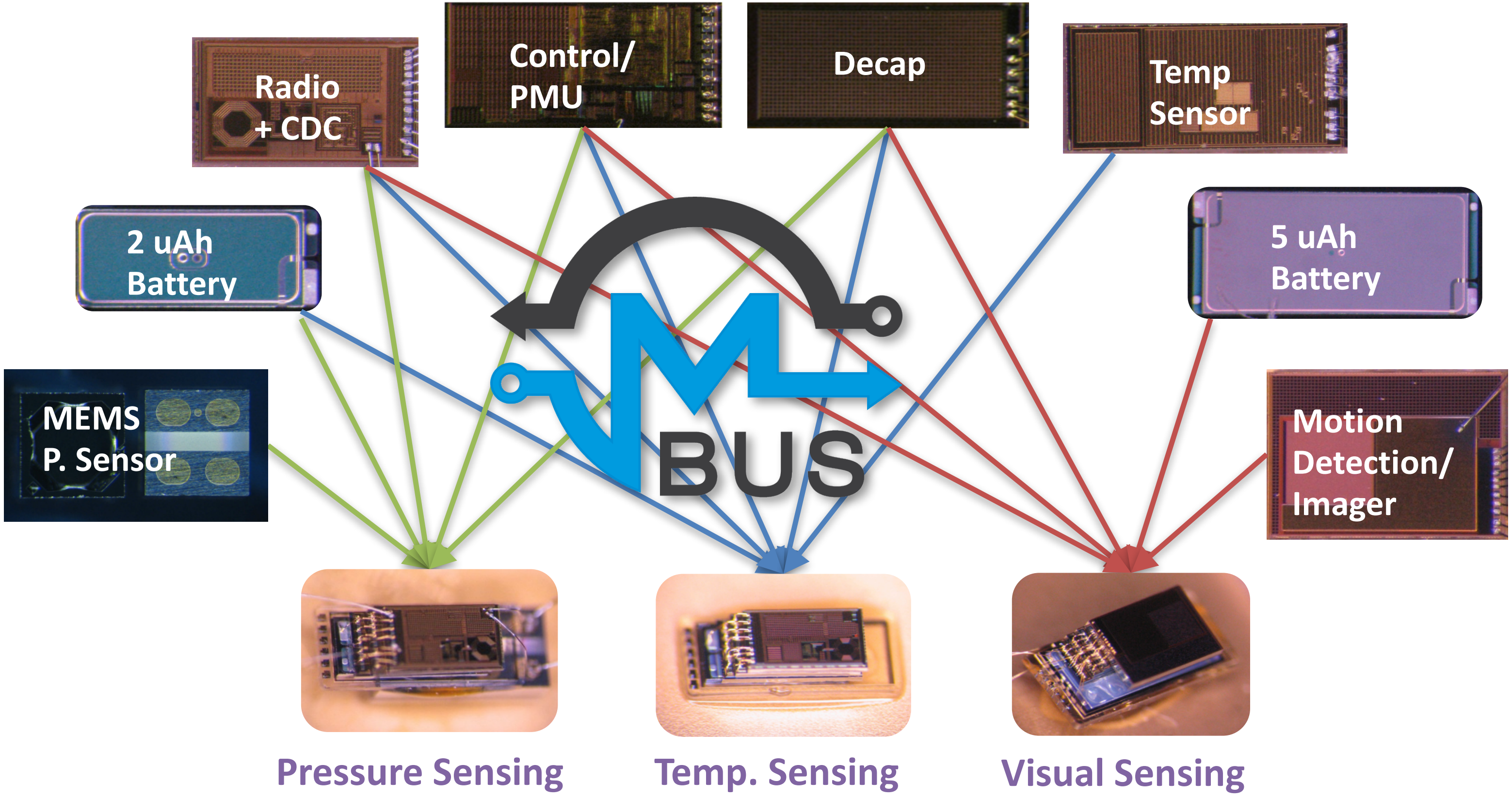
MBus: A power-aware interconnect for ultra-low power micro-scale system design

Pat Pannuto, University of Michigan
ppannuto@umich.edu

Advisor: Prabal Dutta, University of Michigan / UC Berkeley
prabal@umich.edu

PI: Dennis Sylvester, University of Michigan
dmcs@umich.edu

MBus is the interconnect for Michigan's nanopower chips

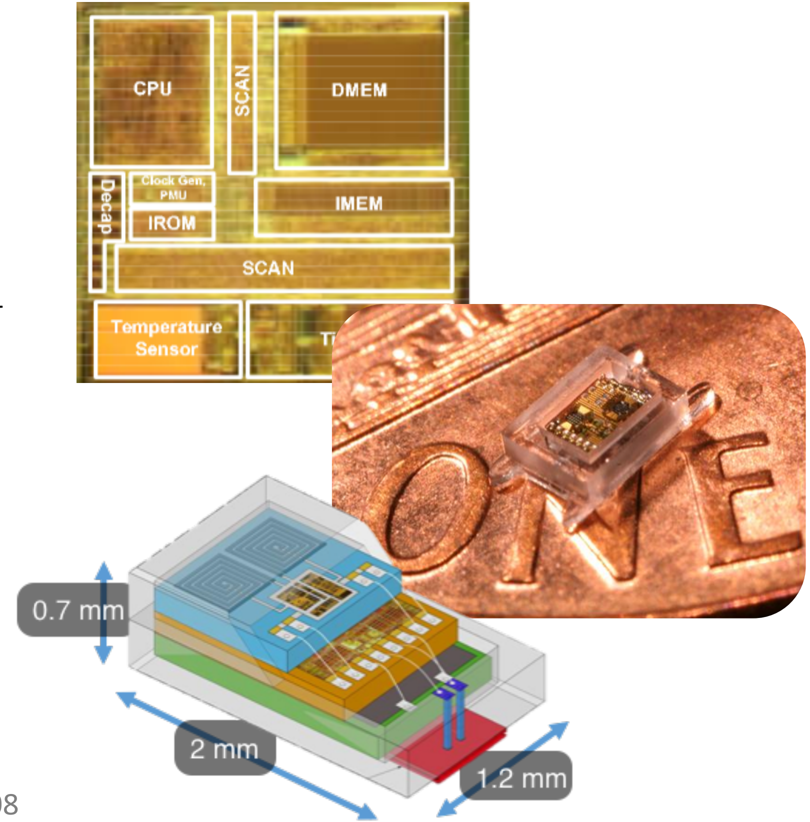


Growing ecosystems of MBus chips and systems

- Processor (PRC/PREv13)
  - ARM Cortex M0
  - 8 generations with MBus
- Radios (RADv10, SIRv2, FFRv1, MRRv1)
  - 900 MHz near field ; med range
  - ~1 GHz far field
- Flash Memory (FLSv2, FLPv1)
  - Long-term data retention
- Sensor (SNSv7)
  - Generic CDC frontend
- Energy Harvesters
  - SOLv5, HRVv4, GAPv3
- Power Management (PMUv2)
  - Power regulation, brown-out detection
- Imager (IMGv3)
  - 160x160 pixel imager with < 1 µW motion detection
- GPS Correlator (CORv2)
  - Acquire & record raw I/Q data
- N-ZERO chips?

MBus addresses a modularity need

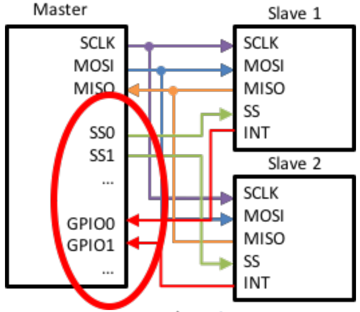
- Michigan has a well-established history of low-power circuit design
  - Next challenge: low-power *systems*
  - Phoenix '08, 30 pW temperature sensor
- Monolithic designs slowed progress
  - Intraocular Pressure '11, semi-modular design
- What would it take get reusable components?

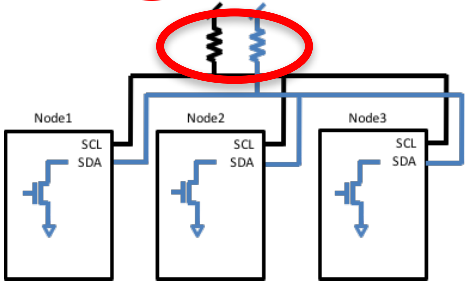


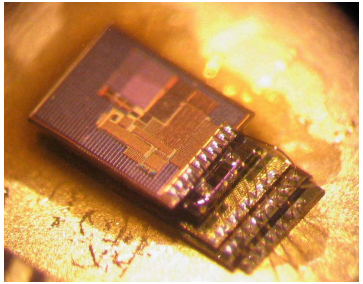
The Phoenix Processor: A 30pW Platform for Sensor Applications, M. Seok, VLSI '08
A Cubic-Millimeter Energy-Autonomous Wireless Intraocular Pressure Monitor, G. Chen, ISSCC '11

No existing embedded interconnect satisfied our needs

- SPI
  - I/O overhead: per-chip select, interrupt lines
  - Centralized architecture inefficient
- I²C
  - Pull-ups consume too much energy (~100 uW)
- **First Try: I²C variant**
  - Not easily synthesizable, energy state tracking
  - Y. Lee, S. Bang, I. Lee, Y. Kim, G. Kim, M. H. Ghaed, P. Pannuto, P. Dutta, D. Sylvester, and D. Blaauw, "A modular 1 mm^3 die-stacked sensing platform with low power I²C inter-die communication and multi-modal energy harvesting," in IEEE Journal of Solid-State Circuits, vol. 48, 2013





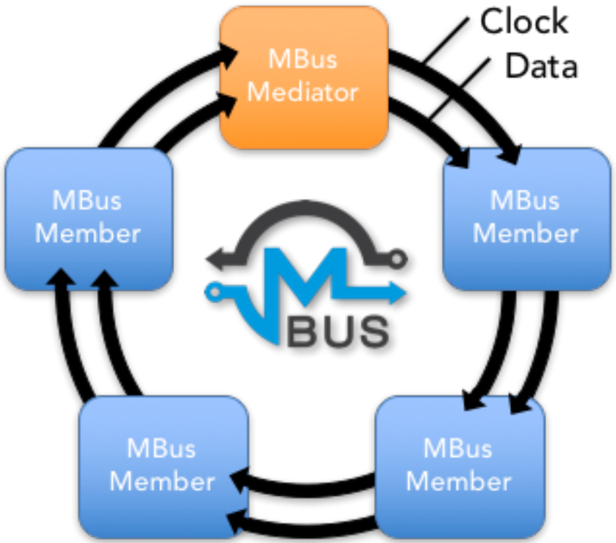


The driving goals of the MBus design

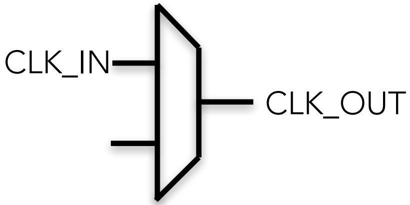
- Three things <u>really</u> motivated the team at first:
  - Power
  - Area
  - Reliability
- Clean slate design compelled rigorous feature evaluation
  - Power, area, reliability, synthesizability, scalability (address space), flexibility (multi-master / interrupt), efficient (broadcast, HW ACK)
  - System design revelation: Power-aware

MBus Overview

- FOM (meas)
  - Active: 22 pJ / bit / chip
  - < 10 pW standby / chip



- Ring Topology
- 2 lines – 4 I/O per node
  - Clock
  - Data
- Transaction oriented
  - Arbitration
  - Address Transmission
  - Data Transmission
  - Interjection
  - Control (ACK/NAK)
- "Shoot-Through"



In a distributed sensing system, automatic power management makes life much, much easier

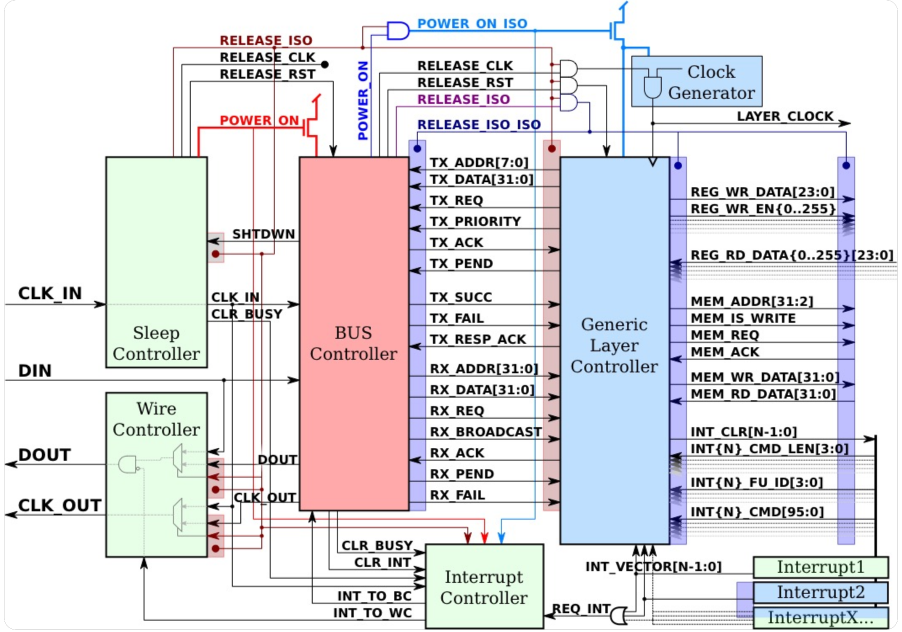
- **Managing power modes presented some of the biggest challenges**
  - Power state: which chips are on? CPU must turn peripherals on to talk to each other
  - Wakeup circuitry: custom clockless cold-boot required for each chip
- **Insight: Interconnect can handle power management**
  - Arbitration protocol puts a few clock edges on the global bus
    - 1. Node is awake and participating in arbitration
    - 2. Node is asleep, bus frontend clocks wakeup circuitry on arbitration edges and powers on having just lost arbitration
  - Hierarchical power domains make this efficient
    - Minimal always-on frontend (7 gates)
    - Bus controller listens for address, provides byte-interface to bus, and powers rest of chip only when needed

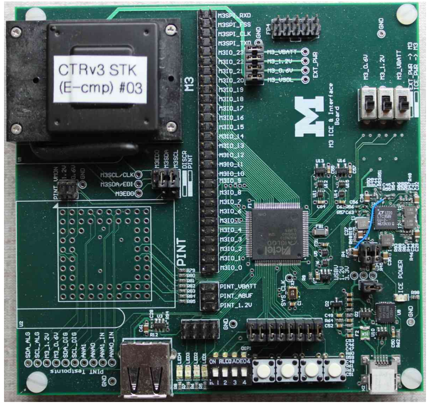
The other big design accelerant for M3 systems was to standardize the layer controller

- Common design pattern:
  - Chip state-machines triggered by register-file interface
  - Some require a small amount of configuration memory (registers)
  - Some require large actual memory (images, audio, etc)
- All M3 chips have the same logical interface: MPQ
  - Essentially a distributed DMA interface
  - Facilitates distributed state machines
    - Send configurable messages on events, very flexible / composable

Status of MBus today

- Synthesizable Verilog
  - Free, open source license
  - No process-specific parameters
    - (ratioed logic, etc)
- FPGA and big-banged MCU implementations
- Protocol Analyzer for Saleae Logic
- Python library + debug board
  - Real-time programmatic interaction, read/write MBus from a PC
- Exploring formal protocol verification





Additional Information

- Overview
  - Pat Pannuto, Yoonmyung Lee, Ye-Sheng Kuo, ZhiYoong Foo, Benjamin Kempke, Gyouho Kim, Ronald G. Dreslinski, David Blaauw, and Prabal Dutta. "MBus: A System Integration Bus for the Modular Micro-Scale Computing Class". In: *vol. 37. Micro Top Picks* 3. May 2016.
- Architectural Design and Protocol Logic
  - Pat Pannuto, Yoonmyung Lee, Ye-Sheng Kuo, ZhiYoong Foo, Benjamin Kempke, Gyouho Kim, Ronald G. Dreslinski, David Blaauw, and Prabal Dutta. "MBus: An Ultra-Low Power Interconnect Bus for Next Generation Nanopower Systems". In: *Proceedings of the 42nd International Symposium on Computer Architecture. ISCA '15*. Portland, Oregon, USA: ACM, June 2015.
- Circuit Design and Power Domains
  - Ye-Sheng Kuo, Pat Pannuto, Gyouho Kim, ZhiYoong Foo, Inhee Lee, Benjamin Kempke, Prabal Dutta, David Blaauw, and Yoonmyung Lee. "MBus: A 17.5 pJ/bit Portable Interconnect Bus for Millimeter-Scale Sensor Systems with 8 nW Standby Power". In: *CICC '14: IEEE Custom Integrated Circuits Conference*. San Jose, California, USA, Sept. 2014.
- Specification
  - http://mbus.io/spec.html
- Verilog
  - https://github.com/mbus/mbus
- Homepage
  - http://mbus.io

For more information, specification, and reference verilog:

http://mbus.io

http://github.com/mbus/mbus

MBus Team: Pat Pannuto, Yoonmyung Lee, Ye-Sheng Kuo, ZhiYoong Foo, Benjamin Kempke, David Blaauw, Prabal Dutta

PI: Dennis Sylvester, University of Michigan

dmcs@umich.edu

Backups

FOM's

	I^{2}C	SPI	UART	Lee-I^{2}C	MBus
Critical					
I/O Pads (n nodes)	**2/4**†	$3 + n$	$2 \times n$	**2/4**†	**4**
Standby Power	**Low**	**Low**	**Low**	**Low**	**Low**
Active Power	High	**Low**	**Low**	Med	**Low**
Synthesizable	**Yes**	**Yes**	**Yes**	No	**Yes**
Global Uniq Addresses	128	—	—	128	$\mathbf{2^{24}}$
Multi-Master (Interrupt)	**Yes**	No	No	**Yes**	**Yes**
Desirable					
Broadcast Messages	No	Option	No	No	**Yes**
Data-Independent	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**
Power Aware	No	No	No	No	**Yes**
Hardware ACKs	**Yes**	No	No	**Yes**	**Yes**
Bits Overhead (n bytes)	$10 + n$	**2**‡	(2-3)§ $\times\ n$	$10 + n$	**19, 43***

† When wirebonding, a shared bus requires two pads/chip (or a much larger shared pad/trace)
‡ Asserting and de-asserting the chip-select line
§ Depending on the stop condition; assumes 8-bit frames and no parity
* Depends on whether short (more common) or long addressing is in use

Module	Verilog SLOC	Gates	Flip-Flops	Area in 180 nm
Bus Controller	947	1314	207	27,376 μm^2
Optional				
Sleep Controller	130	25	4	3,150 μm^2
Wire Controller	50	7	0	882 μm^2
Interrupt Controller	58	21	3	2,646 μm^2
Total	1185	1367	214	37,200 μm^2§
Other Buses:				
SPI Master†	516	1004	229	37,068 μm^2
I^{2}C ‡	720	396	153	19,813 μm^2
Lee I^{2}C [14]	897	908	278	33,703 μm^2

§ Includes a small amount of additional integration overhead area
† SPI Master from OpenCores [32] synthesized for our 180 nm process
‡ I^{2}C Master from OpenCores [10] synthesized for our 180 nm process

		Energy per bit
Member+Mediator Node	sending	27.5 pJ/bit
Member Node	receiving	22.7 pJ/bit
Member Node	forwarding	17.6 pJ/bit
Average		22.6 pJ/bit

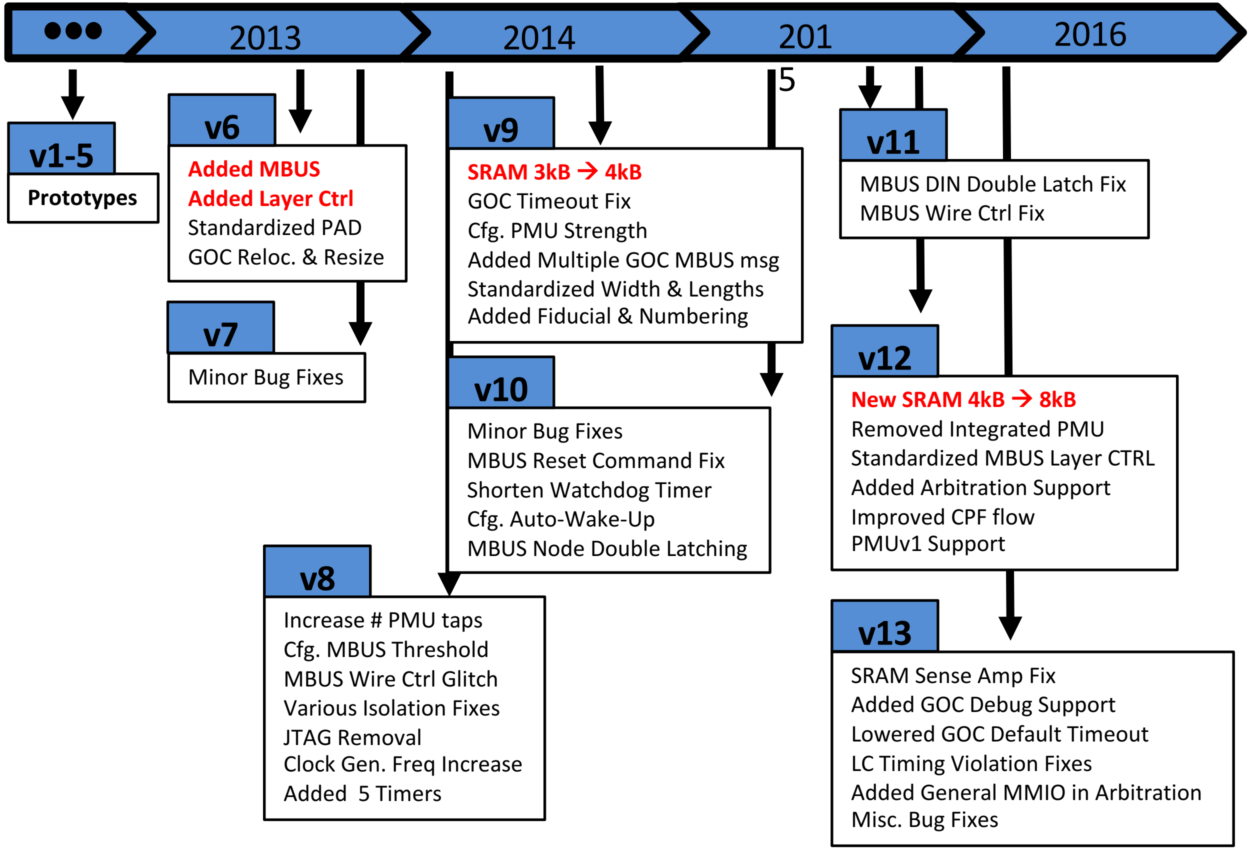
The TODOs in the specification

- Some interoperability questions
  - Minimum drive strength
  - Standards or bounds for bus clock speed
- CPU MMIO interface to MBus / MPQ internals
  - This is a more niche issue that should be in a different spec
- ”Future Extensions”
  - Obviated by MPQ streaming

M3 Evolution

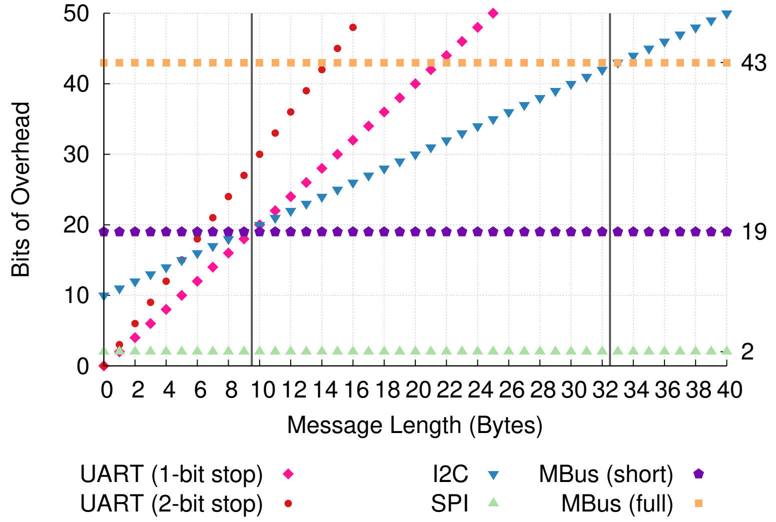
Chip	#
PRC	v13
PRE	v13
RAD	v10
DCP	v8
SNS	v7
SOL	v5
HRV	v4
IMG	v3
GAP	v3
COR	v2
FLS	v2
SIR	v2
PMU	v2
FFR	v1
MRR	v1
ADO	v1
FLP	v1

Processor Evolution



Embedded interconnect technology has not changed in over 30 years

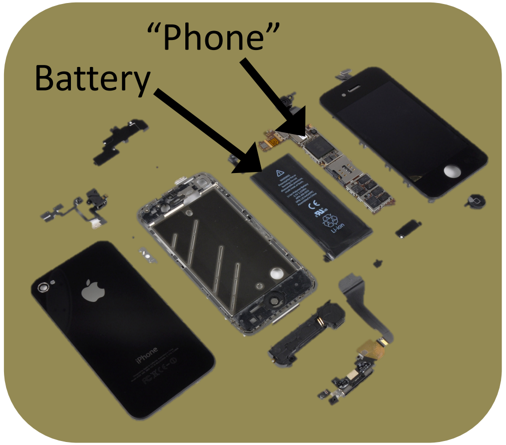
- If we re-examine…
  - Addressing
  - Acknowledgements

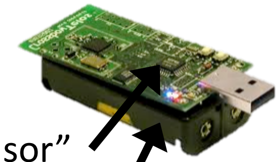


- I²C acknowledges every byte
  - How often do NAKs happen?
    - To a random byte?
  - 12.5% overhead
- MBus ACKs transactions
  - Receiver can interject message

Millimeter-scale systems are *small*

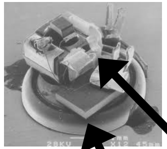
Node volume is dominated by energy storage





"Sensor"

Batteries



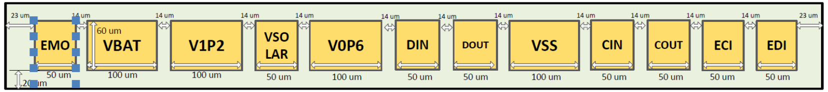
"Sensor"

Battery

And volume is **shrinking cubically**

Budget 10's µW active, 10's nW sleep, DC 0.1%

I/O pads begin to account for non-trivial percentage of node **surface area**



1 mm = 1,000 µm

50 µm, minimum viable bond pad

16-20 maximum I/O pins for 3D stacking