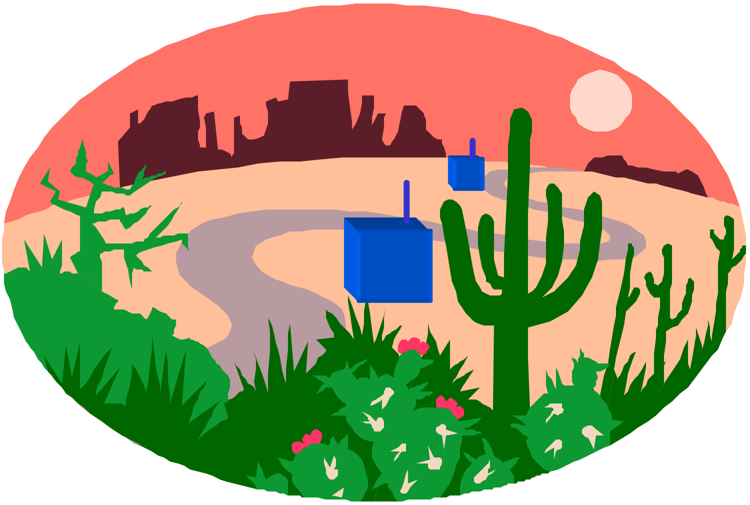
Dynamic Voltage Scaling for Distributed Microsensor Networks

Rex Min, Travis Furrer, and
Anantha Chandrakasan

April 27, 2000

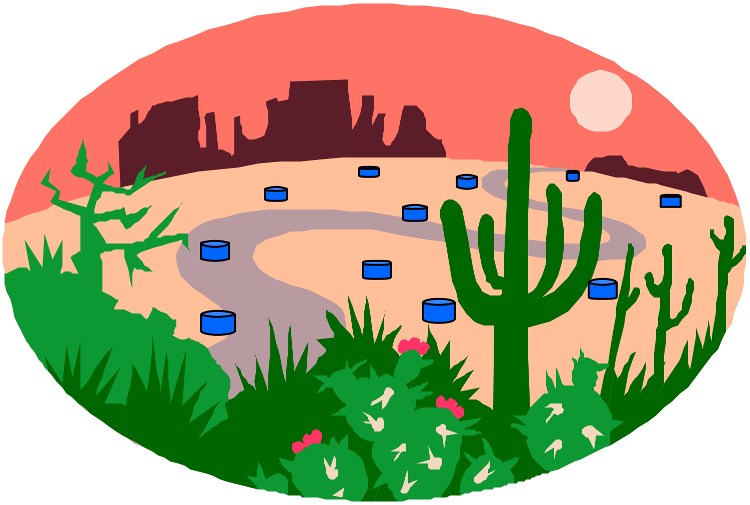
IEEE Workshop on VLSI

Distributed Microsensor Networks



Macrosensor Network

- Large, expensive nodes
- Accurate but high-maintenance
- Large internal battery



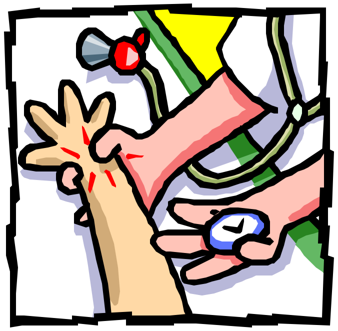
Microsensor Network

- Small, consumable nodes
- Collaborative data gathering
- No maintenance, fault-tolerant
- **Extremely low-power**

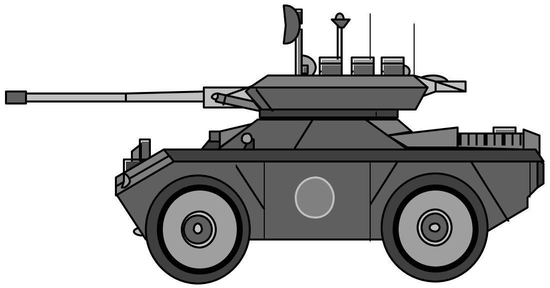
Applications of Distributed Microsensors

Monitoring of industrial/commercial equipment (Xerox, Boeing)

Remote climate data (Lincoln Lab)



Human vital signs and involuntary responses (MIT Media Lab)



Military movement detection/classification (U.S. Army)

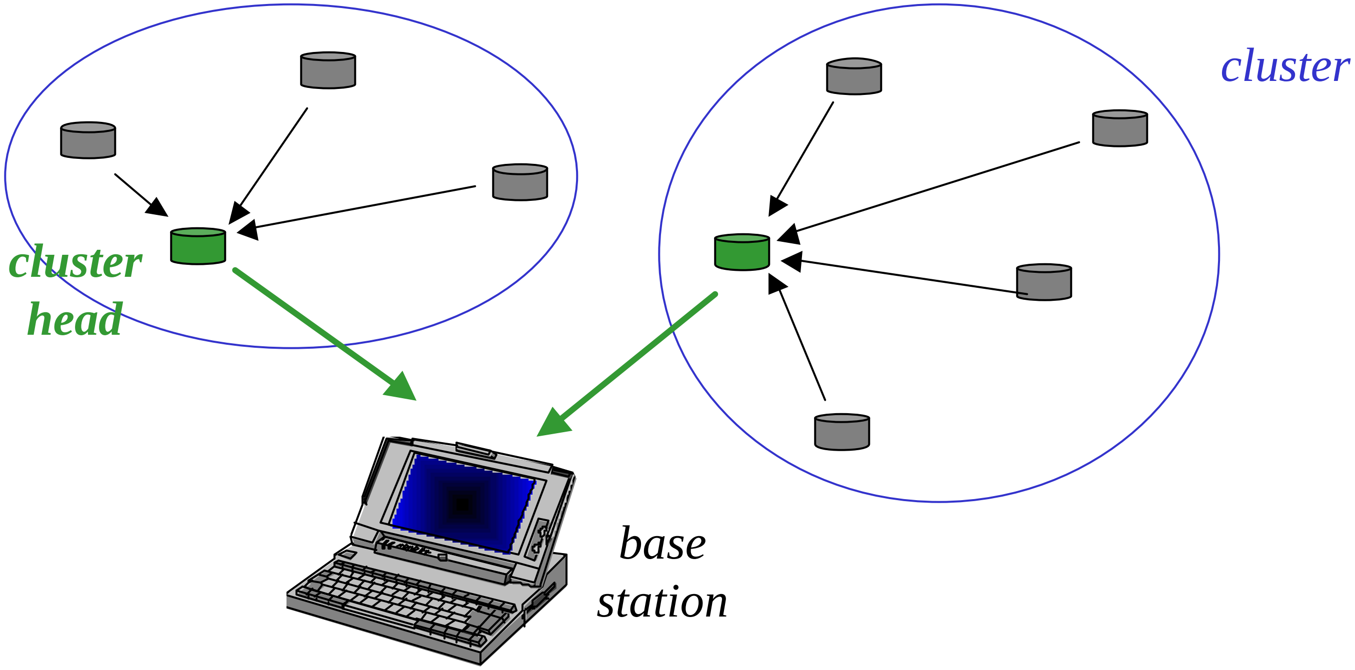
Challenges in Distributed Microsensor Networks

- Routing sensor data to remote base station
  - Combine signal processing with low-energy networking
- Limited battery capacity at the node
  - Harvest energy from the environment
- Low data rate, event-driven computation
  - Low duty cycle electronics, leakage current control
- Time-varying resources
  - Energy/quality scalable computing

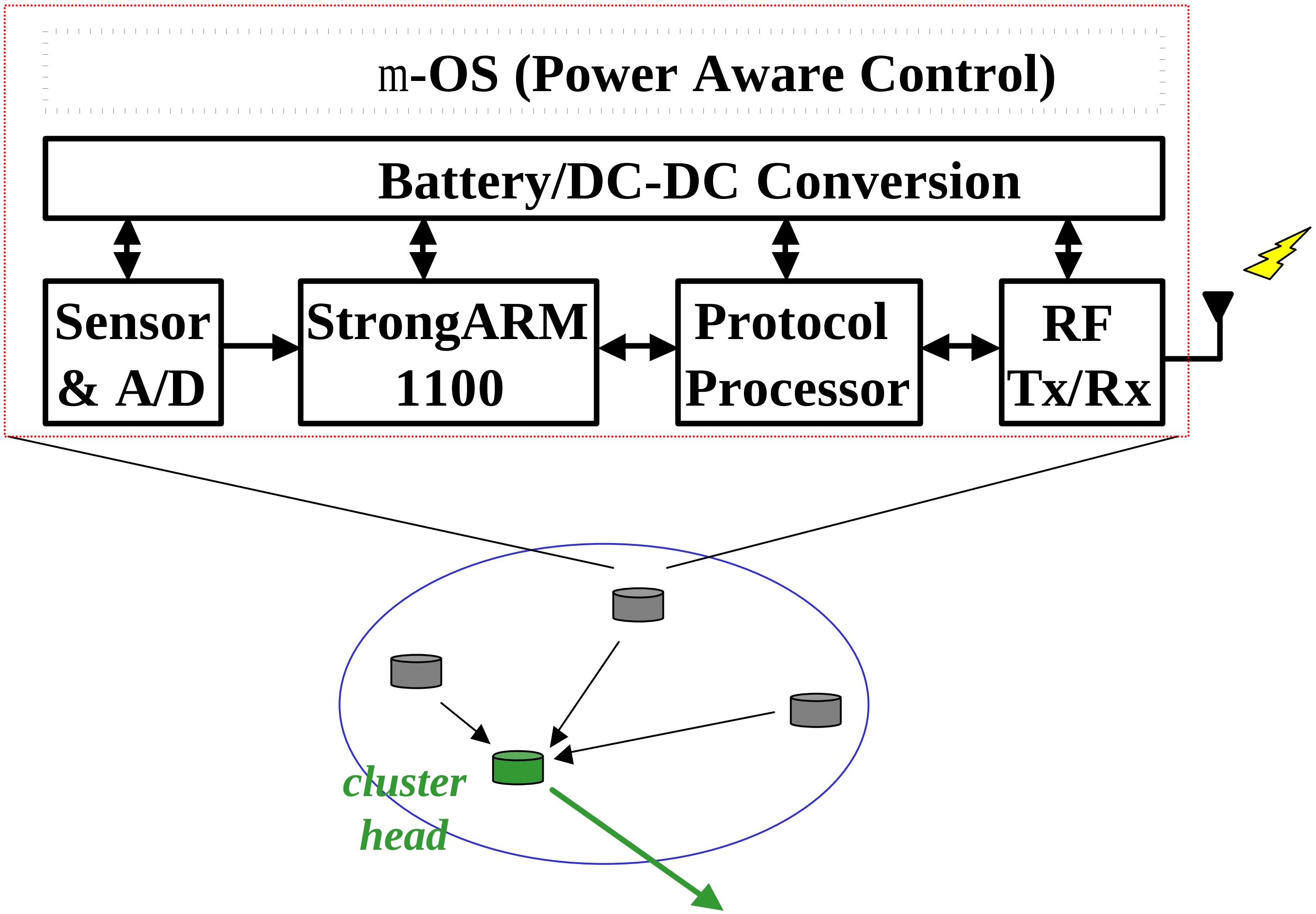
The MIT µAMPS Project

micro-Adaptive Multidomain Power-aware Sensors

- Dynamically formed *clusters*
- Data aggregation at *cluster head*
- Clustering reduces total Tx energy



μAMPS Sensor Node



m-OS (Power Aware Control)

Battery/DC-DC Conversion

Sensor & A/D

StrongARM 1100

Protocol Processor

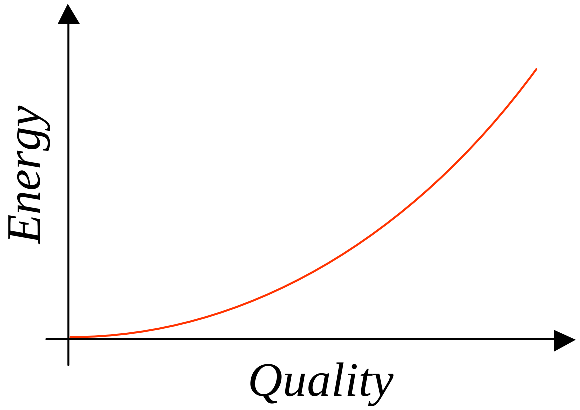
RF Tx/Rx

cluster head

Energy/Quality Scalability

Reducing computation for energy savings

- Exploit scalable algorithms...
  - FIR Filtering: number of taps
  - Computational latency: clock speed
  - Encryption: length of key
  - Data aggregation: number of data channels (sensors)
- …to adapt dynamically to:
  - Available energy
  - User-specified result quality
  - Conditions in environment



Dynamic Voltage Scaling (DVS)

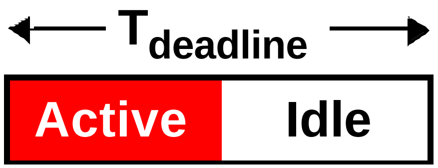
Enabling the energy-quality tradeoff

- *Switching energy* for a given computation is independent of time:

$$E_{switch} = C_{tot} V_{DD}^2$$

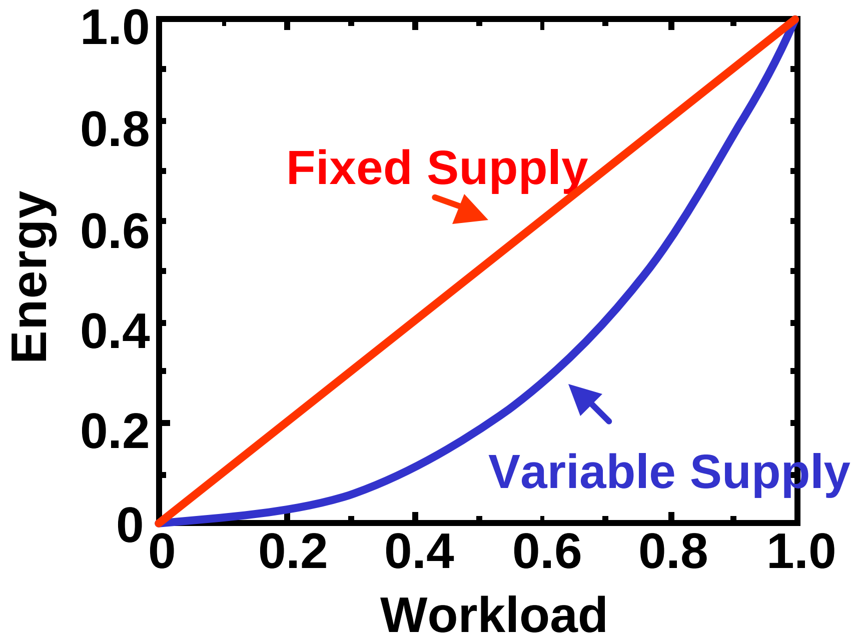
total capacitance switched by computation

- *Latency* of computation increases when voltage decreases



$E_{fixed} = CV_{DD}^2$

$E_{var} = C(V_{DD}/2)^2 = 1/4E_{fixed}$

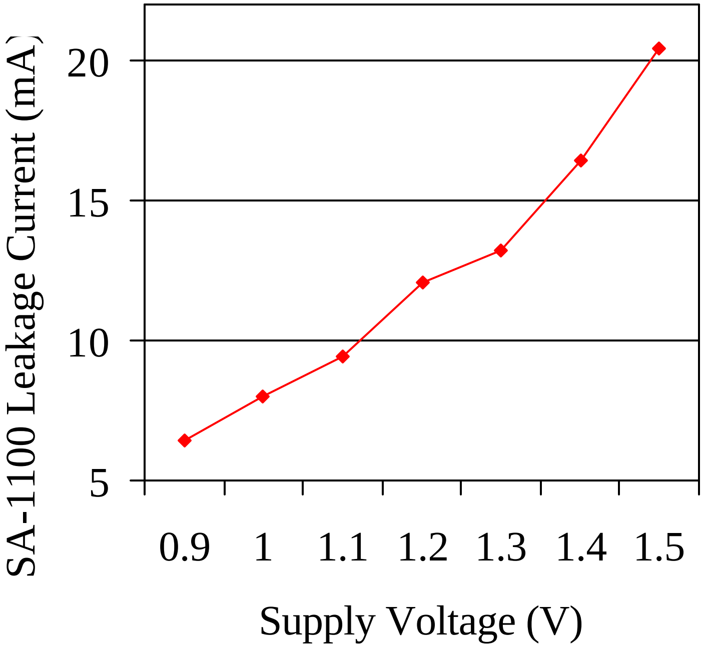


DVS and Leakage Current

- *Leakage energy* for a computation varies with V_{DD} and time:

$$E_{leak} = V_{DD}\,\underbrace{\left(I_0 \exp \frac{V_{DD}}{nV_T}\right)}_{\textit{leakage current}}\,\underset{\textit{time required by computation}}{t}$$

- DVS reduces leakage energy by reducing V_{DD}



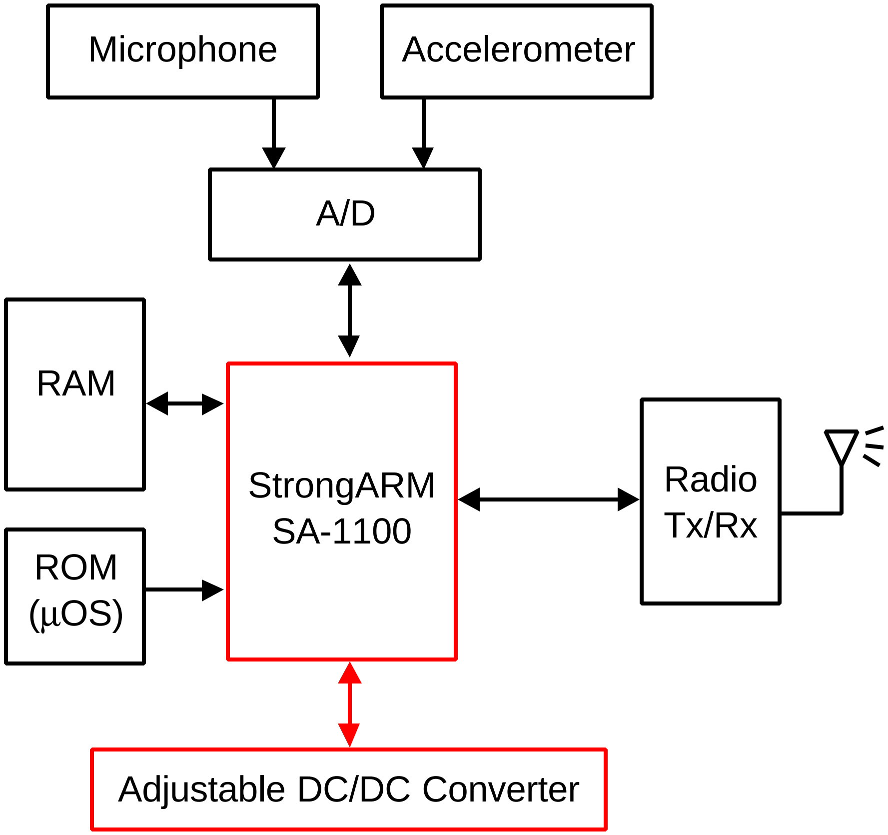
Prior Work on DVS

- ASIC design
  - J. Goodman: encryption processor
  - V. Gutnik: variable-voltage supply
- General-purpose processors
  - T. Burd: ARM core and peripherals with circuit-level modifications
- Voltage scheduling
  - T. Pering, M. Weiser: simulations with interval-based schedulers

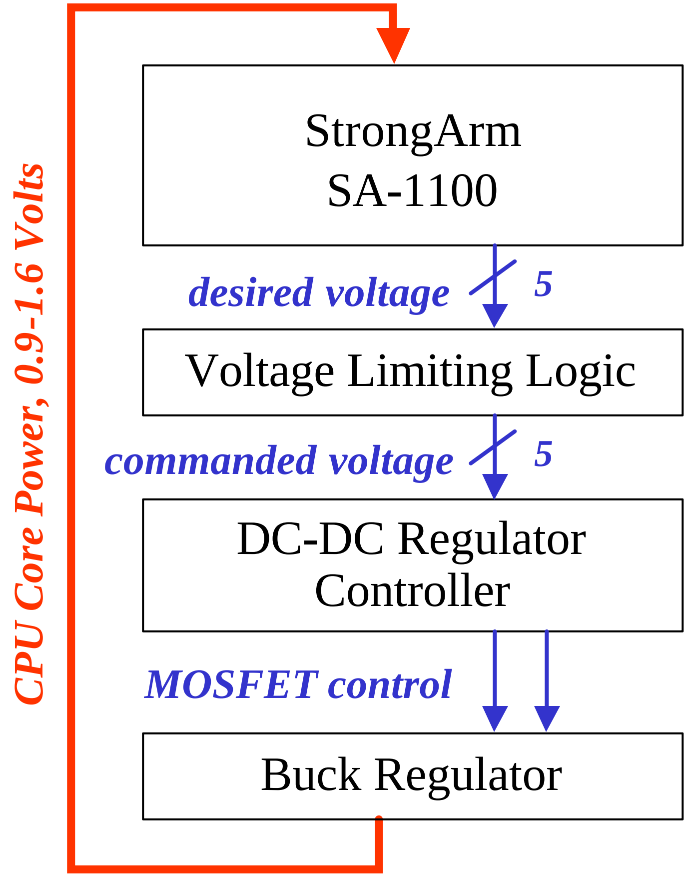
We have developed the necessary hardware and software to enable DVS on a COTS processor

DVS on the μAMPS Sensor Node

- First μAMPS sensor node will be constructed with COTS components
- SA-1100 is low-power and *static*



DVS Demonstration System



CPU Core Power, 0.9-1.6 Volts

StrongArm SA-1100

desired voltage 5

Voltage Limiting Logic

commanded voltage 5

DC-DC Regulator Controller

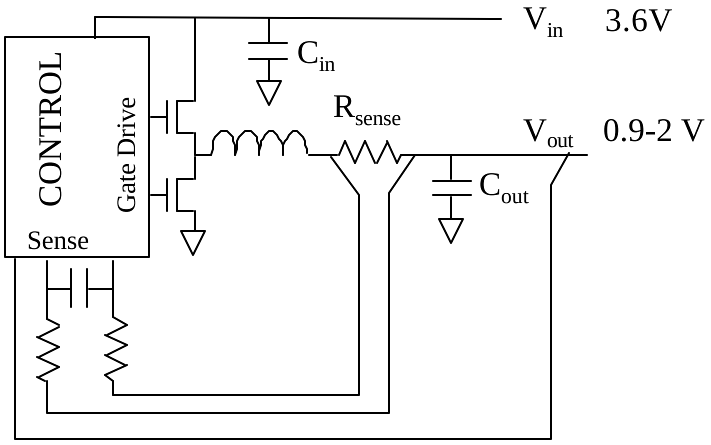
MOSFET control

Buck Regulator

Operating system selects a voltage/clock frequency combination based upon current processor workload

Glue logic prevents overvoltage to core (V_{DD} < 1.6V)

COTS regulator controller drives buck regulator to the commanded voltage



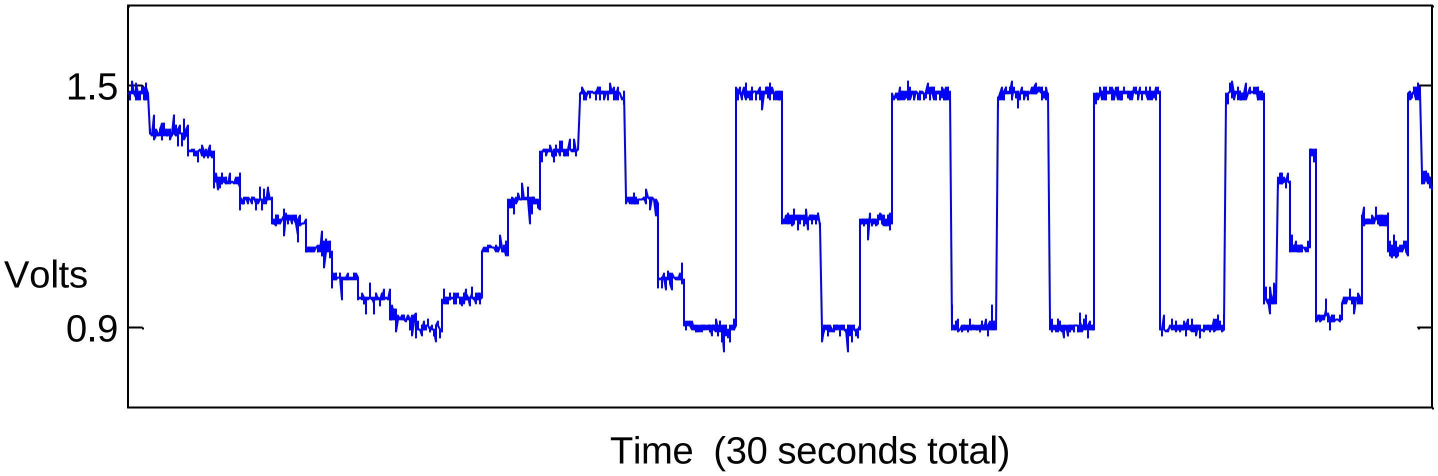
μAMPS Operating System

- μAMPS operating system based on the eCos microkernel
- Multithreaded, real-time, open-source OS
- Lightweight “μOS” (no file system, runtime loader, etc.)
- *Support for dynamic voltage scaling added*

μOS Support for DVS

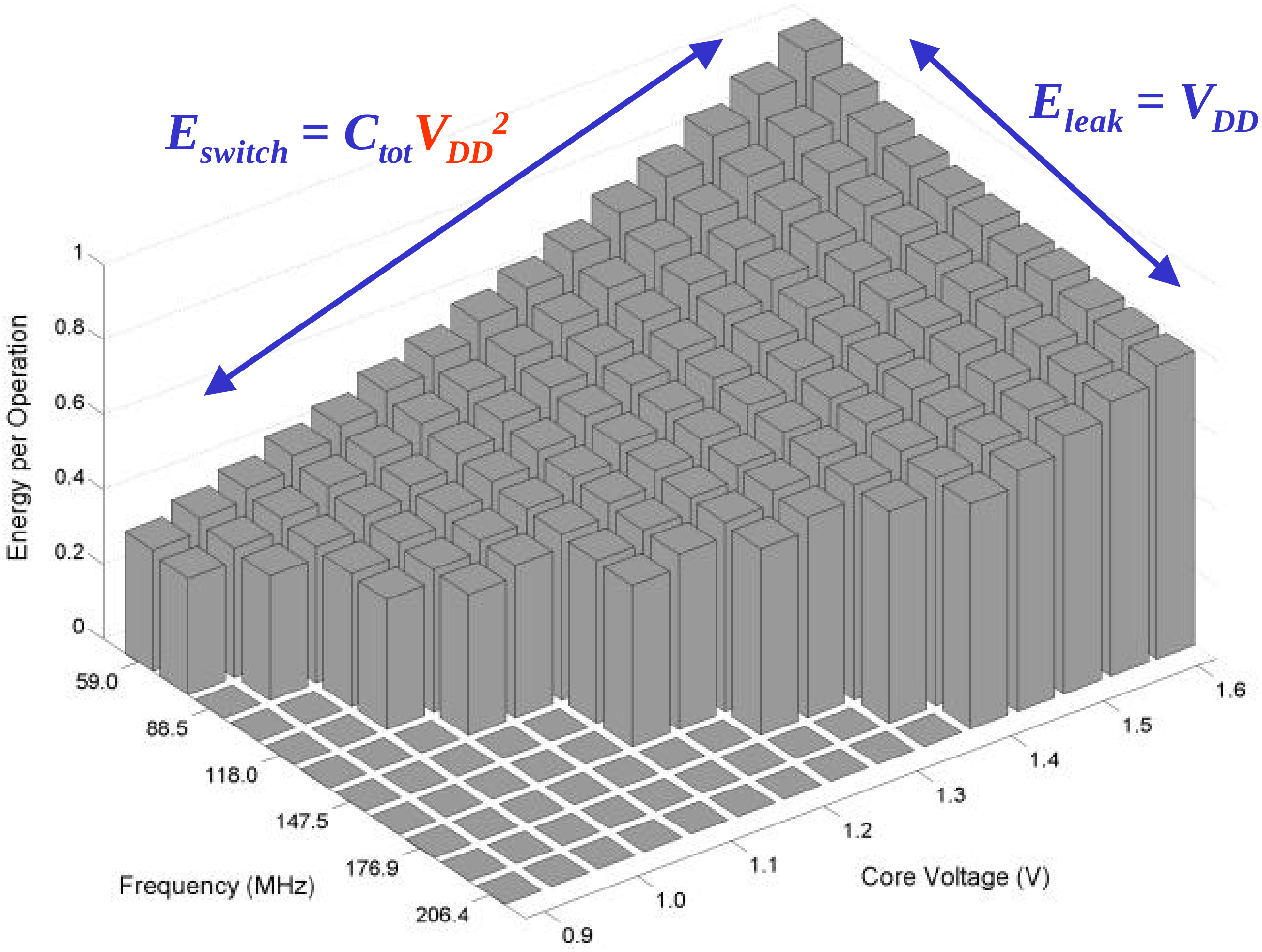
- Goal: Approach the optimal schedule
  - Meet real-time deadlines with minimum required clock frequency and voltage
- Method: Interval-based prediction
  - Assign voltage/frequency based on prior intervals of activity
  - Assumes locality/correlation in workload
  - Straightforward to implement
- Future: Thread-based analysis
  - Characterize energy usage of each thread
  - Accept “hints” from threads

Results from the DVS Test System



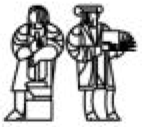
- Typical transient is 100µs and overdamped; StrongARM PLL requires 150µs to re-lock
- µOS adjusts clock frequency simultaneously with voltage
- Energy measurements include regulator and SA-1100 core

Energy Characterization of System



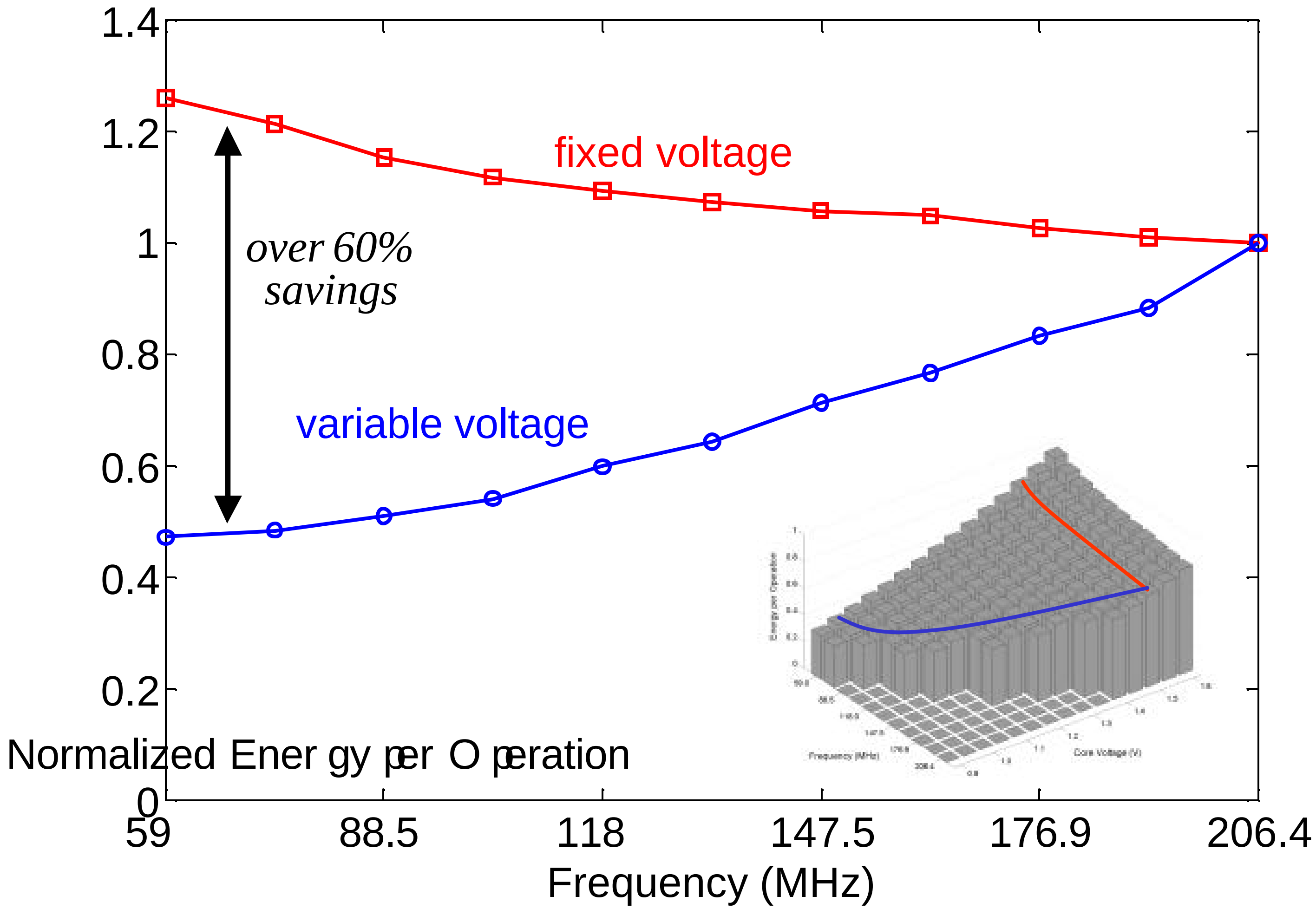
$E_{switch} = C_{tot} V_{DD}^2$

$E_{leak} = V_{DD} (I_0 e^{\frac{V_{DD}}{nV_T}}) t$



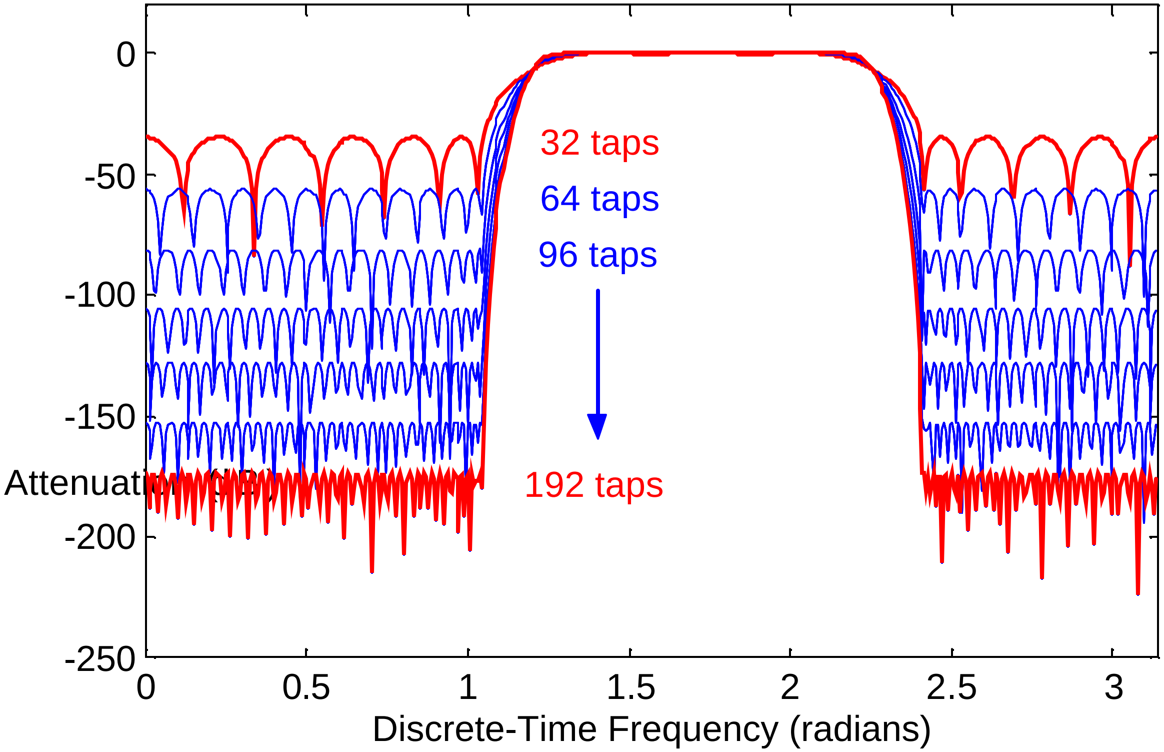
Energy-Quality Tradeoff

Computational latency



Energy-Quality Tradeoff

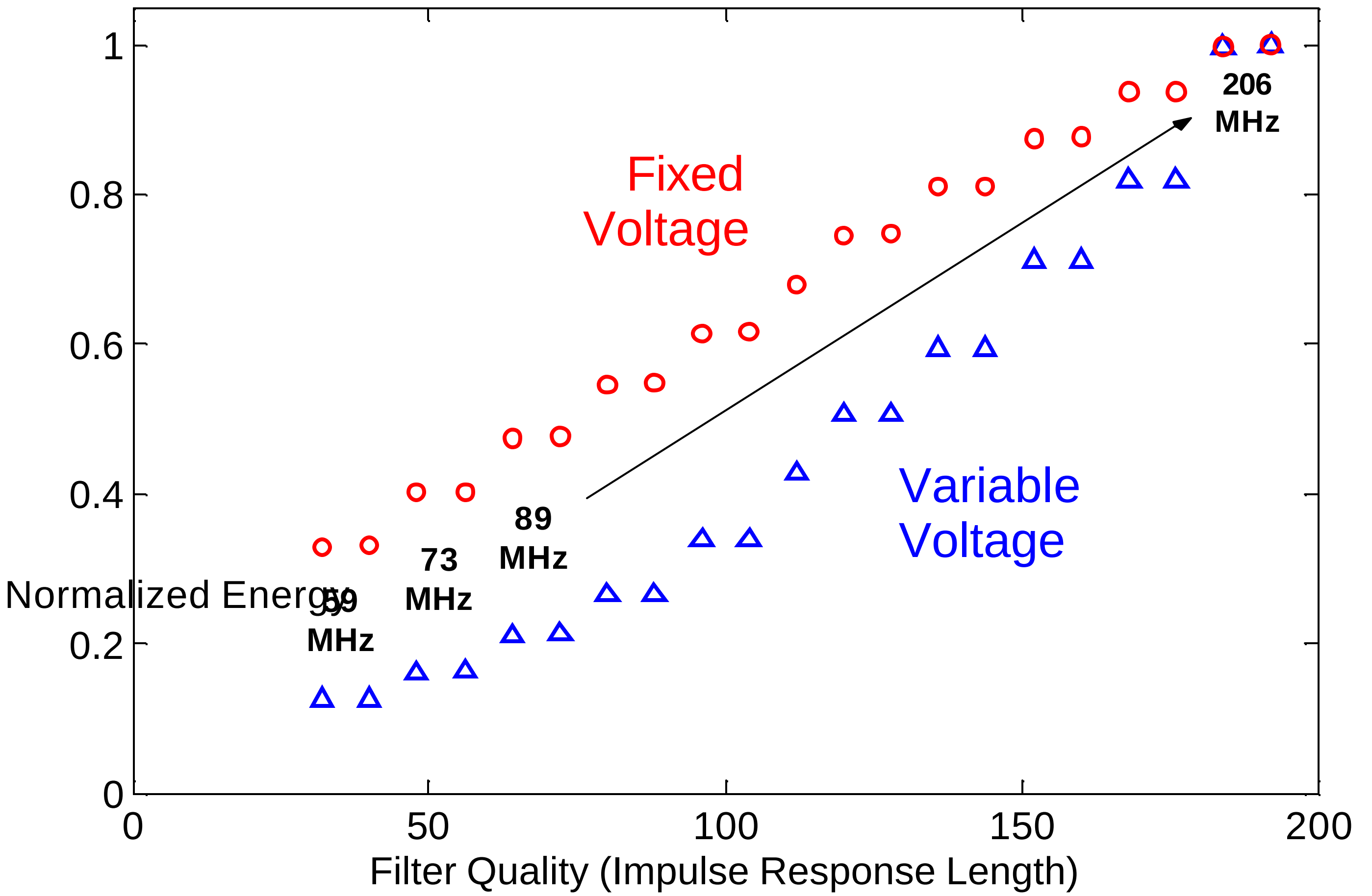
Digital filter



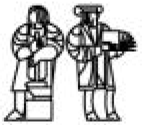
Computational energy (operations in convolution)
is linear with number of taps

Energy-Quality Tradeoff

Digital filter



Clock frequencies are chosen dynamically by mOS in response to load



Conclusions

- Distributed microsensor networks demand *power-aware* algorithms and protocols for long system lifetime
- Dynamic voltage scaling (DVS) enables *energy-quality scaling*, for adaptive energy reduction
- Energy savings from DVS have been realized on a system constructed with commercial, off-the-shelf components
- DVS will be a key enabling technology for the µAMPS distributed microsensor network

We gratefully acknowledge DARPA for funding this research and Cygnus Solutions (now Redhat) for providing the eCos microkernel