
<td>V</td>
</tr>
<tr>
<td>Power Consumption 5</td>
<td>sleep</td>
<td>2</td>
<td>5</td>
<td></td>
<td>( \mu )W</td>
</tr>
<tr>
<td></td>
<td>measuring</td>
<td>3</td>
<td></td>
<td></td>
<td>mW</td>
</tr>
<tr>
<td></td>
<td>average</td>
<td>150</td>
<td></td>
<td></td>
<td>( \mu )W</td>
</tr>
<tr>
<td>Communication</td>
<td>digital 2-wire interface, see Communication</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage</td>
<td></td>
<td>10 – 50°C (0 – 125°C peak), 20 – 60%RH</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Energy

\[ E = P \cdot t \]

Examples:

- Rechargeable battery (NiMH):
  
  \[
  1.2V \times 2000mAh = 2400mWh = 2.4Wh
  \]

- Rechargeable battery (LiPo):
  
  \[
  3.7V \times 1340mAh = 4958mWh = 4.958Wh
  \]

- Communication and computation energy:
  (remember 1 Ws=1 Joule)

<table>
<thead>
<tr>
<th></th>
<th>1999 (Bluetooth Technology)</th>
<th>2004</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
<td>(150nJ/bit)</td>
<td>(5nJ/bit)</td>
</tr>
<tr>
<td></td>
<td>1.5mW*</td>
<td>50uW</td>
</tr>
<tr>
<td>Computation</td>
<td>~ 190 MOPS</td>
<td>(5pJ/OP)</td>
</tr>
</tbody>
</table>
Managing Power Consumption

Consumption vs. capabilities (example 1):

- Disdrometer 1 ("tipping bucket") (≈ 0W)
  - no snow, sleet
  - no robust to freezing
  - no drop statistics
  - resolution
- Disdrometer 2 (laser) (≈ 10W)
  + snow, sleet
  + robust to freezing
  + drop statistics
  - expensive
  - delicate
- Disdrometer 3 (hot plate) (≈ 100W)
  + snow, sleet
  + robust freezing
  + simple
  - drop statistics
Managing Power Consumption

Consumption vs. capabilities (example 2):

- **Anemometer 1 (cup) (≈ 0W)**
  - 1D (main direction)
  - no robust to freezing
  - temporal resolution (0.3 Hz)
  - minimal wind speed high

- **Anemometer 2 (ultrasonic) (≈ 10W)**
  + 2D
  + *some* snow, sleet
  + *some* robustness to freezing
  + temporal resolution
    (up to 60Hz)
  - expensive

- **Anemometer 3 (Anemometer 2+heater) (≈ 100W)**
  + 2D
  + snow, sleet
  + robust to freezing
  + temporal resolution
  - expensive
Managing Power Consumption

Consumption vs. processing speed:

- $P \sim f_{\text{clock}}$
- Energy/operation = const

Consumption vs. transmission power:

- $P = f(P_{RF})$
- sometimes linear: $P \sim P_{RF}$
- often: “sweet spot”

Source: MSP430 data sheet
Sensor Node Energy Roadmap

- Deployed (5W)
- PAC/C Baseline (.5W)
- (50 mW)
- (1mW)

Rehosting to Low Power COTS (10x)
- System-On-Chip
- Adv Power Management Algorithms (50x)

Source: ISI & DARPA PAC/C Program
**Communication/Computation Technology Projection**

<table>
<thead>
<tr>
<th></th>
<th>1999 (Bluetooth Technology)</th>
<th>2004</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication</td>
<td>(150nJ/bit)</td>
<td>(5nJ/bit)</td>
</tr>
<tr>
<td></td>
<td>1.5mW*</td>
<td>50uW</td>
</tr>
<tr>
<td>Computation</td>
<td>~ 190 MOPS</td>
<td>(5pJ/OP)</td>
</tr>
</tbody>
</table>

Assume: 10kbit/sec. Radio, 10 m range.

**Large cost of communications relative to computation continues**

Source: ISI & DARPA PAC/C Program
Free Space Path Loss (Friis Law)

From Week 8 slides

- Signal power decay in air:
  \[ L = \left( \frac{4\pi df}{c} \right)^2 \]
  \[ L_{dB} = 20 \log_{10}(d) + 20 \log_{10}(f) - 147.56 \]

- Proportional to the square of the distance \( d \)
- Proportional to the square of the frequency \( f \)
  - high frequency = high loss
  - low frequency = low throughput
Ex.: DISAL Arduino Xbee kit vs. TinyNode

From Week 8 slides

DISAL Arduino Xbee kit
- Microcontroller:
  - ATMega 2560 (low-power)
- Transceiver:
  - Silicon Labs EM357 (part of the Xbee 802.15.4 module)
  - 2.4 GHz carrier
  - Throughput: up to 250 kbps
  - Range: up to 90 m

TinyNode (Shockfish)
- Microcontroller:
  - TI MSP430 (ultra-low power)
- Transceiver:
  - Semtech XE 1205
  - 868 and 915 MHz carriers
  - Throughput: up to 153 kbps
  - Range: up to 2 km

Sensorscope data logger
- Internal batteries
- GPRS module
- Memory card
- TinyNode module
Power Generation
Power generation methods

- Solar
- Wind
- Temperature difference (Seebeck Effect)
- Vibration
- Hydro
Solar Power Generation

1m²

1000 W/m²

Efficiency: 10-20%

165 W

Efficiency: 90%

135 W

Efficiency: 70%

0.1 m² per W (e.g., 32 x 32 cm²)

100 W/m²

85 W

Efficiency: 90%
Example: SensorScope Station

162x140mm solar panel
12Ah NiMH rechargeable battery

Note based on s. 17 estimation:
average consumption < 220 mW for battery level maintenance!
Compare with s. 31-33 power consumption
Power Storage
Batteries

- primary (non rechargeable, red)
  “often a good idea”
- secondary (rechargeable, blue)

Other important parameters:
- number of cycles (few 100 – few 1000) optimal conditions
- cold temperature behaviour

Interesting alternative: super capacitor
- 30 Wh/kg
- expensive
- very high number of cycles (> 100’000)
From a Single Instrument to a Network
Motivation for Sensor Networks

What if we could monitor events which …

– have a large *spatial* and *temporal* distribution
– require *in-situ* measurements
– take place in hard to access places
– generate data which need to be available in *real-time*
Motivation for Sensor Networks

What would we need for that?

A device which …

– is cheap – so we can distribute many of it
– is reliable – so we can measure for a long time
– uses little power – battery/solar cell powered
– has a radio – so it can communicate
– can potentially move – so it can potentially relocate
The Sensorscope Project
Introduction

Temperature
Humidity
Light
Topology
Topology
Topology
Topology

GPRS (General Packet Radio Services)
Topology

Pros
• Very simple!
• Essentially no restrictions in sensor locations

Cons
• The closest server access point may be quite far from the stations
• A long-range link may consume a lot of energy!
Mean Power consumption [mA over 3.6V]

- **0.05** Sensor (e.g., temperature)
- **0.3** MSP430 (microcontroller)
- **50** XE1205 (short range radio, < 2 km)
Mean Power consumption [mA over 3.6V]

- Sensor: 0.05
- MSP430: 0.3
- XE1205: 50
- GPRS: 700!

14x the XE1205
Topology

Assuming four AA batteries NiMH, i.e. 1.2 V, 2000 mAh (total energy 9.6 Wh) and only one of the components below always active:

- Sensor: 6 years
- MSP430: 1 year
- Short range radio: 2 days
- Long range radio: 4 hours
But makes sense on projects such as OpenSense (see W12 slides)!
Topology

Short range

Sink

GPRS
Topology

Recall Friis law again:

\[ L = \left( \frac{4\pi df}{c} \right)^2 \]

Example: To transmit over 5 Km, using a 868 MHz carrier, we can:

- One hop of 5 km: \( L = 106 \text{ dB} \)
- Two hops of 2.5 km: \( L = 99 \text{ dB} \)
- Five hops of 1 km: \( L = 92 \text{ dB} \)

Energy is the main issue !!!
Multi-hop Sensor Network

GPRS
Multi-hop Sensor Network

Pros
• Only one car battery in the network
• The sensor network has extended monitoring coverage
• Multiple routes for stations to communicate with the sink
• Auto configurable network (robustness)

Cons
• Significantly more complicated
• Data rate is not increased
• Unable to use directional antennas
Multi-hop WSNs

Implementation:

• Neighborhood discovery
• Data routing
• Time synchronization
• Duty-cycling (radio management)
Hello messages (Beacons) are one common method:
1. Node A sends a HELLO message to its neighbors (B, C, and D).
3. Node B sends a HELLO message to its neighbors (A, C, and D).
4. …
Neighborhood

Hello messages (Beacons) are one common method:
1. Node A sends a HELLO message to its neighbors (B, C, and D).
3. Node B sends a HELLO message to its neighbors (A, C, and D).
4. ...
Hello messages (Beacons) are one common method:
1. Node A sends a HELLO message to its neighbors (B, C, and D).
3. Node B sends a HELLO message to its neighbors (A, C, and D).
4. …
Neighborhood

Hello messages (Beacons) are one common method:
1. Node A sends a HELLO message to its neighbors (B, C, and D).
3. Node B sends a HELLO message to its neighbors (A, C, and D).
4. ...
Neighborhood

What information do we need about our neighbors?

- Distance to sink
- Last time heard
- Link quality
Neighborhood

Node E’s neighborhood table

<table>
<thead>
<tr>
<th>Id</th>
<th>Age</th>
<th>Distance</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>2 min</td>
<td>1 hop</td>
<td>87%</td>
</tr>
<tr>
<td>C</td>
<td>2 min</td>
<td>1 hop</td>
<td>98%</td>
</tr>
<tr>
<td>F</td>
<td>4 min</td>
<td>2 hops</td>
<td>74%</td>
</tr>
<tr>
<td>G</td>
<td>1 min</td>
<td>2 hops</td>
<td>93%</td>
</tr>
</tbody>
</table>

A few remarks:

- Only the distance to the sink is stored in # of hops
- Neighborhood discovery can’t be done only once!
- We need to estimate link qualities!
Neighborhood

Variations of simple schema:
- Each node sends $X$ beacons per minute.
- Number of beacons received per minute are stored.
- Quality is estimated over the past $Y$ minutes by counting losses.

Example ($X = 10; Y = 4$):

<table>
<thead>
<tr>
<th></th>
<th>t - 4</th>
<th>t - 3</th>
<th>t - 2</th>
<th>t - 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>7</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>5</td>
</tr>
</tbody>
</table>
Time Synchronization

Weather conditions, especially temperature and humidity, may have a significant effect on hardware.

Crystal oscillators are highly impacted by temperature!
Time Synchronization

![Temperature and time drift graphs]

**Air temperature [°C]**

- Indoor
- Outdoor
- Freezer

**Time drift [ms]**

- Day 1 to Day 8

---

**Legend:**
- Blue line for temperature
- Dotted line for time drift
Time Synchronization

Nodes need to know the time to:
• Timestamp packets
• Synchronize actions (e.g., taking samples, transmitting data)

How do we get time:
• Fully decentralized: Every node gets the time itself
• Partially centralized: Time is propagated from reference nodes
Time Synchronization

Every node gets the time:
- Atomic clock receivers:
  - Cheap (both energy and $)
  - Complexity
  - Limited coverage
- GPS:
  - Coverage
  - Complexity
  - High cost (energy and $)
- GPRS: same as GPS with less coverage

What about a partially centralized approach?
For instance, the sink serve as time reference node leveraging the anyway needed GPRS connection.
SensorScope

Many previous successful deployments

97 stations deployed at EPFL (one year)
SensorScope

Many previous successful deployments

16 stations deployed at Le Génépi to monitor conditions leading to dangerous mudslides (two months)
Visualizing Filter "Public Stations"

Local time: 2:59 (GMT+1)

Meteorological data

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Davis Anemometer Direction</td>
<td>West (283.8°)</td>
</tr>
<tr>
<td>Davis Anemometer Speed</td>
<td>2.7 m/s</td>
</tr>
<tr>
<td>SHT75 Humidity</td>
<td>65.3 %</td>
</tr>
<tr>
<td>SHT75 Temperature</td>
<td>-13.6 °C</td>
</tr>
</tbody>
</table>

Plot recent data

Health status

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Battery - External</td>
<td>Not connected</td>
</tr>
<tr>
<td>Battery - Internal</td>
<td>5.4 V</td>
</tr>
<tr>
<td>CPU - Temperature</td>
<td>-17.9 °C</td>
</tr>
<tr>
<td>CPU - Voltage</td>
<td>3 V</td>
</tr>
<tr>
<td>MMC Card Free Space</td>
<td>964 MB</td>
</tr>
</tbody>
</table>
Conclusion
Take Home Messages

• Power is most often the key design constraint in embedded systems, even more for field embedded systems/instruments if operating in unattended mode.
• Efficient power management strategies can decrease the consumption by several orders of magnitude.
• Power is comparably difficult to generate and store.
• Sensor networks enable environmental monitoring in remote locations and of difficult-to-measure processes.
• Real-world deployments may be highly unpredictable!
Additional Literature – Week 11

Books

Papers

Pointers
• Sensorscope: http://sensorscope.epfl.ch/
• Permasense: http://www.permasense.ch/
• GITEWS: http://www.gitews.de
• CENS: https://en.wikipedia.org/wiki/Center_for_EMBEDDED_Network_Sensing