A Component-Based Architecture for Power-Efficient Media Access Control in WSNs

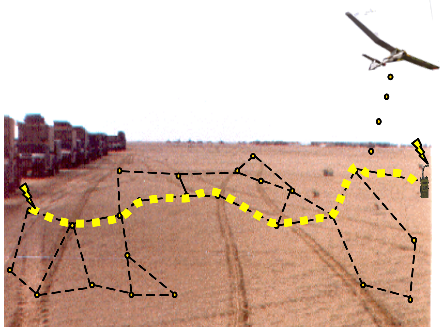
Kevin Klues, Greg Hackmann,
Octav Chipara, Chenyang Lu
Department of Computer Science and Engineering

Washington University in St.Louis

Problem

- Conflicting application requirements
  - Energy
  - Latency
  - Throughput

Habitat Monitoring



Tracking

Structural Health

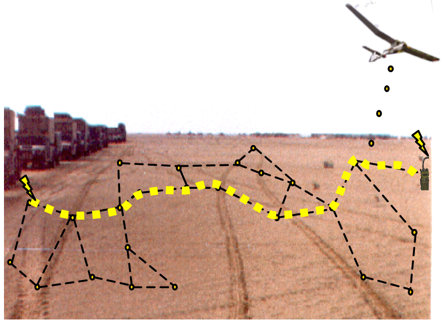
Health Care

Problem

- Conflicting application requirements
  - Energy
  - Latency
  - Throughput

Need different low-level protocols to meet demands of each requirement

Habitat Monitoring



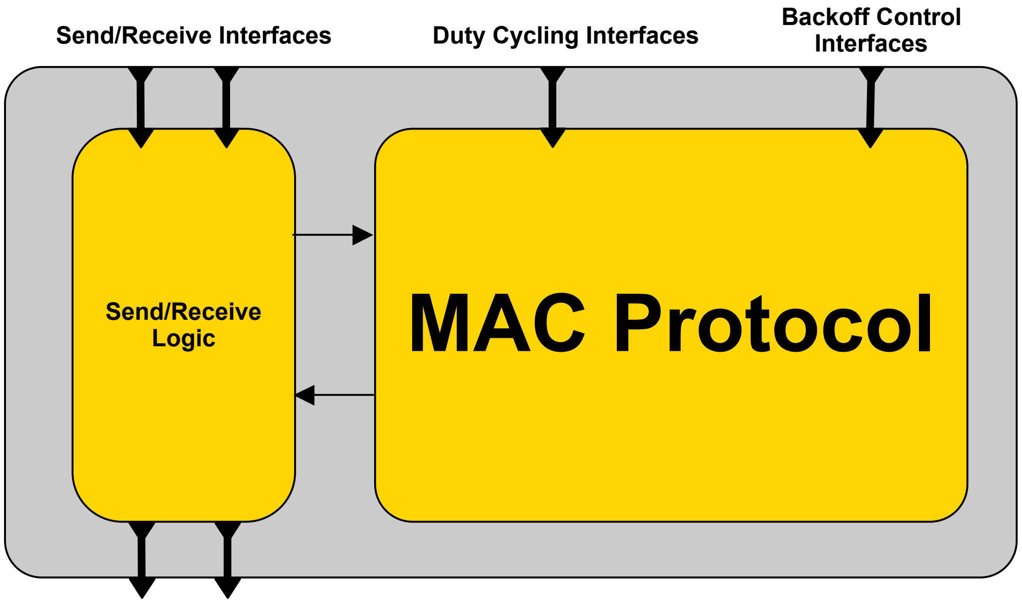
Tracking

Structural Health

Health Care

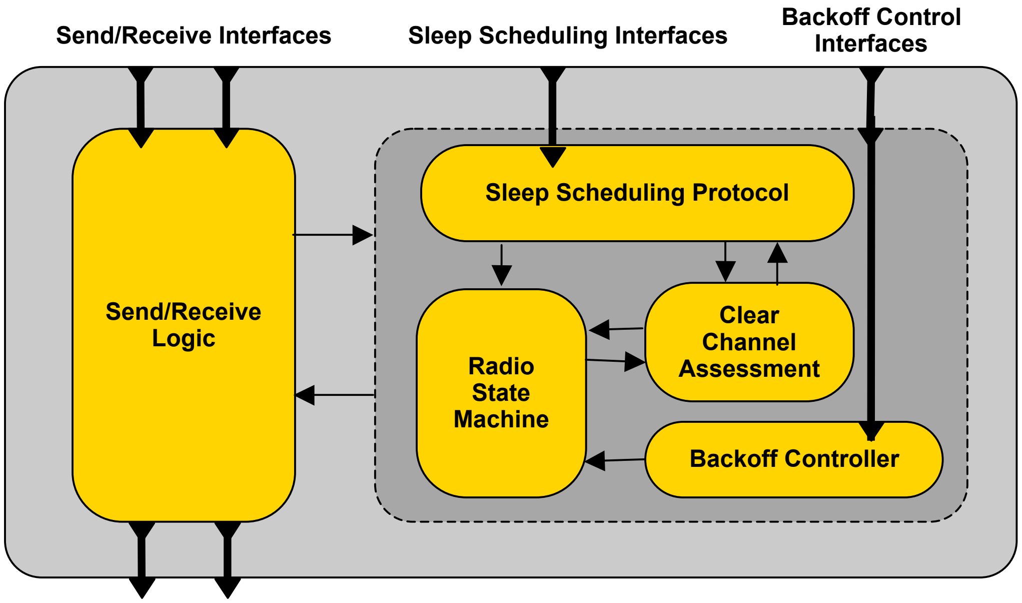
Current Solution

- Design a new MAC protocol
  - ❑ S-MAC
  - ❑ BMAC
  - ❑ ZMAC
  - ❑ XMAC
  - ❑ WiseMAC
  - ❑ T-MAC
  - ❑ SCP
  - ❑ Funnel-MAC
  - ❑ Crankshaft
  - ❑ 802.15.4
  - ❑ DRAND
  - ❑ ....



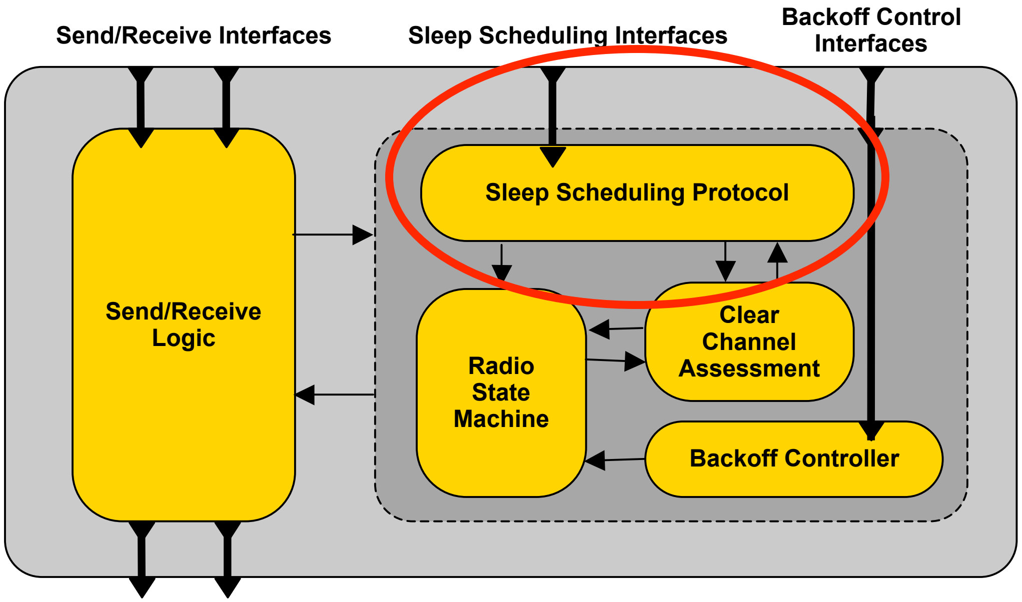
Problem with Current Solution

- Design a new MAC protocol
  - ❑ S-MAC
  - ❑ BMAC
  - ❑ ZMAC
  - ❑ XMAC
  - ❑ WiseMAC
  - ❑ T-MAC
  - ❑ SCP
  - ❑ Funnel-MAC
  - ❑ Crankshaft
  - ❑ 802.15.4
  - ❑ DRAND
  - ❑ ....



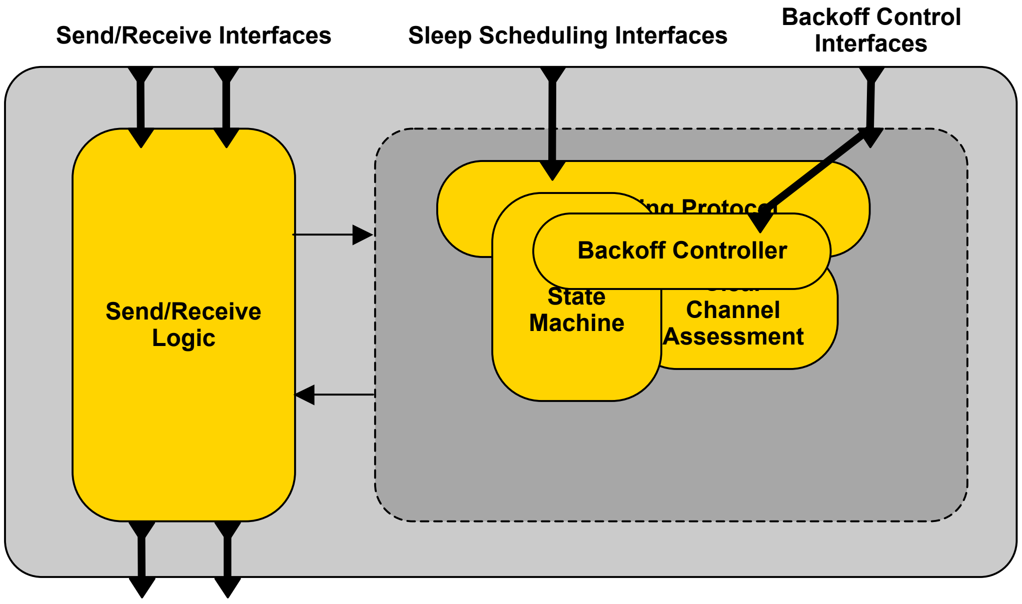
Problem with Current Solution

- Design a new MAC protocol
  - S-MAC
  - BMAC
  - ZMAC
  - XMAC
  - WiseMAC
  - T-MAC
  - SCP
  - Funnel-MAC
  - Crankshaft
  - 802.15.4
  - DRAND
  - ……………



Problem with Current Solution

- Design a new MAC protocol
  - S-MAC
  - BMAC
  - ZMAC
  - XMAC
  - WiseMAC
  - T-MAC
  - SCP
  - Funnel-MAC
  - Crankshaft
  - 802.15.4
  - DRAND
  - ....



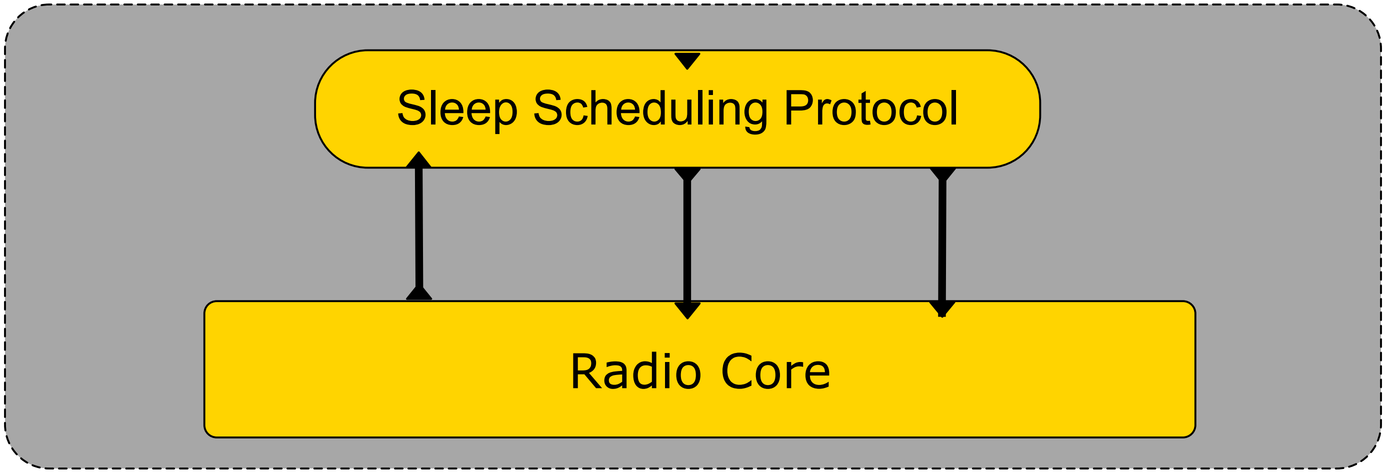
All functionality jumbled into one big monolithic implementation

Problem with Current Solution

- What's wrong with a monolithic radio stack?
  - Hard to develop new MAC protocols
    - No clear separation of concerns
    - Need intimate knowledge of entire stack
  - Protocols not reusable across radio platforms

Problem with Current Solution

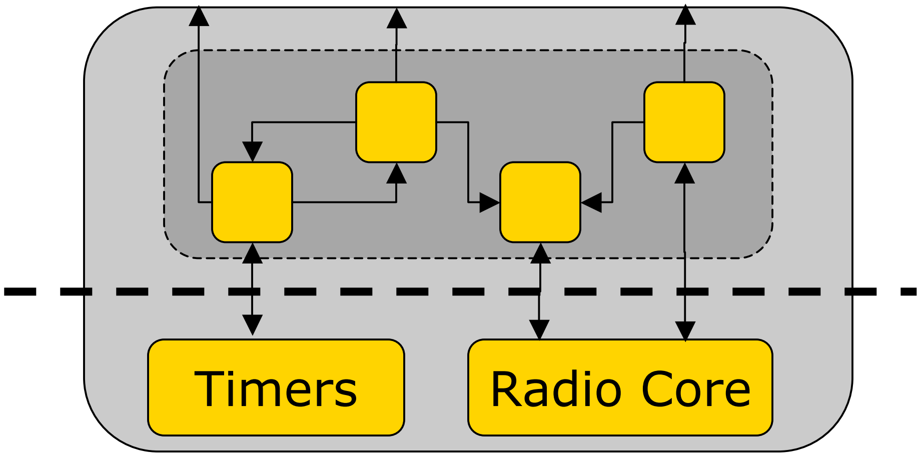
- What's wrong with a monolithic radio stack?
  - Hard to develop new MAC protocols
    - No clear separation of concerns
    - Need intimate knowledge of entire stack
  - Protocols not reusable across radio platforms



Link Layer Support for Flexible Radio Power Management in Wireless Sensor Networks
Kevin Klues, Guoliang Xing, and Chenyang Lu.
In proceedings for The Fifth International Conference on Information Processing in Sensor Networks (IPSN 2007).

Our Contributions

- Extend previous work to componentize sleep scheduling protocols
  - Enables fast development of new protocols
  - Reduces porting effort
- Introduce definition of low-level components enabling more sophisticated protocol development

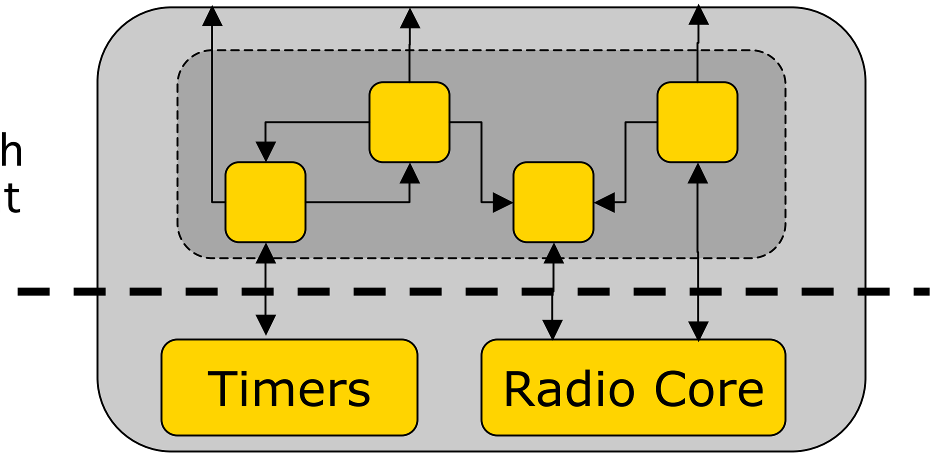


Outline

- MLA: MAC Layer Architecture
- Example B-MAC Implementation
- Evaluation
- Conclusions

MLA: MAC Layer Architecture

- Component-based architecture for MAC protocol development
- Defines two types of components
  - Hardware-independent (portable across platforms)
  - Hardware-dependent (portable interfaces, platform specific implementations)
- Components implement common features of MAC protocols
  - Previous work focused on supporting duty cycling for CSMA protocols
  - Now provide support for TDMA and time-sync through multiple hardware-dependent components



Component Library

Hardware Independent	Hardware Dependent
Preamble Sender	Radio Core
LPL Listener	Local Time
Channel Poller	Alarm
Slot Handlers (TDMA/CSMA)	
Time Synchronization	
Low Level Dispatcher	
Async I/O Adapter	

Component Library

CSMA Protocols (including RTS/CTS above)

Hardware Independent	Hardware Dependent
Preamble Sender	Radio Core
LPL Listener	Local Time
Channel Poller	Alarm
Slot Handlers (TDMA/CSMA)	
Time Synchronization	
Low Level Dispatcher	
Async I/O Adapter	

Component Library

TDMA Protocols

Hardware Independent	Hardware Dependent
Preamble Sender	Radio Core
LPL Listener	Local Time
Channel Poller	Alarm
Slot Handlers (TDMA/CSMA)	
Time Synchronization	
Low Level Dispatcher	
Async I/O Adapter	

Component Library

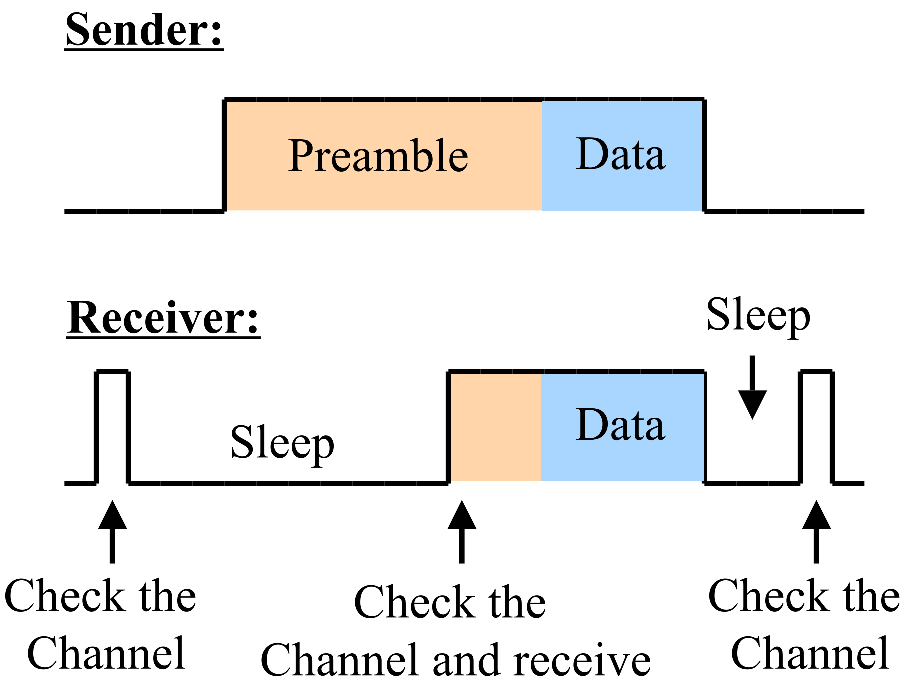
Hybrid Protocols

Hardware Independent	Hardware Dependent
Preamble Sender	Radio Core
LPL Listener	Local Time
Channel Poller	Alarm
Slot Handlers (TDMA/CSMA)	
Time Synchronization	
Low Level Dispatcher	
Async I/O Adapter	

MLA: MAC Layer Architecture

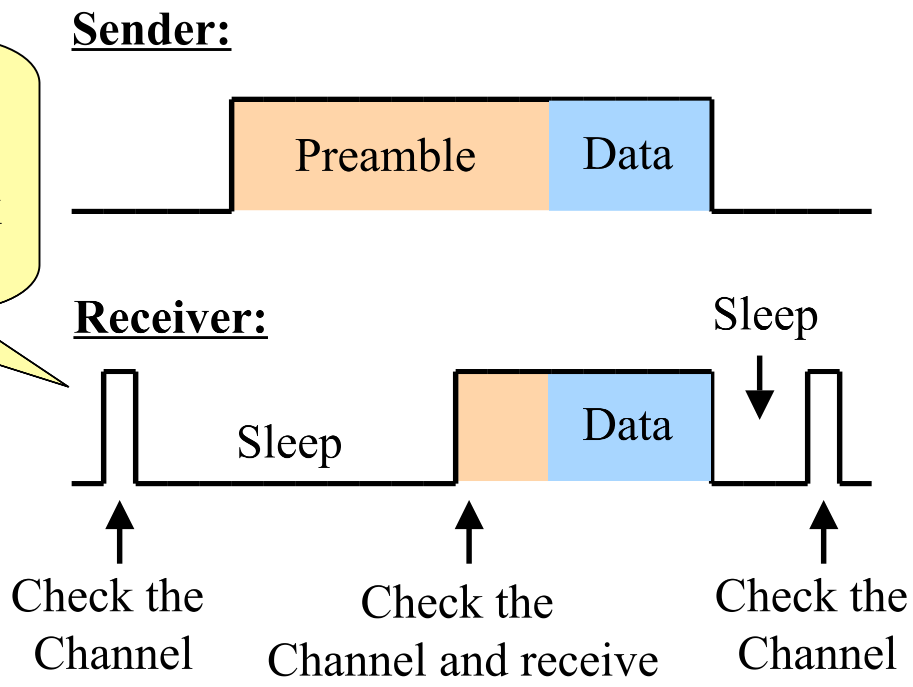
- Allows swappable MAC implementations for differing application requirements
- Enables fast development of new MAC protocols
- Simplifies porting to a new platform
  - Reimplement hardware-dependent components
  - Hardware independent components stay the same
- Comparable to monolithic implementations
  - Memory Footprint
  - Performance
  - Energy efficiency

B-MAC

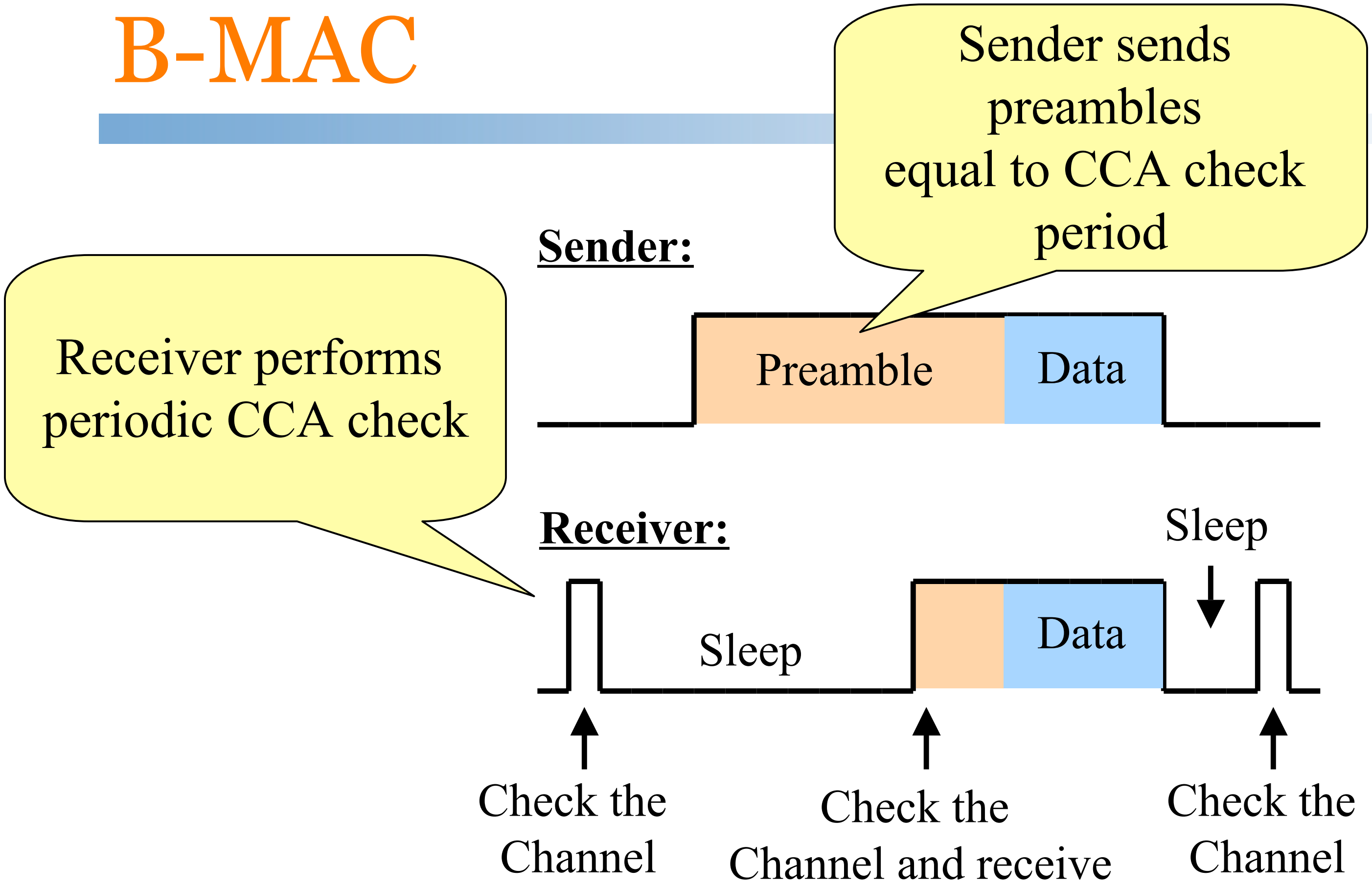

Sender:
Preamble
Data
Receiver:
Sleep
Sleep
Data
Check the Channel
Check the Channel and receive
Check the Channel

B-MAC

Receiver performs periodic CCA check



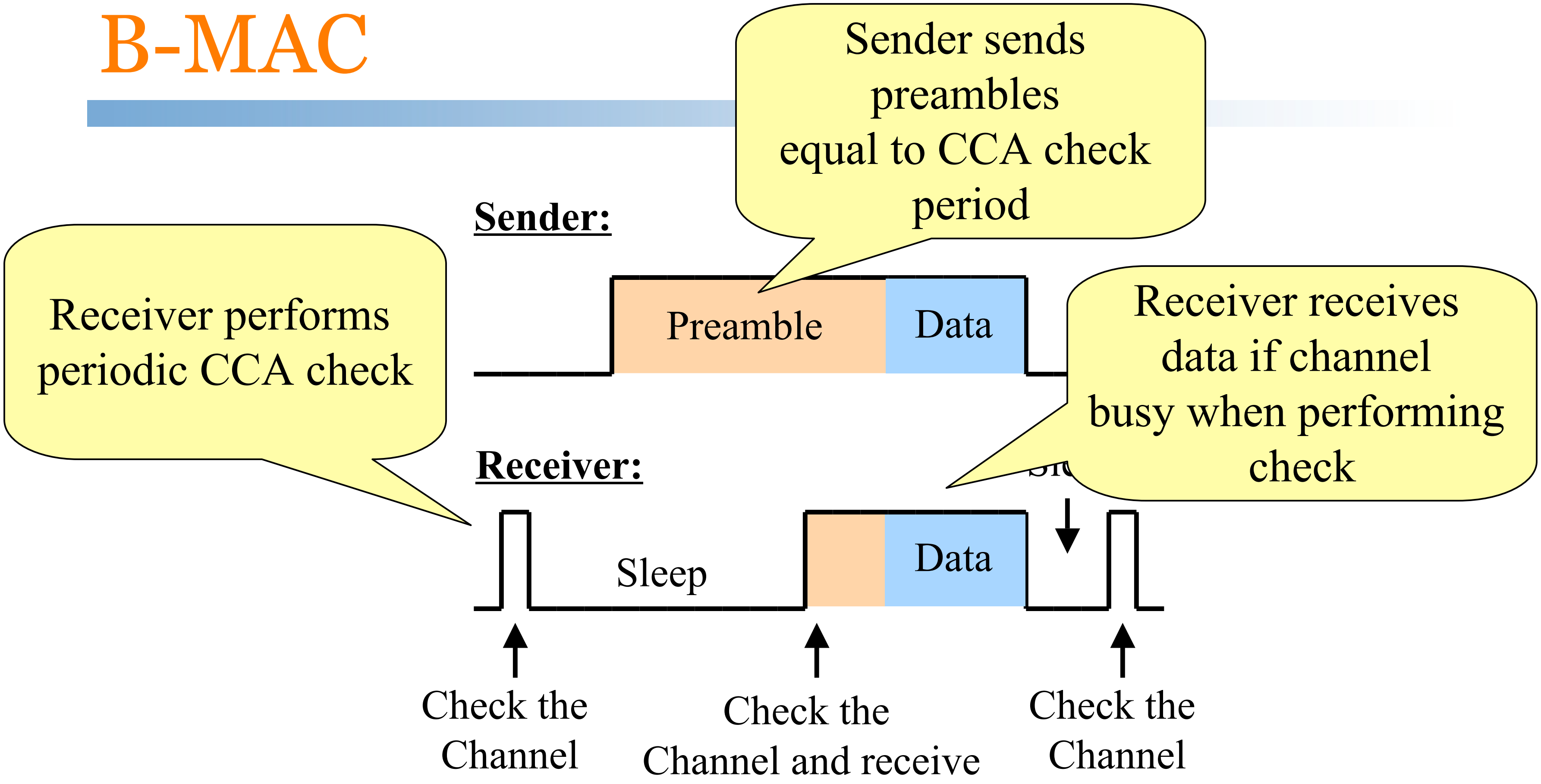
B-MAC



Sender sends preambles equal to CCA check period

Receiver performs periodic CCA check

B-MAC



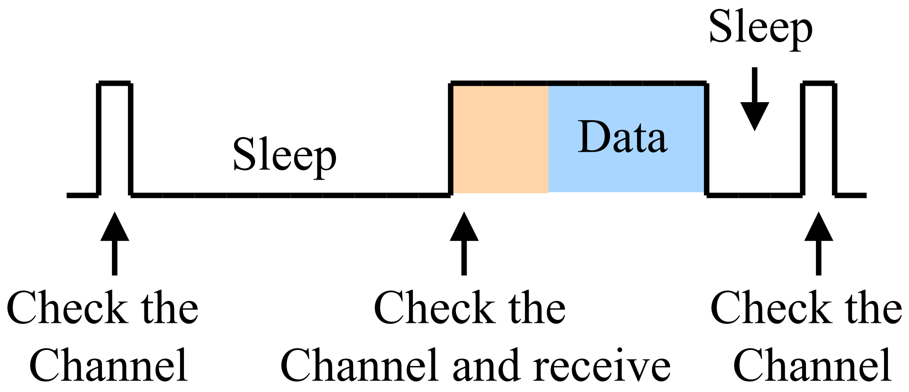
Sender sends preambles equal to CCA check period

Receiver performs periodic CCA check

Receiver receives data if channel busy when performing check

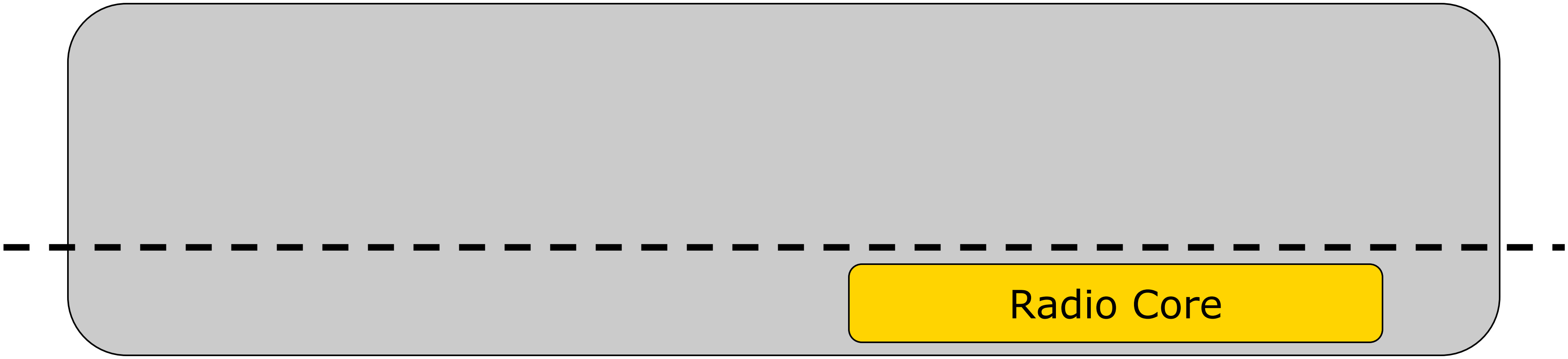
Breakdown of B-MAC

- What does it need?
  - Method of turning the radio on and off
  - Method of checking the channel for radio activity (CCA)
  - Periodic Timer to listen for radio activity
  - A way of sending / receiving preambles
  - A way of sending / receiving data



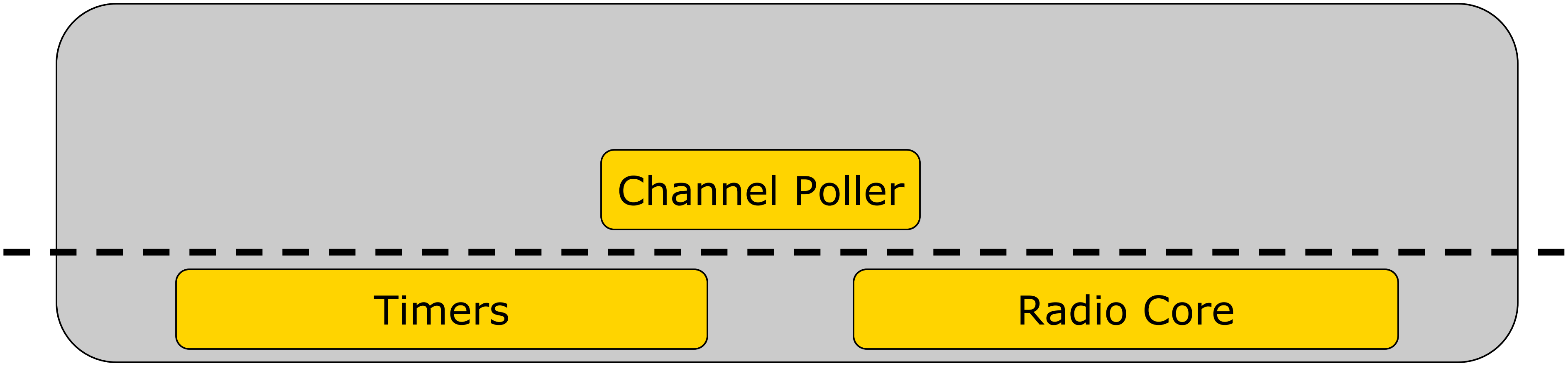
Breakdown of B-MAC

- What does it need?
  - Method of turning the radio on and off
  - Method of checking the channel for radio activity (CCA)



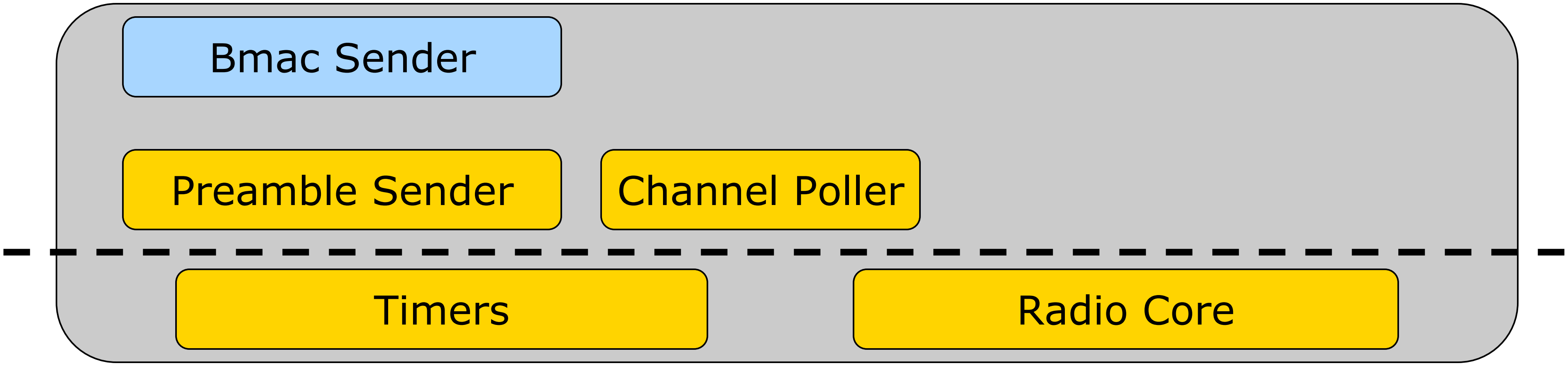
Breakdown of B-MAC

- What does it need?
  - Method of turning the radio on and off
  - Method of checking the channel for radio activity (CCA)
  - Periodic Timer to listen for radio activity



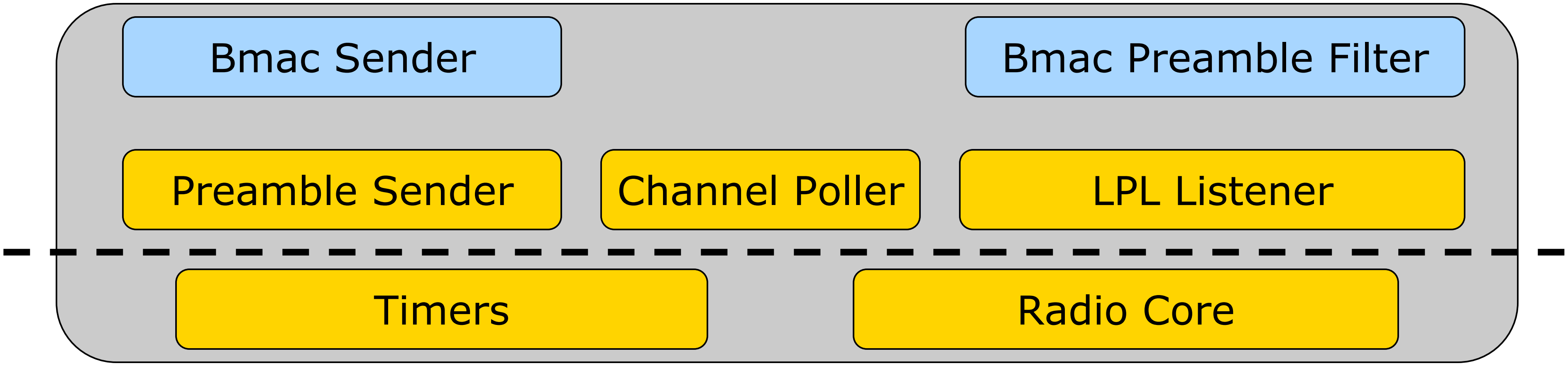
Breakdown of B-MAC

- What does it need?
  - Method of turning the radio on and off
  - Method of checking the channel for radio activity (CCA)
  - Periodic Timer to listen for radio activity
  - A way of sending preambles and data



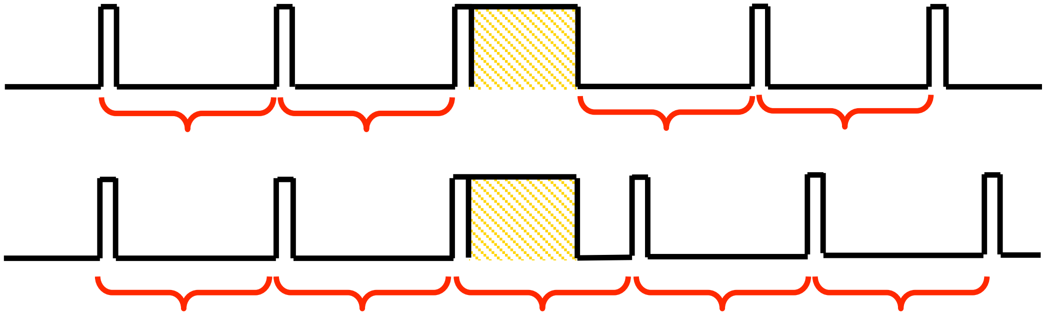
Breakdown of B-MAC

- What does it need?
  - Method of turning the radio on and off
  - Method of checking the channel for radio activity (CCA)
  - Periodic Timer to listen for radio activity
  - A way of sending preambles and data
  - A way of receiving data and filtering out preambles



Other Topics Worth Mentioning

- Customizable CCA checks
  - Uses LocalTime component to control duration of CCA check
  - Larger memory footprint
  - More energy consumption
- Fixed Sleep vs. Periodic LPL Listener



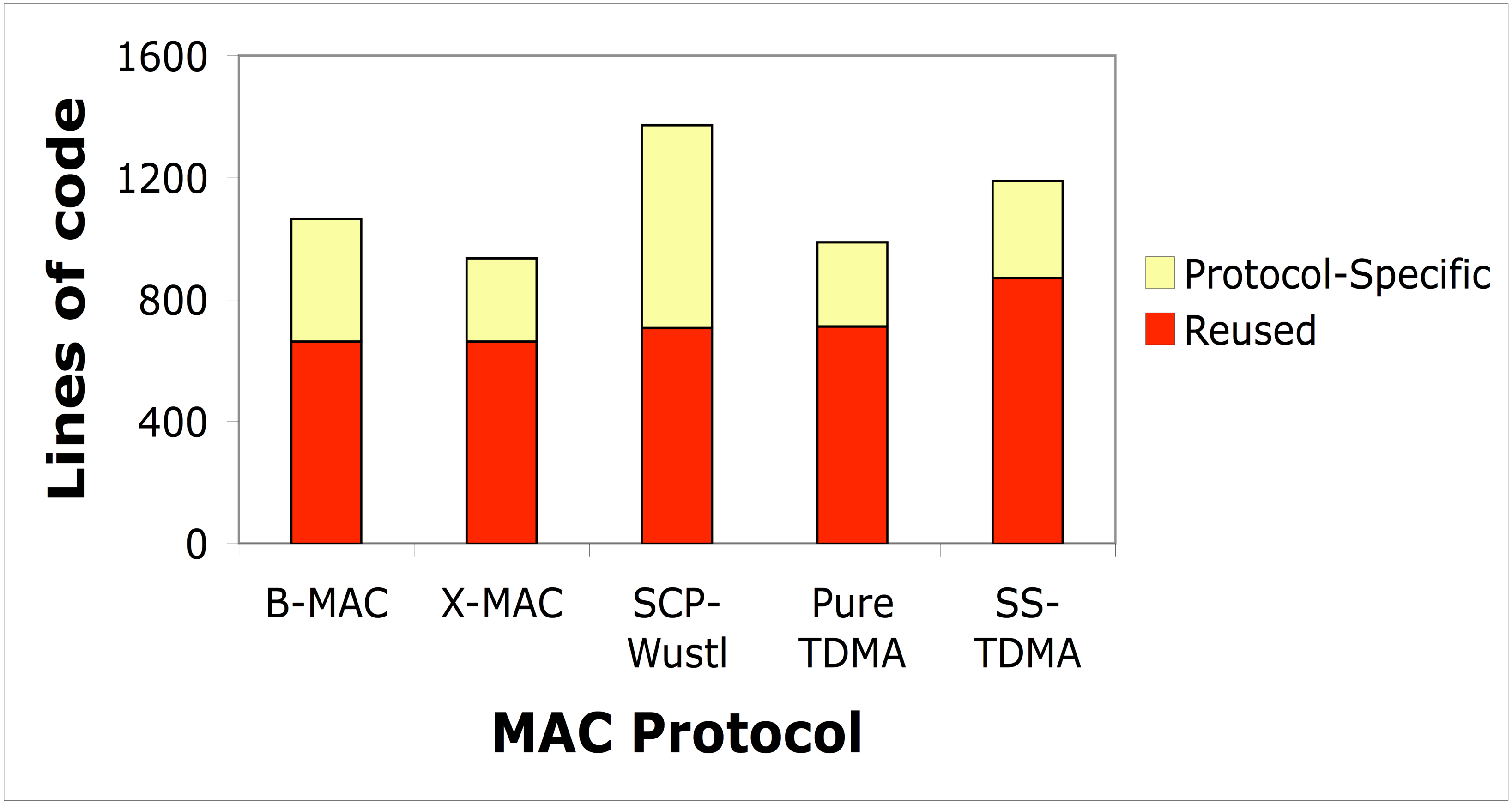
Evaluation

- All evaluations performed on TelosB motes in TinyOS 2.0.1
- Implemented 5 MAC protocols
  - B-MAC, X-MAC, SCP-Wustl, Pure TDMA, SS-TDMA
- Measure
  - Reusability of components among protocols
  - Memory footprint compared to monolithic implementations
  - Throughput
  - Latency
  - Energy Consumption

Reusability of Components

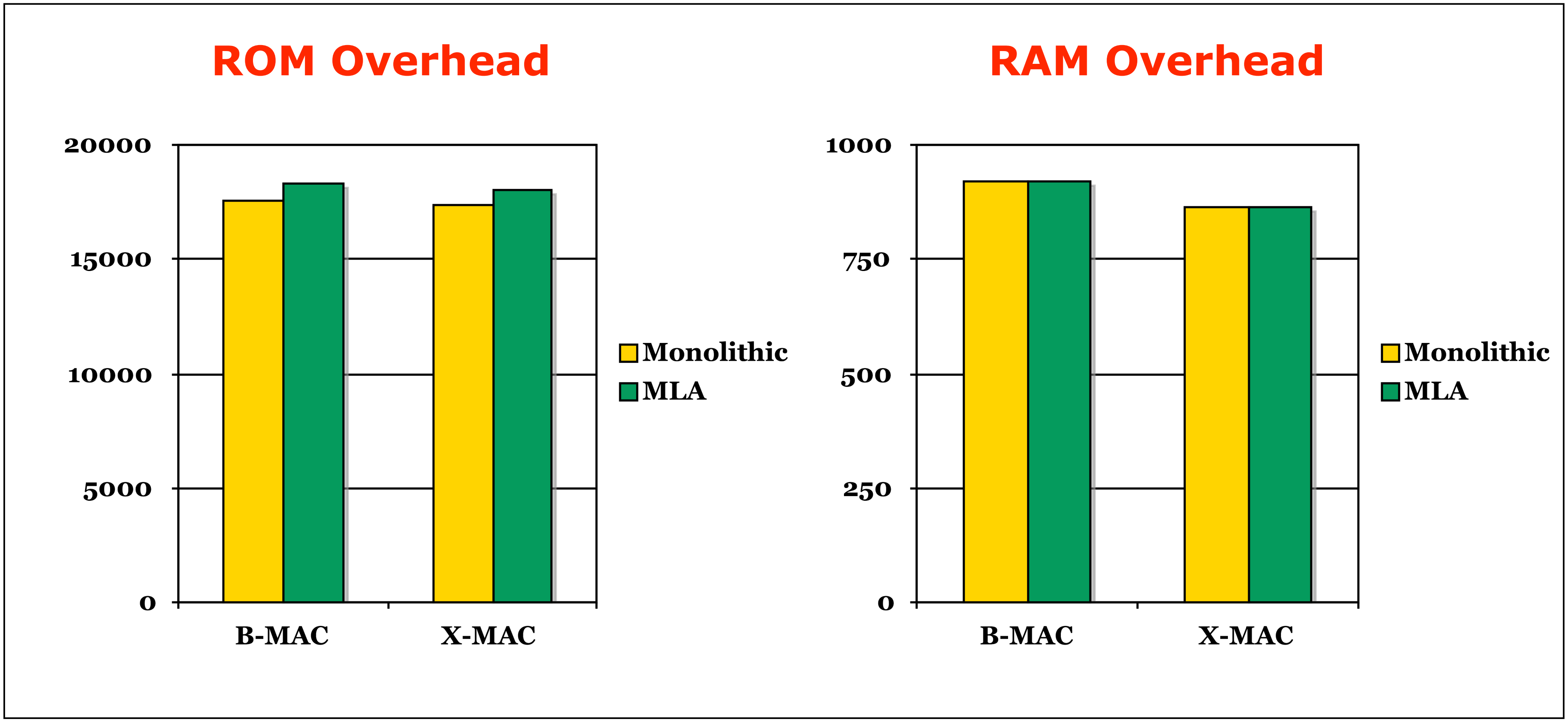
	B-MAC	X-MAC	SCP-Wustl	Pure-TDMA	SS-TDMA
Channel Poller	●	●	●		
LPL Listener	●	●	●		
Preamble Sender	●	●	●		
Time Synchronization			●	●	●
TDMA Slot Handler				●	●
CSMA Slot Handler					●
Low Level Dispatcher				●	●
Async I/O Adapter	●	●	●	●	●
Alarm	●	●	●	●	●
Local Time			●	●	●
Radio Core	●	●	●	●	●
Other Components	3	3	4	2	2
Reused Components	6	6	8	7	8

Code Reuse



Lines of code
1600
1200
800
400
0
B-MAC
X-MAC
SCP-Wustl
Pure TDMA
SS-TDMA
MAC Protocol
Protocol-Specific
Reused

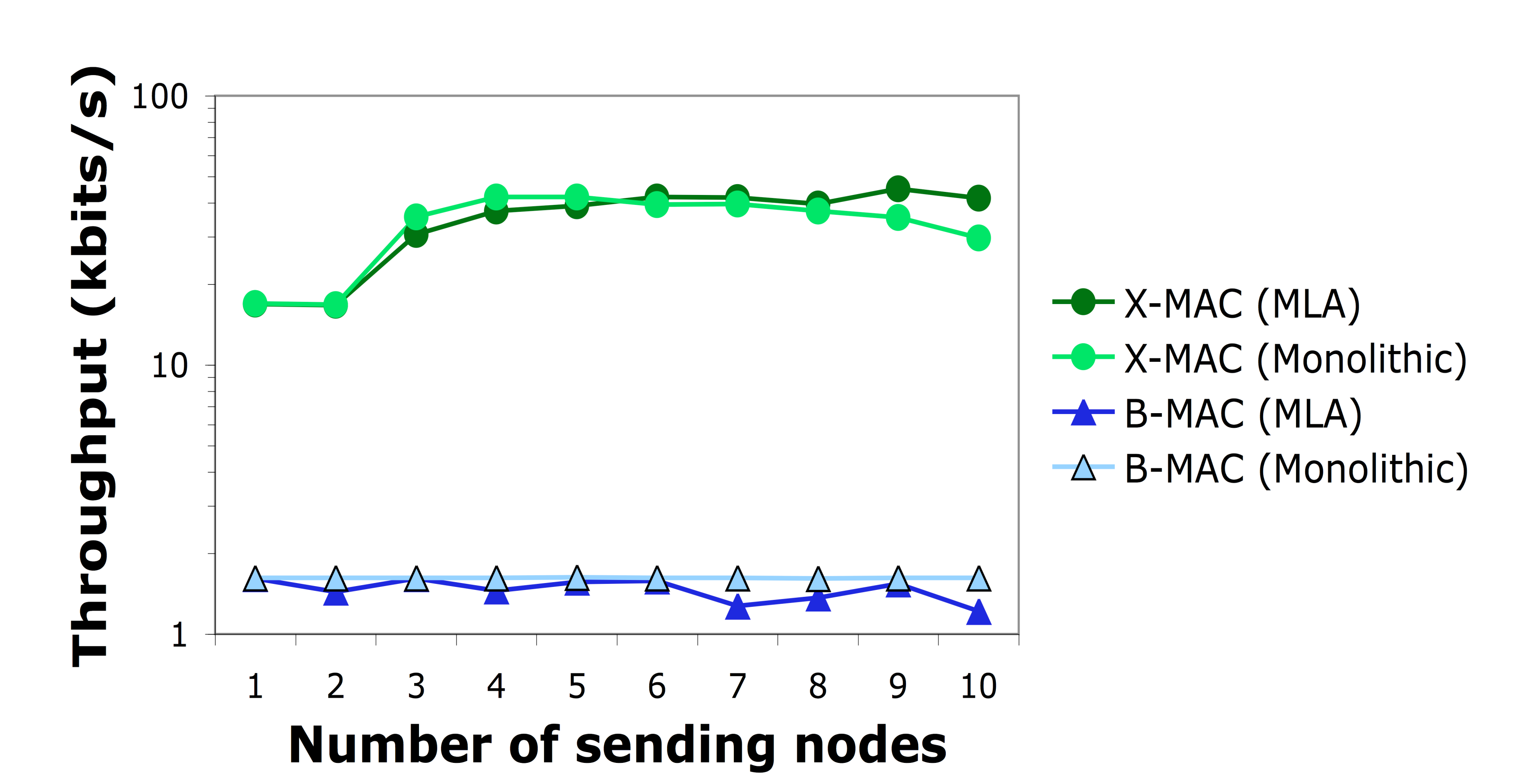
Memory Footprint (TelosB)



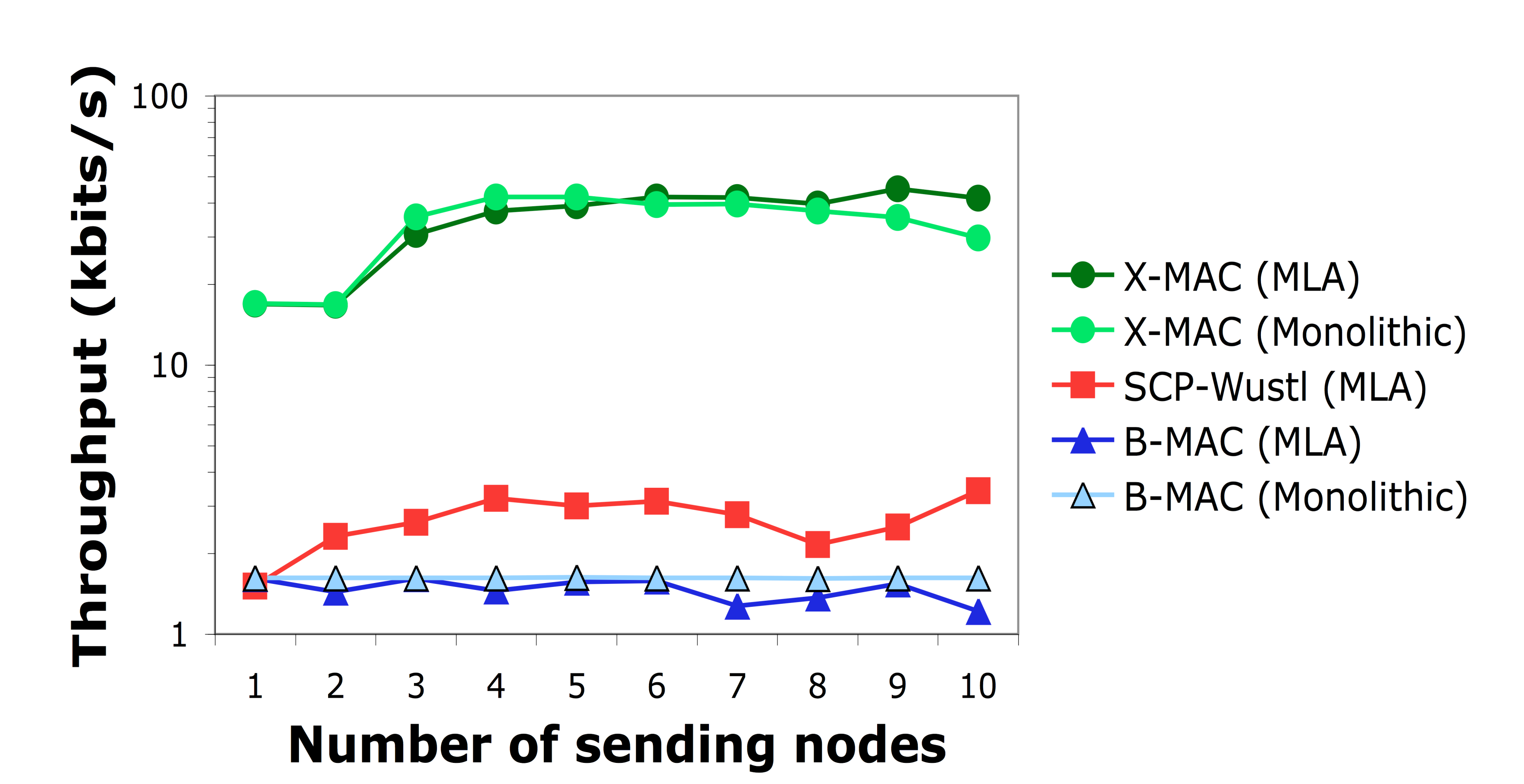
ROM Overhead

RAM Overhead

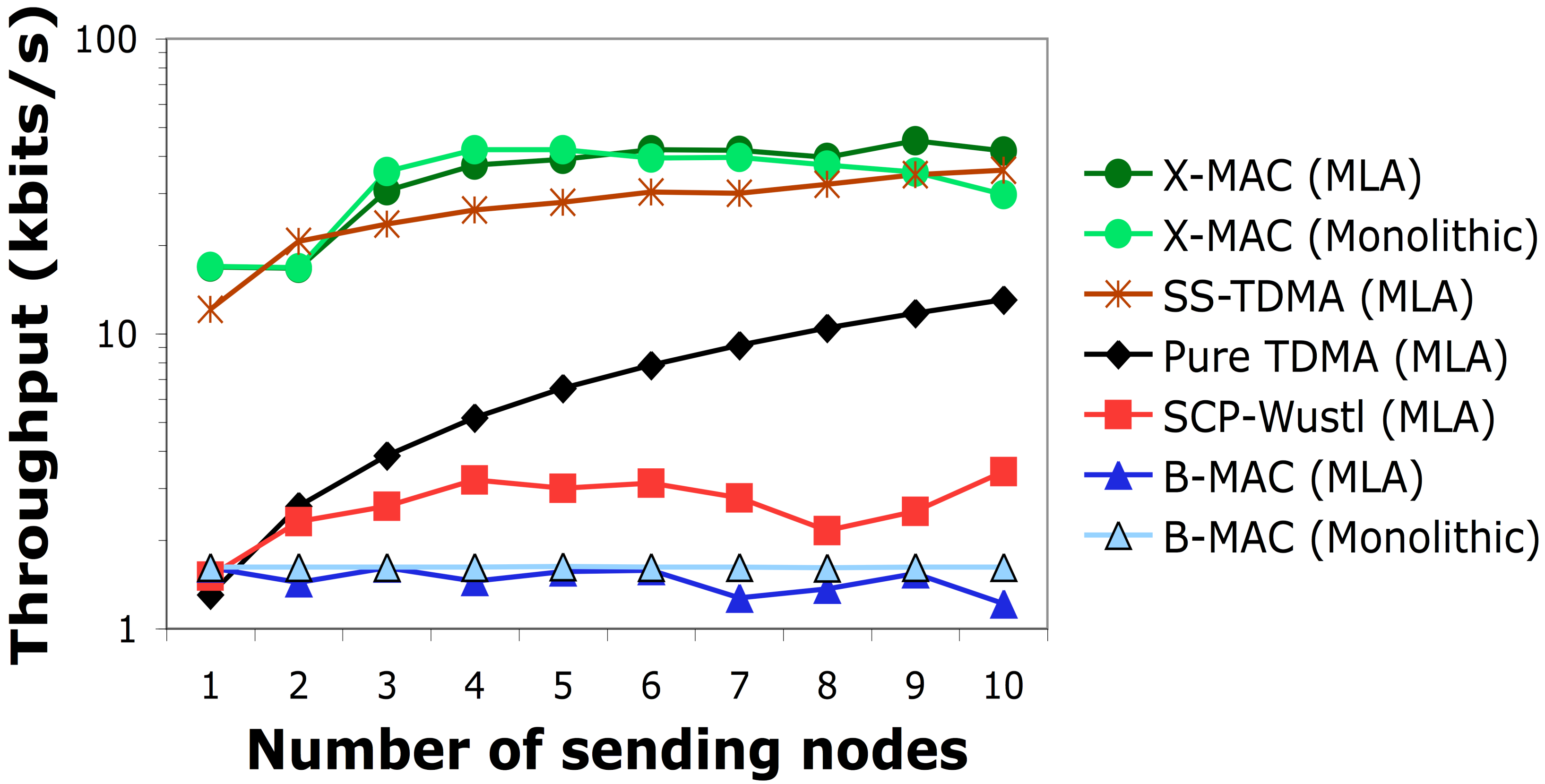
Throughput


Throughput (kbits/s)
100
10
1
1
2
3
4
5
6
7
8
9
10
Number of sending nodes
X-MAC (MLA)
X-MAC (Monolithic)
B-MAC (MLA)
B-MAC (Monolithic)

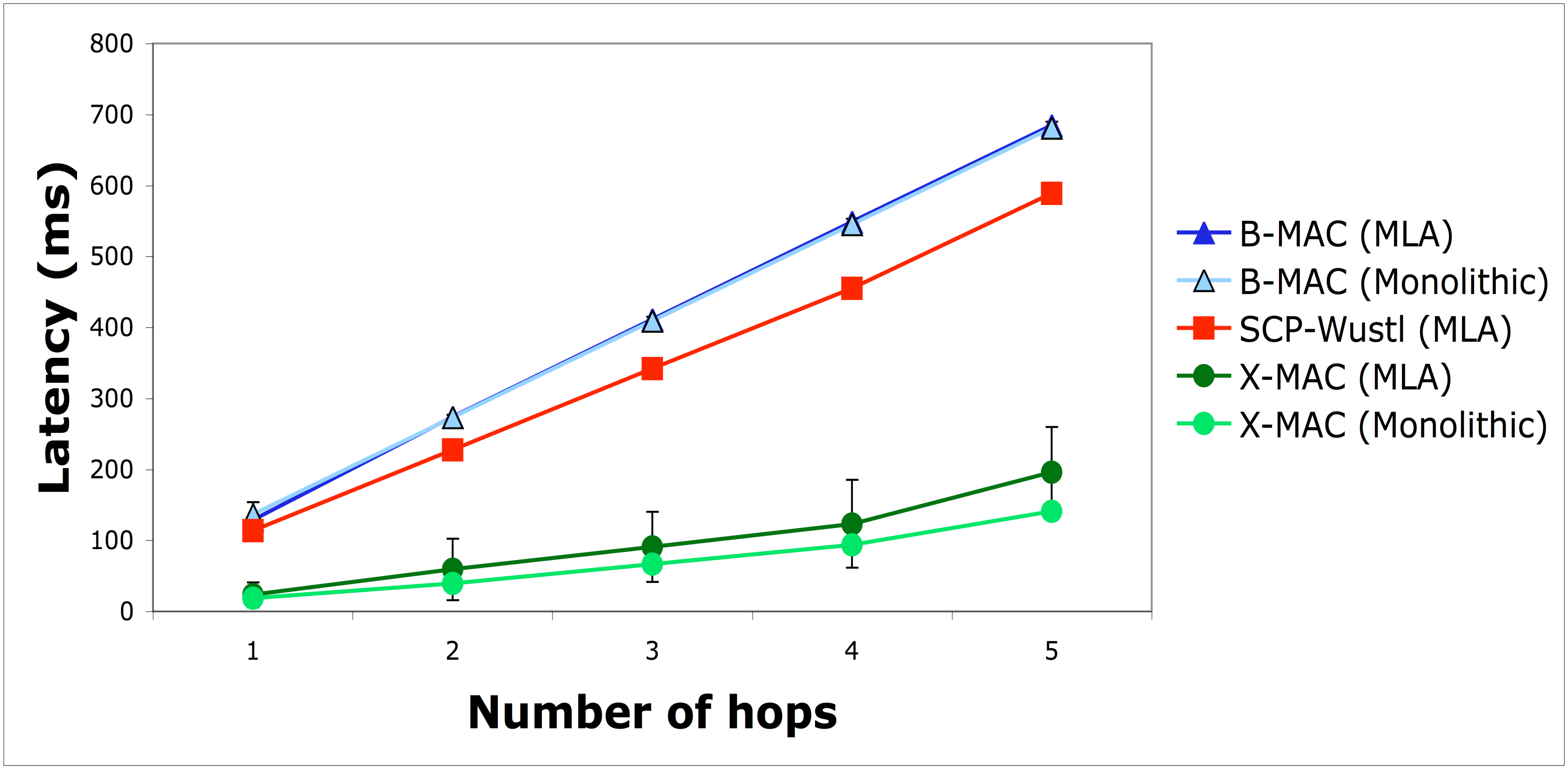
Throughput


Throughput (kbits/s)
100
10
1
1
2
3
4
5
6
7
8
9
10
Number of sending nodes
X-MAC (MLA)
X-MAC (Monolithic)
SCP-Wustl (MLA)
B-MAC (MLA)
B-MAC (Monolithic)

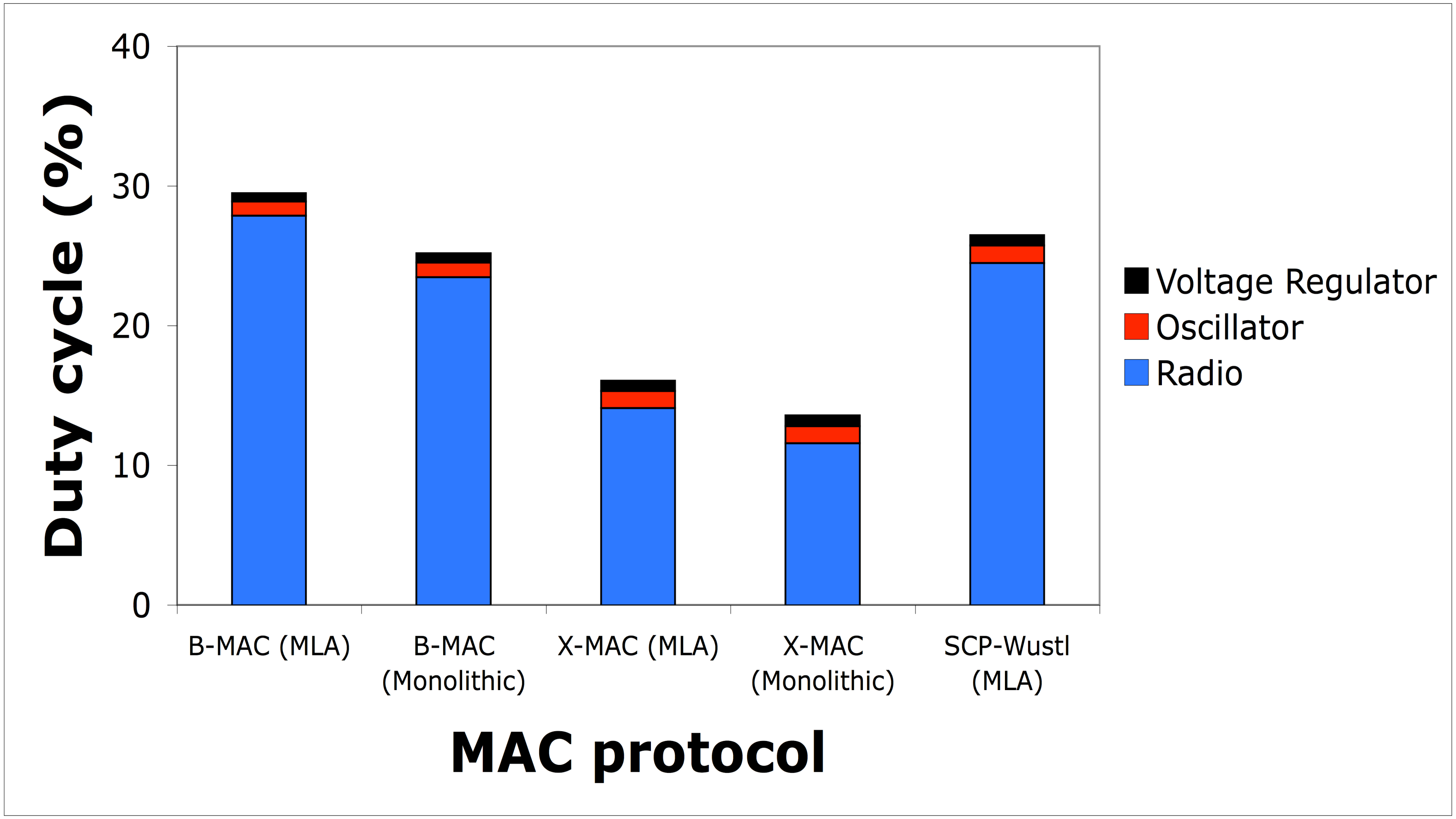
Throughput


Throughput (kbits/s)
100
10
1
1
2
3
4
5
6
7
8
9
10
Number of sending nodes
X-MAC (MLA)
X-MAC (Monolithic)
SS-TDMA (MLA)
Pure TDMA (MLA)
SCP-Wustl (MLA)
B-MAC (MLA)
B-MAC (Monolithic)

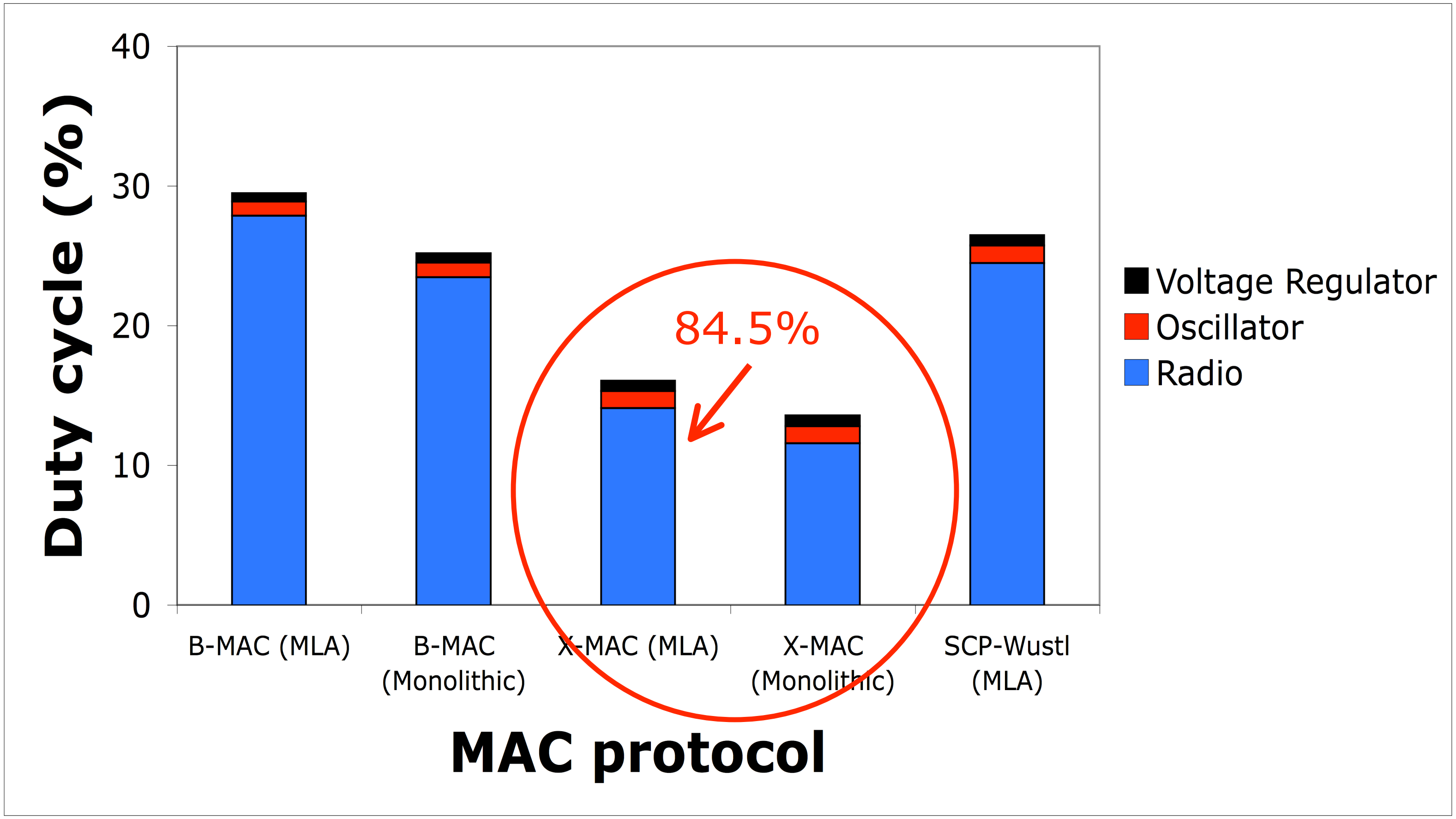
Message Latency



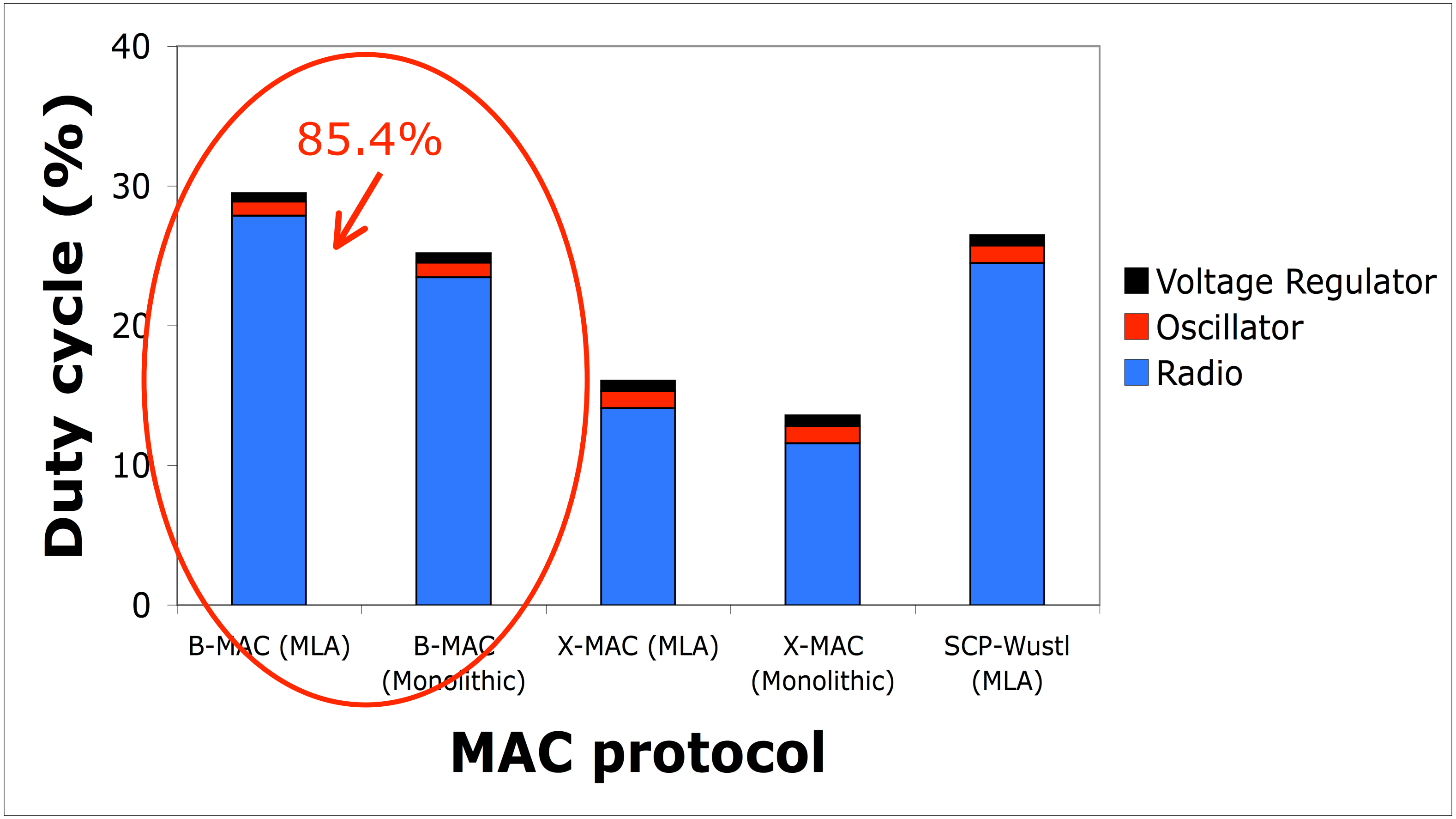
Energy Consumption



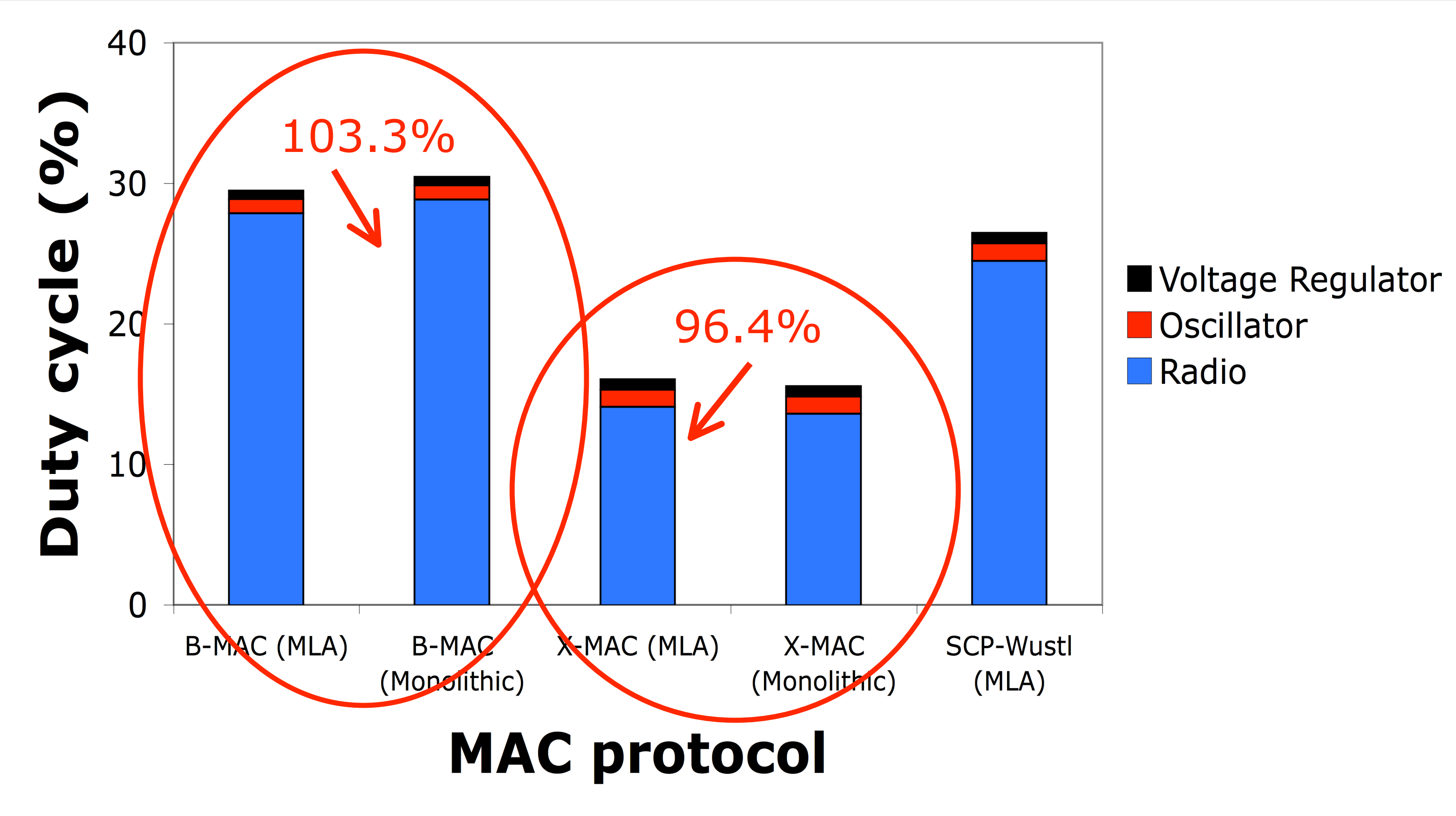
Energy Consumption



Energy Consumption



Energy Consumption



Conclusion

- MLA: Component-based, Low-Power MAC architecture for wireless sensor networks
  - Increases Flexibility
  - Simplifies development
  - Reduces porting effort
- Provides evidence contrary to the existing philosophy that radio stacks must be monolithic to be efficient

Questions?

Code available under

tinyos-2.x-contrib/wustl/upma

Don't forget, you can contribute too...

tinyos-contrib-caretakers@millennium.berkeley.edu