Transiently Powered Wireless Embedded Systems

Kasım Sinan Yıldırım

Embedded Software Group, Delft University of Technology

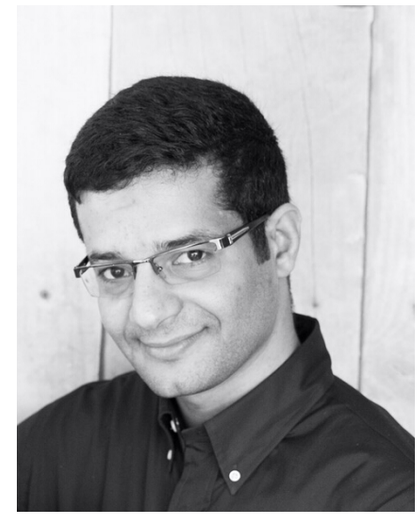
December 9, 2016

4TU Embedded Systems Student Event

Battery-less Computers

Przemysław Pawełczak

Kasım Sinan Yıldırım



Amjad Yousef Majid

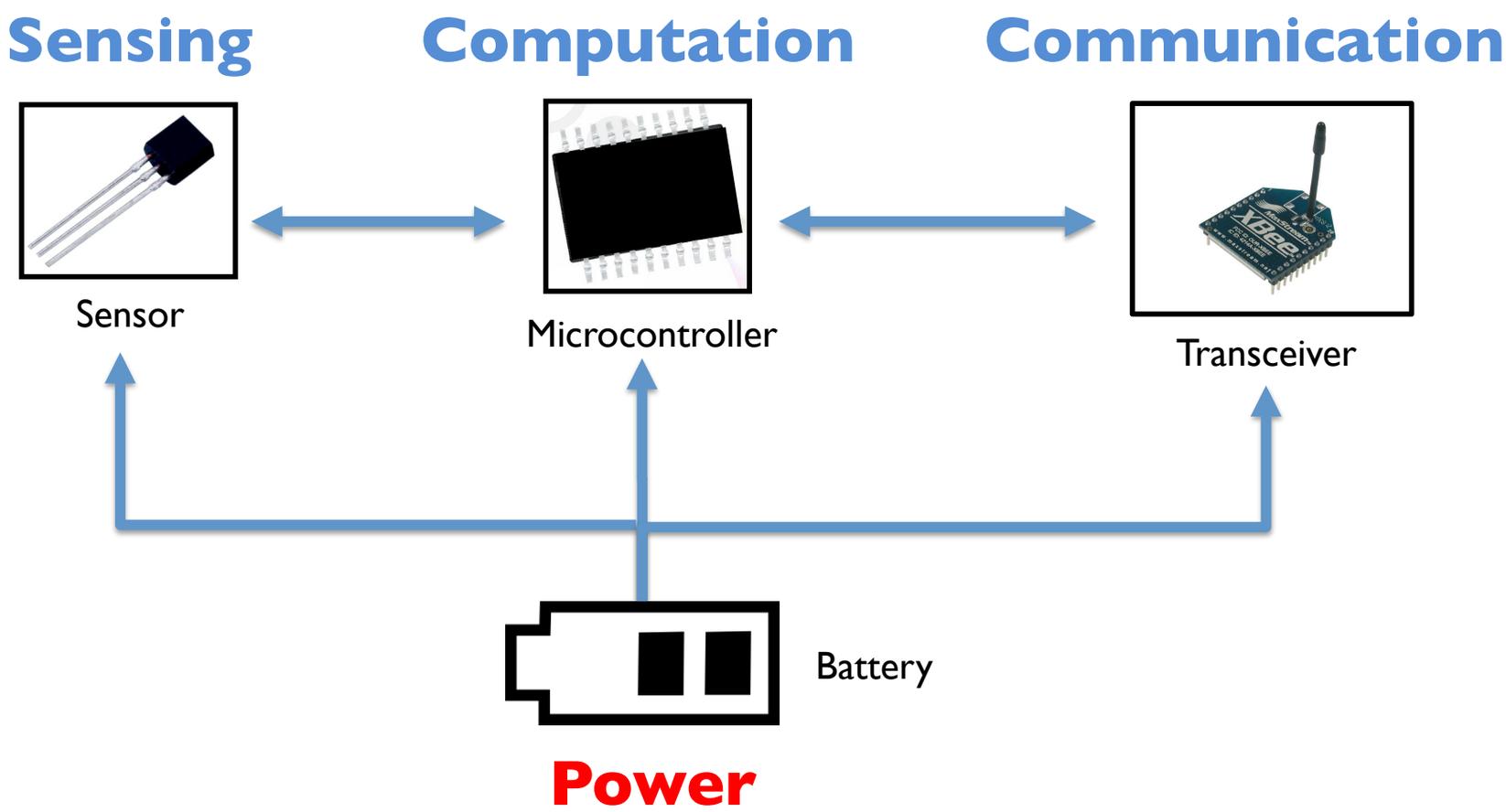
Michel Jansen

Koen Schaper

embedded software

TU Delft

IoT – Wireless Embedded Systems



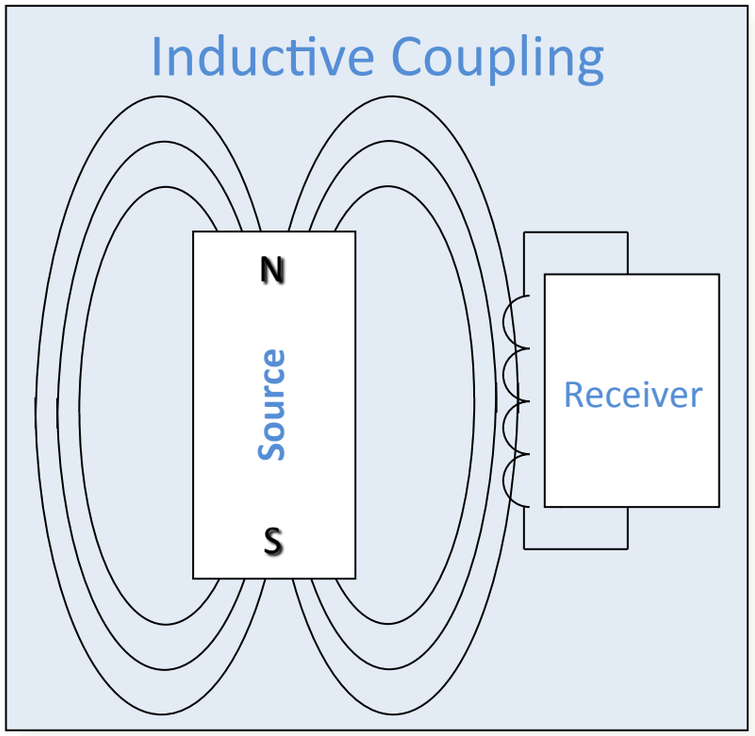
Powering IoT

- Powering cyber-physical systems is a challenge
  - **By 2025:** >100 billion IoT devices
  - sustainable operation
  - large-scale deployment
- Batteries
  - increase weight, cost of the hardware
  - replenishment is generally impractical
  - ecological footprint
- Transfer of electromagnetic energy
  - from a power source to receiver devices over the air
  - wireless power transfer

Iris Mote

Wireless Power Transfer (WPT) - I

- Non-radiative techniques
  - either inductive or magnetic resonant coupling
    - varying magnetic flux induces current
  - transfer power over short distances

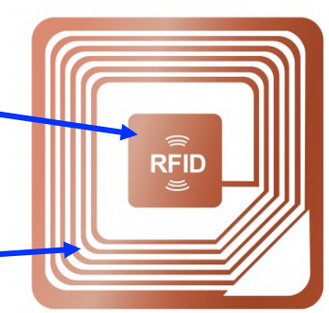


Wireless Power Transfer (WPT) - II

- Radiative techniques
  - use the electric field of the electromagnetic waves
    - radio frequency (RF) waves as an energy delivery medium
  - transfer power over longer distances
  - provision of energy to many receivers simultaneously
    - broadcast nature
  - low complexity, size and cost for the energy receiver hardware
  - suitability for mobility
  - charge low-power embedded devices
    - RFID (Radio Frequency Identification) tags

chip

antenna



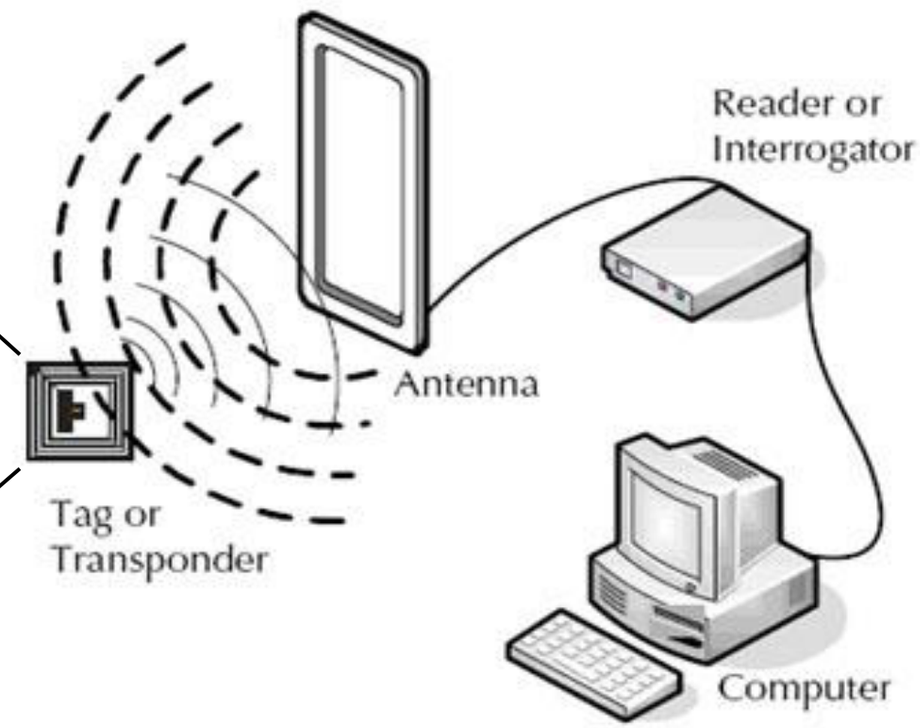
RFID System Overview

Antenna

IC

Tiny memory

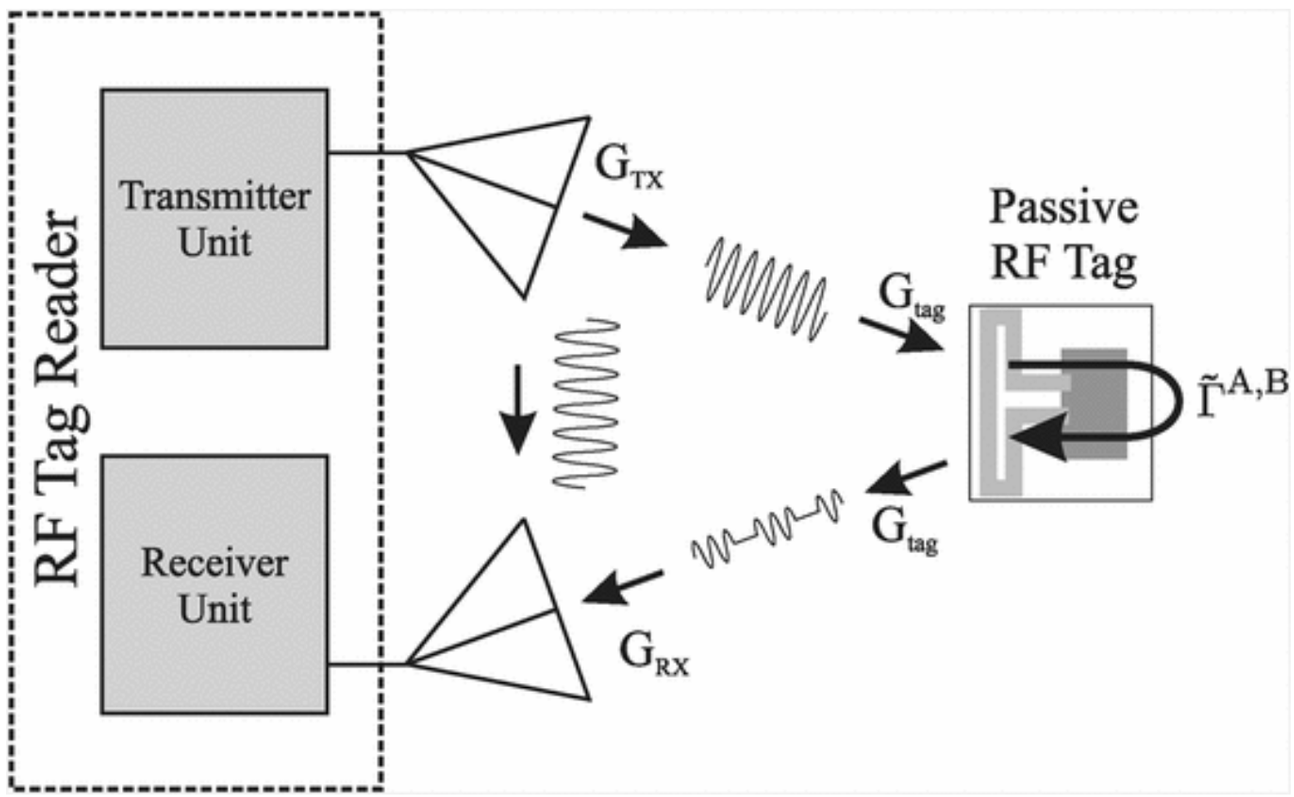
Energy Harvester



http://ieeexplore.ieee.org/xpls/icp.jsp?arnumber=4447347

Backscatter Communication

- Transmitting data requires too much energy for a battery-free device
- Modulate the reflections of an incident RF signal
  - use the signal sent by the reader to communicate data back
  - not by generating radio waves



http://ieeexplore.ieee.org/xpls/icp.jsp?arnumber=4447347

RF-Powered Computing

- A new class of low-power battery-less embedded systems
  - Transiently Powered Computers (TPCs)
- CRFIDs (Computational RFIDs)
  - RFID technology as a foundation
  - Allow sensing, computation and communication without batteries
    - Charge a super capacitor using harvested rf energy
  - Equipped with a backscatter radio
    - simple circuitry for the receiver
    - allows communication to come almost for free

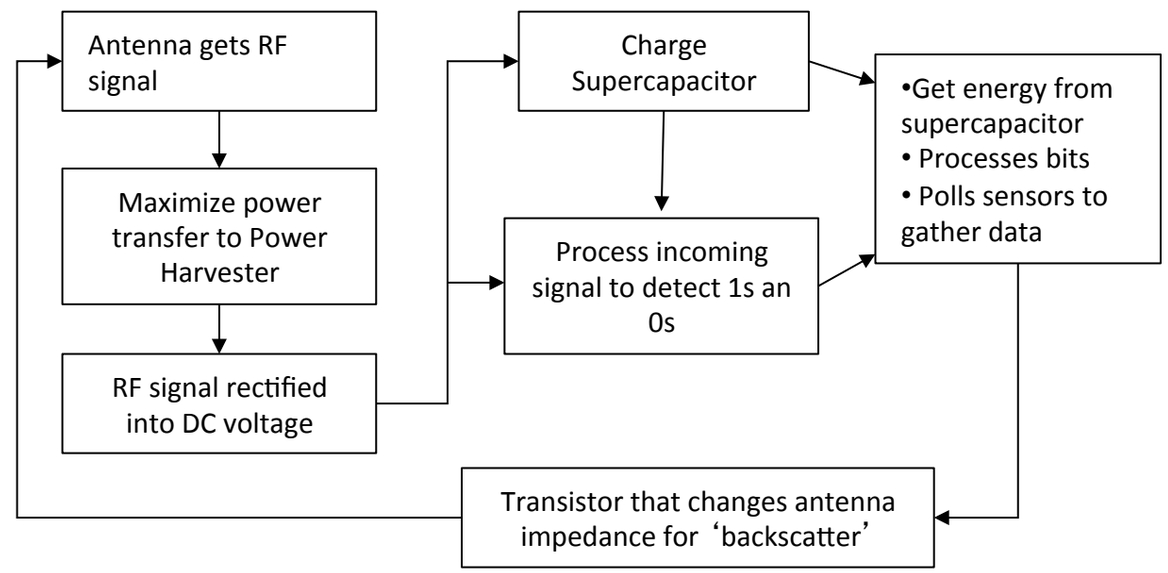
A CRFID platform: WISP - Wireless Identification and Sensing Platform (University of Washington)

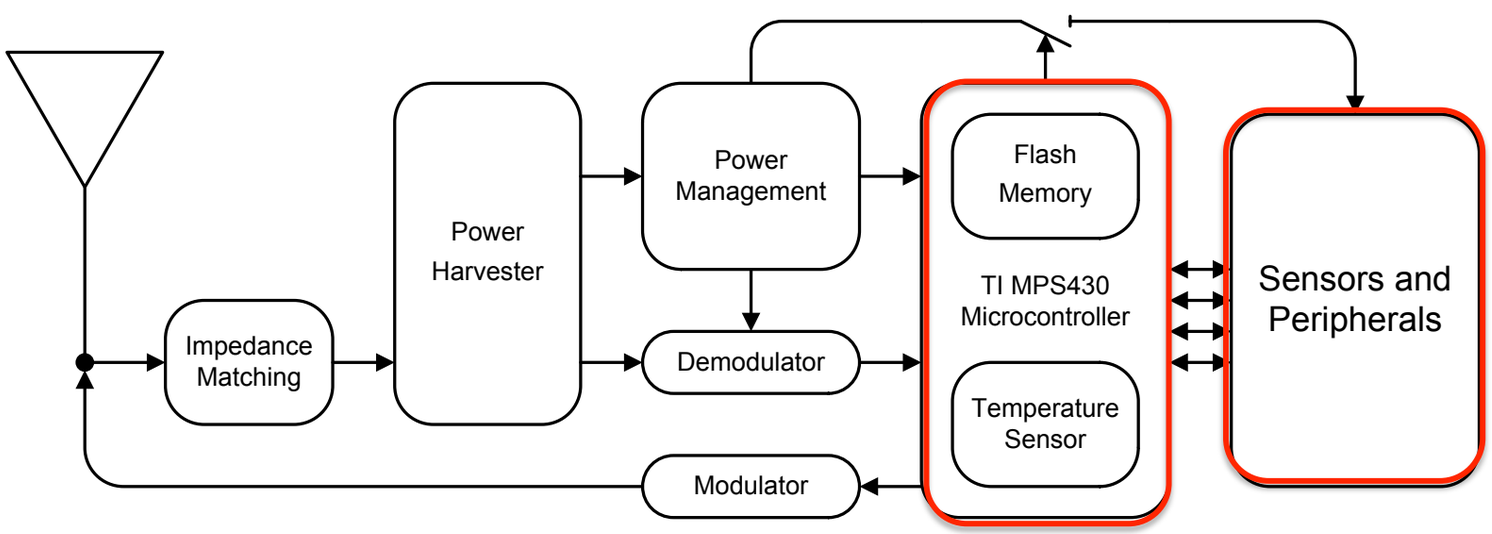
Ultimate goal: replacing existing battery-powered wireless sensor networks

WISP Hardware - Overview

RFID reader

Antenna

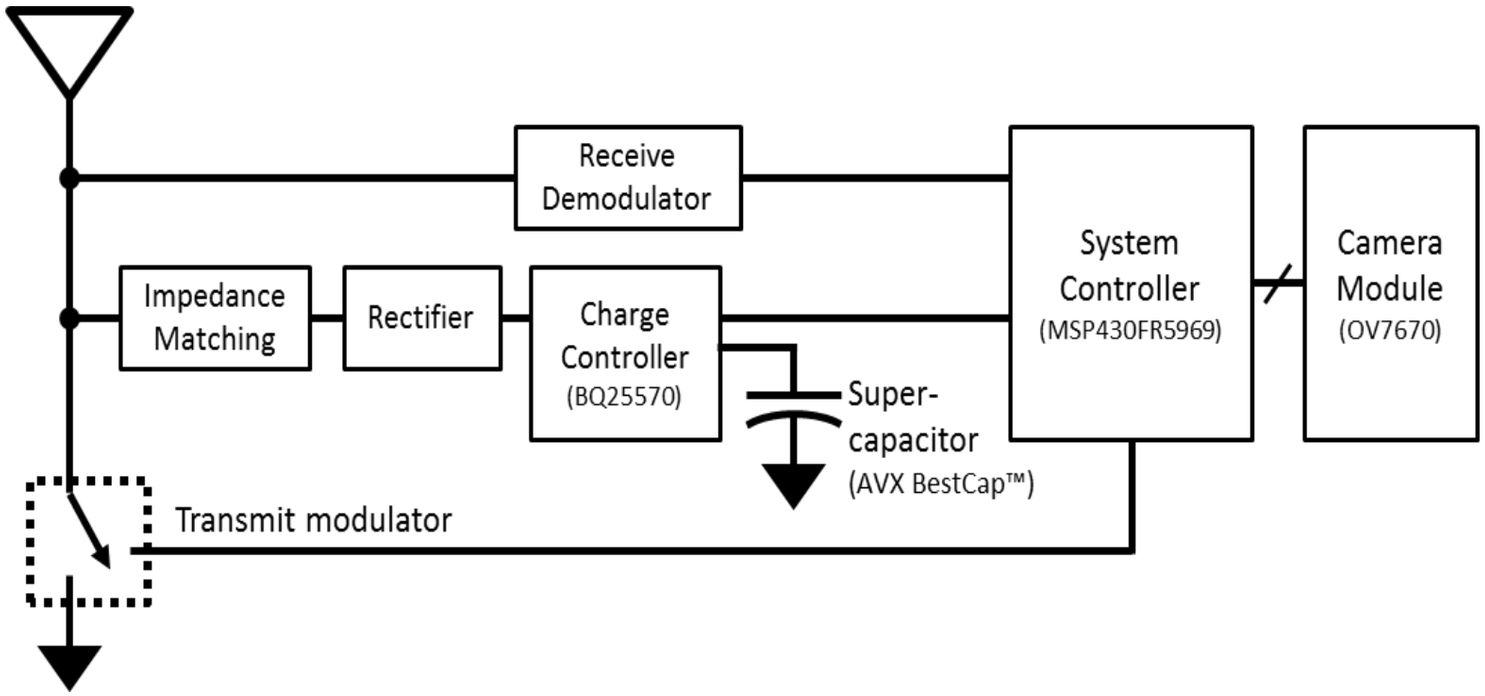


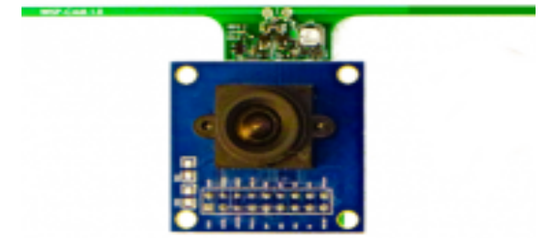


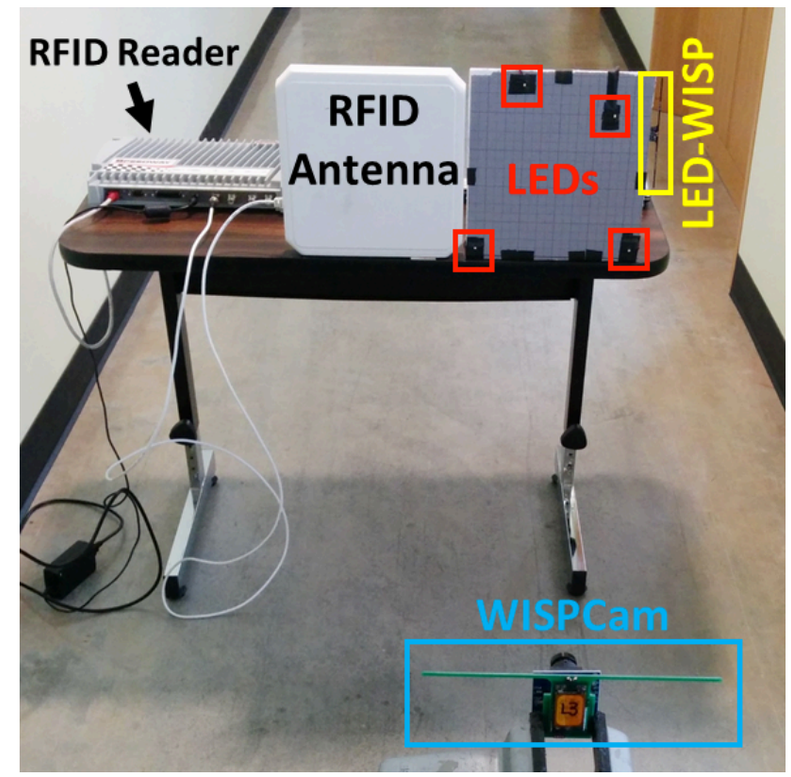
Sample, Alanson P., et al. "Design of an RFID-based battery-free programmable sensing platform." IEEE Transactions on Instrumentation and Measurement 57.11 (2008): 2608-2615.

WISPCam: Battery-less Camera

- WISPCam - University of Washington





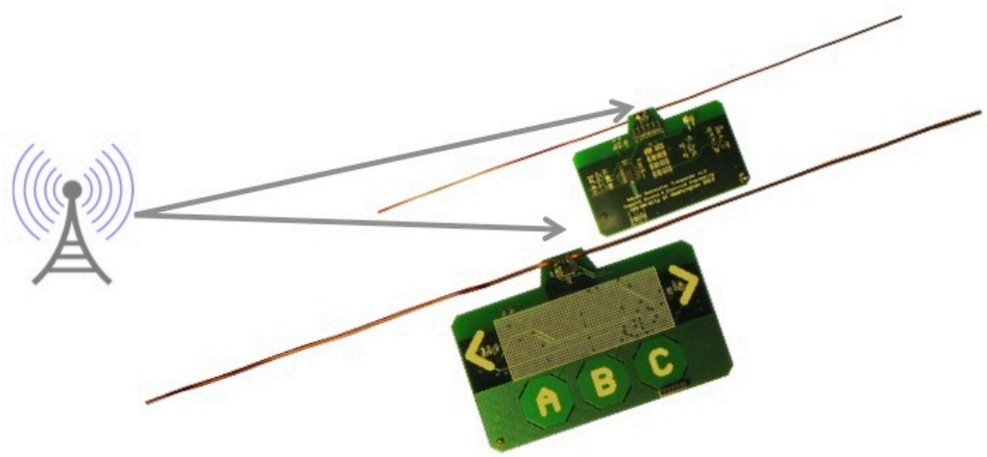


WISPCam captures a 160x120 low resolution image for face detection

Naderiparizi, Saman, et al. "Wispcam: A battery-free rfid camera." 2015 IEEE International Conference on RFID (RFID). IEEE, 2015.

Ambient Backscatter

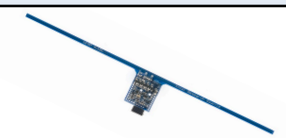
- Traditional backscatter communication, (e.g. in RFID)
  - a device communicates by modulating its reflections of an incident RF signal - not by generating radio waves
- Ambient backscatter
  - Communicate using ambient RF signals as the only source of power
    - Ambient RF from TV and cellular communications



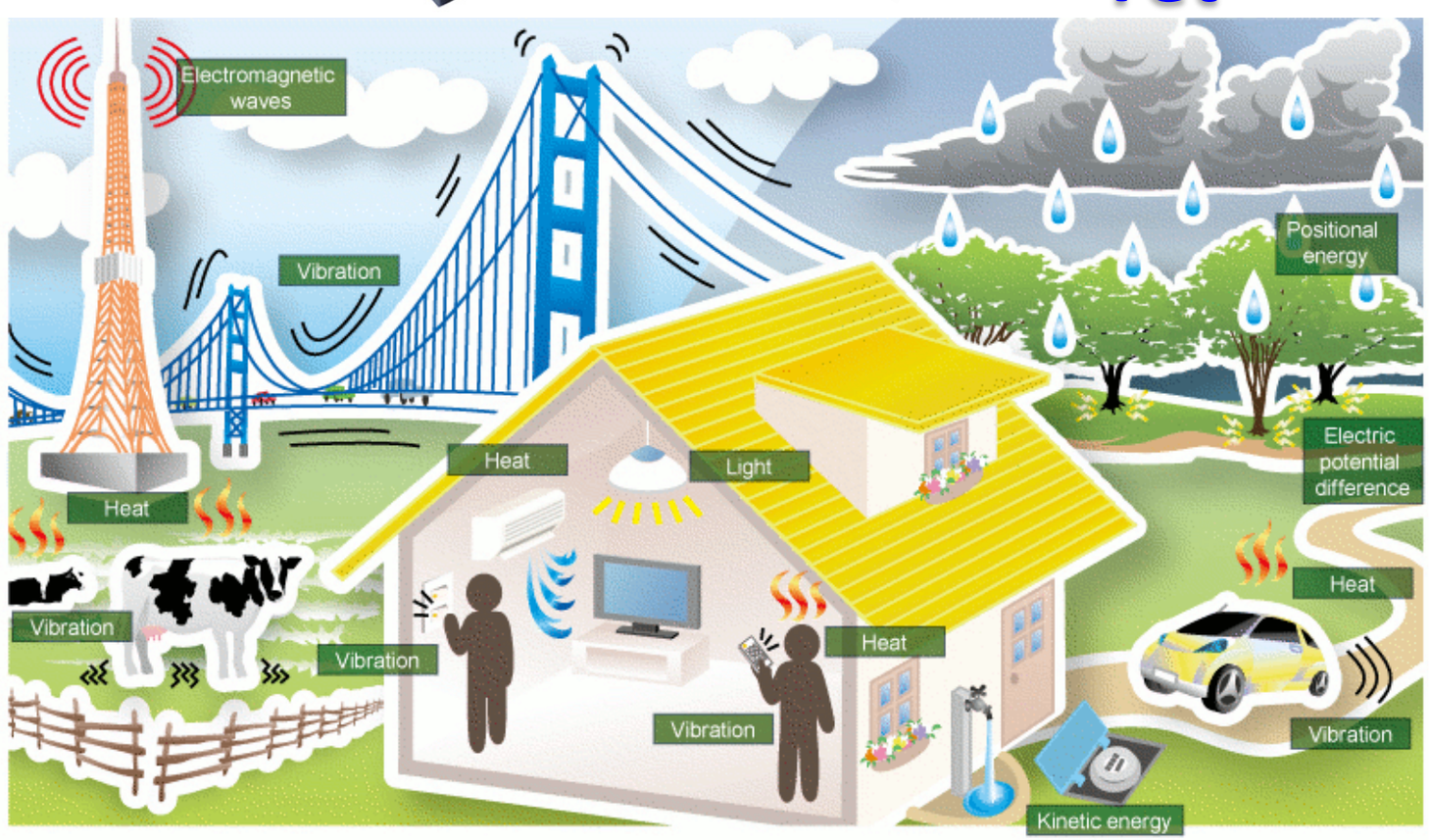
Vincent Liu et al. "Ambient Backscatter: Wireless Communication Out of Thin Air ", ISIGCOMM, August 2013

Future…

	Active	Passive	Ideal
Power Source	Battery-powered	RF-powered (battery-free)	RF-powered (battery-free)
Physical Operating Range	Unlimited	Requires proximity to RF power source	Unlimited
Lifespan	Months to years	No fundamental limitation	No fundamental limitation



Not Yet

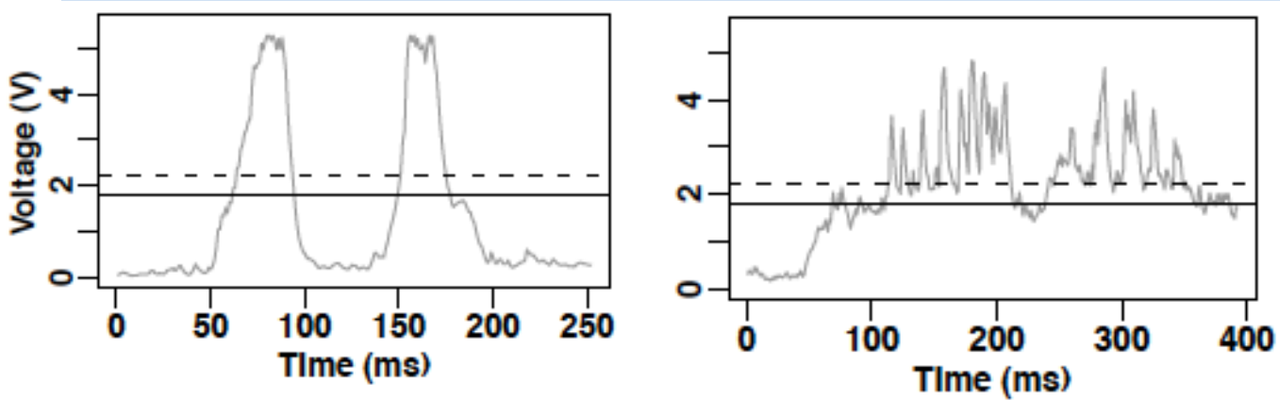


Programming Challenges

WISP tags vs Battery-powered Nodes - I

- Continuously varying voltage level
  - WSNs: stable voltage levels in the short term (battery-powered)
  - WISP: fluctuating input voltage[1]

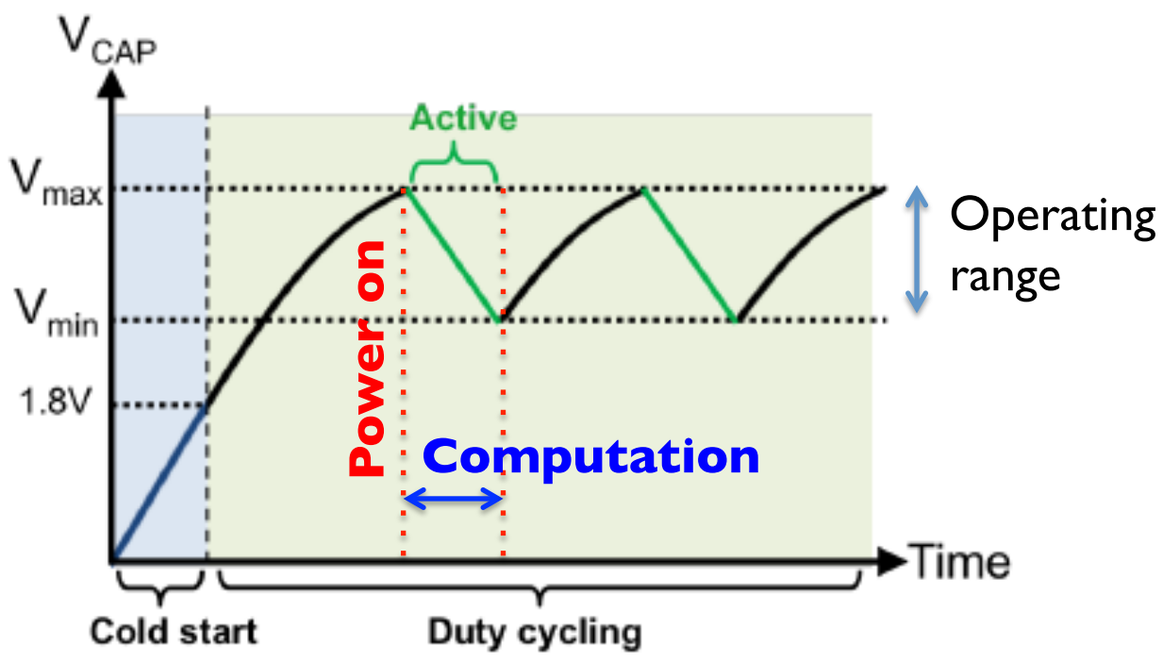
Different voltage levels at different distances to the reader



[1]*Benjamin Ransford et al., "Mementos: system support for long-running computation on RFID-scale devices." Acm Sigplan Notices 47.4 (2012): 159-170.*

WISP tags vs Battery-powered Nodes - II

- Frequent loss of computation state
  - frequently ``die'' due to power loss
    - need to save the computation state into the non-volatile memory
    - recover when they harvested sufficient energy to start up
    - saving computational state to non-volatile memory is also energy consuming



Energy is available intermittently

Computation is intermittent

[1]*Naderiparizi, Saman, et al. "Wispcam: A battery-free rfid camera." 2015 IEEE International Conference on RFID (RFID). IEEE, 2015.*

WISP tags vs Battery-powered Nodes - III

- The classical motto
  - ``compute instead of communicate whenever possible''
  - No longer valid for the WISP platform
    - backscatter communication comes almost for free
- Intermittent power
  - lightweight methods in terms of computation are desirable

A Simple Program: Fibonacci

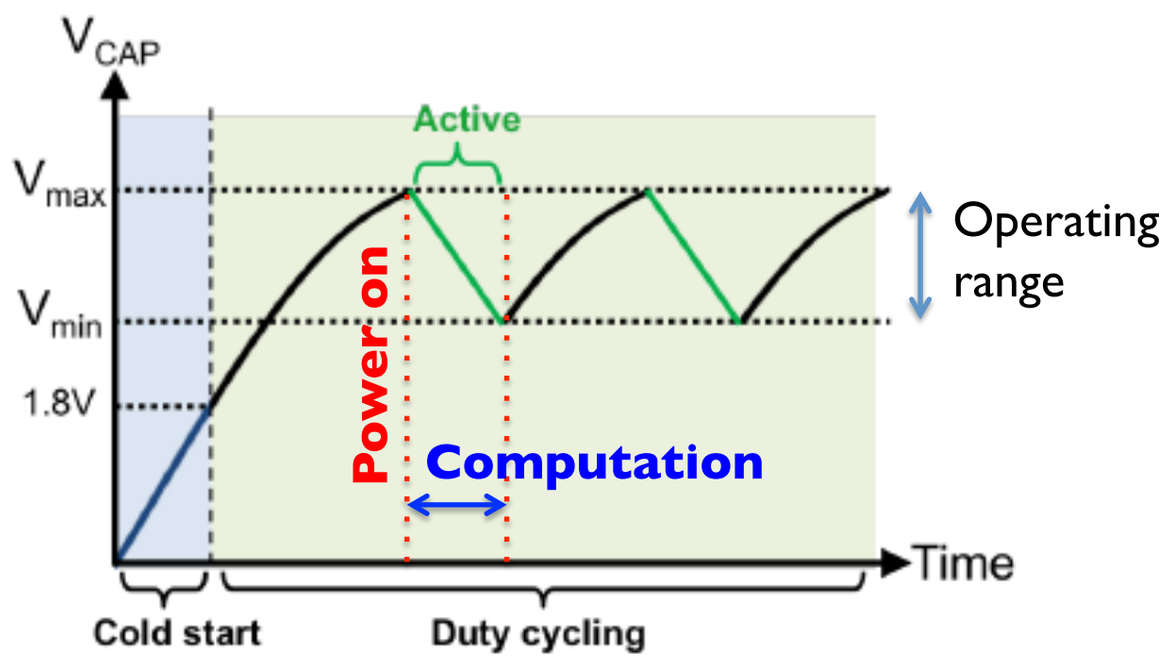
```
function fibonacci(n){
   a= 1;
   b= 1;

   while(i<=n){
      c = a + b;
      a = b;
      b = c;
   }

  return b;
}
```

```
main {
   i = 0;
   while(TRUE){
      fibonacci(i++);
    }
}
```

1, 1, 2, 3, 5, 8, 13, 21, 34, 55,..

TPCs – How?

- How to design programs under the effect of frequent power interruptions?
  - How to ensure
    - Consistency of the non-volatile memory?
    - Correctness of the program?
- How to determine when and what to save in non-volatile memory
  - Energy consuming

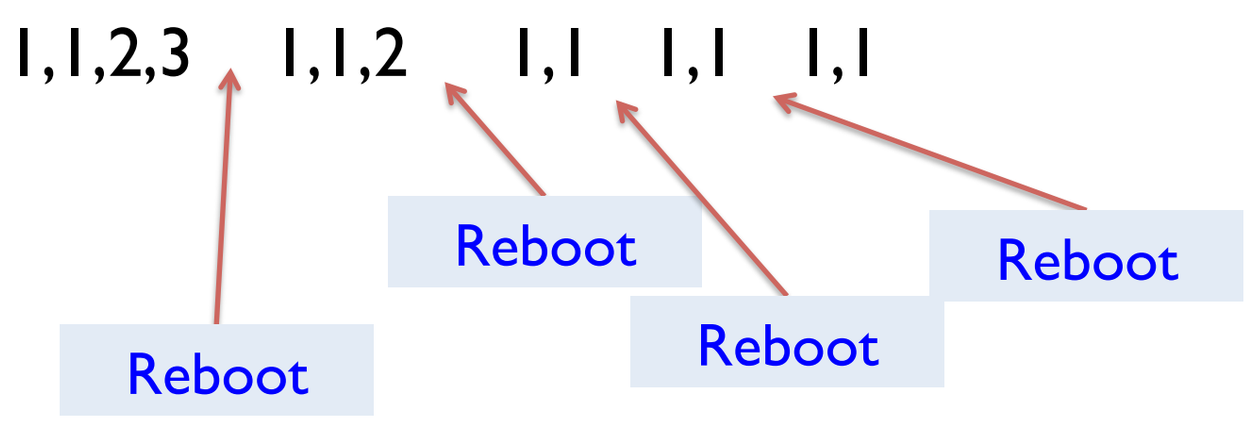


[1] *Chain: Tasks and Channels for Reliable Intermittent Programs*
Alexei Colin, Brandon Lucia, OOPSLA 2016

Execution Under Intermittent Power

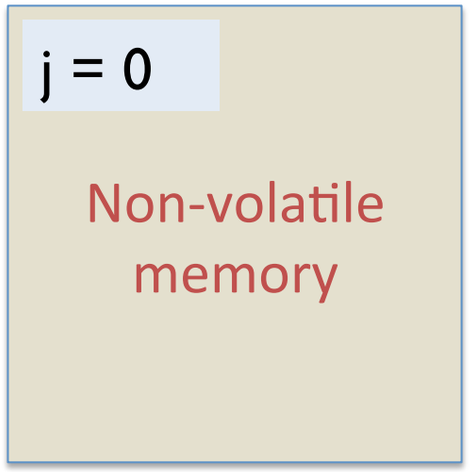
```
function fibonacci(n){
   a= 1;
   b= 1;

   while(i<=n){
      c = a + b;
      a = b;
      b = c;
   }

  return b;
}
```

```
main {
   i = 0;
   while(TRUE){
      fibonacci(i++);
    }
}
```



How to Design Programs

```
main {
   i = j;
   while(TRUE){
      print fibonacci(i++);
      j++;
    }
}
```

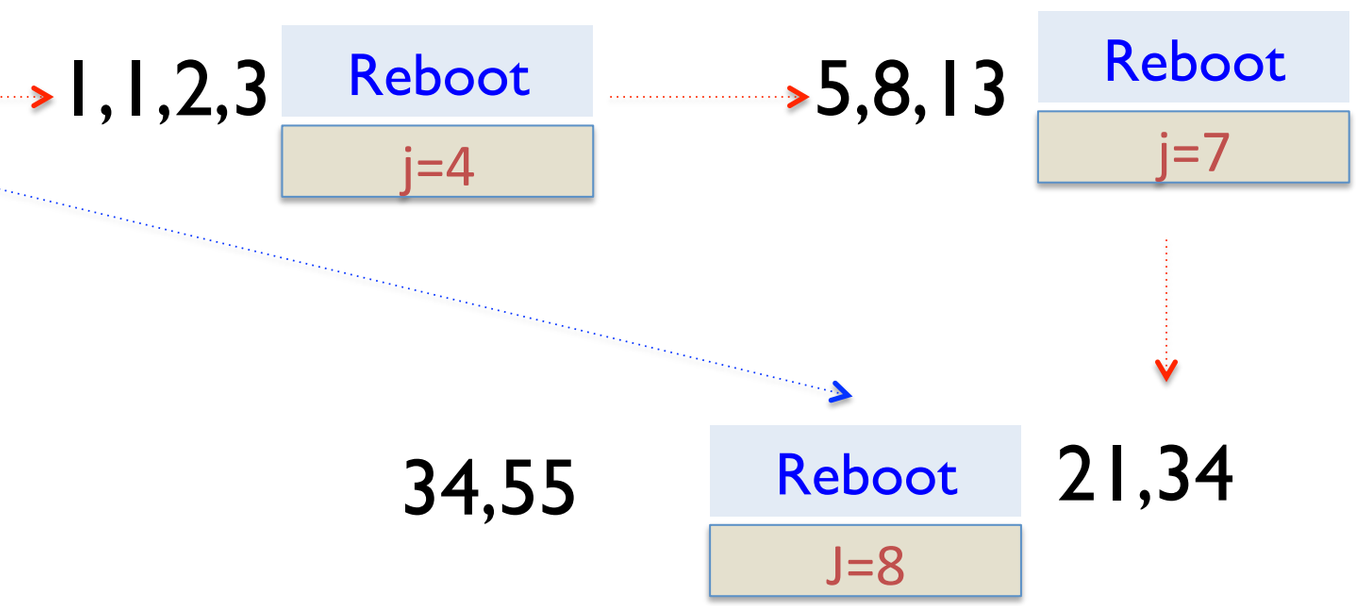


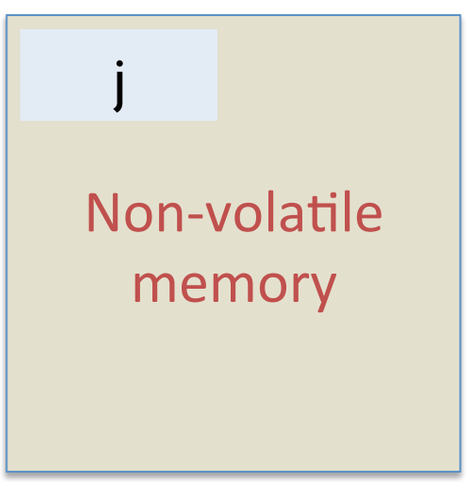
Computational state

Save computational state into non-volatile memory.
Recover state when there is sufficient energy and continue.

Execution Under Intermittent Power

```
main {
   i = j;
   while(TRUE){
      print fibonacci(i++);
      j++;
    }
}
```





Computational state

The program can be interrupted at any time!

Trade-offs

The more frequent you save the computational state, the more overhead you introduce!

The less frequent you save the computational state, the more redundant computations you need to do!

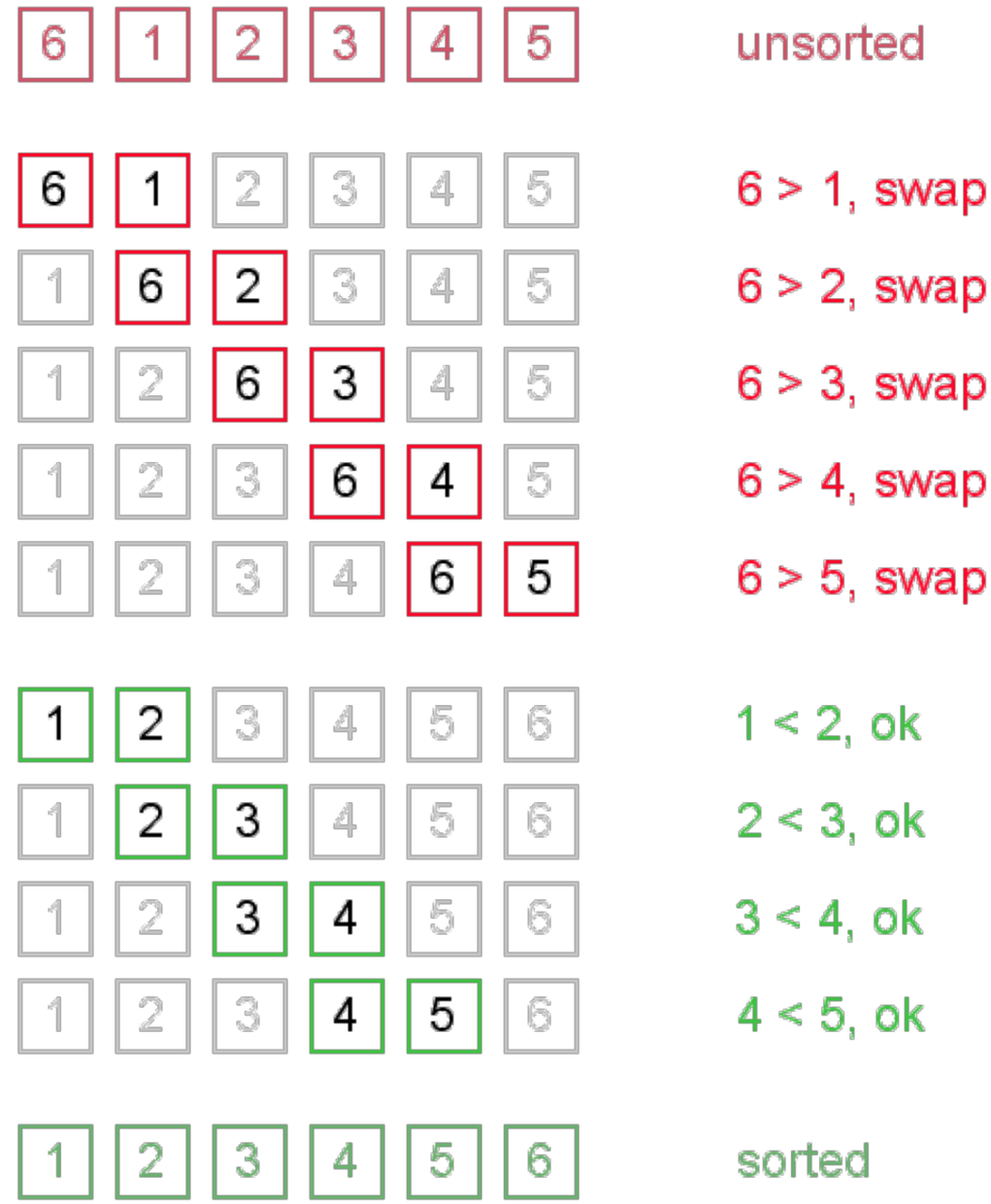
How to ensure consistency of the non-volatile memory?
When and what to save?

Exercise: Buble Sort

Implement Power-Aware Bubble Sort on array A in non-volatile memory

Which variables to store in non-volatile memory?

When to update the variables in non-volatile memory?



Conclusions

- RF-Based TPCs are emerging
- There are lots of research opportunities in this domain
  - Communication Protocols
    - Physical layer
    - MAC layer
    - Routing, Synchronization
  - Programming Platforms
  - Operating Systems
  - Safety, security
  - Many more…

Thank You!