MBus
An Ultra-Low Power Interconnect for Next Generation Nanopower Systems

Pat Pannuto, Yoonmung Lee, Ye-Sheng Kuo, ZhiYoong Foo, Benjamin Kempke, Gyouho Kim, Ronald G. Dreslinski, David Blaauw, and Prabal Dutta
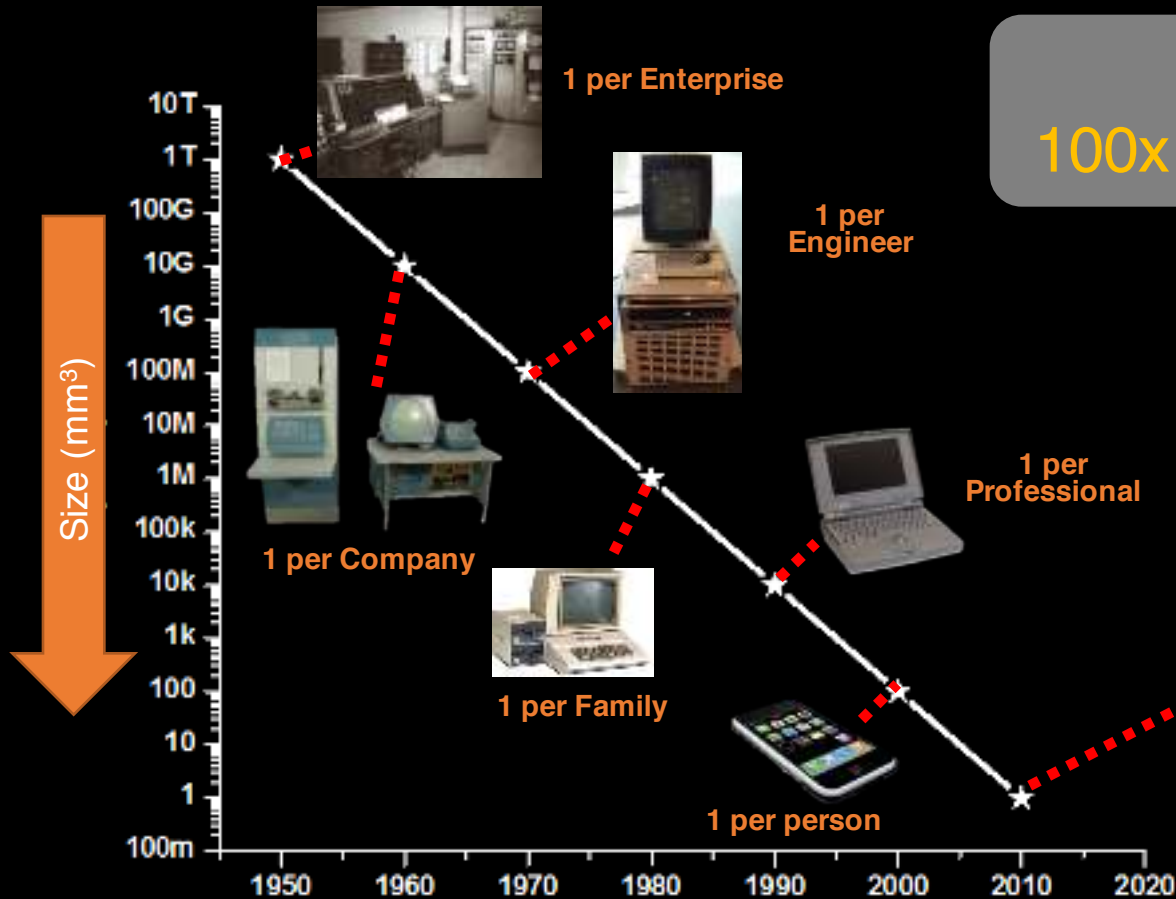
University of Michigan

The 42nd International Symposium on Computer Architecture (ISCA'15)

June 13-17, Portland, Oregon, USA

Bell's Law:
A new computing class every decade

Corollary:
100x smaller / decade

BY GORDON BELL

BELL'S LAW FOR THE BIRTH AND DEATH OF COMPUTER CLASSES

A theory of the computer's evolution.

"Smart Dust" should arrive by ~2017

1 per Enterprise

1 per Engineer

1 per Professional

1 per Company

1 per Family

1 per person

Size (mm³)

10T
1T
100G
10G
1G
100M
10M
1M
100k
10k
1k
100
10
1
100m

1950 1960 1970 1980 1990 2000 2010 2020

Graphic originally from Dennis Sylvester

2

We have a diverse array of mm-scale components

Timers

Power Management

CPUs
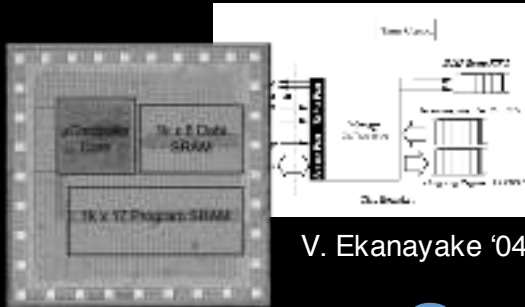
Y.-S. Lin '07

Y. Lee '13

N. Sturcken '13

V. Ekanayake '04
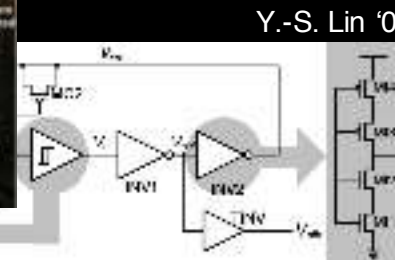
B. Warneke '04

Communication

ADCs and Sensors

A. Ricci '09
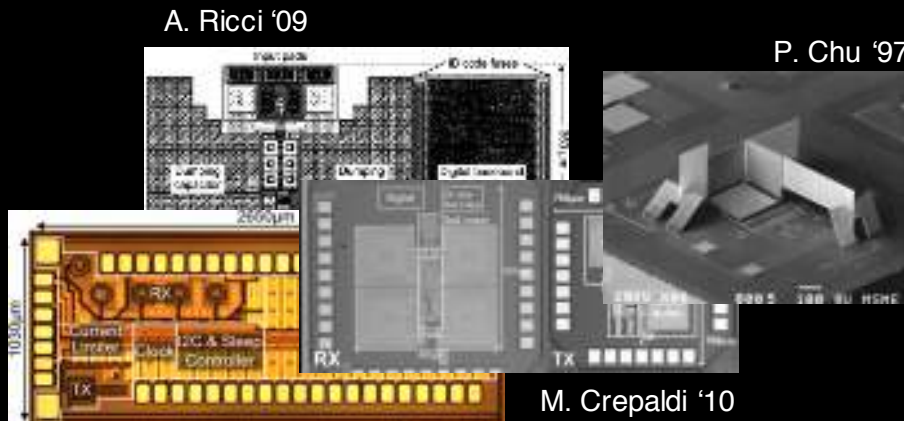
P. Chu '97

S. Hanson '09

M. Scott '03

M. Crepaldi '10

J. Brown '13

Y.-S. Lin '08

3

And a few mm-scale systems…

The 1cc Computer
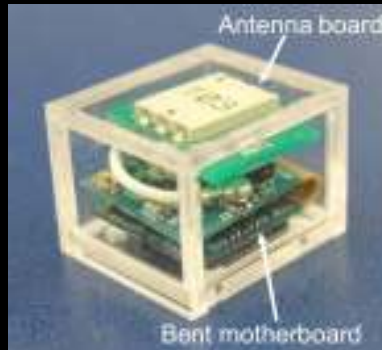
T. Nakagawa '08

Smart Dust

B. Warneke '07

Smart Dew

Y. Shapira '08

4

But where is the next class of computing?

1 per Enterprise

1 per Engineer

1 per Professional

1 per Company

1 per Family

1 per person

Size (mm³)

10T
1T
100G
10G
1G
100M
10M
1M
100k
10k
1k
100
10
1
100m

1950 1960 1970 1980 1990 2000 2010 2020

? 100 – 1000's per person
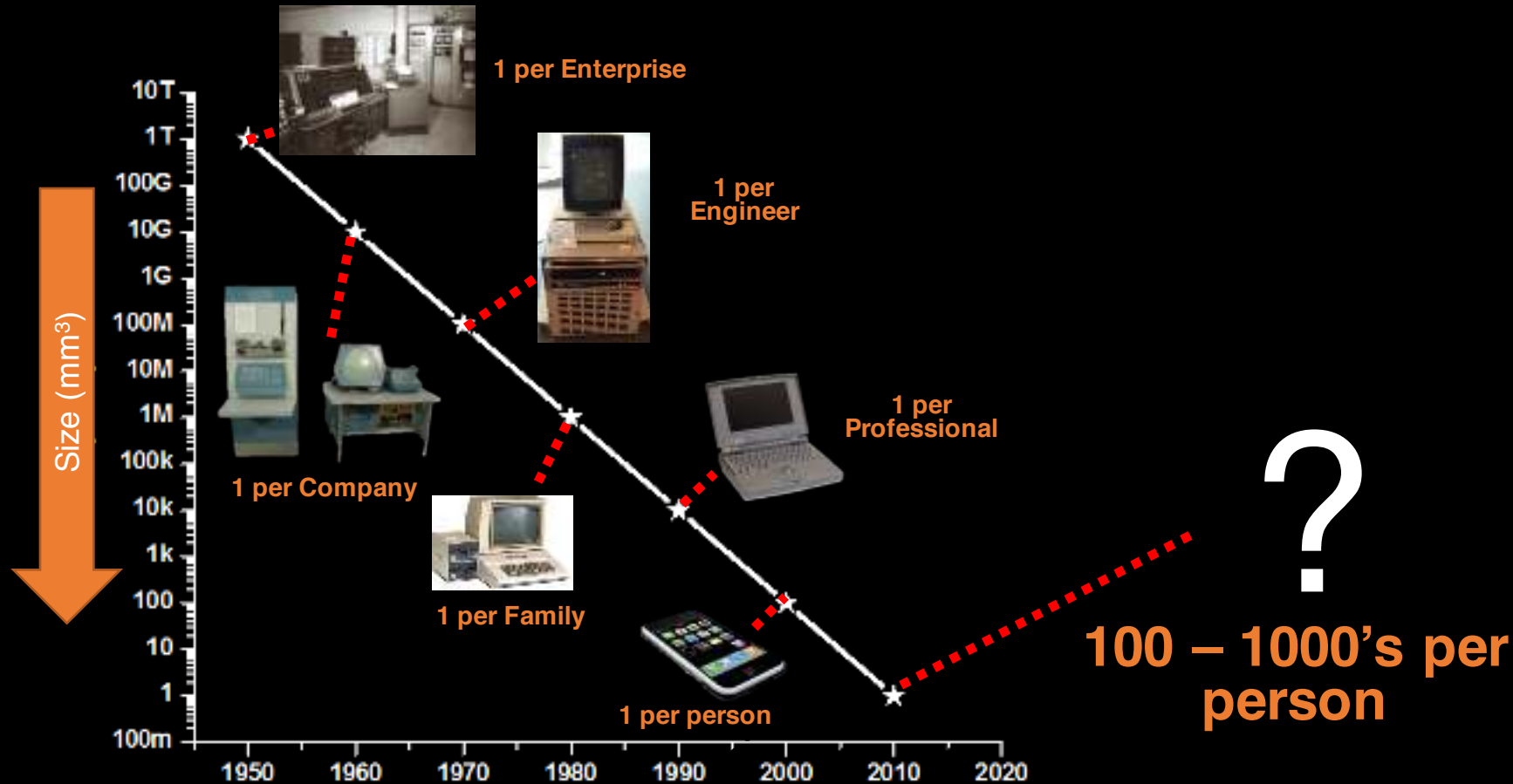
MBus is the missing interconnect that enables the mm-scale computing class

- 22.6 pJ / bit / chip, < 10 pW standby / chip
- Single-ended (push-pull) logic
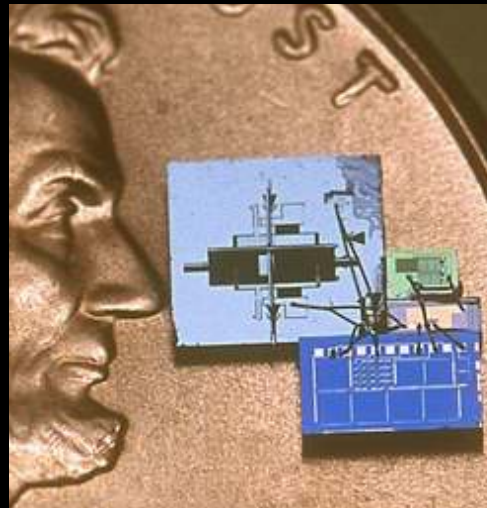- Low, fixed wire count (4)
- Multi-master
- **Power-aware**

- Implemented in over a dozen (and growing) mm-scale **chips**
  - CPU, Radio
  - Flash Memory
  - Temperature, Pressure, Imager
- To make half a dozen (and growing) mm-scale **systems**

Modularity is key for fast, iterative design, but was previously absent from mm-scale systems
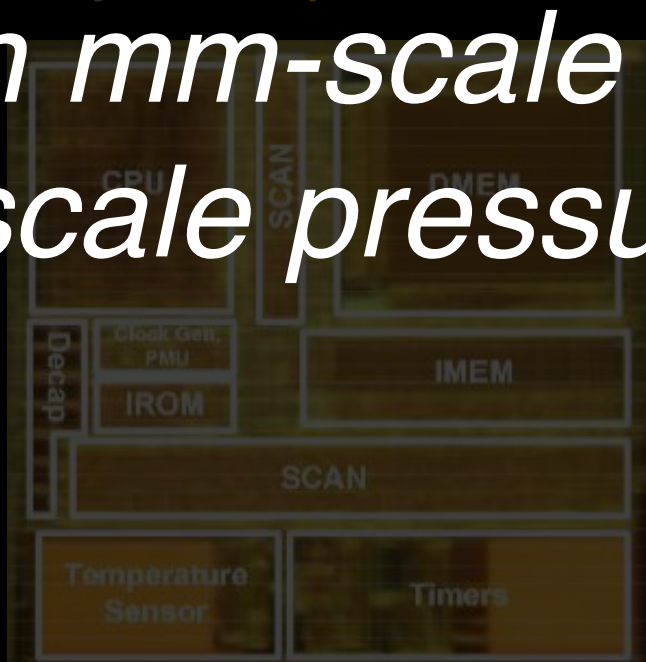
- Phoenix 2008:
 - World's lowest power computer
  - Basically a temperature sensor

- Intraocular Pressure 2011:
  - Collaboration for glaucoma health
  - A pressure sensor

From mm-scale temperature sensor to mm-scale pressure sensor took 3 years

The Phoenix Processor: A 30pW Platform for Sensor Applications, Mingoo Seok, Scott Hanson, Yu-Shiang Lin, Zhiyoong Foo, Daeyeon Kim, Yoonmyung Lee, Nurrachman Liu, Dennis Sylvester, David Blaauw , VLSI '08
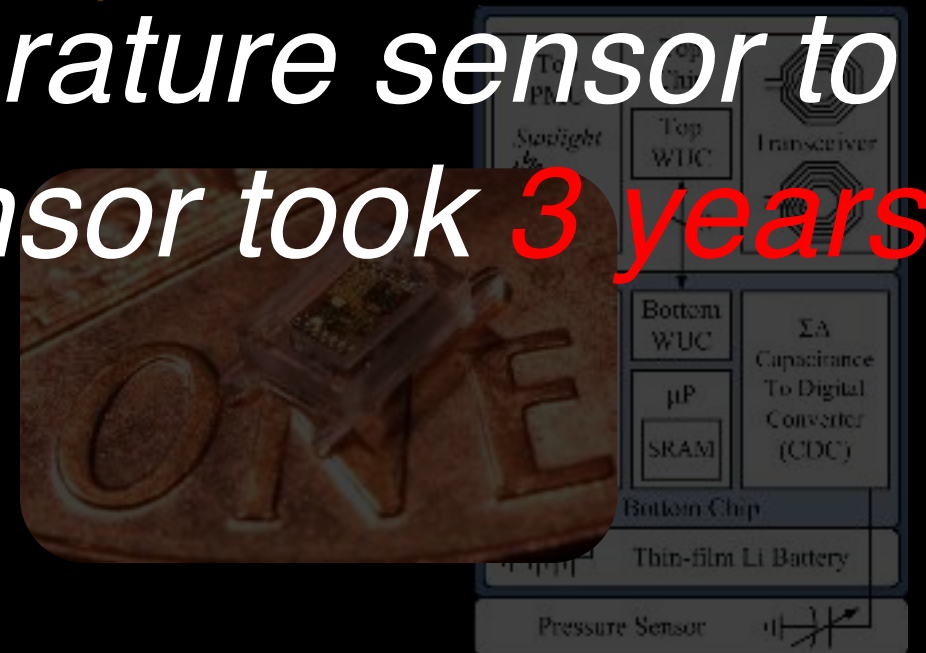
A Cubic-Millimeter Energy-Autonomous Wireless Intraocular Pressure Monitor, Gregory Chen, Hassan Ghaed, Razi-ul Haque, Michael Wieckowski, Yejoong Kim, Gyouho Kim, David Fick, Daeyeon Kim, Mingoo Seok, Kensall Wise, David Blaauw, Dennis Sylvester, ISSCC '11

MBus enables a modular, composable ecosystem of mm-scale components
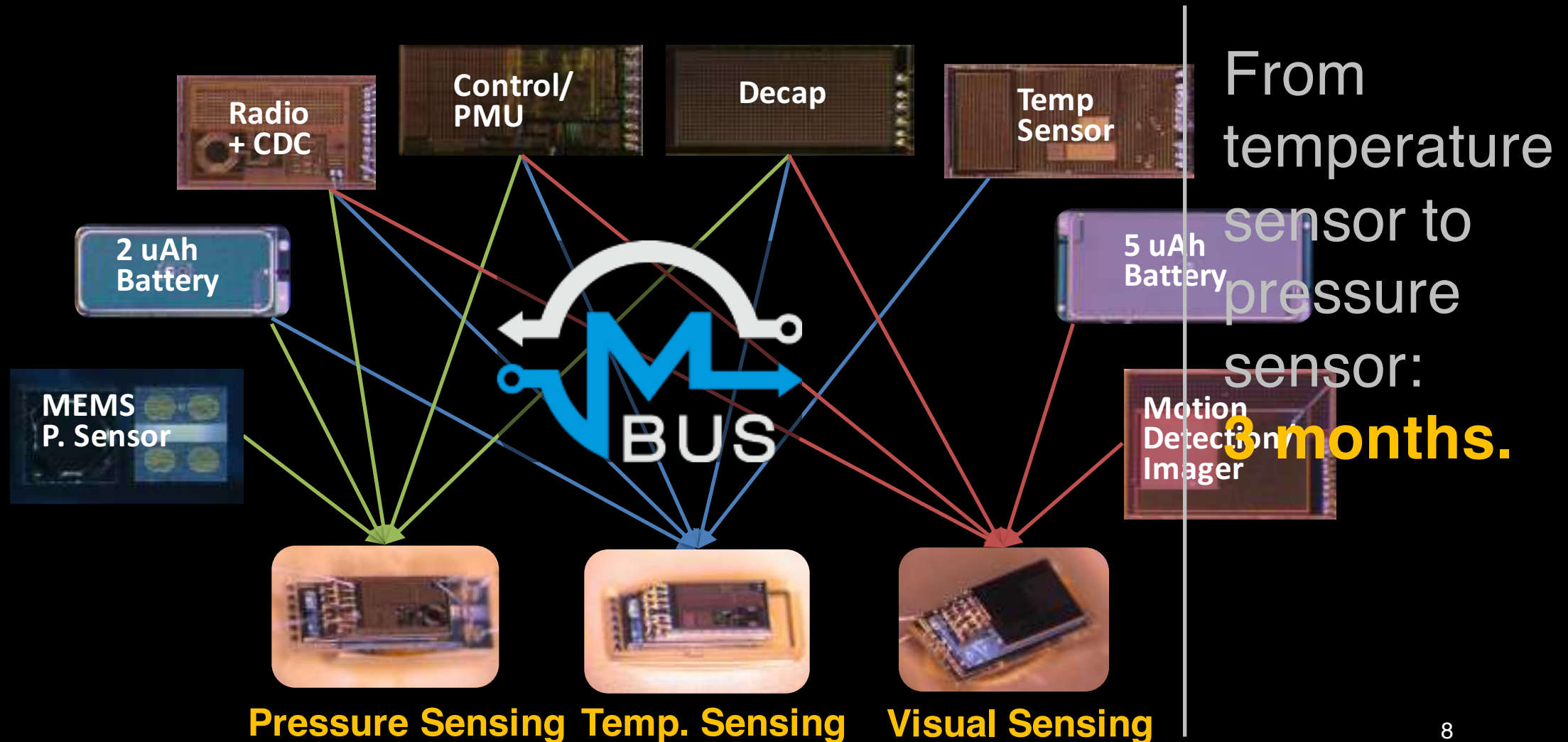
Radio + CDC

Control/ PMU

Decap

Temp Sensor

2 uAh Battery

5 uAh Battery

MEMS P. Sensor

Motion Detection Imager

From temperature sensor to pressure sensor: **3 months.**

Pressure Sensing   Temp. Sensing   Visual Sensing

Existing interconnects have served us well for 30 years. What makes mm-scale systems unique?

Node **volume** is dominated by **energy storage**

"Phone"
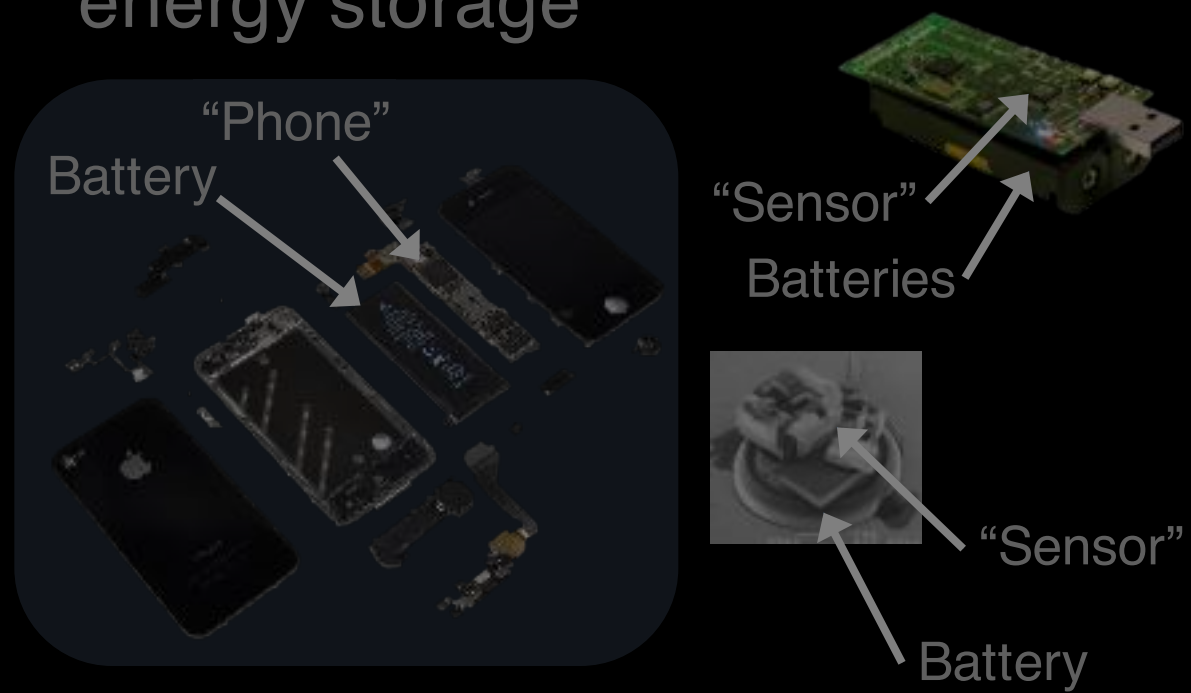
Battery

"Sensor"
Batteries

"Sensor"

Battery
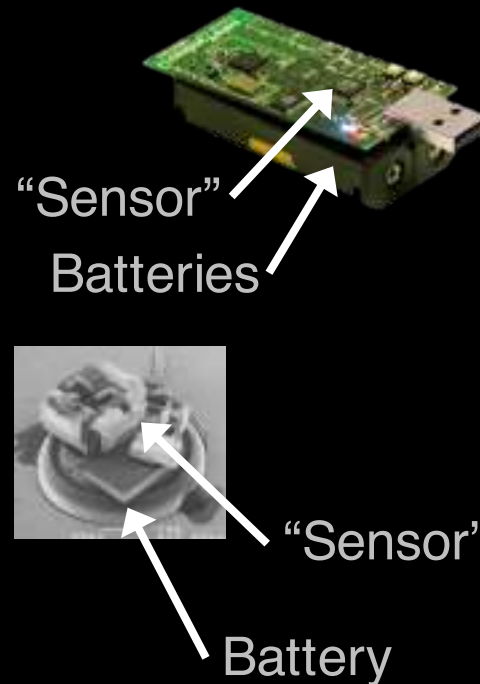
And volume is **shrinking cubically**

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

Existing interconnects have served us well for 30 years. What makes mm-scale systems unique?

Node volume is dominated by energy storage

"Phone"
Battery

"Sensor"
Batteries

"Sensor"

Battery

And volume is **shrinking cubically**

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

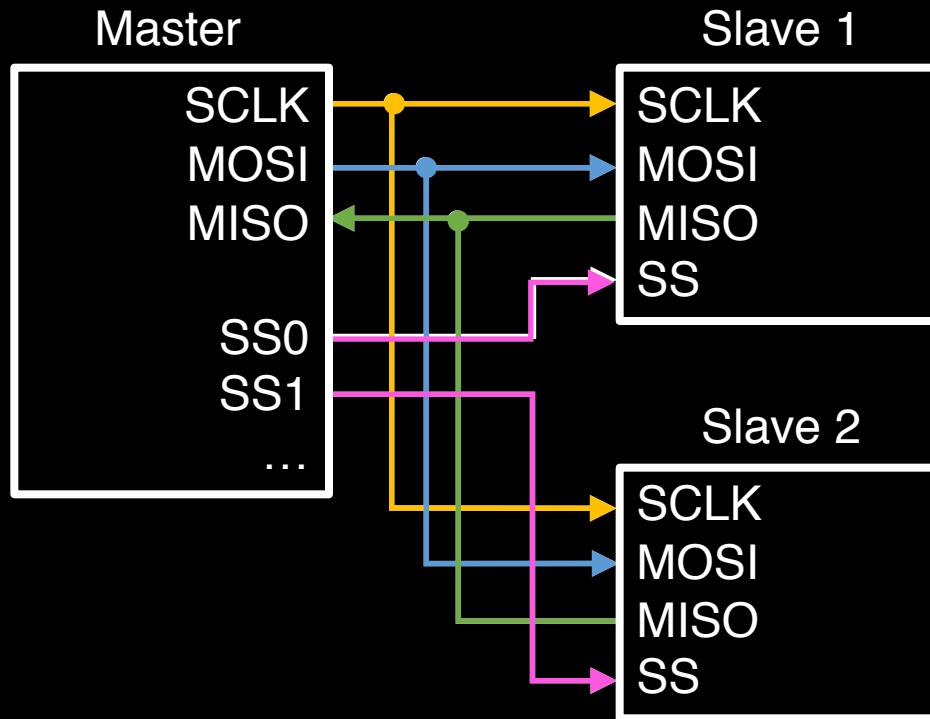I/O pads begin to account for non-trivial percentage of node **surface area**

1 mm = 1,000 µm

50 µm, minimum viable bond pad
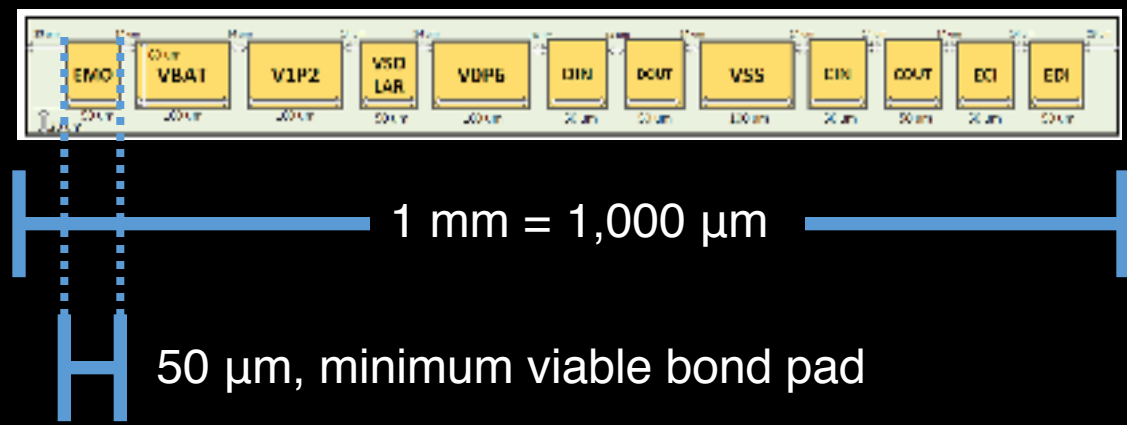
16-20 maximum I/O pins for 3D stacking

SPI and I²C are like the USB and Firewire of embedded interconnects

- Nearly every microcontroller has both
- Nearly every peripheral has one or the other
- Very few use anything else
  - (except maybe UART)
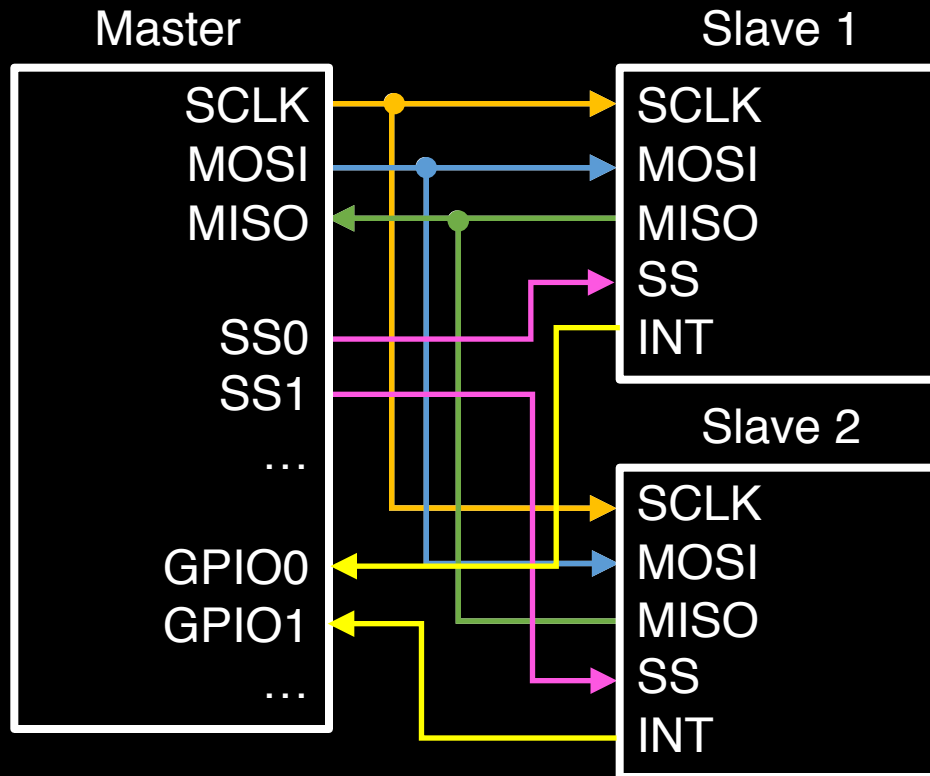
What is wrong with how are systems composed today?

- SPI, invented by Motorola in ~1979

Master

| SCLK |
| MOSI |
| MISO |
| |
| SS0 |
| SS1 |
| … |

Slave 1

| SCLK |
| MOSI |
| MISO |
| SS |

Slave 2

| SCLK |
| MOSI |
| MISO |
| SS |

- One master, N slaves
- Shared clock: **SCLK**
- Shared data bus: **MOSI**
- Shared data bus: **MISO**
- One **Slave Select** line per slave

- **Key Properties**
  - One dedicated I/O line **per slave**
  - Master controls all communication

# What is wrong with how are systems composed today?

- SPI, invented by Motorola in ~1979

Master

Master		Slave 1
SCLK		SCLK
MOSI		MOSI
MISO		MISO
		SS
SS0		INT
SS1		
…		**Slave 2**
		SCLK
GPIO0		MOSI
GPIO1		MISO
…		SS
		INT

- One master, N slaves
- Shared clock: **SCLK**
- Shared data bus: **MOSI**
- Shared data bus: **MISO**
- One **Slave Select** line per slave
- One **interrupt line** per slave

- **Key Properties**
  - ~~One~~ Two dedicated I/O lines **per slave**
  - Master controls all communication
    - Interrupts must be out-of-band

# SPI's I/O overhead and centralized architecture **do not scale** to mm-scale systems

- SPI, invented by Motorola in ~1979
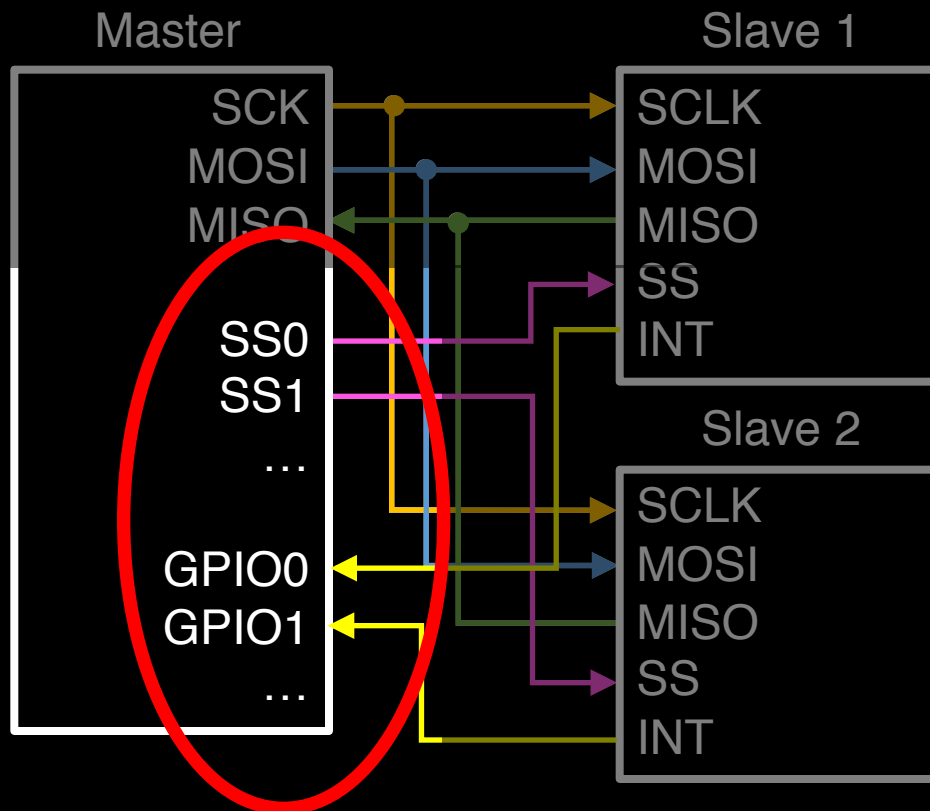
Master

- One master, N slaves
- Shared clock: **SCLK**
- Shared data bus: **MOSI**
- Shared data bus: **MISO**
- One **Slave Select** line per slave
- One **interrupt line** per slave

- **Key Properties**
  - ~~One~~ Two dedicated I/O lines **per slave**
  - Master controls all communication
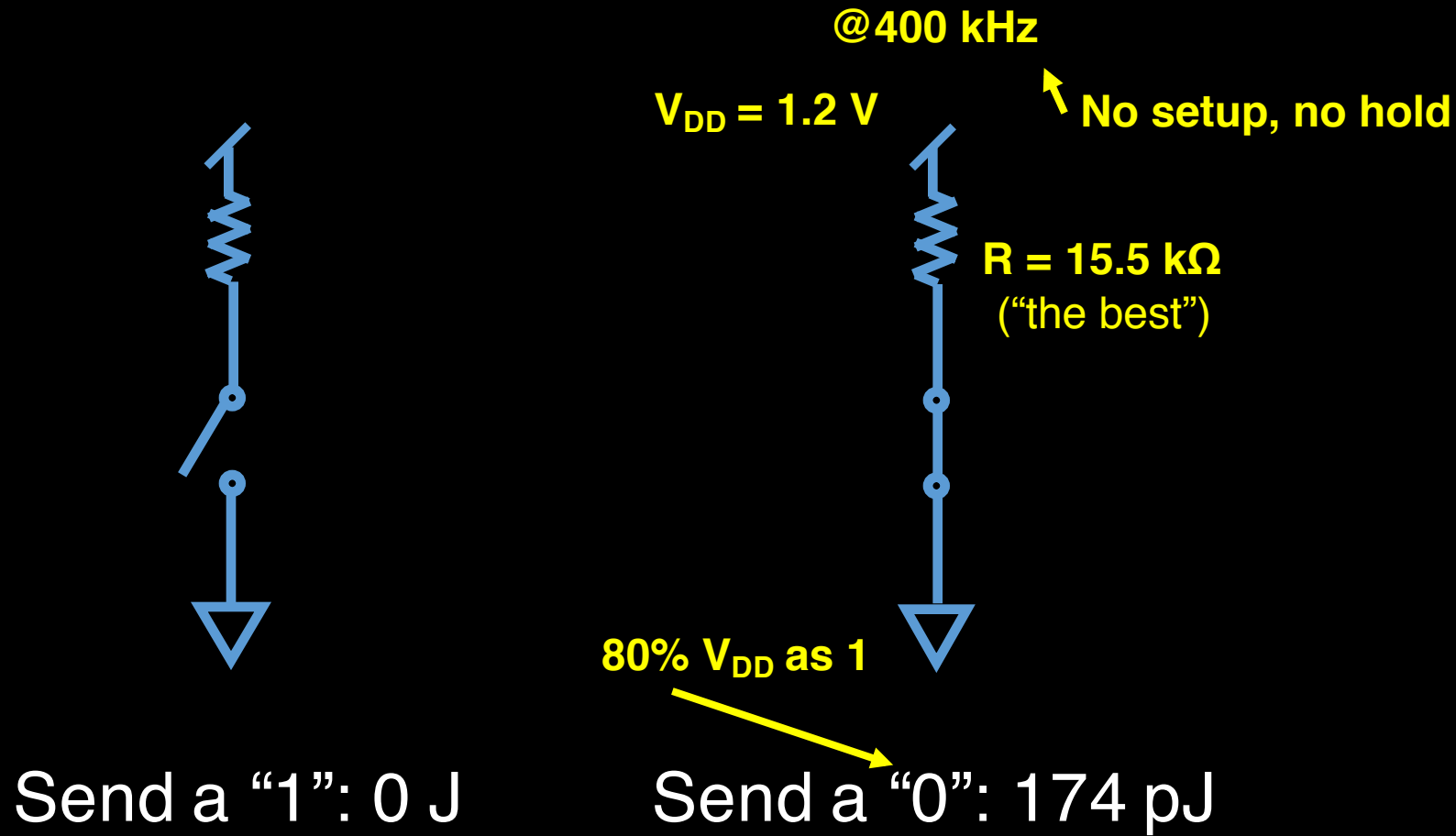   - Interrupts must be out-of-band

I^{2}C has fixed I/O requirements and a decentralized architecture

- I^{2}C, invented by Phillips in 1982
  - Any-to-(m)any on one shared bus

- **Key Properties**
  - Fixed wire count (2)

Node1

Node2

Node3

SCL
SDA

SCL
SDA

SCL
SDA
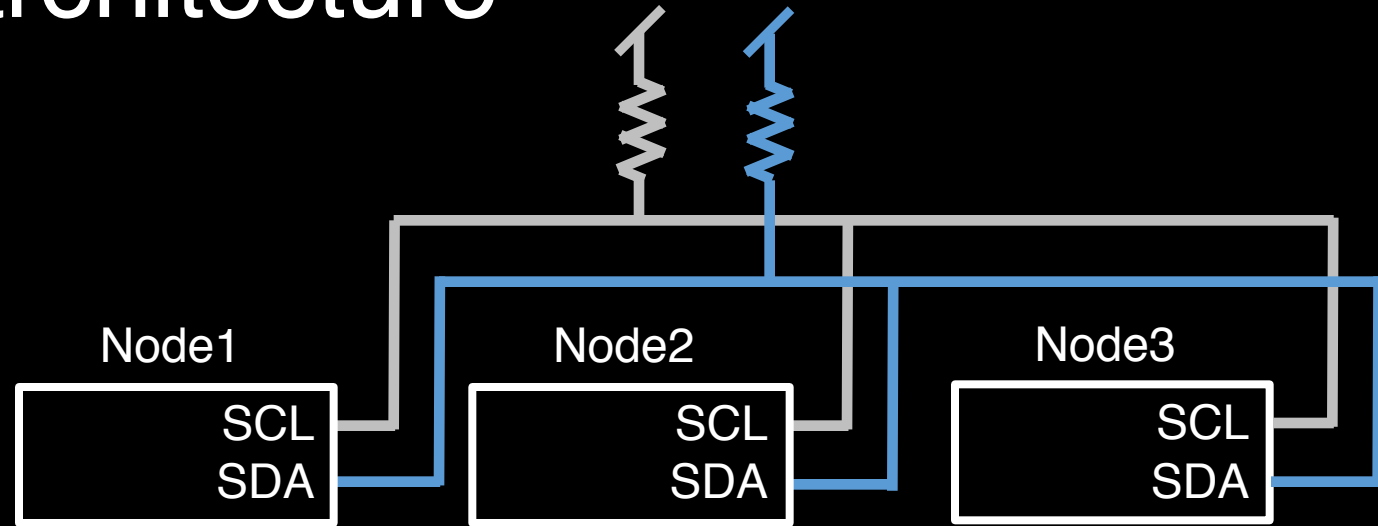
I²C has fixed I/O requirements and a decentralized architecture

- I²C, invented by Phillips in 1982
  - Any-to-(m)any on one shared bus

- **Key Properties**
  - Fixed wire count (2)
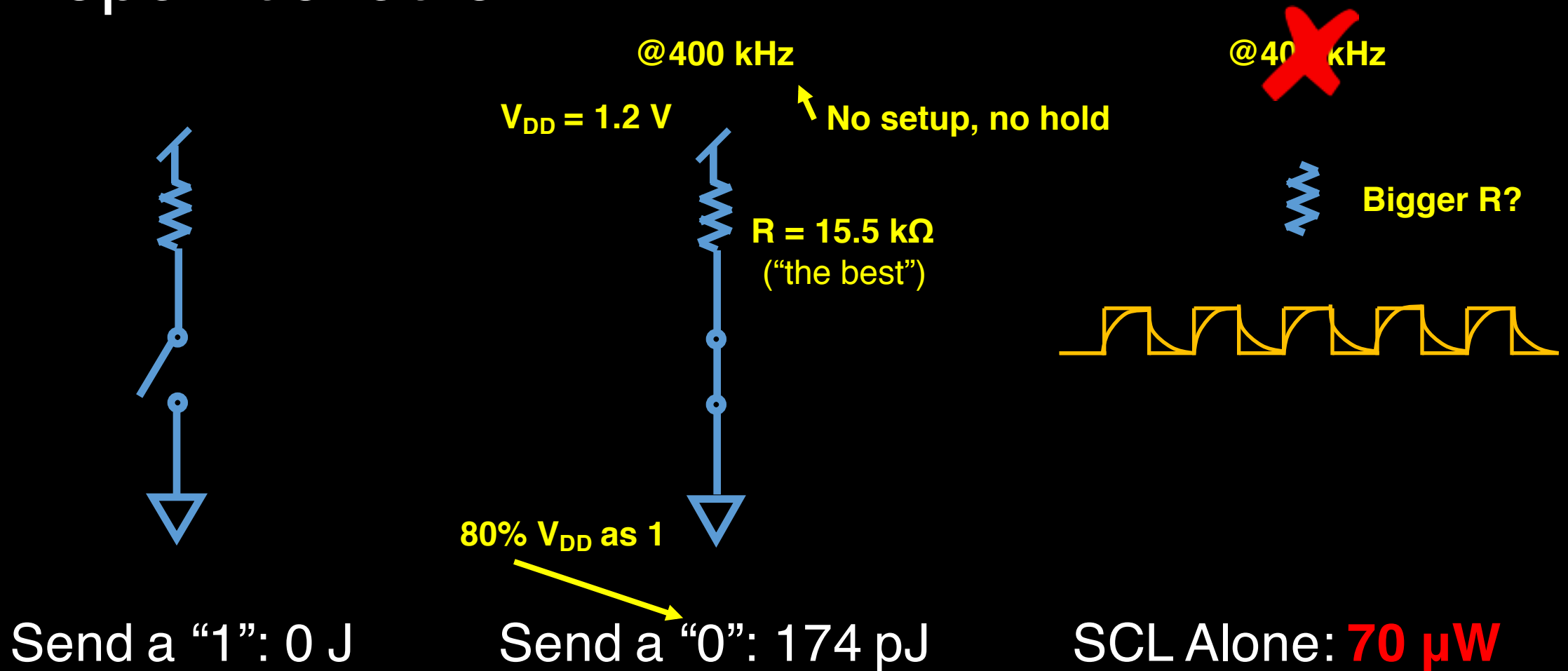 - Open-collector
    - Multi-master
    - Flow Control

Node1   Node2   Node3
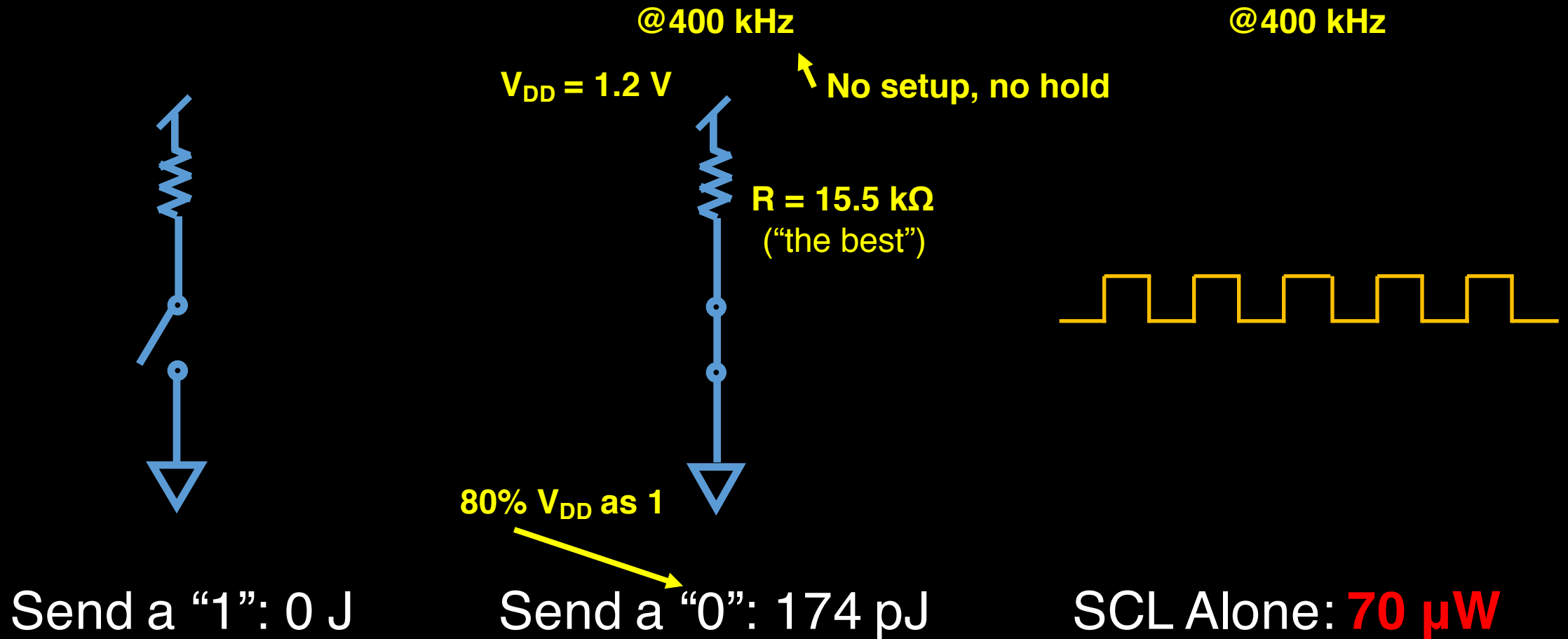
SCL   SCL   SCL
SDA   SDA   SDA
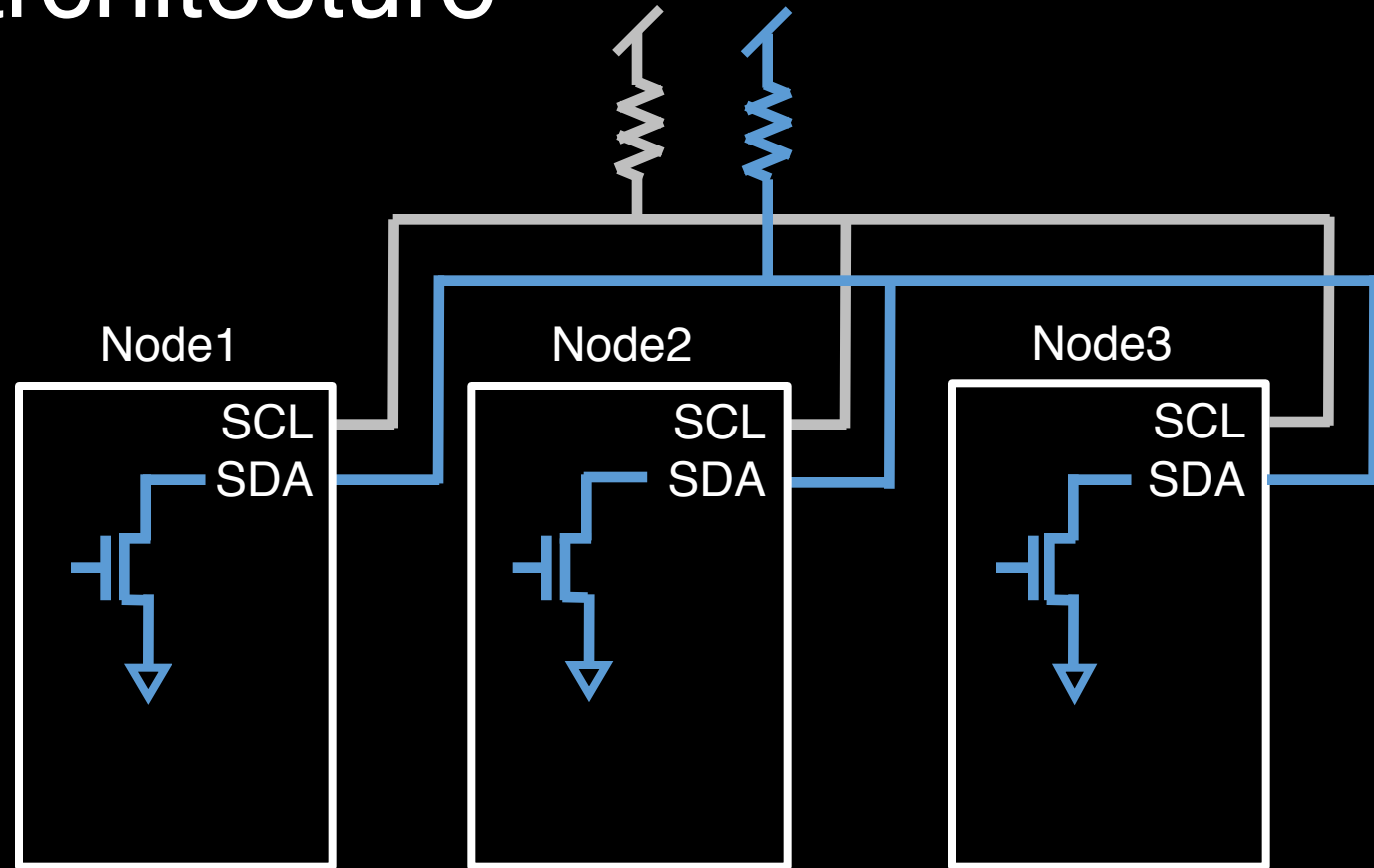
Open-collector (aka wired-AND)

The problem is the energy costs of running an open-collector

@400 kHz

V_{DD} = 1.2 V

No setup, no hold

R = 15.5 kΩ
("the best")

80% V_{DD} as 1

Send a "1": 0 J   Send a "0": 174 pJ

The problem is the energy costs of running an open-collector

@400 kHz

V_{DD} = 1.2 V

No setup, no hold

@400 kHz

R = 15.5 kΩ
("the best")

Bigger R?

80% V_{DD} as 1

Send a "1": 0 J   Send a "0": 174 pJ   SCL Alone: **70 µW**

The problem is the energy costs of running an open-collector

@400 kHz

@400 kHz

V_{DD} = 1.2 V

No setup, no hold

R = 15.5 kΩ
("the best")

80% V_{DD} as 1

Send a "1": 0 J   Send a "0": 174 pJ   SCL Alone: **70 µW**

The energy demands of open-collectors make them unsuitable for mm-scale systems

- Active Energy Budget: **20 µW**
- Not an arbitrary number:
  - Volume Target
  - Lifetime Target
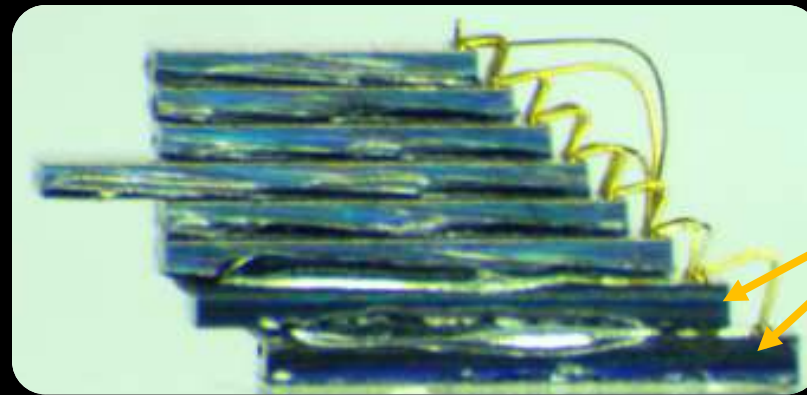
Two batteries

~ 3.1 mm

SCL Alone: **70 µW**

Can we modify I²C to bring energy costs in line with mm-scale?
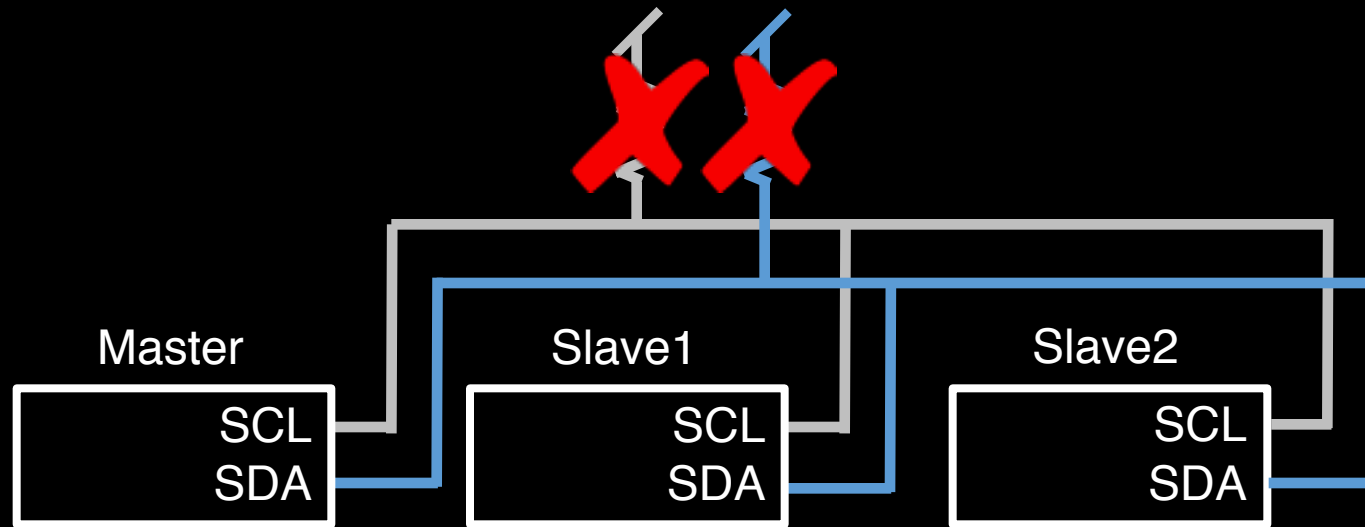
Replace the passive pull-up resistor with active circuitry
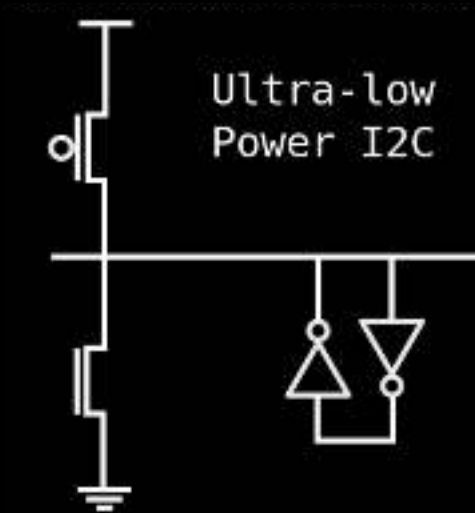
A Modular 1 mm^3 Die-Stacked Sensing Platform with Low Power I2C Inter-die Communication and Multi-Modal Energy Harvesting
Yoonmyung Lee, Suyoung Bang, Inhee Lee, Yejoong Kim, Gyouho Kim, Mohammed Hassan Ghaed, Pat Pannuto, Prabal Dutta, Dennis Sylvester, David Blaauw
IEEE Journal of Solid-State Circuits

The Good : Able to achieve **88 pJ / bit (measured)**

The Bad : Required clocks running at **5x bus clock** on every chip

Replace the passive pull-up resistor with active circuitry
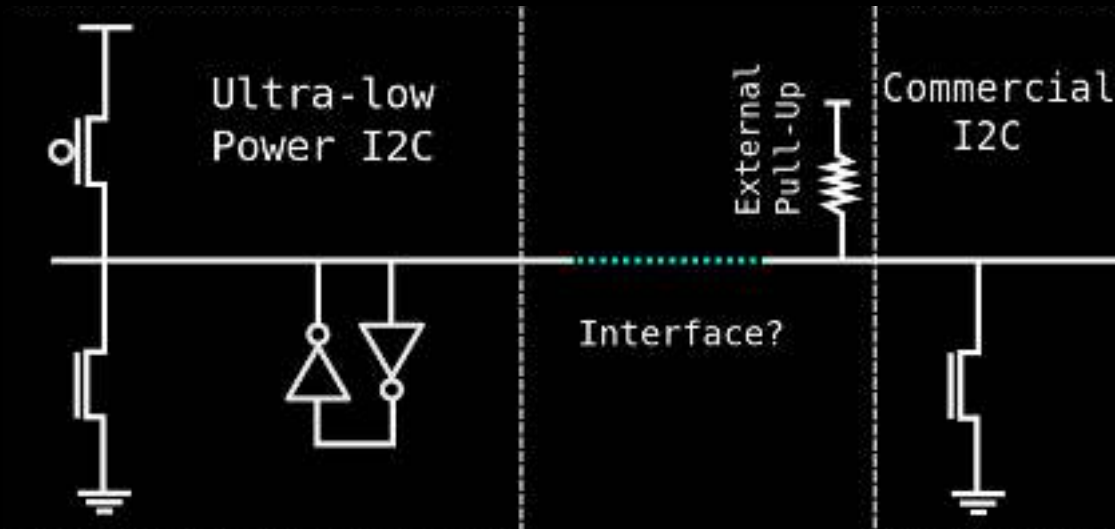
[Ultra-Constrained Sensor Platform Interfacing](#)
Pat Pannuto, Yoonmyung Lee, Benjamin Kempke, Dennis Sylvester, David Blaauw, and Prabal Dutta
IPSN 2012 (Demo)

The Good : Able to achieve **88 pJ / bit (measured)**

The Bad : Required clocks running at **5x bus clock** on every chip

The Bad : "I^{2}C-like" is not I^{2}C – required FPGA to integrate with COTS chips

The Ugly : Hand-tuned, ratioed logic on every chip – **Not synthesizable**

"Dark silicon" is more like "dimly lit silicon"

- Clock-gated modules still exhibit static leakage
  - Blows mm-scale power budget
- mm-scale systems perform power-gating
  - This means modules are cold-booting all the time
- Manageable for monolithic designs because *something* always powered on

CPU Timer	Radio
Sensor	Flash
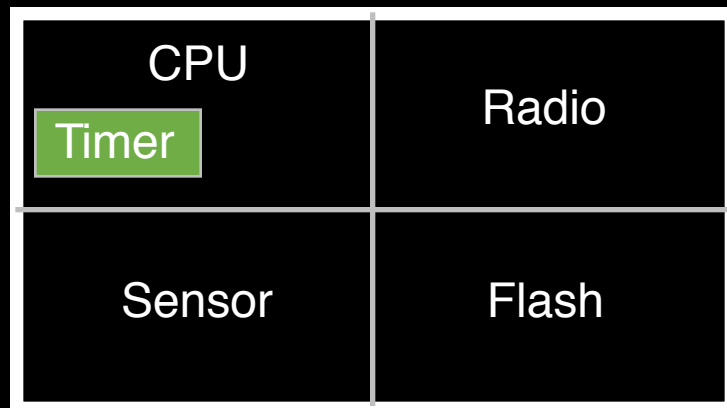
Modular mm-scale components introduce novel circuits problems and novel systems problems

- Clockless cold boot circuits are tricky

- How do you know what's awake?

- How do you communicate with "pitch black" silicon to wake it up?
  - I^2C-variant: custom "wakeup" signal

CPU Timer	Radio
Sensor	Flash
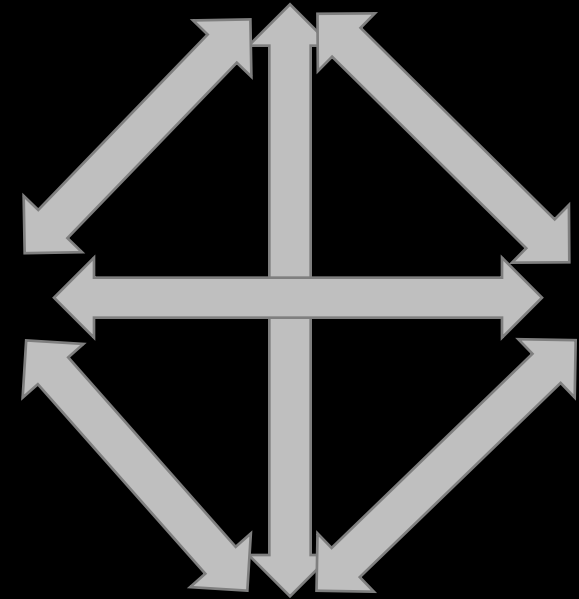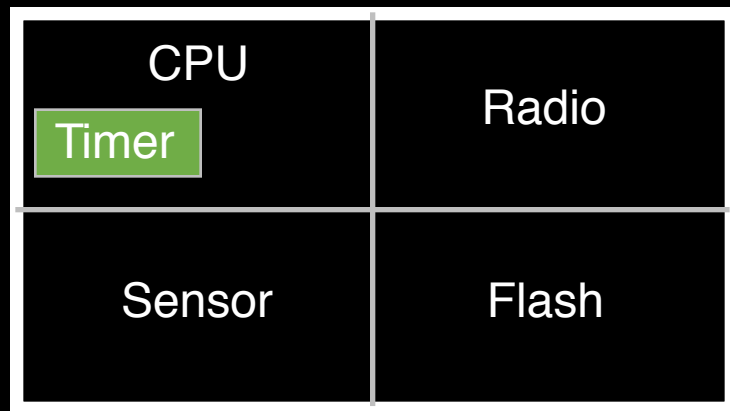
The MBus design follows from a careful consideration of all the requirements for modular, mm-scale systems
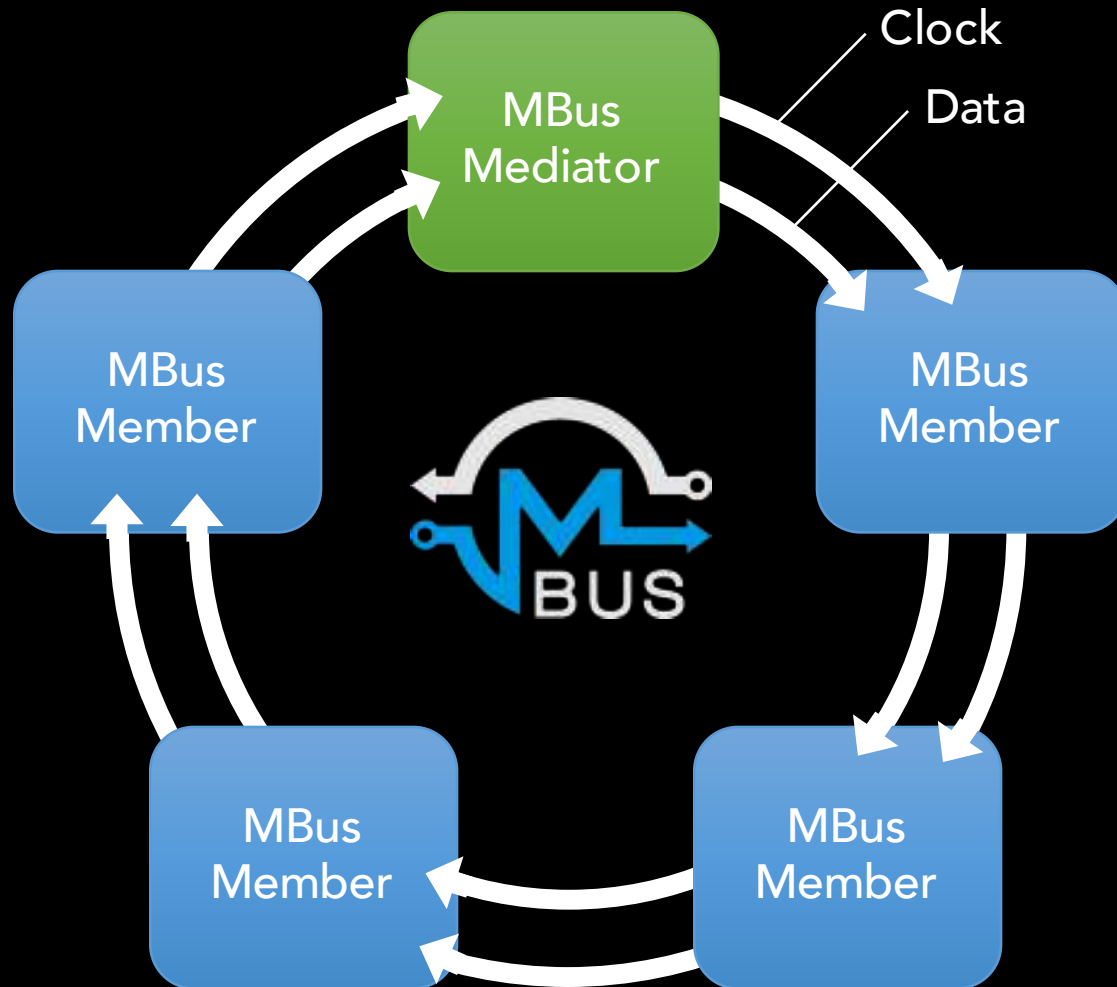
- Ring Topology

- 2 lines – 4 I/O per node
  - Clock
  - Data

- "Shoot-Through"
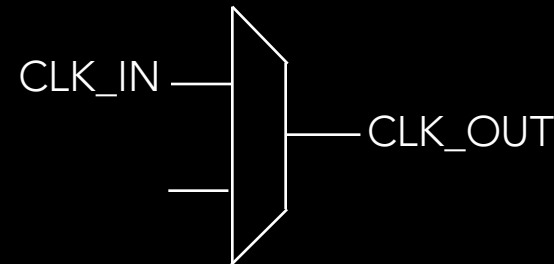
To be extensible and respect I/O constraints, wire count must be independent of node count
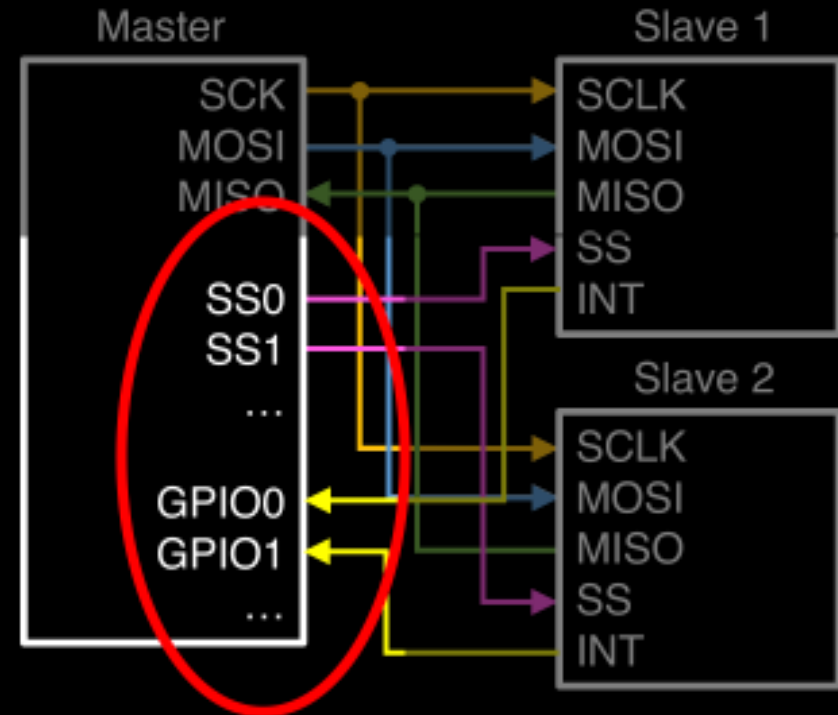
Clock

Data

Recall SPI:

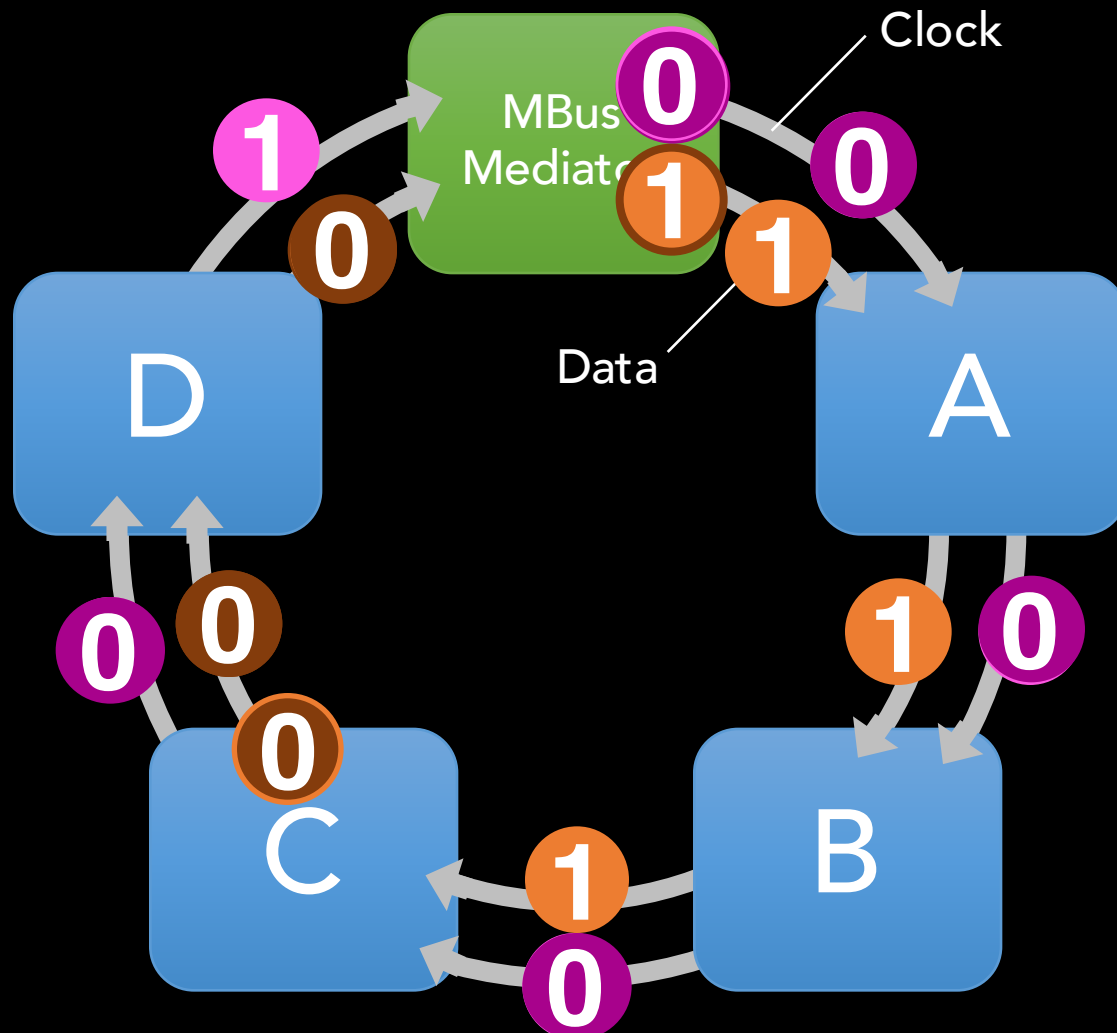# Supporting interrupts with a fixed number of single-ended connections requires an arbitration protocol
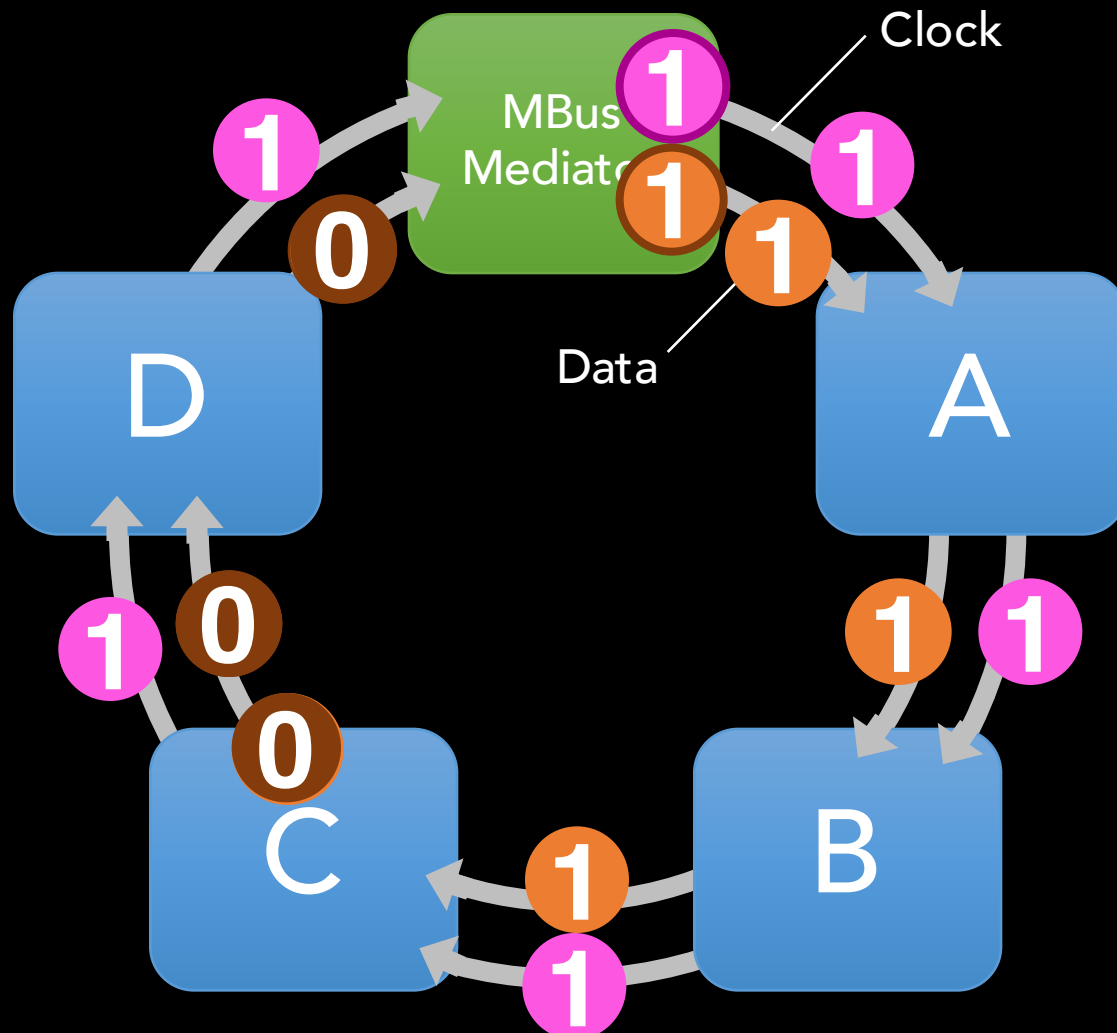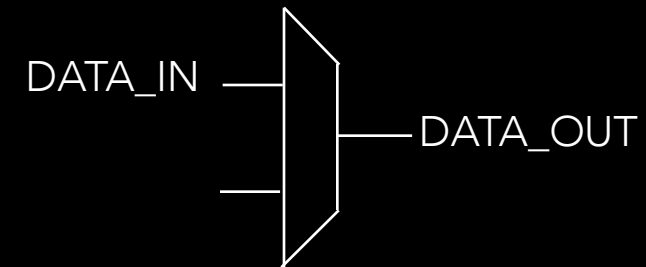

- Recall: "shoot through"
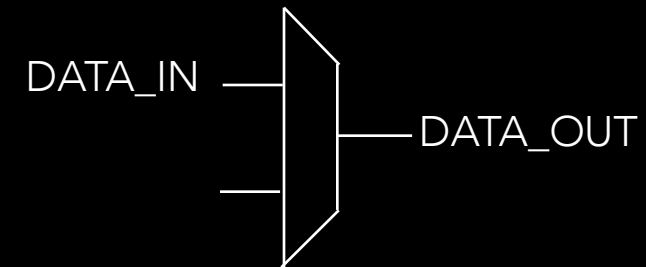
- C wants to send a message
  - Stop forwarding, drive 0

- The **mediator** does not forward during arbitration
 - Also generates the bus clock

Supporting interrupts with a fixed number of single-ended connections requires an arbitration protocol

- Recall: "shoot through"

- C wants to send a message
  - Stop forwarding, drive 0

- The **mediator** does not forward during arbitration
  - Also generates the bus clock

29

Supporting interrupts with a fixed number of single-ended connections requires an arbitration protocol
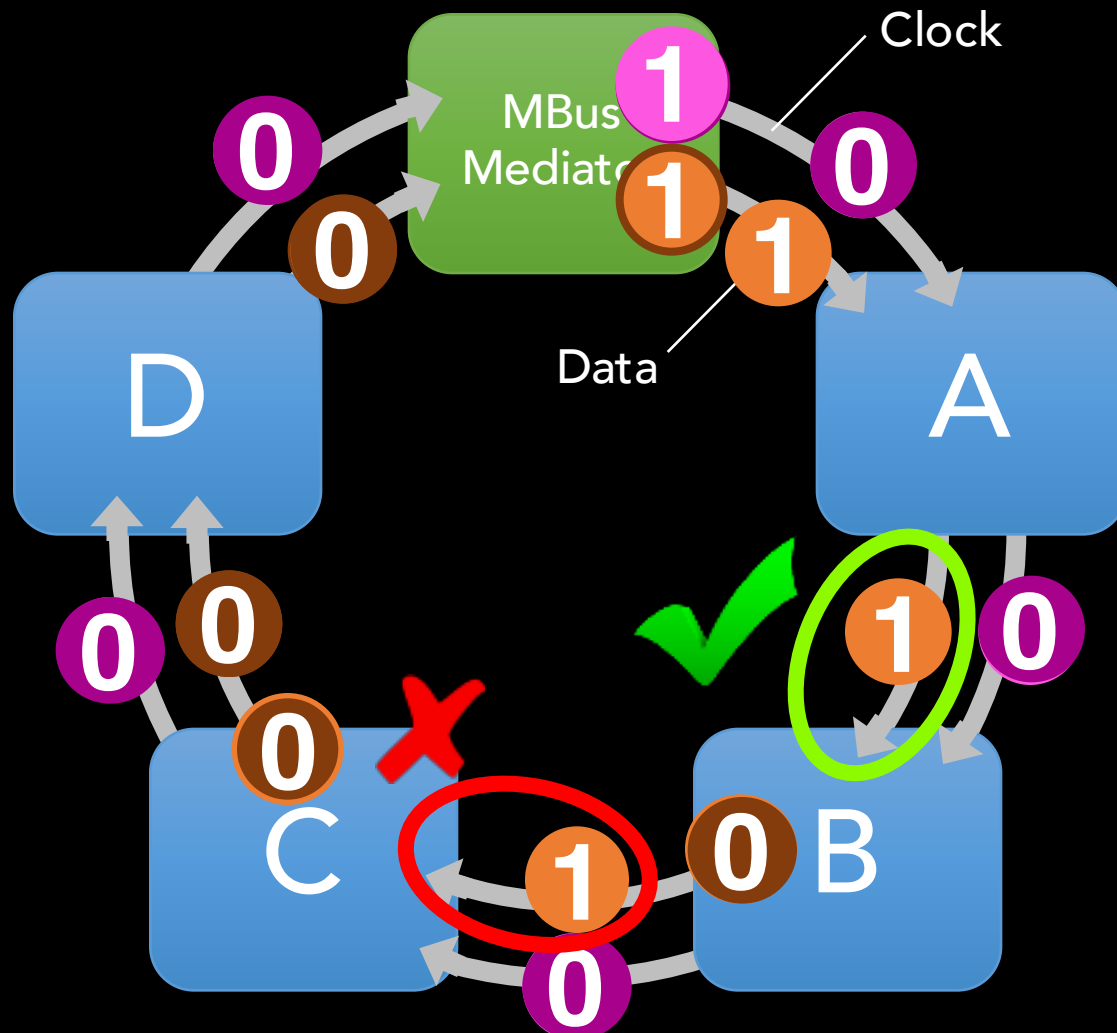
- What changes if B tries to send as well as C?
  - B and C drive DATA_OUT to 0

- B's DATA_IN high, wins

- C's DATA_IN low, loses

- MBus has topological priority

Tradeoff between globally unique addresses, address length, and overhead

- Embed addresses in message frames
  - Overhead proportional to number of uniquely addressable device
- I^2C uses 7-bit device addresses with design-time LSBs
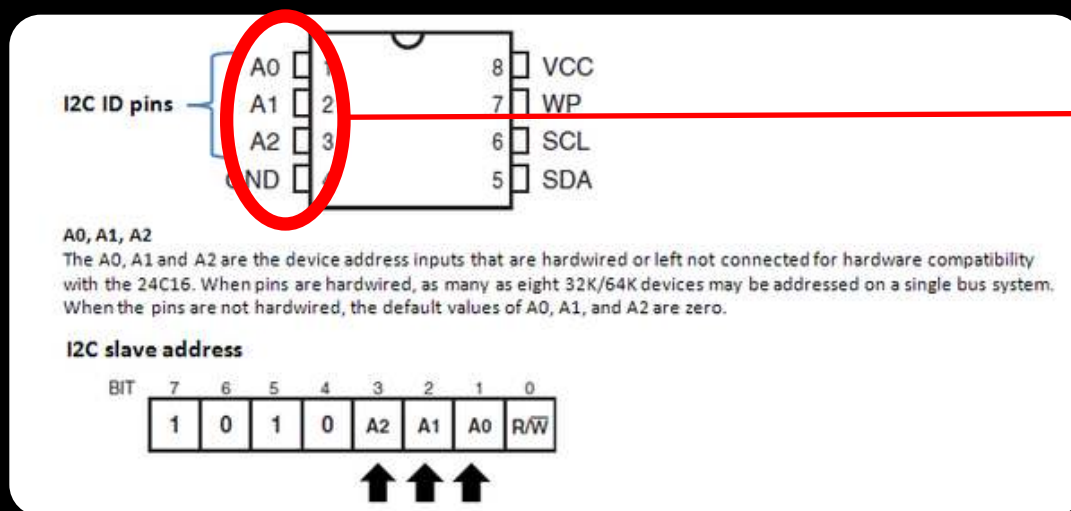
- Requires I/O not available
- Makes packaging assumptions
  - mm-scale systems not always PCB
  - Routing may not be easy
    - 3D stack
    - Flip-chip + TSVs

31

Tradeoff between globally unique addresses, address length, and overhead

- 3 Options
  - Short static addresses and allow device conflicts
  - Long static addresses to avoid device conflicts
  - Non-static addresses

- MBus does all 3
  -   4-bit: Static short prefixes (device class)
  - 24-bit: Static long prefixes (unique device ID)
 -   4-bit: Runtime enumeration protocol (replaces short prefix)

Unbounded messages maximize flexibility and minimize overhead

- An MBus message is 0…N bytes of data
- Embed length in message
  - Imposes large overhead for short messages
  - Forces fragmentation of long messages
- "End-of-message" sentinel byte(s)
  - Imposes large overhead for short messages
  - Requires escaping if sentinel is in transmitted
  - Data-dependent behavior, hard to reason about
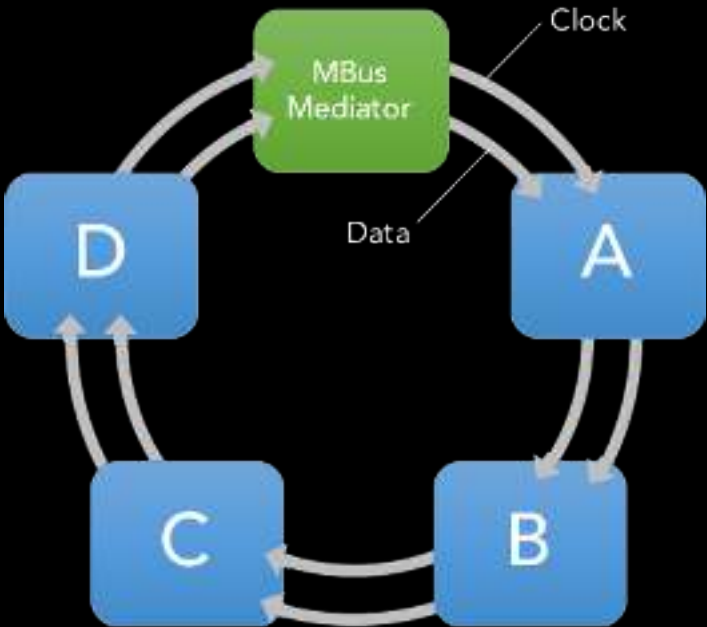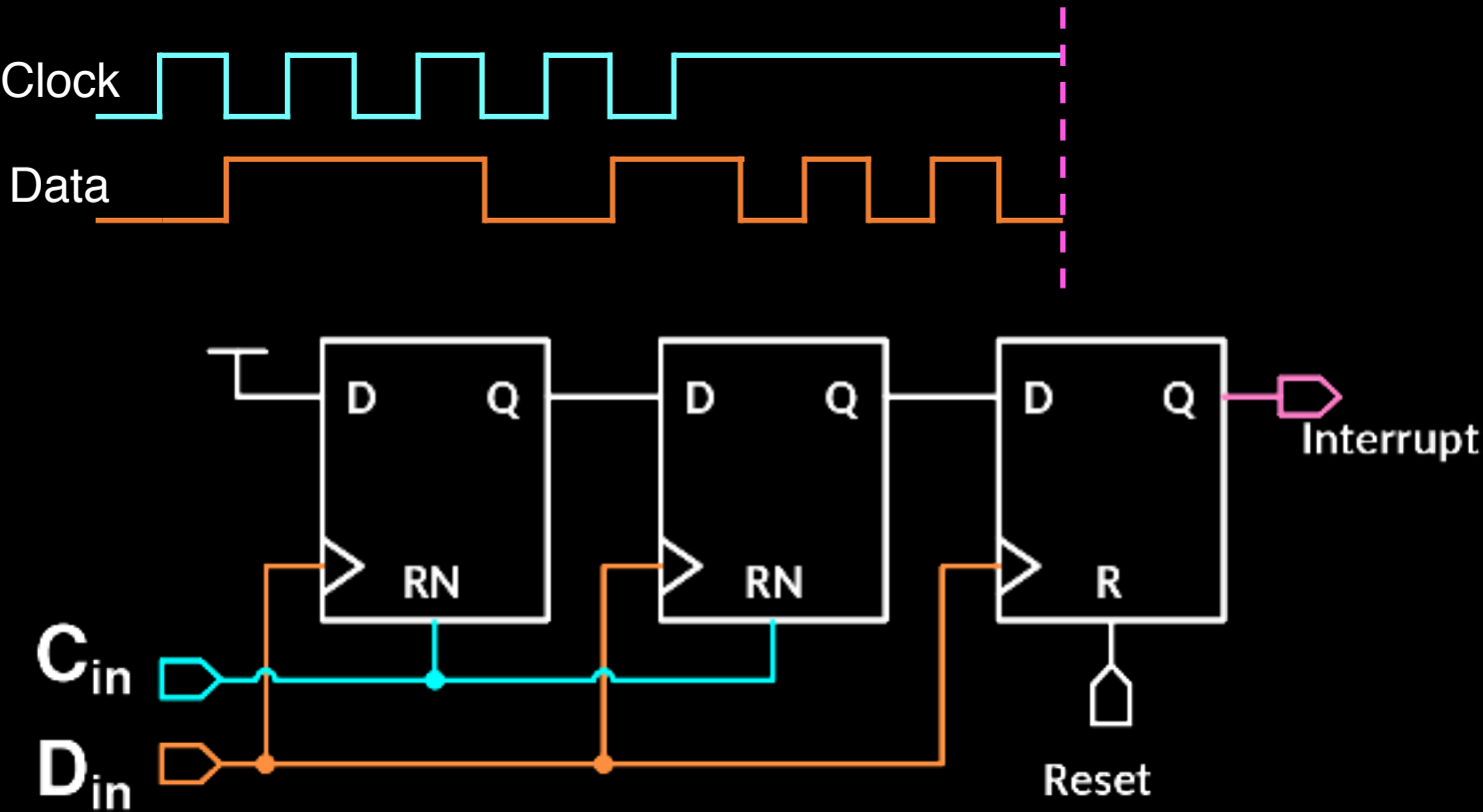   - Worst case 2x overhead!
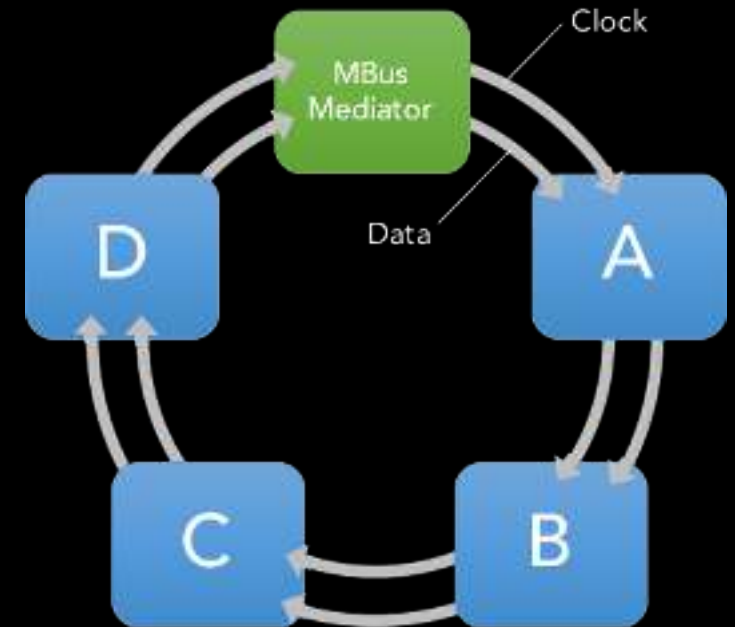
MBus "interjections" provide an in-band end-of-message with minimal overhead

• During normal operation, Data toggles slower than Clock

MBus "interjections" provide an in-band end-of-message with minimal overhead
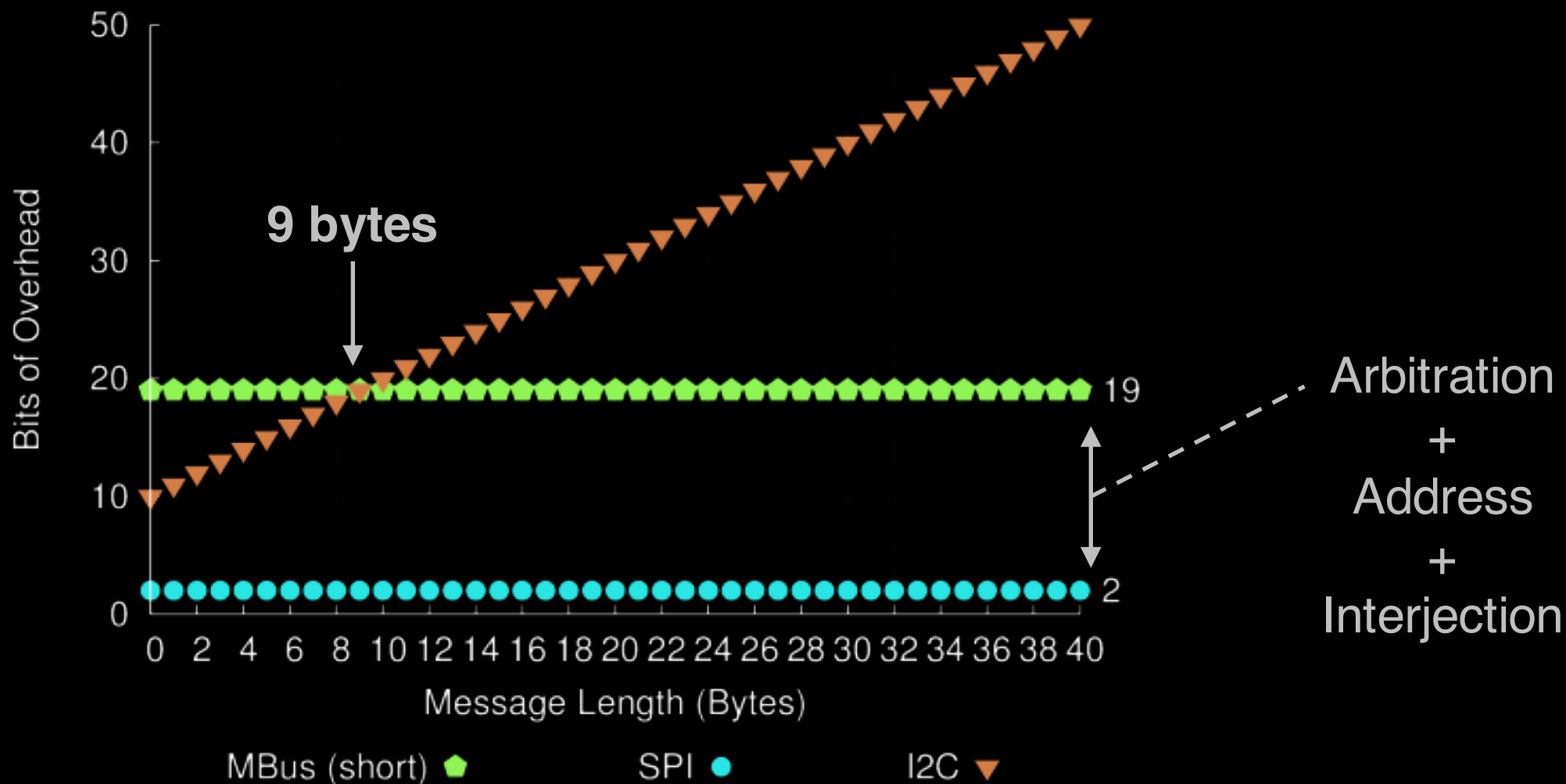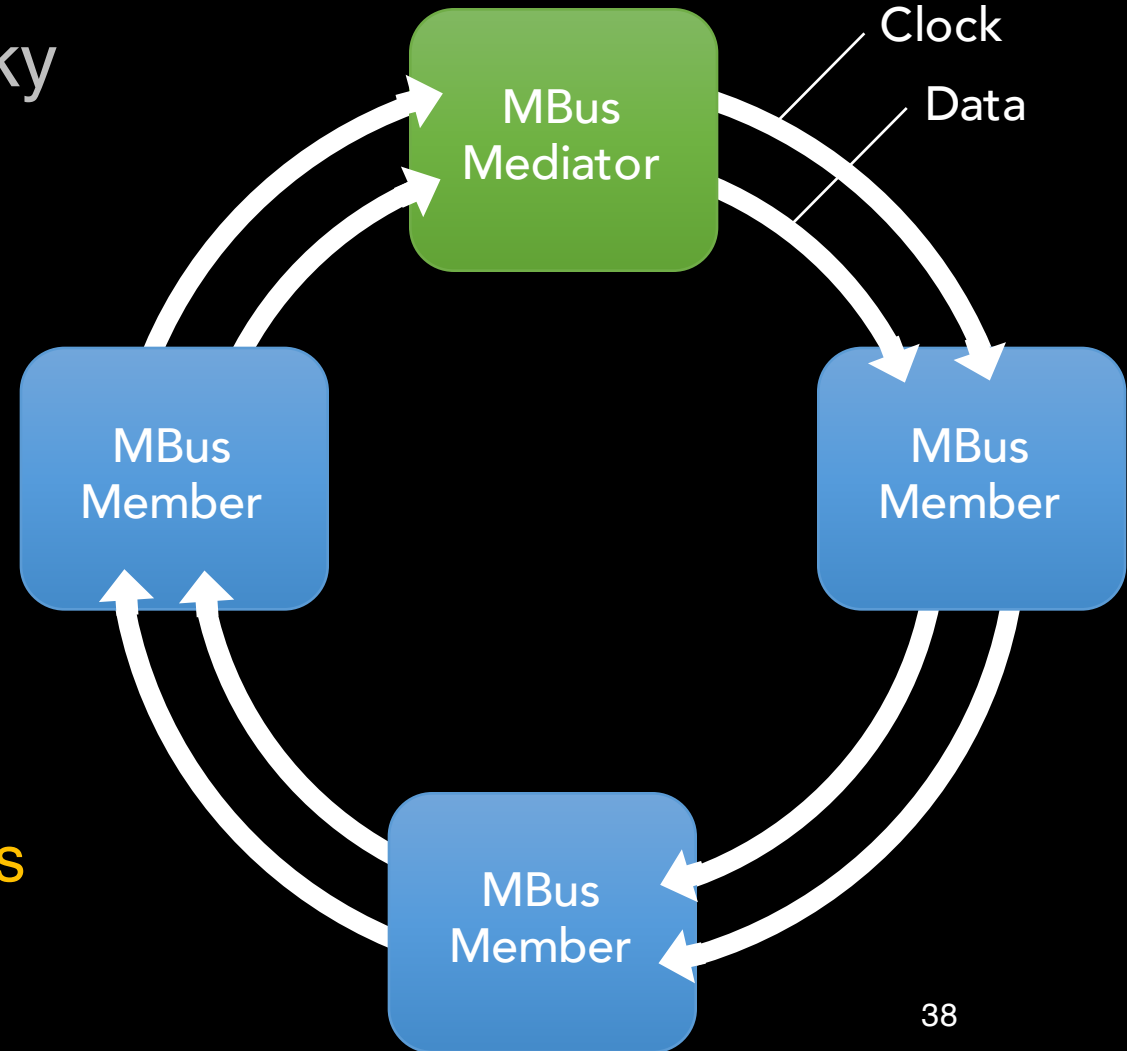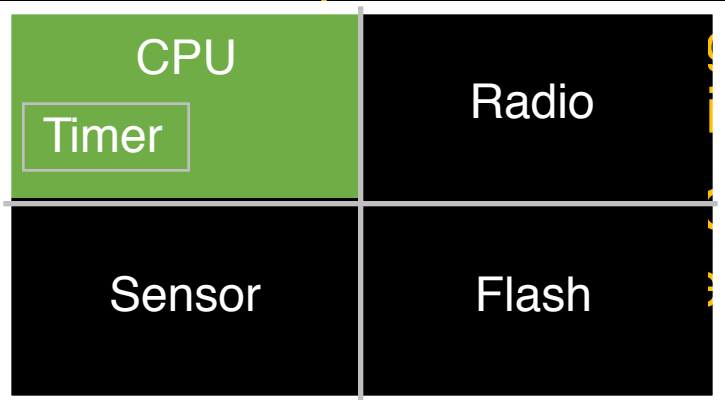
- During normal operation, Data toggles slower than Clock

MBus "interjections" provide an in-band end-of-message with minimal overhead

- During normal operation, Data toggles slower than Clock

ACK

Clock

Data

Idle

C_{in}

D_{in}

D Q

RN

D Q

RN

D Q

R

Interrupt

Reset

36
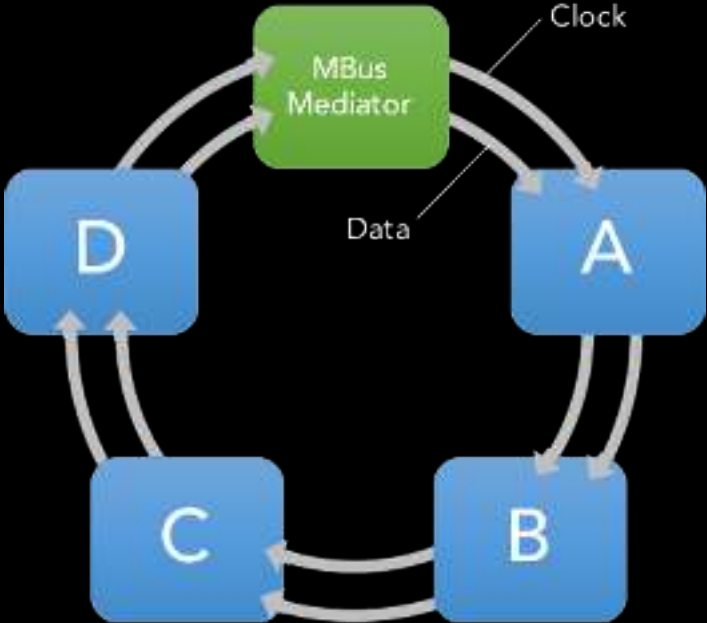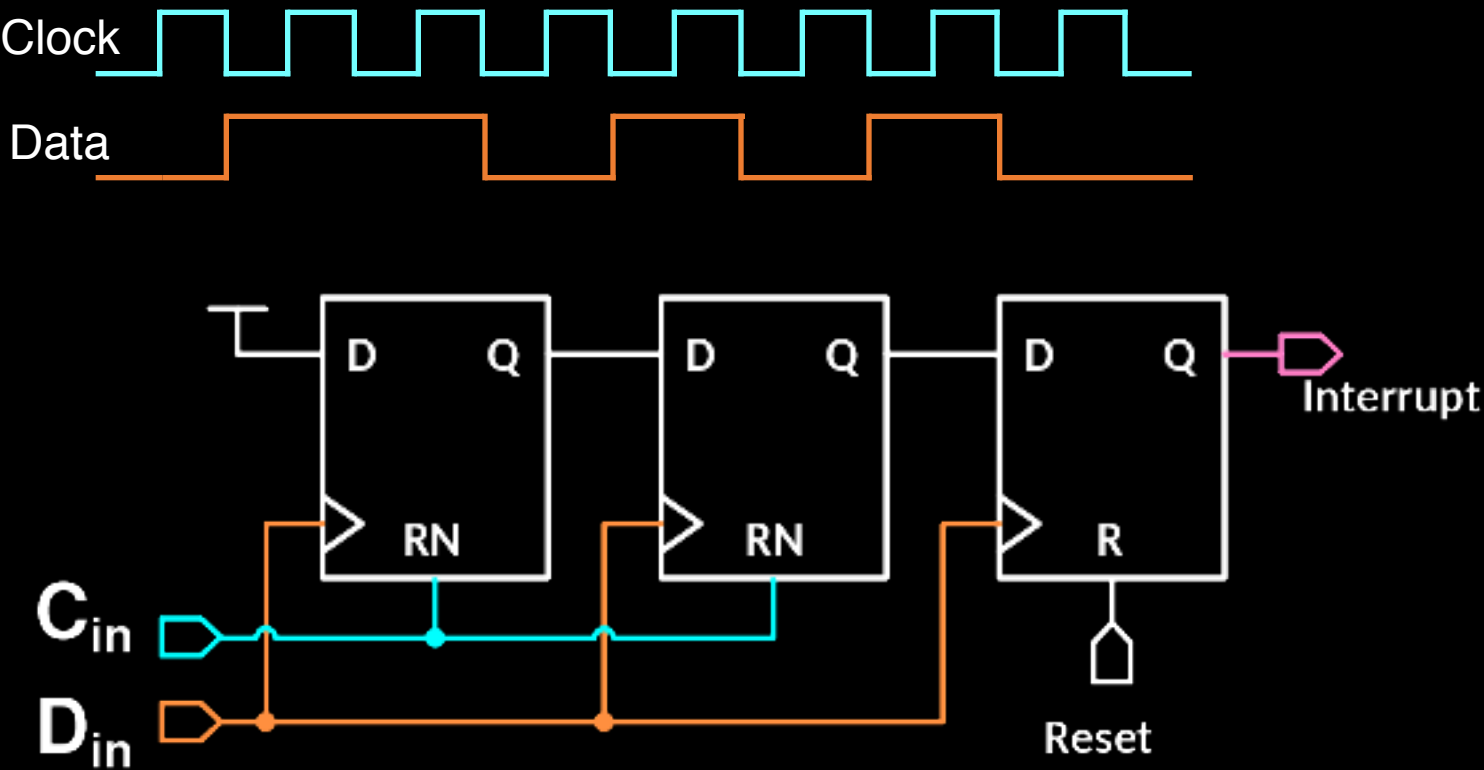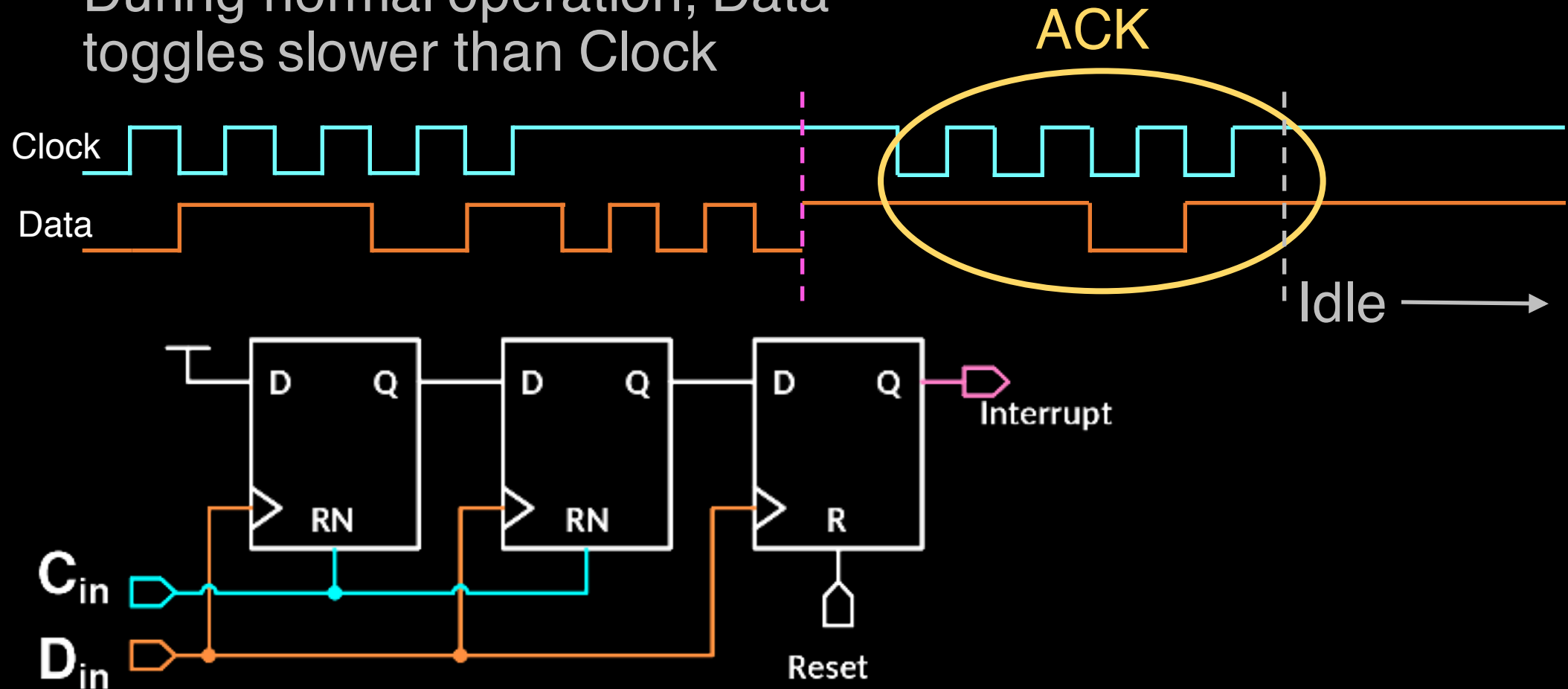
Transaction-level ACKs minimize common-case overhead while interjections preserve flow control

# The modularity enabled by MBus created a circuits problem and a systems problem

- Clockless cold boot circuits are tricky
- How do you communicate with "pitch black" silicon to wake it up?
  - How do you know what's awake?

- A power-gated node cannot send
  - Use arbitration edges to drive the cold-boot circuitry
    - (M̶~~~~~~~~~~gs)
  - Nodes~~~~~~~~~~ing
- Nodes a~~~~~~~~~~on"
  - No ne~~~~~~~~~~r states

CPU	Radio
Timer	
Sensor	Flash

Clock
Data

MBus Mediator

MBus Member

MBus Member

MBus Member

These primitives enable an architectural shift in system design

- CPU acts as configurator instead of overseer
  - Preprogram temperature sensor to send radio packets
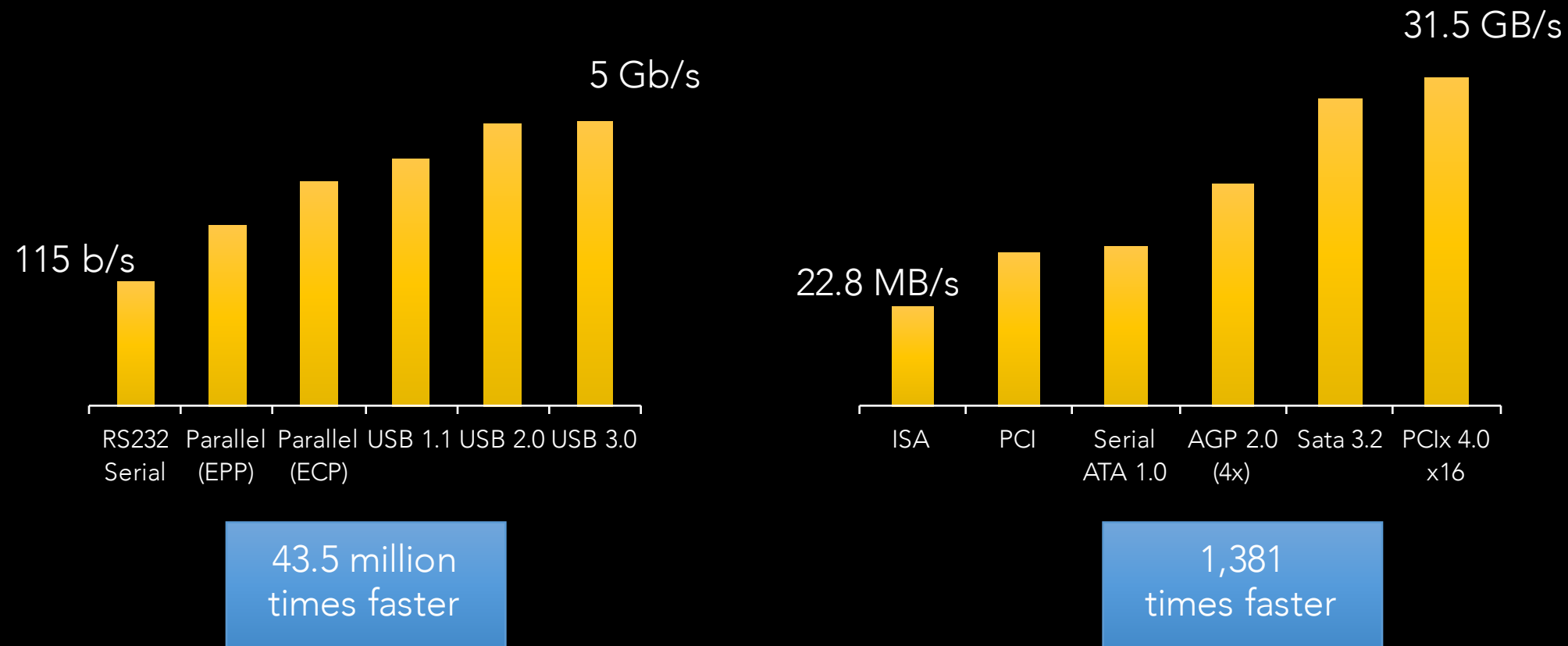 - Not unlike modern SOCs (sleepwalking, µDMA), but distributed
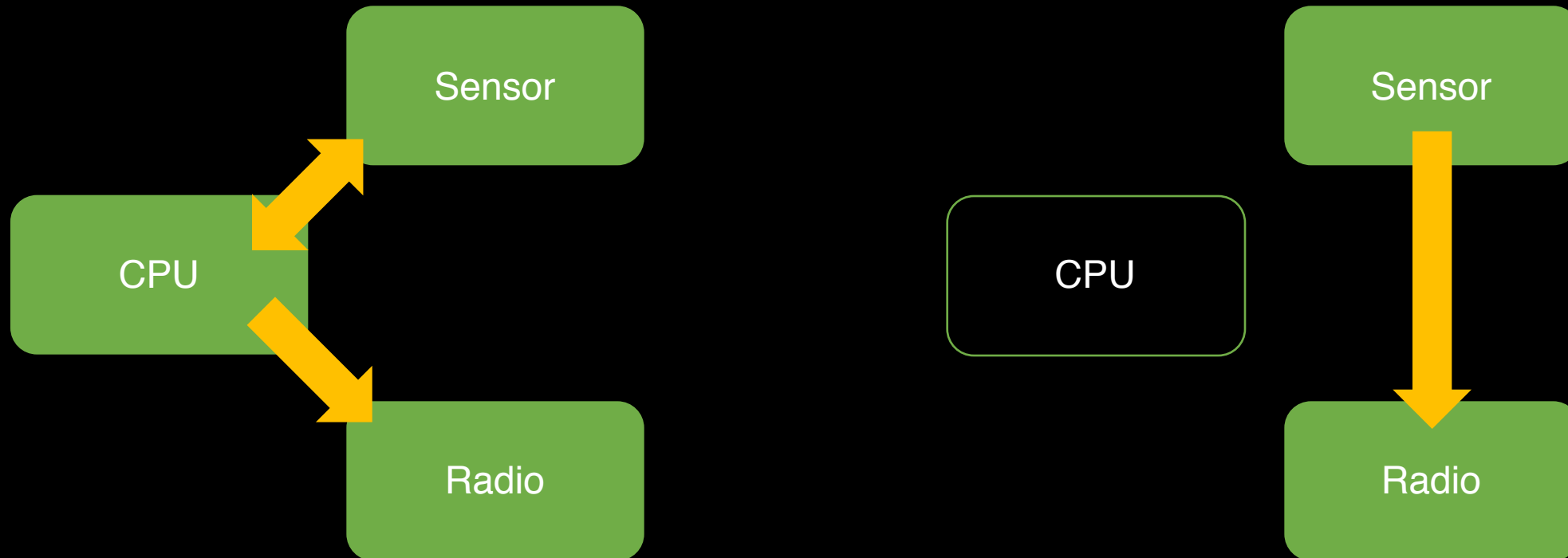
Seamless and transparent interaction between power-aware and power-oblivious chips
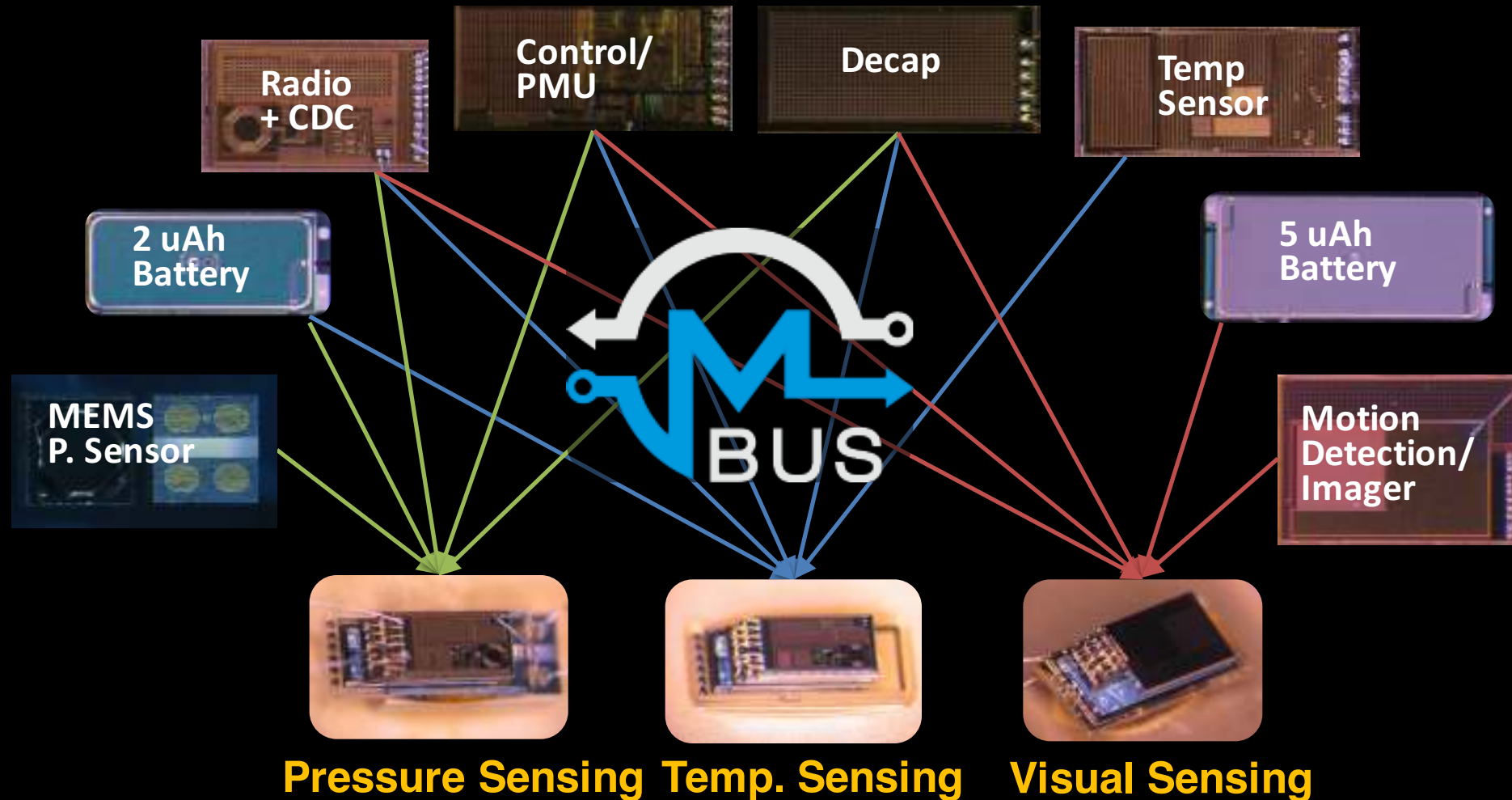
• Facilitates integration with COTS chips
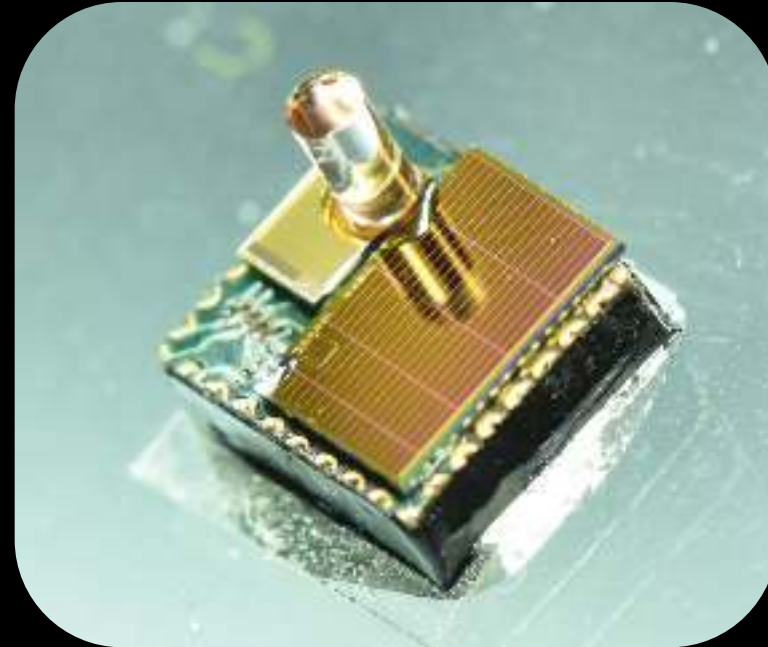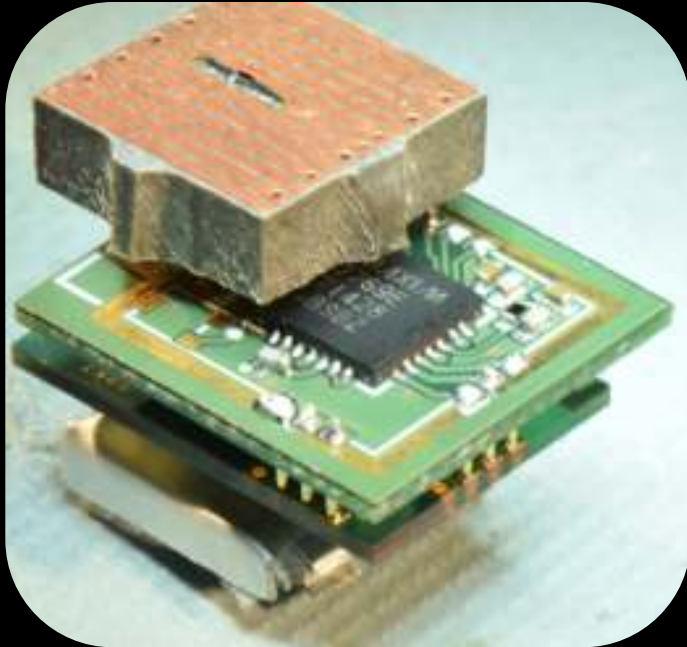
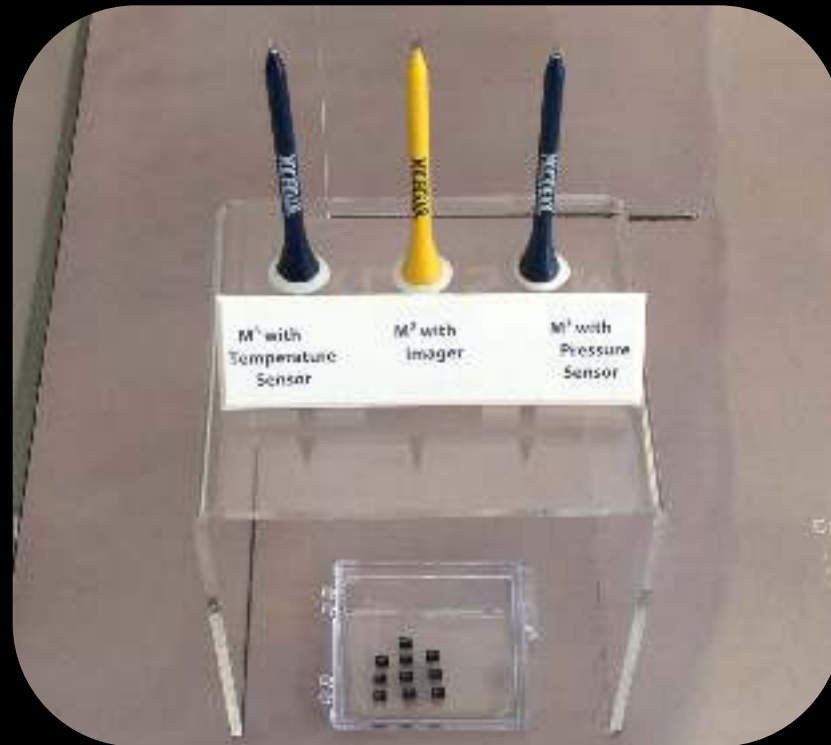*No current COTS chip support MBus, these integrations leverage more traditional buses still

# The majority of interconnect research is focused on performance at the expense of power and area

5 Gb/s

115 b/s

RS232 Serial	Parallel (EPP)	Parallel (ECP)	USB 1.1	USB 2.0	USB 3.0

43.5 million times faster

31.5 GB/s

22.8 MB/s

ISA	PCI	Serial ATA 1.0	AGP 2.0 (4x)	Sata 3.2	PCIx 4.0 x16

1,381 times faster

Specification and Verilog at http://mbus.io

Pressure Sensing **Temp. Sensing** **Visual Sensing**

MBus-based smart dust now on display at the computer history museum

Protocol Overhead and message length
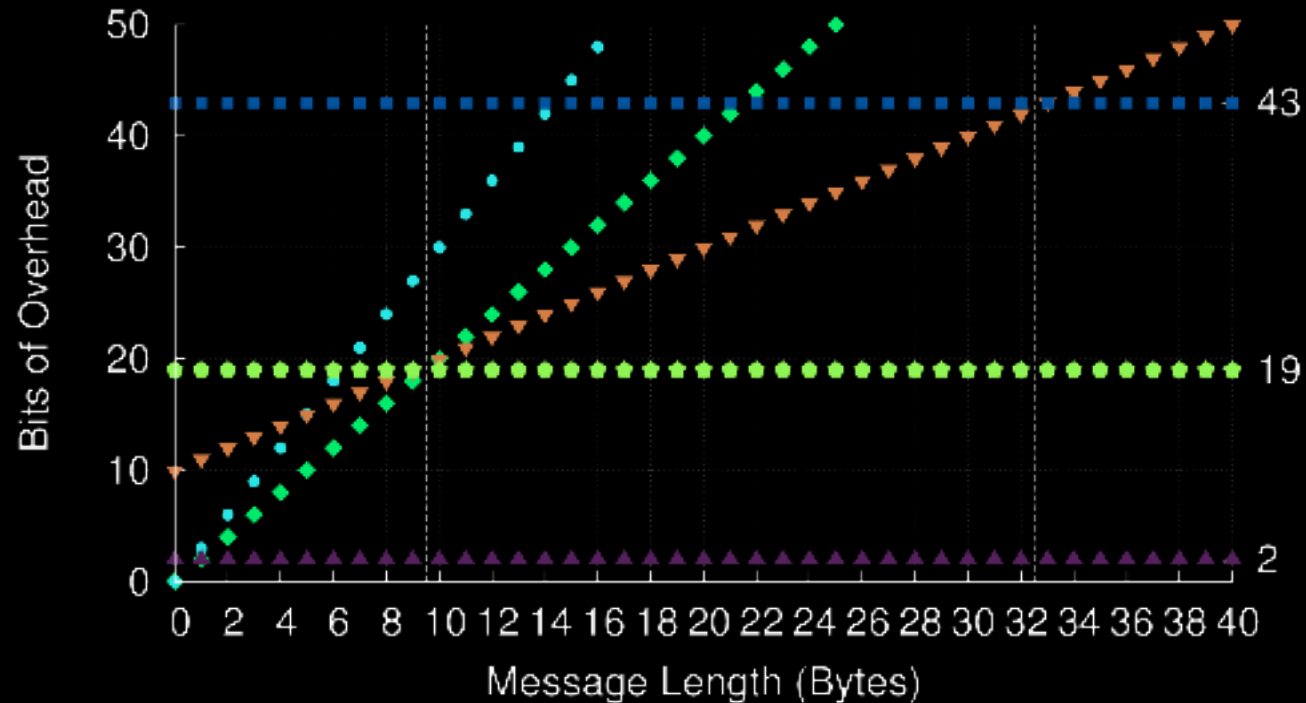
Energy per bit of goodput (useful data)
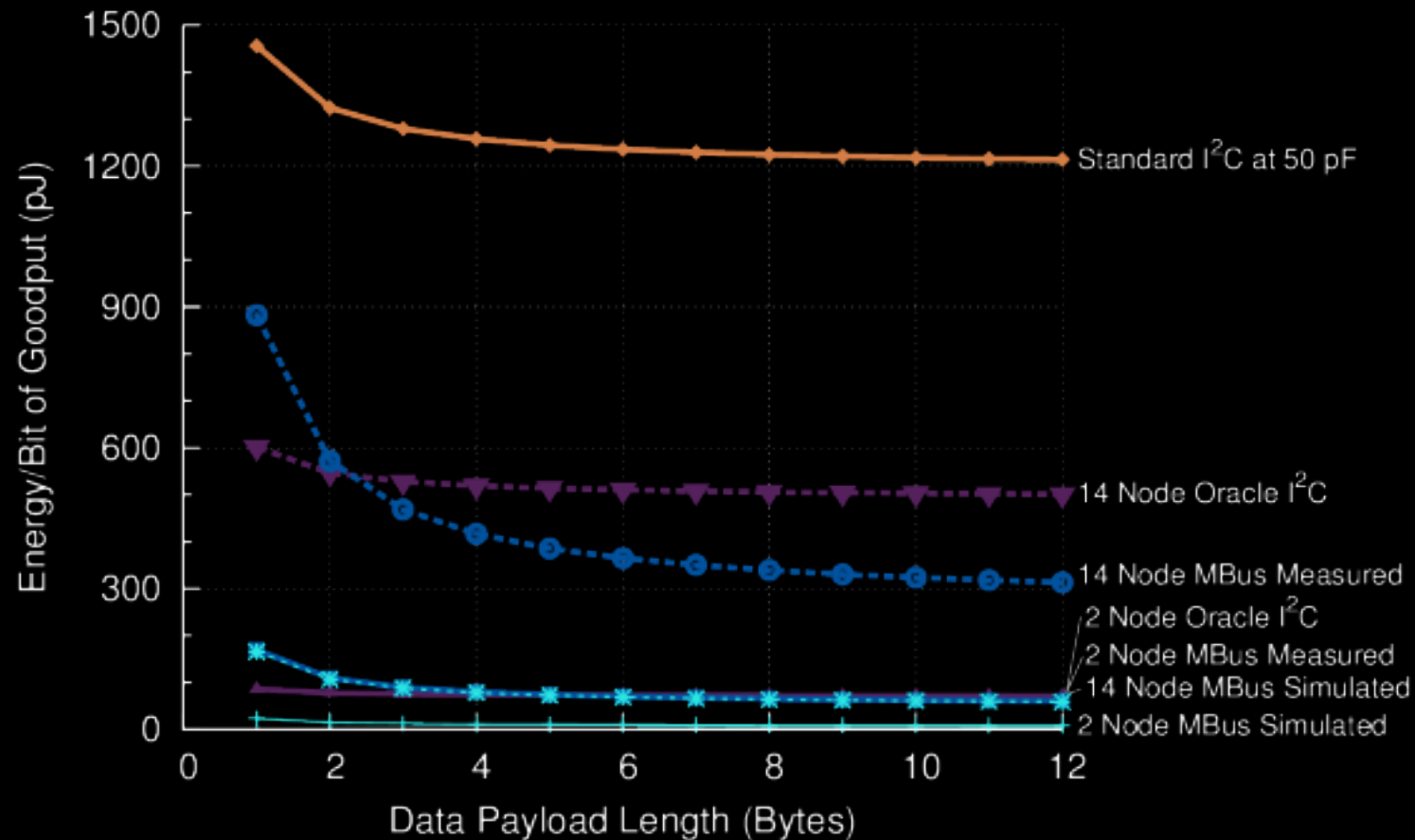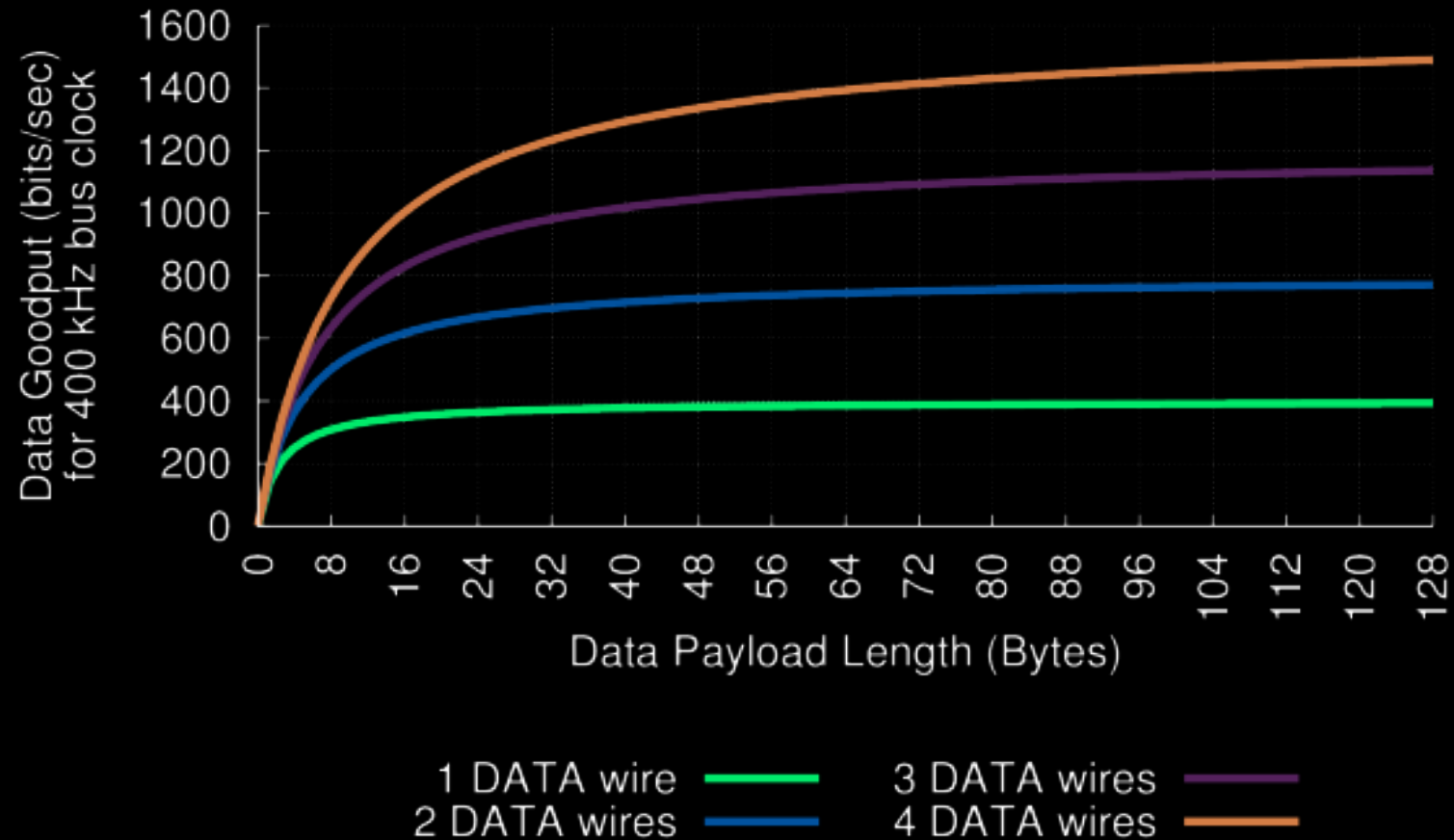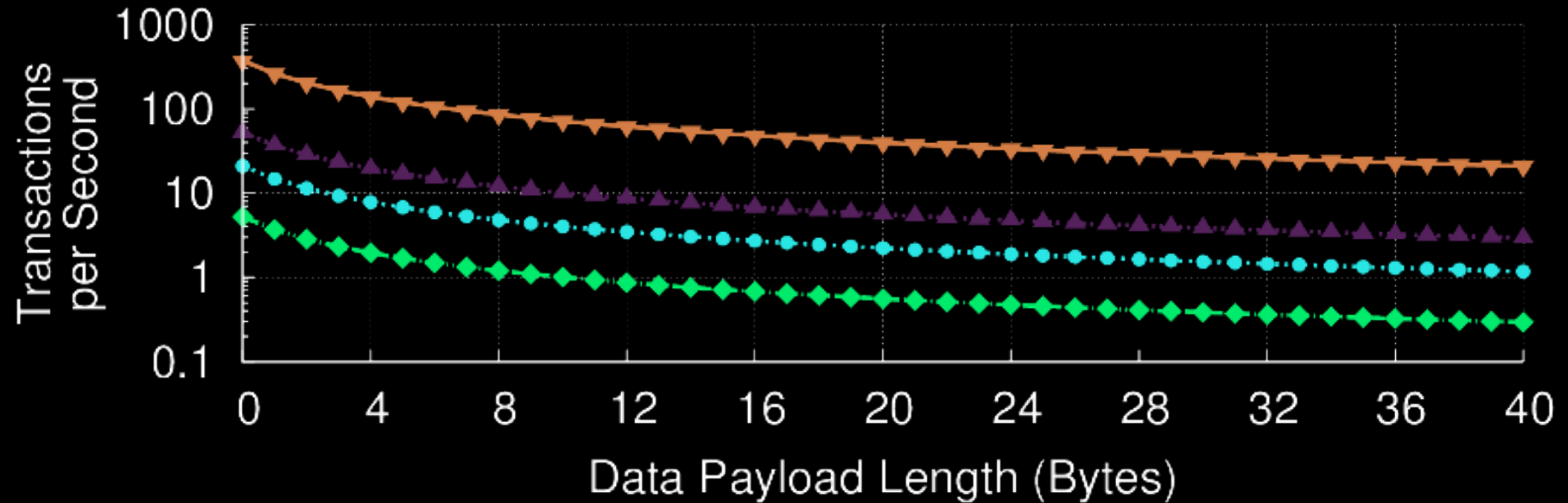
Power Draw Comparison
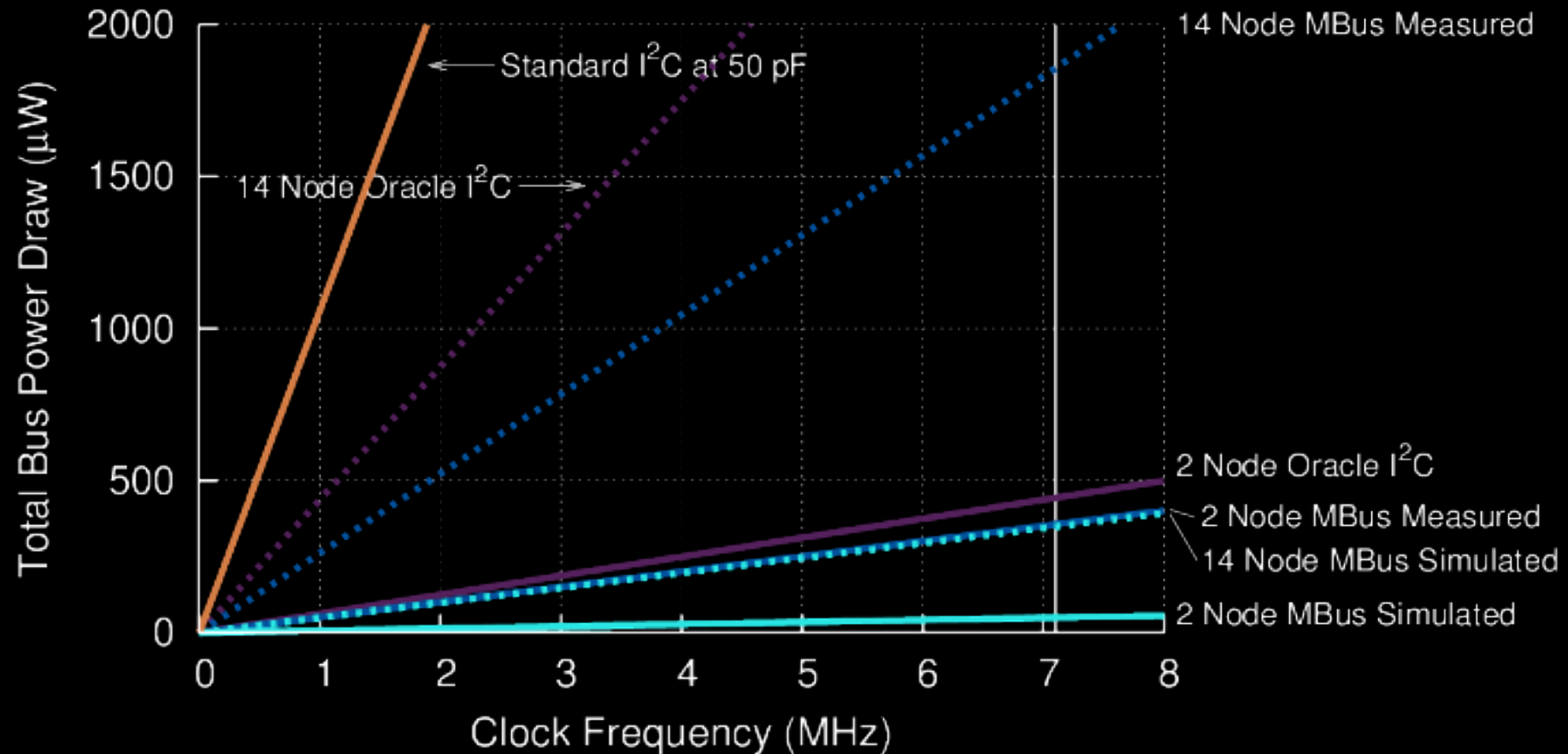
Goodput of parallel MBus

Saturating Transaction Rate

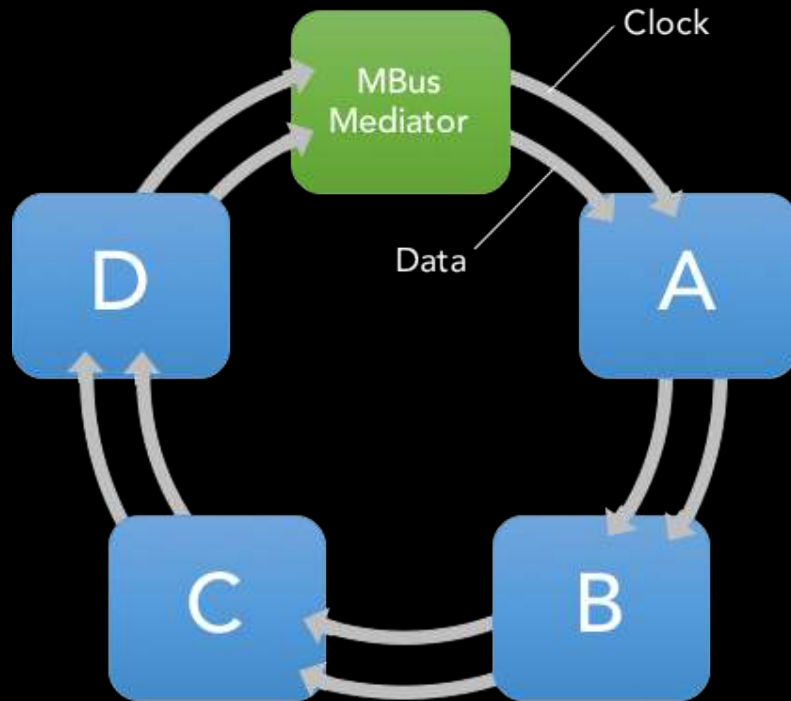# Adding additional nodes does not have significant impact on MBus latency
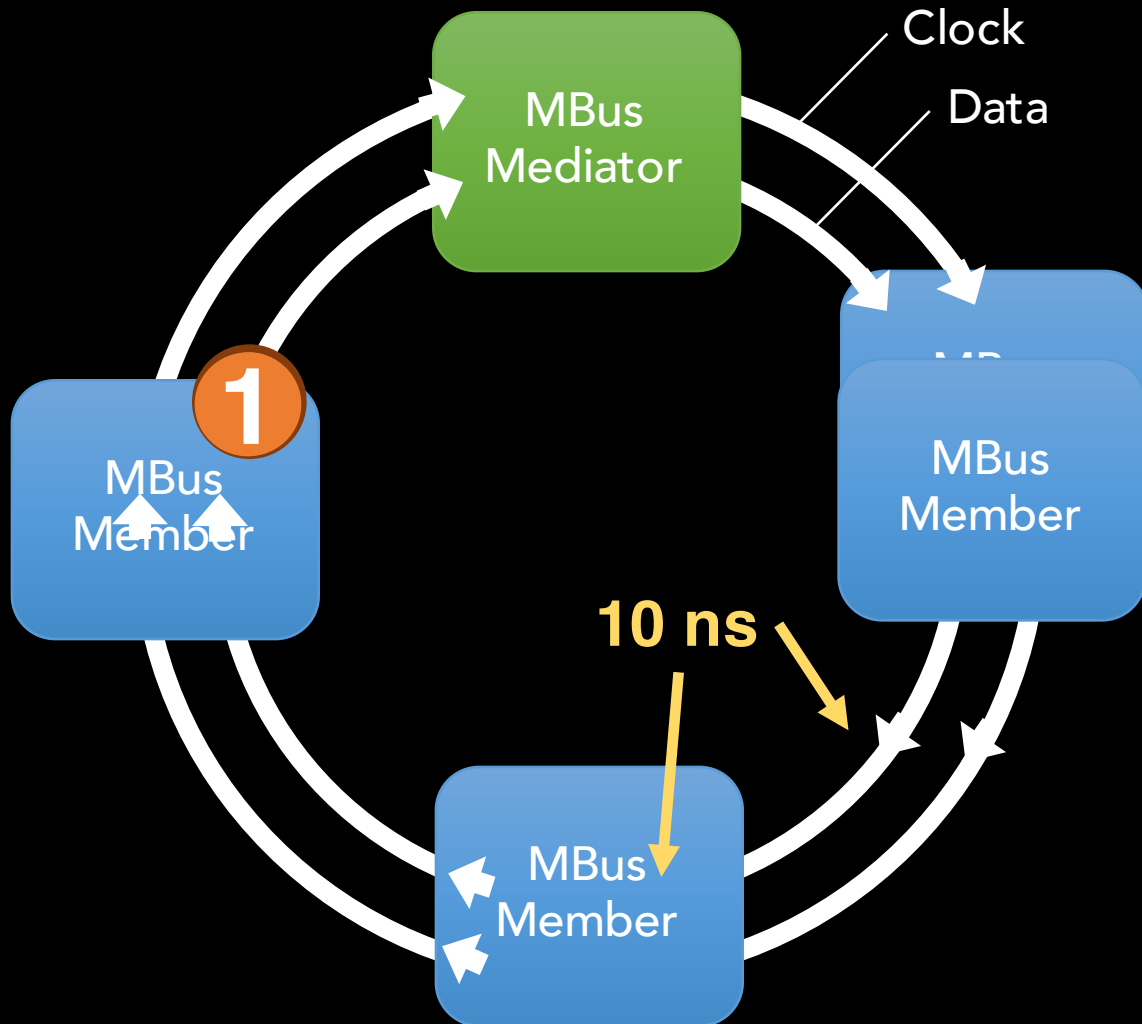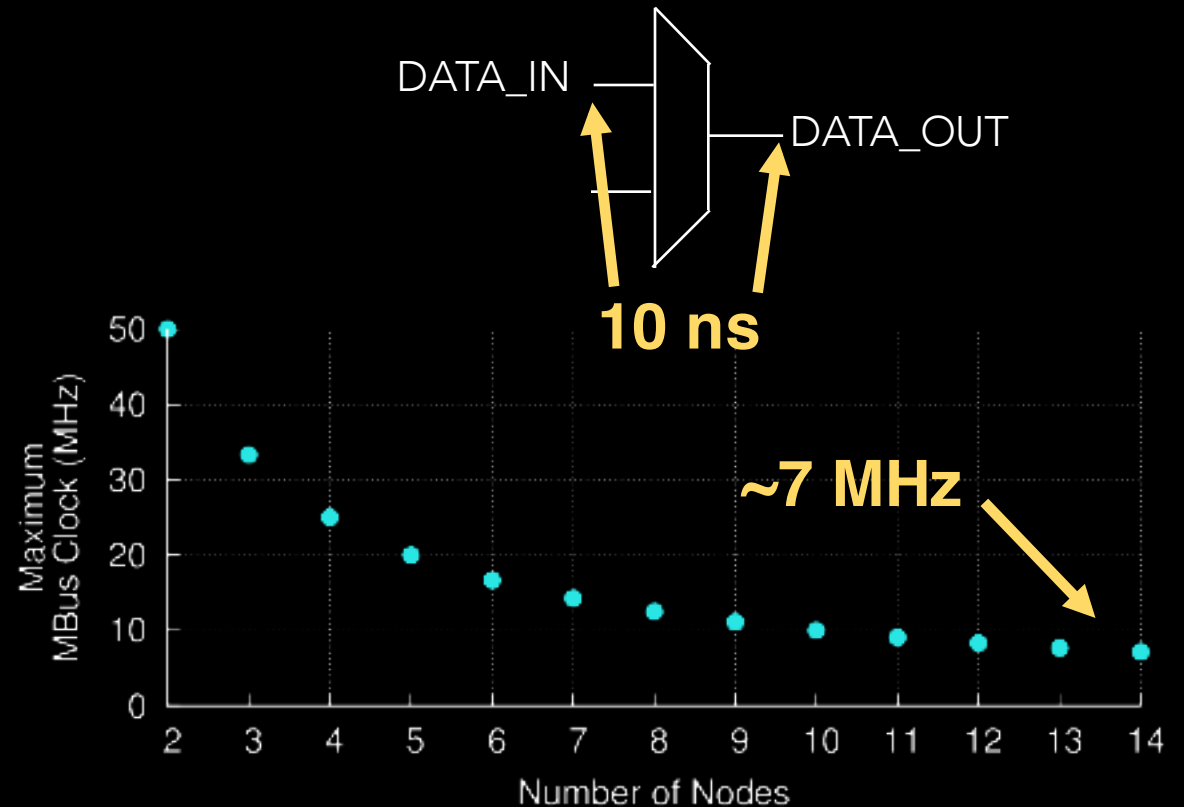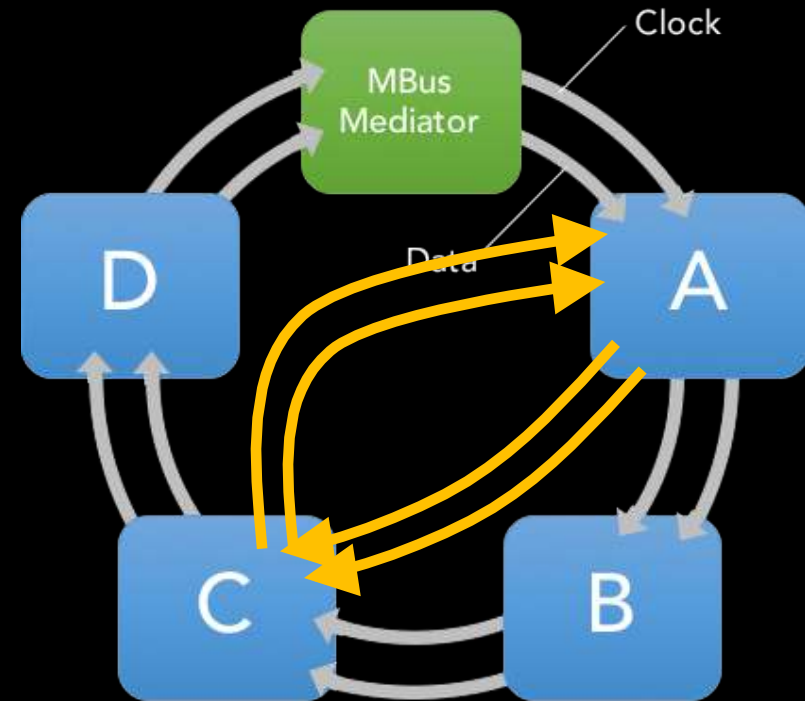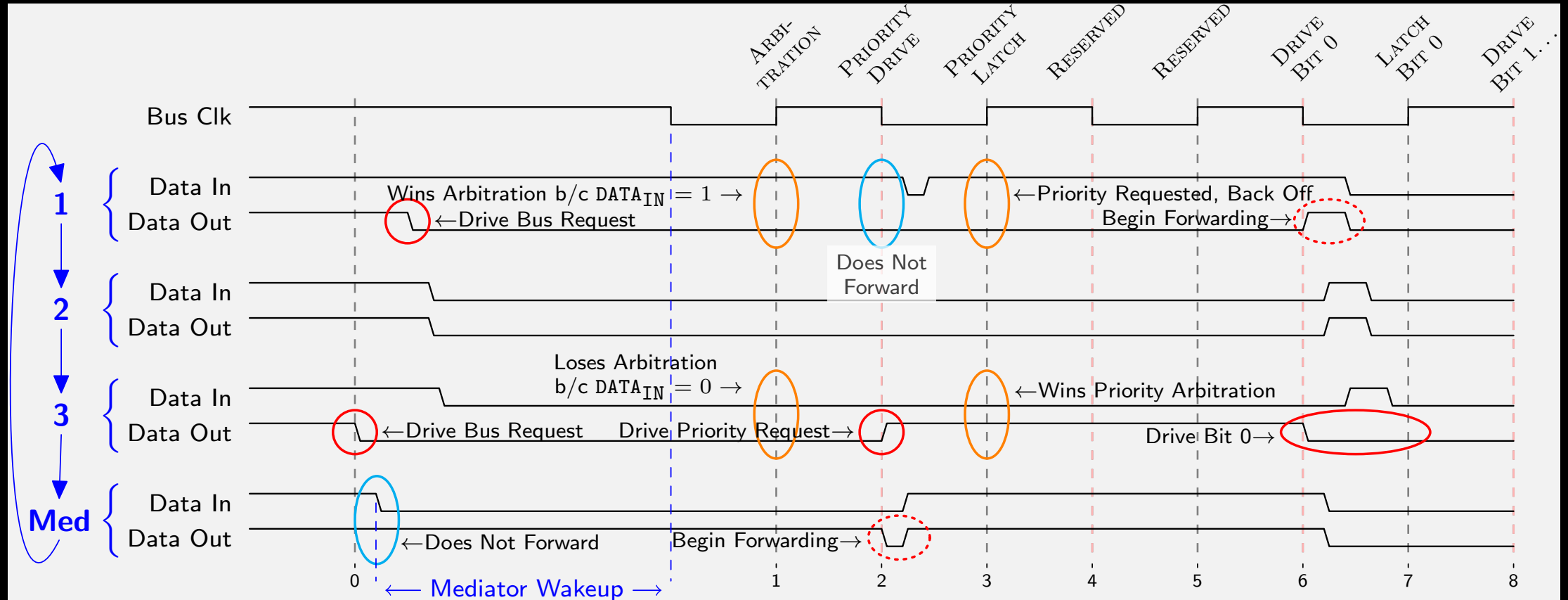
Some low-hanging fruit…

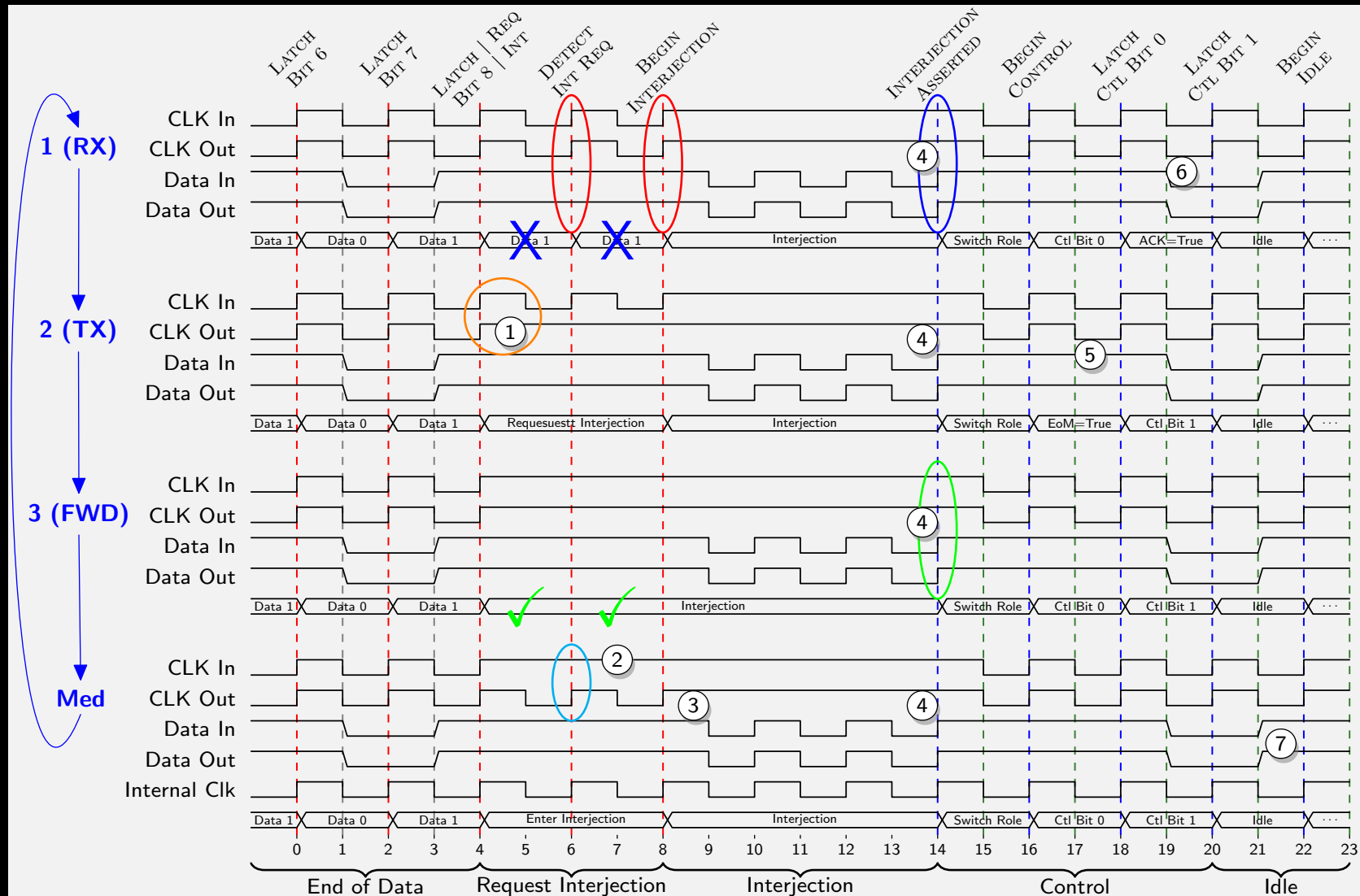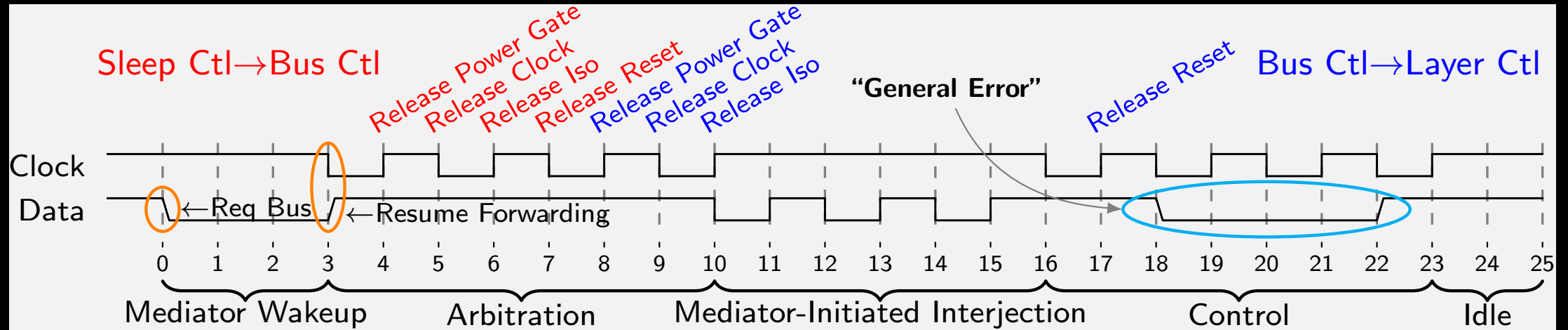- Full Duplex is trivial

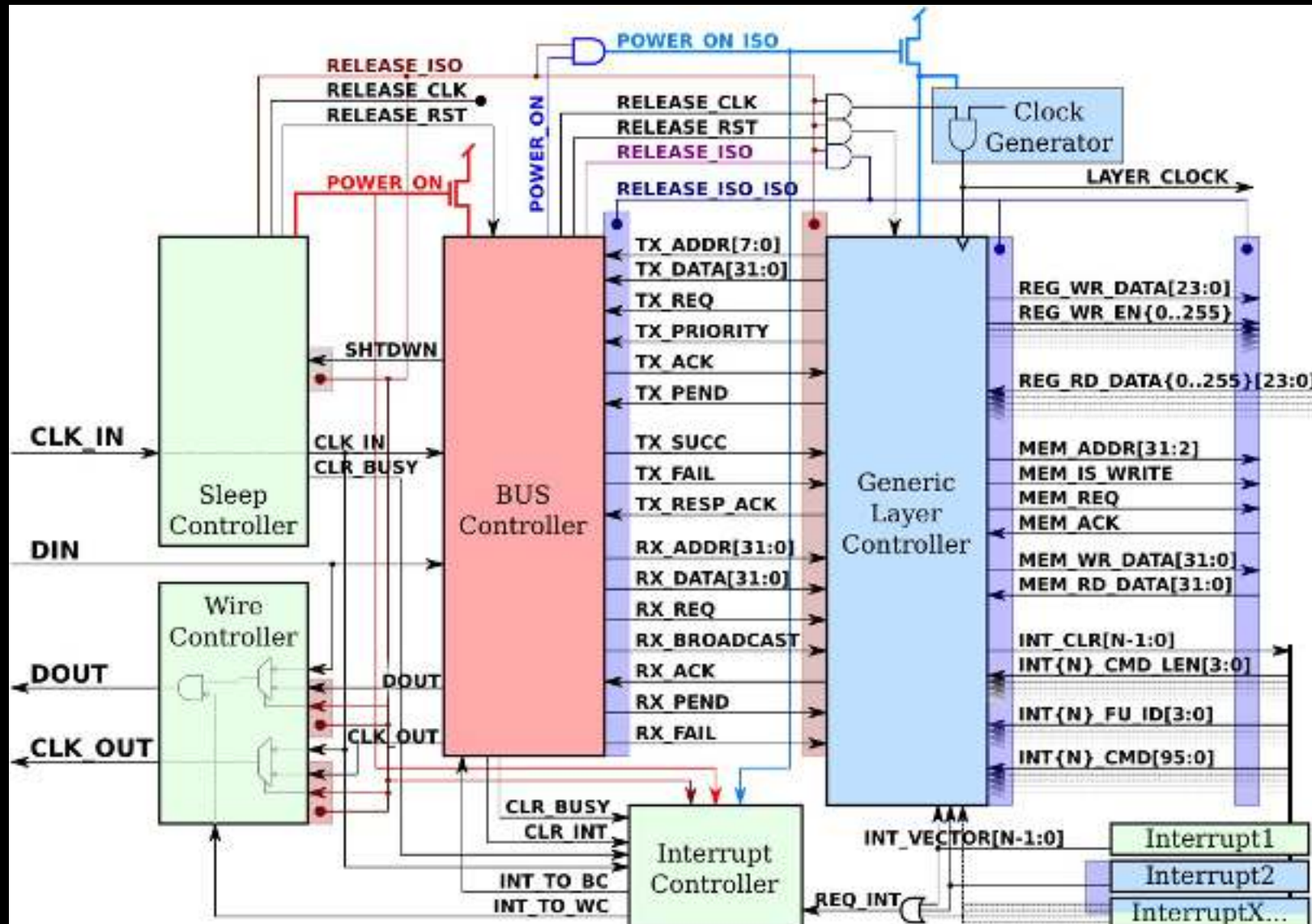- "Selectively parallel"

Arbitration Detail

Interjection Detail

Tertiary node power-on request

Hierarchical Power Domains

Implementation

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