Options for EEE in 100G

Draft

Hugh Barrass

IEEE P802.3bj

January, 2012

Contributors, reviewers and supporters

Stephen Bates	PMC Sierra
Mike Bennett	LBL
Matt Brown	Applied Micro
Mark Gustlin	Xilinx
Oren Sela	Mellanox
Alexander Umnov	Huawei
Pedro Vasallo	U. Nebrija

Agenda

- Background
- PHY power breakdown
- EEE options
- Simulated performance
- Conclusions?
- Questions…

Energy Efficient Ethernet

- 802.3az – Defined EEE for 100M-10G
- Wake times ~ max length packet
- Includes definition for longer wake time negotiation

- All PHY definitions include quiescent state
- Signals stop/start - parameters kept refreshed
- Measured PHY power savings up to 80%

… but how effective is it?

- How widely will it be used & how much energy will it save?
  - The answer is “it depends”

- Two critical parameters – wake time; % power in LPI state
- Time spent in LPI depends on wake time & traffic profile
- Wake time defines latency hit (& whether it gets disabled)
- These considerations will become more important for 100G

Please do not print!

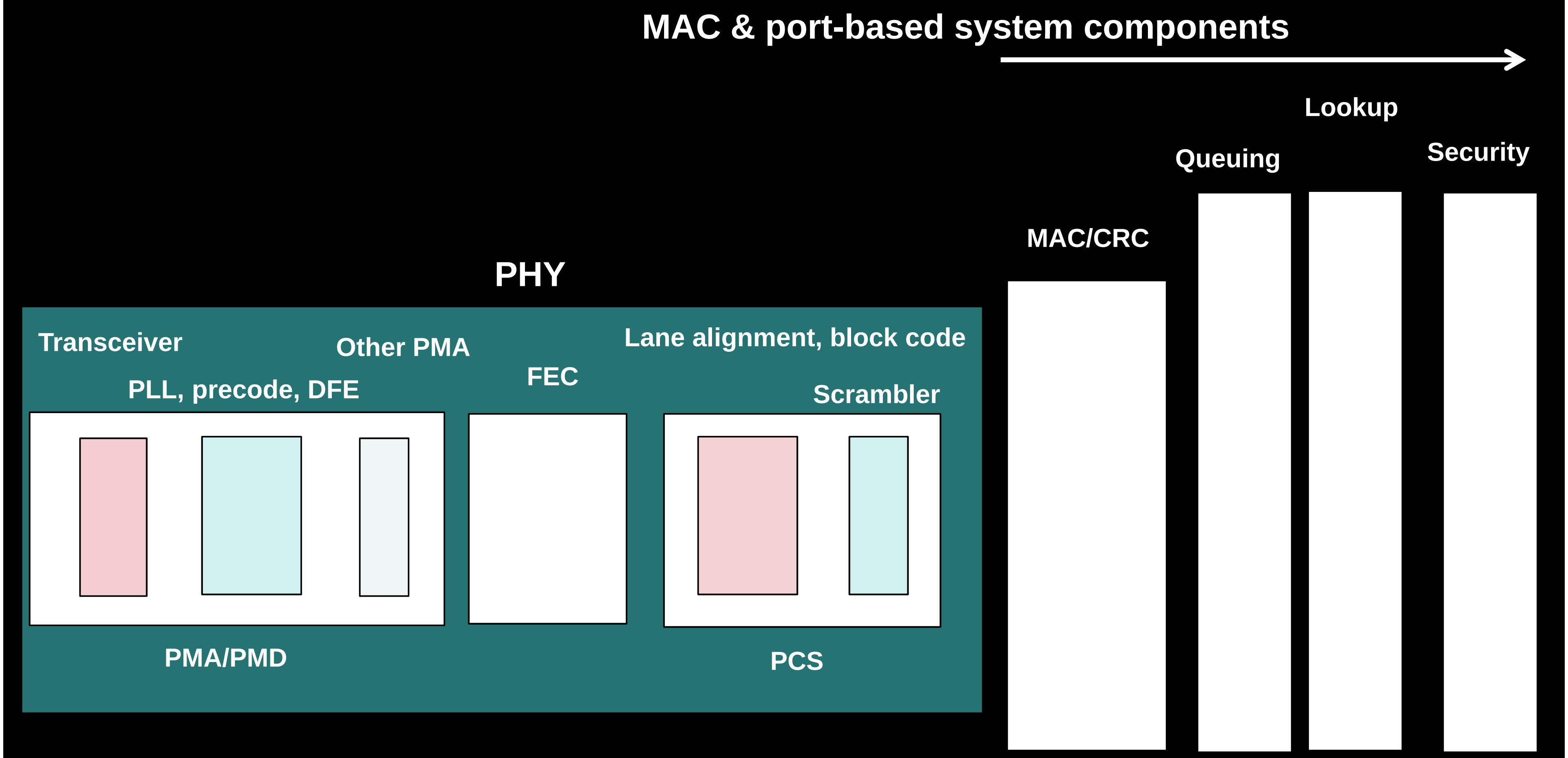
Issues for 100G EEE

- V. short max packet time (~150ns)
- Problems to reduce wake time:
  - Time to remove/reapply power constant (no scaling)
  - Unclear how quickly 25GHz PLL can capture
  - Lane alignment must be re-established
- Ultra-high speed designs require “aggressive” silicon libraries (high leakage)
  - Clock stop alone doesn’t save as much power
- Perhaps there will not be a single answer…

Agenda

- Background
- PHY power breakdown
- EEE options
- Simulated performance
- Conclusions?
- Questions…

PHY Components/Functions



MAC & port-based system components

Lookup

Queuing

Security

MAC/CRC

PHY

Transceiver

Other PMA

Lane alignment, block code

FEC

PLL, precode, DFE

Scrambler

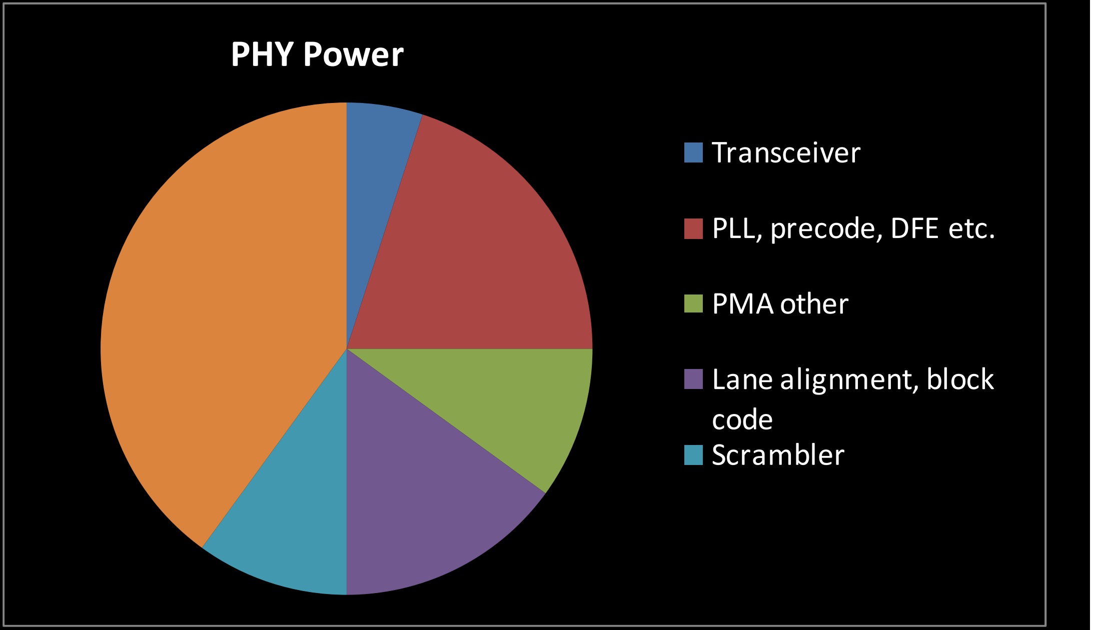
PMA/PMD

PCS

Relative power for components

PHY Function	Power
Transceiver	5
PLL, precode, DFE	20
Other PMA	10
FEC	40
Lane alignment, block code	15
Scrambler	10

Normalized to PHY power = 100



MAC & port-based system components

Function	Power
MAC	20
Lookup	20
Queuing	10
Security	40

Reduced power scenarios

- For each component – consider three scenarios:
  - Normal operation (data mode)
  - Clock only – synchronization maintained, no data present
  - Clock stopped – no synchronization
- Note that complex scenarios may be possible: e.g.
  - External clock stopped, internal clock maintained
  - External synchronization maintained, internal clock stopped
  - Functions deeper into the port allow more complex solutions
- Numbers based on assumed design structures and arbitrary (ASIC) library choice

Reduced power scenarios

PHY Function	Power, operating	Clock only	Clock stopped
Transceiver	5	5	1
PLL, precode, DFE	20	20	4
Other PMA	10	10	2
FEC	40	20	8
Lane alignment, block code	15	10	2
Scrambler	10	5	2

MAC & port-based system components

Function	Power	Clock only	Clock stopped
MAC	20	10	4
Lookup	20	10	4
Queuing	10	5	2
Security	40	20	8

Agenda

- Background
- PHY power breakdown
- EEE options
- Simulated performance
- Conclusions?
- Questions…

EEE options

- Effectively, different levels of sleep during LPI
  - A) Line stays active with clock; LPI sent during refresh intervals
  - B) All signaling stopped; quiescent state on line
- Notes:
  - 802.3az defined B) – considered as default choice for 100G
  - MAC and other system components not considered
  - LLDP renegotiation might allow change - particularly where wakeup sequence is unchanged
- Consider LPI requirements (assumptions) for scenarios

Continue clocking

- PMA continues to send clock
  - Maybe with data pattern (e.g. PMA, PRBS test pattern)
  - Refresh not needed for alignment (but may keep s/m simple)
  - Wake time includes some rapid alignment markers
- Transceiver & PMA power at full level
- V. low probability of lane re-alignment during wake
- Most transmit PCS functions may freeze
- Some receive functions need to maintain phase
- Most of PHY is in clock stop state

Please do not print!

Clock stopped

- Same as 802.3az – used as basis for early 100G work
  - Assumes full power down – v. slow wake
  - Some state preserved (e.g. DFE taps; alignment fifo depths)
  - Refresh used to update state – keeps changes minimal
- Most transmit & receive functions fully off
- Requires slow power-up, plus rapid alignment markers

Agenda

- Background
- PHY power breakdown
- EEE options
- Simulated performance
- Conclusions?
- Questions…

Simulated performance

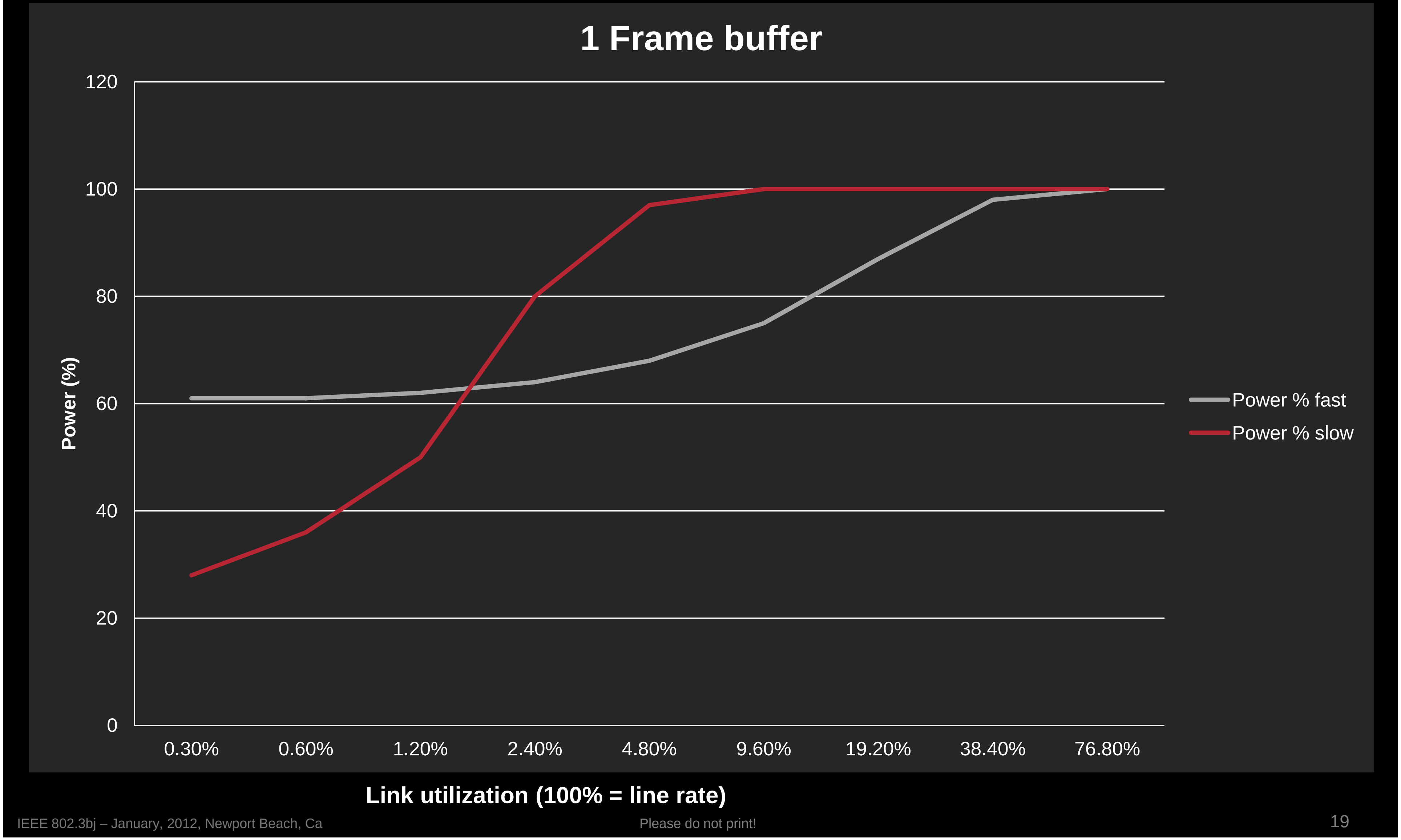
- Using arbitrary structural design assumptions…
- … along with ASIC library power as guideline
- Everything normalized to 100% of operational PHY power
- 2 scenarios:
  - Clock only: Waketime = 250nS; Power saving = 40%
  - Clock stopped: Waketime = 4.5uS; Power saving = 80%
- Modified Poisson traffic
- PHY power only considered – further savings: MAC etc.

Please do not print!

Simulation provisos

- Traffic model scaled up from much slower
  - Results in very pessimistic savings (no long IPGs)
- Heuristic simulation, v. simplistic behavior
- Actual power savings, v. design dependent
  - Leakage losses, fast/slow power switching, etc.
- Other assumptions can be explored
- Effect of buffer & burst
  - Modeled simply as longer packets
  - May be useful for core devices

Power savings

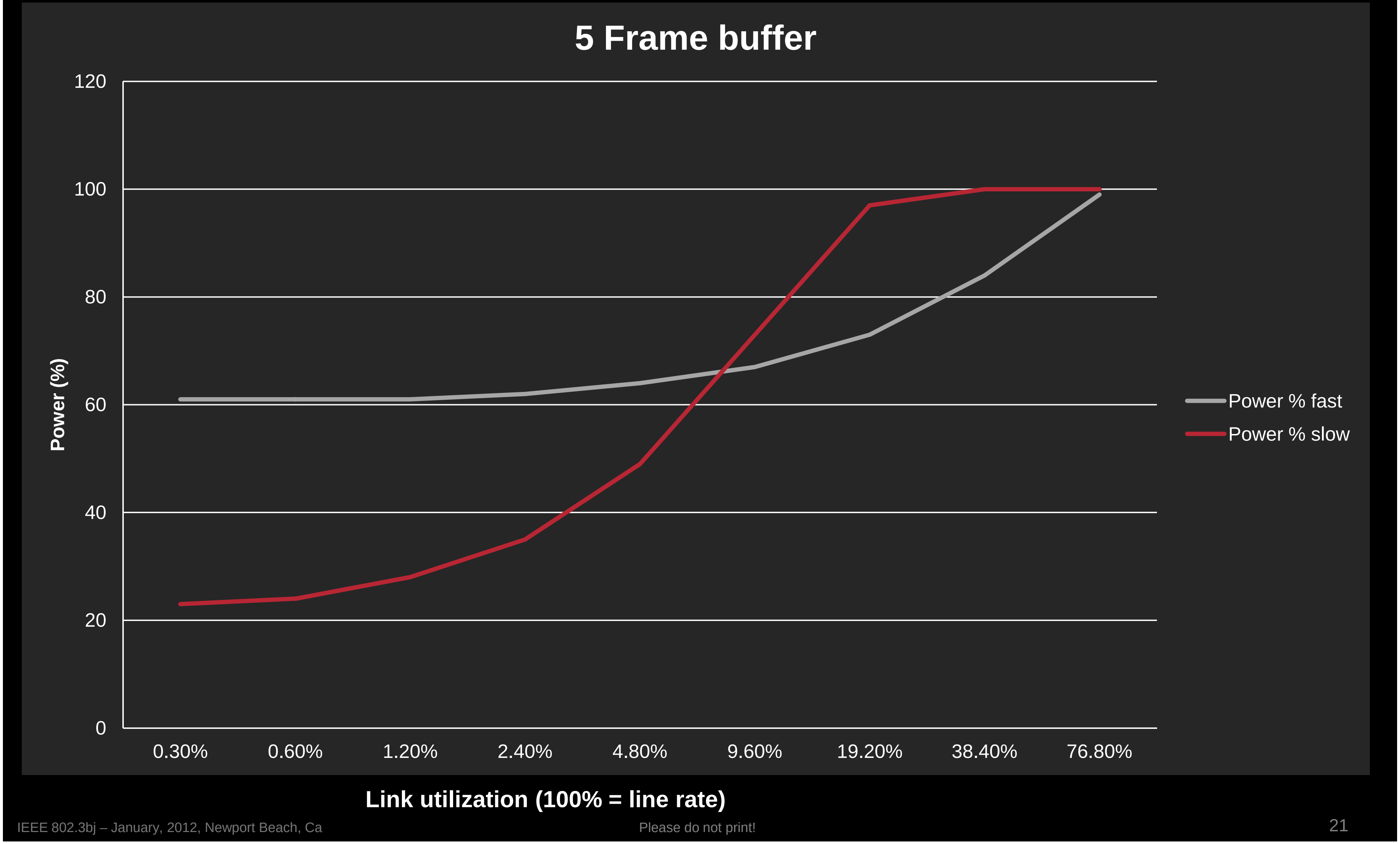


Please do not print!

Notes

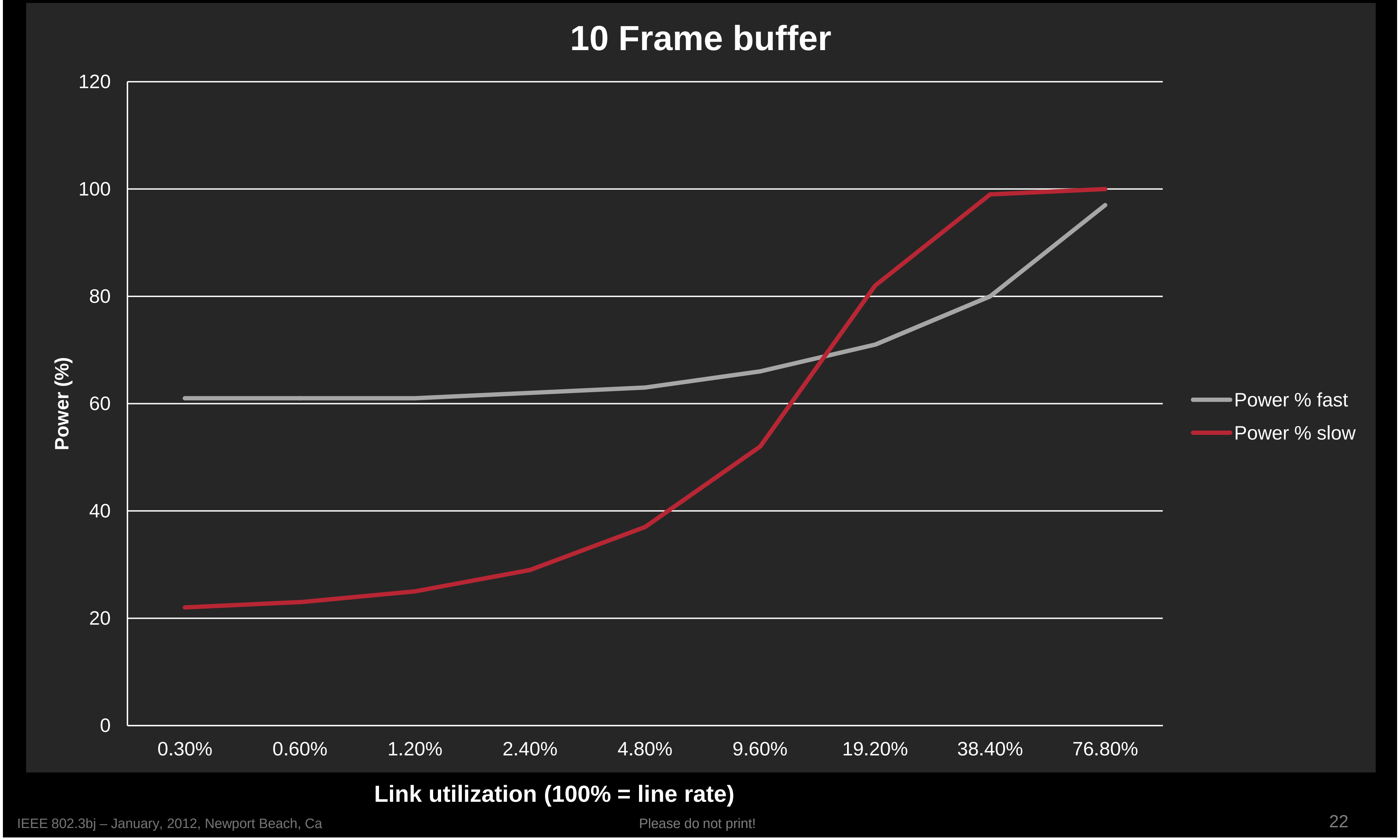
- Fast mode – saves power (20-30%) from 2-20%
  - Key range for aggregation devices
- Slow mode – saves power (up to 80%) less than 2%
  - Ideal for edge devices
  - (and off peak mode – nights & weekends)
- Buffer and burst may help for medium loads
  - Particularly for core devices

Buffer and burst performance



Please do not print!

Buffer and burst performance



Please do not print!

Agenda

- Background
- PHY power breakdown
- EEE options
- Simulated performance
- Conclusions
- Questions…

Conclusions…

- Physical limitations will require an unacceptably long wake time for “classic LPI”
- Faster wake time possible if signaling is maintained
  - But the power savings insufficient for edge/night mode
- Define two LPI modes: fast & slow
  - Expand baseline (gustlin_01_0112) to include both
  - (suggest) support for both mandatory for EEE (which is optional)
  - LLDP to negotiate fast/slow changes – without link drop
- Detailed state machine & functional proposal for March
  - Fast mode added to EEE baseline (slow mode already defined)

Agenda

- Background
- PHY power breakdown
- EEE options
- Simulated performance
- Conclusions?
- Questions…

Please do not print!