#### **Electronics for IoT**

# Hardware for IoT

#### Bernhard E. Boser University of California, Berkeley <u>boser@eecs.berkeley.edu</u>



## **IoT Nodes**

- Cloud
- Processors
- Wireless connectivity



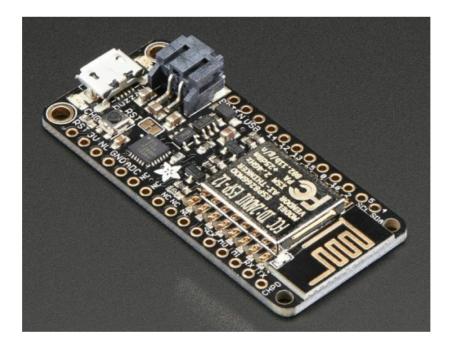
# Cloud



#### Ref: J. Rabaey, UC Berkeley Swarm Lab

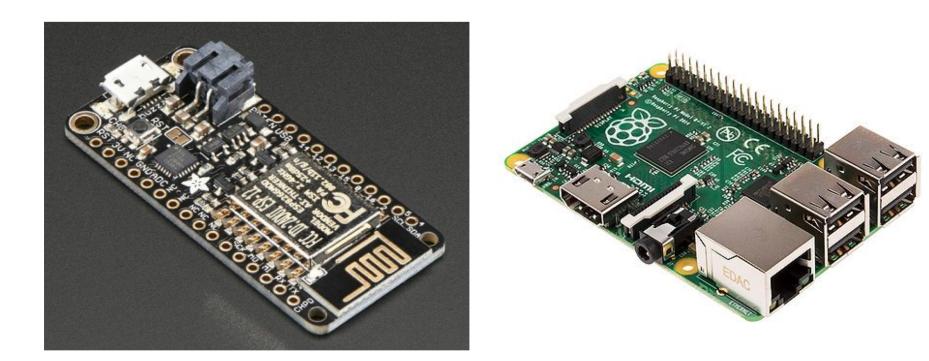
#### Processors







#### **Processors II**



Memory:

Clock rate:

Throughput:

B. E. Boser

# **Power Dissipation**

#### ESP32

• <1 ... 200 mA

#### Rpi 3B+

• 350 ... 980 mA

https://www.pidramble.com/wiki/benchmarks/power-consumption

1 Ah battery (assume same supply voltage)

• 5 hours ... days • 1 ... 3 hours



# **CPU Power Dissipation**

- Computation only
  - no wireless,
  - no peripherals
- $P_{CPU} = P_{static} + P_{dyn}$

• 
$$P_{CPU} = P_{static} + \frac{f_{clk} \times \frac{P_{dyn}}{f_{clk}}}{\frac{P_{dyn}}{P_{dyn}}}$$

•  $\rightarrow$  Dynamically adjust  $f_{clk}$  to save power

**Power Efficiency:** 

P<sub>dyn</sub>/<sub>fclk</sub>

- Examples (trend: down)
  - ESP32
  - STM32F4
  - RISC-V
  - High performance Intel i7
  - E.g. (STM32F4)

~300 μW/MHz 240 μW/MHz 34 μW/MHz ... 5 mW/MHz 10 MHz → 2.4 mW 100 MHz → 24 mW

- Low power:
  - Dynamically vary clock rate
  - (Deep) sleep when not in use
    - Many modes (e.g. RAM retention, peripherals, ...)

# Speed

- Maximum clock rate (e.g.)
  - Laptop, data center CPU
  - Rpi
  - Microcontroller

4 GHz 1.2 GHz 240 MHz



# Speed II





# **Interrupt Latency**

• Time from sensor detecting an obstacle to program reacting



# Laptop (or Datacenter CPU) Latency

- ms ... seconds (rarely)
- Mostly ms, but
  - Occasional "freeze-up"
  - Annoying ... to fatal
  - Do not use a laptop to control a quadcopter
- Why is this?



#### **Multitasking**

Θ	0 *	~	CPU	Memory	Energy	Disk	Netw	ork		Q Search
oces	s Name		% CPU ~	CPU Time	Threads	Idle Wake	Ups	PID	User	
æ	Dropbox		77.5	11:51:23.81	131		0	19842	boser	
×	backupd		11.8	17:12.28	6		28	36800	root	
	kernel_task		9.7	7:24:31.21	157		2,568	0	root	
	WindowServe	ər	6.1	1:58:51.31	9		156	223	_windowserv	er
0	Google Chron	ne	3.4	23:55.50	38		62	74083	boser	
	launchd		3.0	1:31:33.89	4		0	1	root	
4	Activity Moni	tor	2.7	4.96	5		4	38412	boser	
	Box Sync Mo	nitor	2.0	8:00:36.33	1		9	996	boser	
	airportd		1.8	22:42.02	12		0	225	root	
	coreservices	d	1.3	3:00.42	4		0	129	root	
1	Mailplane 3		1.2	1:27:32.46	16		1	74133	boser	
	locationd		1.0	32:11.82	8		3	104	_locationd	
	mds		0.8	1:08:33.06	7		22	85	root	
4	Kite Engine		0.7	59:41.45	18		642	7431	boser	
	sysmond		0.7	1:27.06	3		1	230	root	
	deleted		0.6	12:40.71	6		0	466	boser	
	Microsoft Por	werPoint	0.5	16:24.54	16		12	72360	boser	
	hidd		0.5	7:35.65	5		0	118	_hidd	
	Box Sync		0.5	2:46:55.91	36		132	867	boser	
	Adobe Deskt	op Service	0.4	27:39.50	21		2	936	boser	
	Finder		0.4	35:38.76	9		1		boser	
	Moom		0.4	5:07.88	3		0		boser	
	mdworker		0.3	1:22.99	4			37627		
	mDNSResponder		0.3	47:26.26	5		6		_mdnsrespor	nd
	CleanMyMac 3		0.2	13:53.10	10			37198		
	screensharin	-	0.2	1:11.32	14			33917		
	CleanMyMac		0.2	13.30	8			37201		
	opendirector		0.2	30:13.29	10		0		root	
	Google Chron	ne Helper	0.2	58.19	17			37672		
	Dock		0.2	1:28.18	4		7		boser	
	GoogleTalkPl	-	0.2	18:13.79	10			74135		
	Google Chron	ne Helper	0.1	1:31.78	17			38121		
	distnoted		0.1	7:25.65	9		0		boser	
	mds_stores			1:05:12.46	5		0		root	
	fseventsd		0.1	11:17.41	13		16		root	
	suggestd		0.1	38:26.04	8		1		boser	
	diskarbitratio	nd	0.1	2:30.37	2		0		root	
			0.1	10-15 06	2		^	275	bacar	
		System:	10.665	6	CPU LOAD		Threa	ds:	20	010
		User:	22.86	%			Proces	sses:	4	139
		Idle:	66.48	-						
			00.40.	-						

# **Parallel Processing**

- Multi-core CPU
- Pre-emptive multitasking
  - Operating system periodically (every few ms) switches to next task waiting for execution (transparent to task)
  - Varies with system load
  - Desktop OS'es: Windows, OSX, Linux
- Co-operative multitasking
  - Tasks relinquish CPU when done
  - Interrupts for "immediate attention" (< 10  $\mu$ s)
- Variants, e.g. "real time" operating systems
  ESP32



# Determinism

- Real time applications:
  - 1. Require "guaranteed" response time (e.g. driverless car)
  - 2. In addition to throughput (e.g. image scene analysis)
- Operating systems generally do not satisfy (1)
- Solution:
  - No operating system (aka "bare metal")
  - Application program as full control
    - Cooperative multitasking
    - Schedule tasks that are not time critical when appropriate
      - E.g. check for MQTT subscriptions between time critical events
    - BEWARE of
      - Service interrupt handlers (e.g. WiFi)  $\rightarrow$  dedicated hardware?
      - Resource hogs (bugs)
      - "pre-emptive"

# **Hierarchical Solutions for Real Time Control**

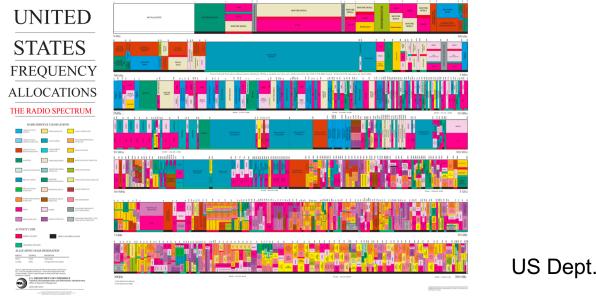
- Small dedicated processor(s) for real time control
  - E.g. ESP32, ARM, ...
  - No OS (real time OS in some cases)
- Middle level for more complex tasks, more compute power (e.g. image analysis)
  - E.g. Rpi, Beagle Bone, ...
- Application processor, Cloud for number crunching, data storage, ...
  - Resource intensive
  - Not real time
- Tends to also optimize power dissipation
  - Adapted e.g. in smartphones

# (Wireless) Connectivity

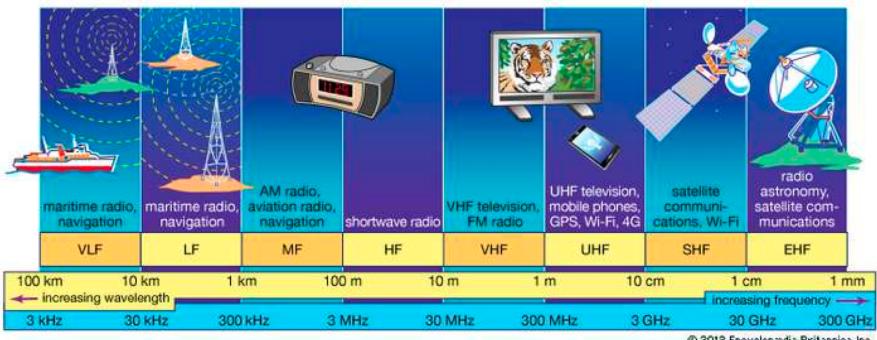
- Spectrum
- Standards
- Performance
  - Range
  - Throughput
  - Power dissipation
  - Cost
- Complex optimization
  - Fragmentation
  - Interoperability

# **RF Spectrum**

- Electromagnetic waves
- Characterized by frequency (wavelength)
- Finite resource
- Carefully controlled by governments → licensing



# **RF Spectrum Properties**



© 2013 Encyclopædia Britannica, Inc.

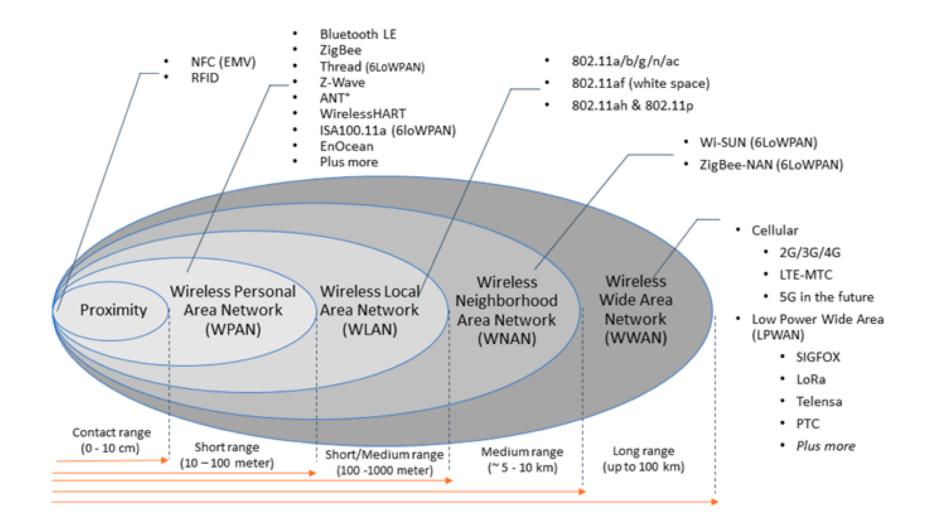
- Range
- Obstacles, line of sight
- Antenna size

Encyclopedia Britannica

# 2.4 GHz Unlicensed Band

- 2.4 GHz == frequency of operation
  - 2.4 ... 2.483 GHz
  - Not speed (bps)
- Open to all
  - Subject to conditions (e.g. max transmit power)
- Crowded
  - Wifi
  - Bluetooth
  - Zigbee
  - Microwaves (2.5GHz)
- Fixed capacity: more users, lower throughput
  - cf lab ...

# **Wireless Communication Standards**



# Example: IEEE 802.11 (WLAN)

- Ubiquitous
- Direct internet connectivity
  - Versatile
  - Difficult to manage security
- Medium
  - Range (~ 20 … 100+ meter, + roaming), "LAN"
  - Fast (Mbps+)
  - Power dissipation
    - Transmit: ~ 800mW
    - Receive: ~ 200mW
  - Very situation & metric dependent
    - Energy per bit, packet

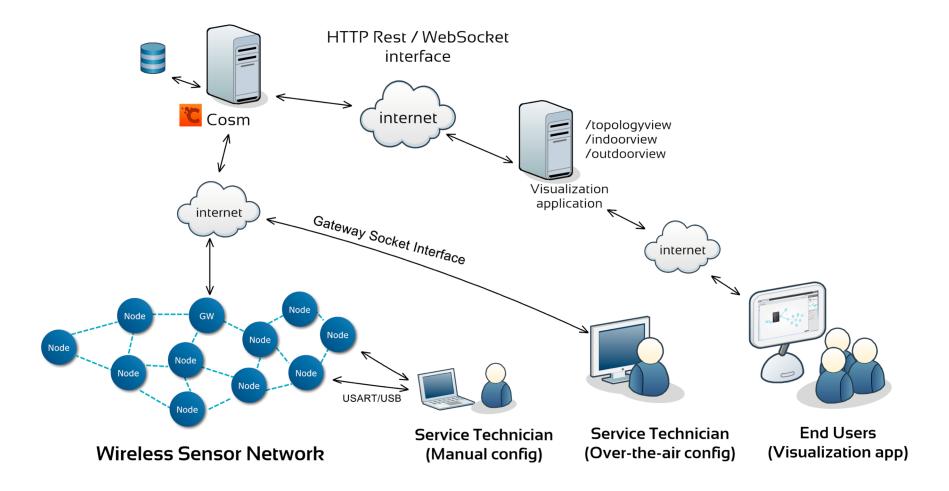
# WWAN

- Cellular
  - Ubiquitous in urban areas
  - Roaming for wide coverage
  - Licensed
  - Not optimized for small packets
    - 5G promises to fix this
  - High power dissipation (1W+)
- LPWAN
  - Low power, high range (many km)
  - Fragmented
  - LoRa, Sigfox, ...

# Low Power: 802.15

- Bluetooth, ANT, Zigbee, ...
- Low power
  - E.g. nRF52840
    - Transmit & Receive: 15mW
    - Fast power cycling
    - (Scales with TX power ...)
- Low range, throughput
  - 10 ... 50 meter
  - 100 ... 2000 kbps

## **Wireless Sensor Networks**



# **Further Power Optimization**

- Network "sleep"
  - Radio on only at scheduled times
  - Requires synchronization (good clocks)
  - Plus a whole host of problems solved
- Really soon, now ...

# Summary

- Processor
  - GPIO
  - Performance: throughput, latency
  - Power dissipation
  - Size, cost
  - A complex optimization problem
    - → many options
- (Wireless) Connectivity
  - Range, throughput, power dissipation
  - Size, cost
  - Standards
  - A complex optimization problem ...
- EE 149, ...