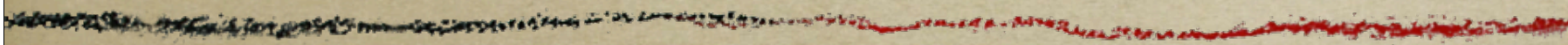


# Embedded Sensor Architecture



Philip Levis  
Stanford EE282  
Fall 2005

# What is Architecture?

---

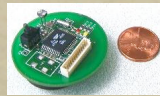
- OED: “The conceptual structure and logical organization of a computer or computer-based system.”
  - Hardware architecture, network architecture, software architecture...
- Hardware component/interface standardization
  - Researching within those parameters (declarativity)
  - Huge markets have well understood metrics

# Hardware Architecture

---

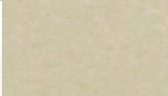
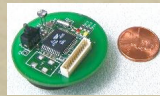
- Computer architecture
  - Processors, buses, clocks, peripherals
- Processor architecture
  - Execution model, data model, memory model
- Hardware architectural {hetero,homo}geneity
  - High performance: x86+PCI+USB+IDE/SCSI
  - Embedded: MSP/avr/ARM/PIC/8051 + ....
  - Price point, complexity, quantity
- Current sensor platforms are mostly COTS

# With What Resources?



Mote Type	WeC	René	René2	Dot	Mica	Mica2Dot	Mica 2	Telos
Year	1998	1999	2000	2000	2001	2002	2002	2004
<b>Microcontroller</b>								
Type	AT90LS8535		ATmega163		ATmega128		TI MSP430	
Program memory (KB)	8		16		128		48	
RAM (KB)	0.5	<b>4-10K RAM</b>			4		10	
Active Power (mW)	15				8	33	3	
Sleep Power ( $\mu$ W)	45	75	75	15				
Wakeup Time ( $\mu$ s)	1000	36	180	180	6			
<b>Nonvolatile storage</b>								
Chip	24LC256			AT45DB041B		ST M25P80		
Connection type	I <sup>2</sup> C			SPI		SPI		
Size (KB)	32			512		1024		
<b>Communication</b>								
Radio	<b>40-256 Kbps</b>			TR1000	CC1000	CC2420		
Data rate (kbps)				40	38.4	250		
Modulation type	OOK			ASK	FSK	O-QPSK		
Receive Power (mW)	9			12	29	38		
Transmit Power at 0dBm (mW)	36			36	42	35		
<b>Power Consumption</b>								
Minimum Operation (V)	2.7		2.7		2.7		1.8	
Total Active Power (mW)	24			27	44	89	41	
<b>Programming and Sensor Interface</b>								
Expansion	none	51-pin	51-pin	none	51-pin	19-pin	51-pin	16-pin
Communication	IEEE 1284 (programming) and RS232 (requires additional hardware)						USB	
Integrated Sensors	no	no	no	yes	no	no	no	yes

# Why So Little?

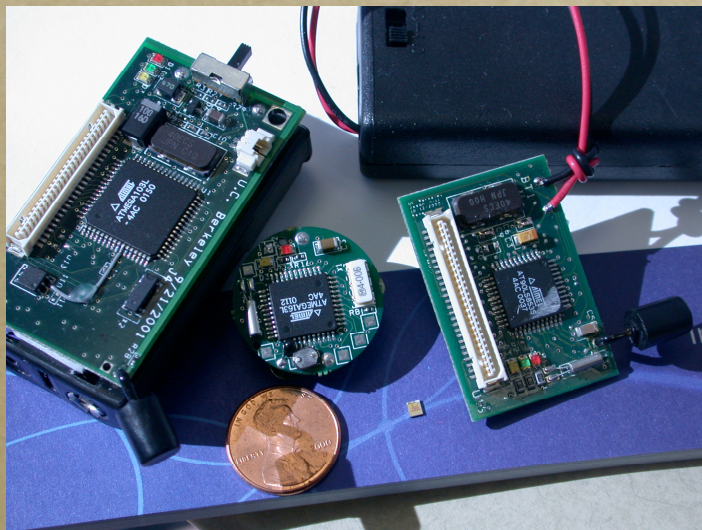
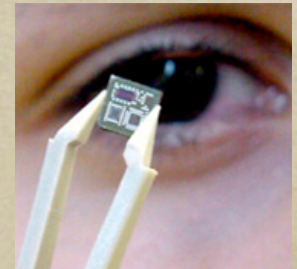


Mote Type	WeC	René	René2	Dot	Mica	Mica2Dot	Mica 2	Telos
Year	1998	1999	2000	2000	2001	2002	2002	2004
Microcontroller								
Type	AT90LS8535		ATmega163		ATmega128		TI MSP430	
Program memory (KB)	8		16		128		48	
RAM (KB)	0.5		1		4		10	
Active Power (mW)	15		15		8		33	
Sleep Power ( $\mu$ W)	45		45		75		75	
Wakeup Time ( $\mu$ s)	1000		36		180		180	
Nonvolatile storage								
Chip					AT45DB041B		ST M25P80	
Connection type					SPI		SPI	
Size (KB)					512		1024	
Communication								
Radio	TR1000		TR1000		CC1000		CC2420	
Data rate (kbps)	10		40		38.4		250	
Modulation type	OOK		ASK		FSK		O-QPSK	
Receive Power (mW)	9		12		29		38	
Transmit Power at 0dBm (mW)	36		36		42		35	
Power Consumption								
Minimum Operation (V)	2.7		2.7		2.7		1.8	
Total Active Power (mW)	24		27		44		89	
Programming and Sensor Interface								
Expansion	none	51-pin	51-pin	none	51-pin	19-pin	51-pin	16-pin
Communication	IEEE 1284 (programming) and RS232 (requires additional hardware)						USB	
Integrated Sensors	no	no	no	yes	no	no	no	yes

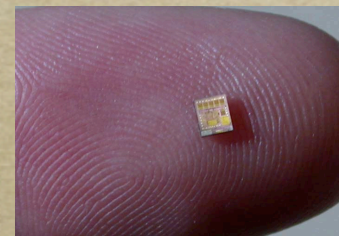
**Power**

# Energy Defines Everything

- Embedment and scale preclude wires
  - Wires are expensive to install and maintain
  - Energy reserves define form factor

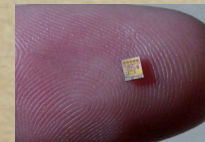


=



# A Great New World

- Performance/energy tradeoff is a principal goal
  - But performance has a different meaning
  - Workload-driven (not maximization)
- Heterogeneity of design space allows new ideas
  - New silicon is \$?0k, not \$?m
  - The spec platform
- Disparity between COTS and possibilities
  - COTS processors driven by large markets



# Outline

---

- A Great New World
- Where the Energy Goes
- A Survey of Proposals
- Looking Forward

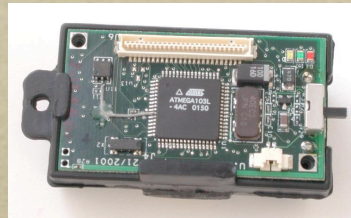
# Energy vs. Power

---

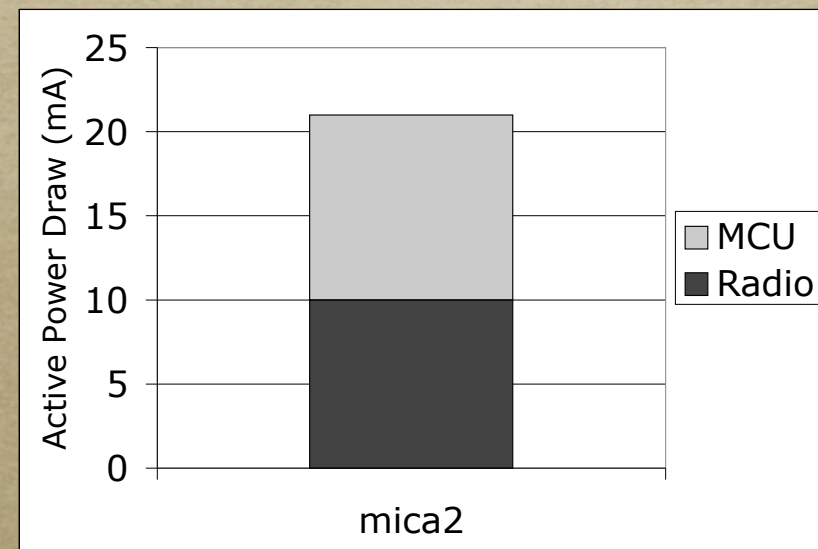
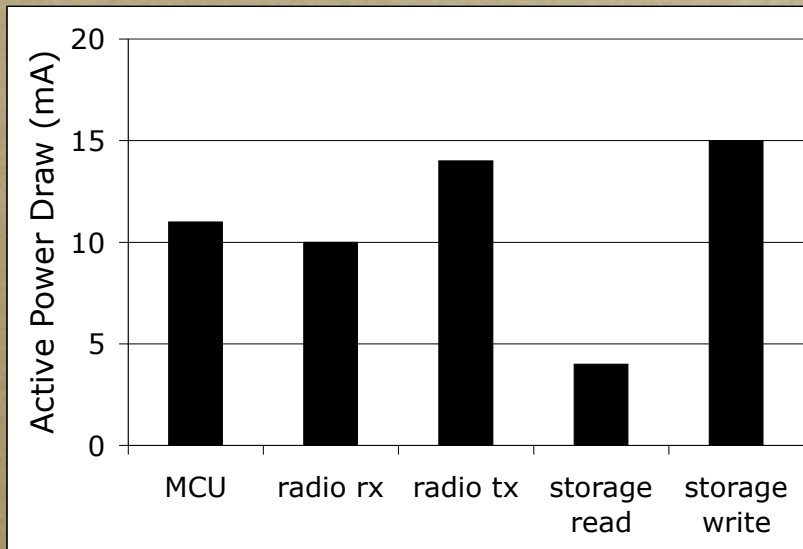
- Power is current draw \* voltage (W)
- Energy is power \* time (J)
- But power can be misleading
  - Regulating the voltage can be expensive
- And so can energy
  - Irregular battery behavior
- What often matters is the current profile.

# Where the mica2 Energy Goes

Active	20-25mA
Idle	13-18mA
Idle, radio off	3mA
Power-down	110µA

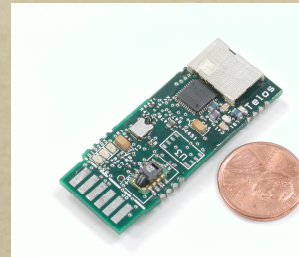


Module Type	Wec 1998	Rem1 1999	Rem2 2000	Dot 2000	Mica 2001	Mica2Dot 2002	Mica 3 2002	Yelow 2004
Microcontroller								
Type	AT90LS8535	Atmega163			Atmega128			HT MSP430
Program memory (KB)	8	16			128			48
RAM (KB)	0.5	1			4			10
Active Power (mW)	15	15			8		33	3
Sleep Power (µW)	45	45			75		75	15
Wakeup Time (µs)	1000	36			180		180	6
Nonvolatile storage								
Chip	24LC256				AT45DB04B			ST M25P80
Connection type	I <sup>2</sup> C				SPI			SPI
Size (KB)	32				512			1024
Communication								
Radio	TR1000				TR1000		CC1000	CC2420
Data rate (kbps)	10				40		38.4	250
Modulation type	OOK				ASK		FSK	O-QPSK
Receive Power (mW)	9				12		29	38
Transmit Power at 0dBm (mW)	36				36		42	35
Power Consumption								
Minimum Operation (V)	2.7	2.7			2.7			1.8
Total Active Power (mW)	24				27		44	59
Programming and Sensor Interface								
Expansion	none	51-pin	51-pin	none	51-pin	19-pin	51-pin	16-pin
Communication		IEEE 1284 (programming) and RS232 (requires additional hardware)						USB
Integrated Sensors	no	no	no	yes	no	no	no	yes

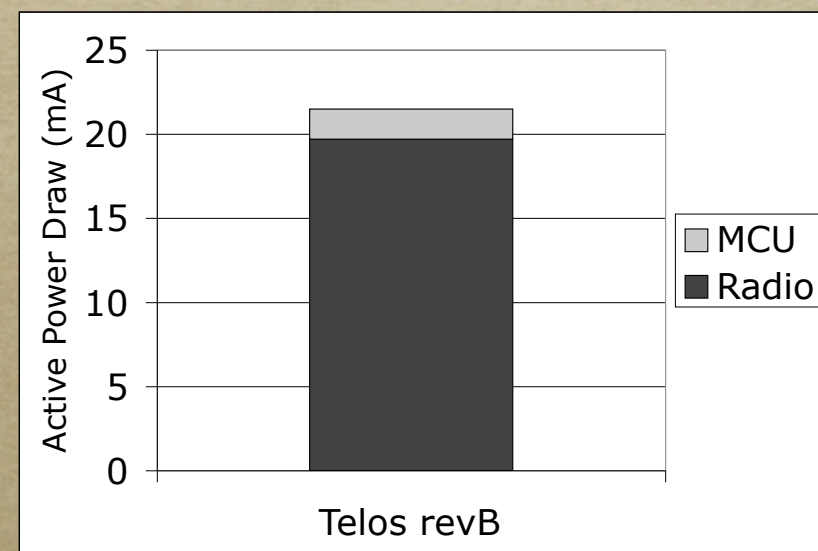
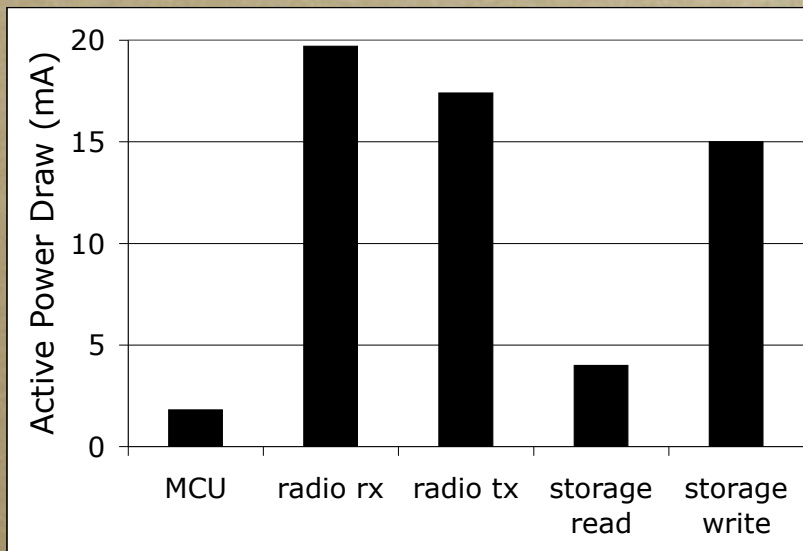


# Where the Telos Energy Goes

Active	18-21mA
Idle	17-20mA
Idle, radio off	50 $\mu$ A
Power-down	10 $\mu$ A



Mote Type	Wec 1998	Rem1 1999	Rem2 2000	Dot 2000	Mica 2001	Mica2Dot 2002	Mica 2 2002	Telos 2004
<b>Microcontroller</b>								
Type	AT90LS8535		Atmega163		Atmega128		TI MSP430	
Program memory (KB)	8		16		128		48	
RAM (KB)	0.5		1		4		10	
Active Power (mW)	15		15		8		33	
Sleep Power ( $\mu$ W)	45		45		75		75	
Wakeup Time ( $\mu$ s)	1000		36		180		180	
<b>Nonvolatile storage</b>								
Chip	24LC256		AT45DB04B		ST M25P80		ST M25P80	
Connection type	I <sup>2</sup> C				SPI		SPI	
Size (KB)	32		512		512		1024	
<b>Communication</b>								
Radio	TR1000		TR1000		CC1000		CC2420	
Data rate (kbps)	10		40		38.4		250	
Modulation type	OOK		ASK		FSK		O-QPSK	
Receive Power (mW)	9		12		29		38	
Transmit Power at 0dBm (mW)	36		36		42		35	
<b>Power Consumption</b>								
Minimum Operation (V)	2.7		2.7		2.7		1.8	
Total Active Power (mW)	24		27		44		89	
<b>Programming and Sensor Interface</b>								
Expansion	none	51-pin	51-pin	none	51-pin	19-pin	51-pin	16-pin
Communication	IEEE 1284 (programming) and RS232 (requires additional hardware)						USB	
Integrated Sensors	no	no	no	yes	no	no	no	yes



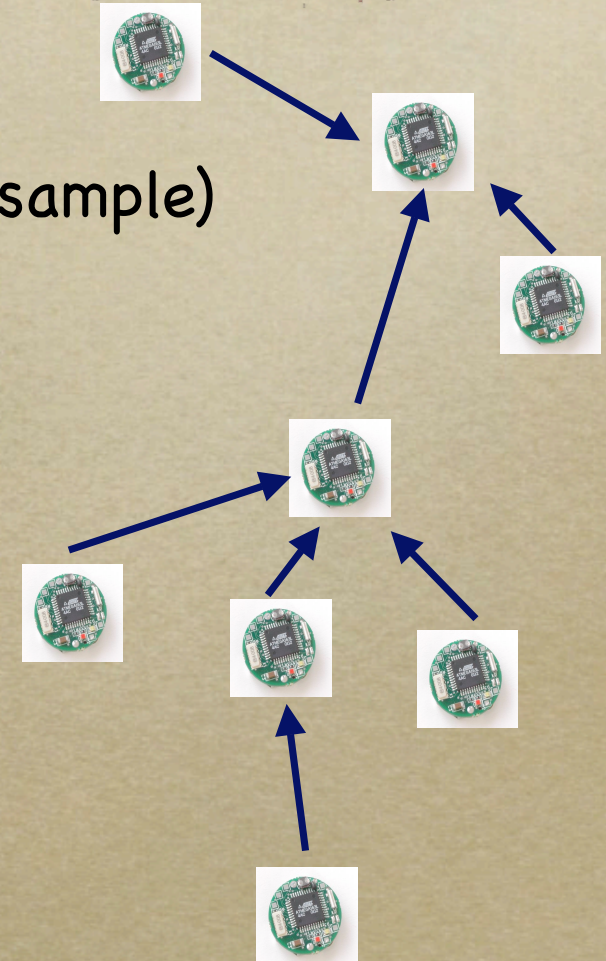
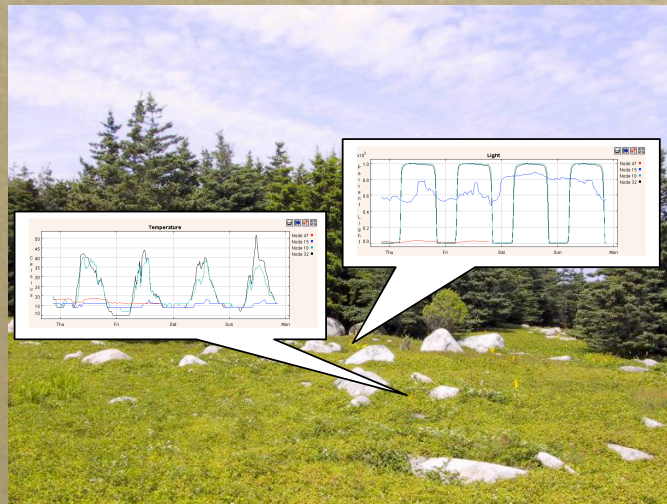
# Lifetime

- 2AA batteries  $\approx$  2700 mAh
- To last a year, average draw must be  $\sim 2\text{-}300\mu\text{A}$
- Radio is principal cost

Platform	Draw	Lifetime
mica2 active	20mA	5.5 days
mica2 idle	13mA	8 days
mica2 power-down	110 $\mu$ A	$\sim$ 3 years
Telos active	18mA	6 days
Telos idle	17mA	6 days
Telos power-down	10 $\mu$ A	$\sim$ 30 years

# Workload: Habitat Monitoring

- Low, periodic data rate (minutes/sample)
- Ad-hoc collection tree
- Lasts months-year



# Three Steps to a Long Life

---

- Sleep: very low duty cycle, wake up rarely
  - Maximize time in power-down
- Wakeup: when you do wake up, do so quickly
  - Minimize transition times
- Active: minimize work
  - No idle activity (e.g., listening)

# Sleeping

- Node must spend almost all of its time asleep
- To last 1 year:

Difference between 10 $\mu$ A and 100 $\mu$ A



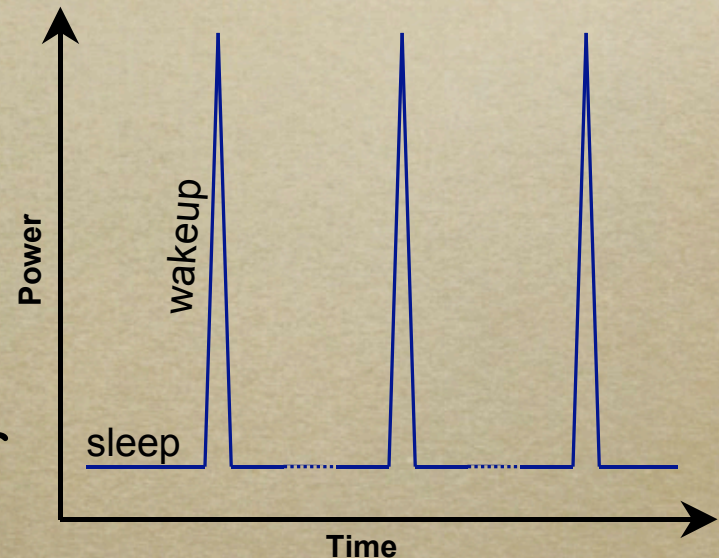
Platform	Duty Cycle	Awake	Sleep	Active
mica2	1%	15 min/day	35%	65%
telos	1.6%	22 min/day	3%	97%

- To last 2 years:

Platform	Duty Cycle	Awake	Sleep	Active
mica2	0.2%	3 min/day	87%	13%
telos	0.8%	11 min/day	6%	94%

# Sleeping, Continued

- Minimize sleep current by turning off all non-essential circuits
  - A single low-speed oscillator sounds about right
- Drive peripheral operations through low-speed oscillator
  - Turn off core whenever possible



# Waking Up

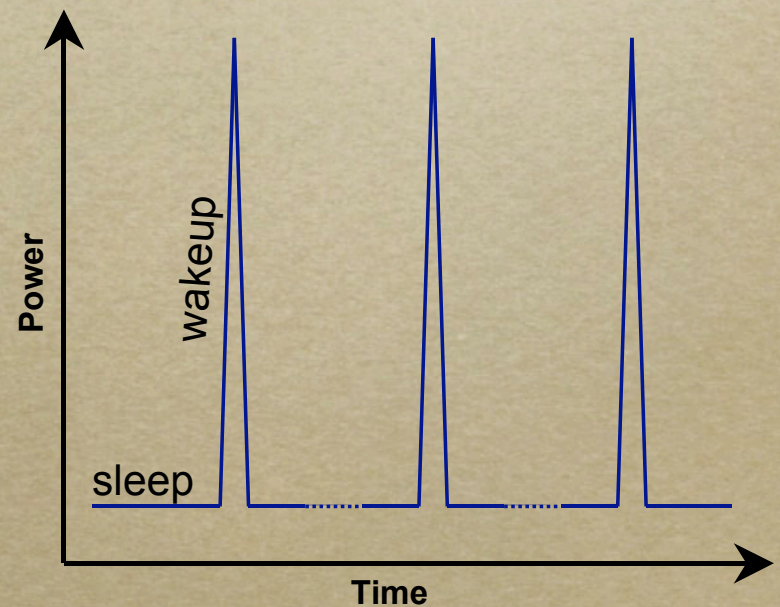
- Wake up latencies are wasted energy
- First step: wake up MCU
  - mica2 (atmega128):  $180\mu s$
  - Telos (MSP430):  $6\mu s$
- Second step: wake up radio
  - mica2 (CC1000): 2ms
  - Telos (CC2420):  $600\mu s$
- Constant costs that determine how many times the node can wake up (more on that later)

# Active Power

- Get work done quickly
  - Tradeoff between cost of resources and their power
  - Powerful CPUs have high sleep costs
- But what work to do?
  - Transmitting is easy... what about reception?

# Low Power Reception

- Scheduling: I know when someone will send to me, I'll wake up then
- Low power listening (LPL)
  - I don't know when someone will send to me, I'll wake up periodically and check
  - How long does it take to wake up and check?
  - How often do you do so?



# Transmit/Receive Tradeoff

---

- Longer LPL packets have lower receive cost
- Shorter LPL packets have lower transmit cost
- LPL wakeups are the common wakeups

# Where the Energy Goes

(mica2 numbers)

Application	CPU idle	CPU active
Beacon	35.86	0.58
Blink	742.50	0.25
BlinkTask	742.50	0.27
CntToLeds	743.72	0.57
CntToLedsAndRfm	741.90	1.61
CntToRfm	741.90	1.54
Oscilloscope	742.65	1.46
OscilloscopeRF	741.90	1.85
Sense	742.21	0.38
SenseLightToLog	741.90	0.81
SenseTask	742.21	0.42
SenseToLeds	743.72	0.73
SenseToRfm	741.90	1.77
TinyDB (idle)	693.29	10.41
TinyDBApp (select)	742.85	11.20
Surge	727.28	1.50

CPU is active 0.01% - 0.4%  
of the time for app

(Note: paper clumps radio  
interrupt handling with radio)

- "Simulating the Power Consumption of Large-Scale Sensor Network Applications," Shnayder et al., SenSys 2004

# Outline

---

- A Great New World
- Where the Energy Goes
- A Survey of Proposals
- Looking Forward

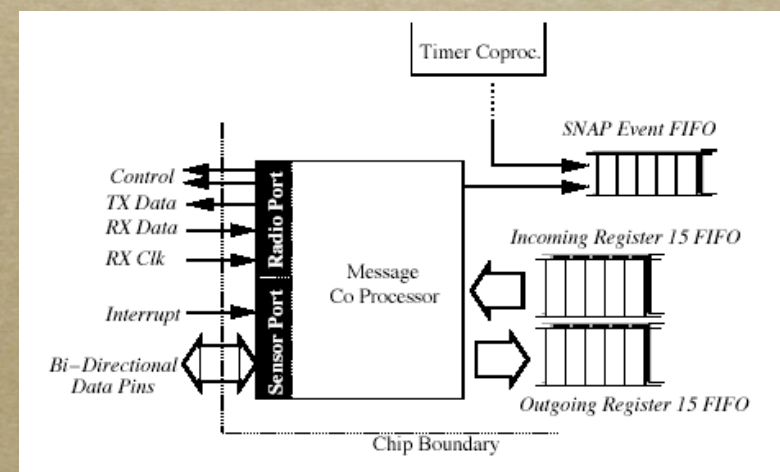
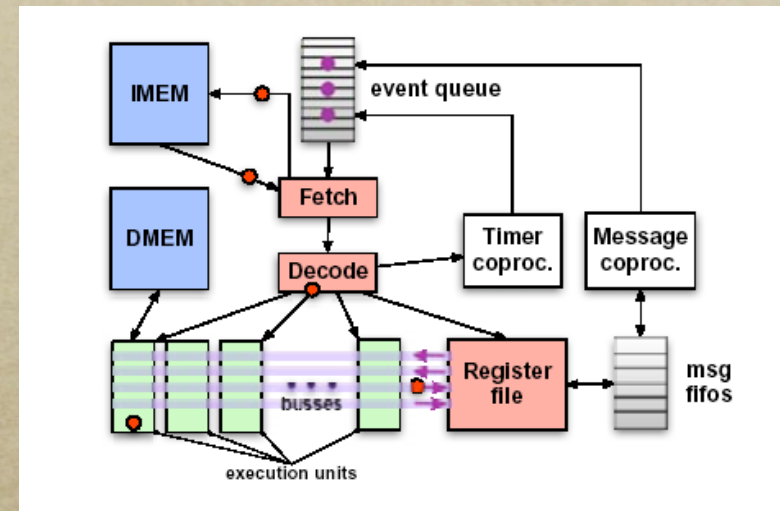
# Proposed Architectures

---

- "An Ultra Low-Power Processor for Sensor Networks," Ekanayake et al., ASPLOS 2004
  - Event-driven processor (read: atomic interrupts)
  - Asynchronous logic
- "An Ultra Low Power System Architecture for Sensor Networks," Hempstead et al. ISCA 2005
  - Architecture for habitat monitoring
  - Interconnect with minimal MCU operation
- Both driven by habitat monitoring

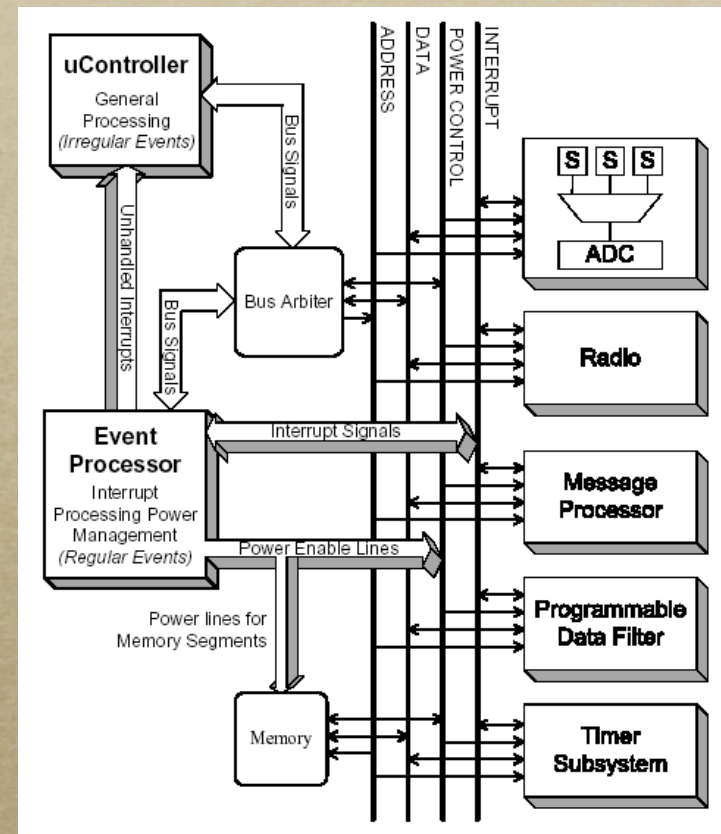
# A Processor

- FIFO event queue
- Message coprocessor
- Optimized ISA
- 150–550 nW @ 1.8V



# An Architecture

- Data interconnect
- MCU runs rarely
  - Regular events
  - Irregular events
- Optimized ISA
- Hardware network stack
- $25 \mu\text{W}$  @ 1.2V



# Missing the Boat

---

- It's easy to optimize a processor for a specific application
  - Silicon gives 1-2 orders of magnitude energy savings
- The issue is that there are *MANY* applications
  - Hardware vs. software timers
  - Timer fidelity, width, wake-up cycle
- Low voltage is great, but voltage regulators can actually be expensive...

# Outline

---

- A Great New World
- Where the Energy Goes
- A Survey of Proposals
- Looking Forward

# Looking Forward

---

- OED: “The conceptual structure and logical organization of a computer or computer-based system.”
- Sensor nodes are truly about architectures: there’s much more to them than a chip
  - Issues like port leakage are critical
- Current proposals have a very narrow view of application space, preclude useful programming
  - No pre-emption is a non-starter

# Further Reading

---

- "Processor Choice for Wireless Sensor Networks," Ciarán Lynch et al. (great, simple COTS intro with no sales pitch)
- "An Ultra Low-Power Processor for Sensor Networks," Ekanayake et al., ASPLOS 2004
- "An Ultra Low Power System Architecture for Sensor Networks," Hempstead et al. ISCA 2005
- "Telos: Enabling Ultra-Low Power Wireless Research," Polastre et al., IPSN/SPOTS 2004
- "Simulating the Power Consumption of Large-Scale Sensor Network Applications," Shnayder et al., SenSys 2004
  - (but look at the numbers carefully)

# Questions

---

- Will be answered with ultra-ultra-ULTRA low power!