



Introduction to Energy Efficient Ethernet

Adam Healey
Joint T11.2/T11.3 Ad Hoc Meeting
March 31, 2010
T11/10-158v0



Motivation for this presentation

- Recent discussions on the topic of FC-EE (Energy Efficient)
 - It now appears in the FCIA 32GFC MRD
- Energy Efficient Ethernet (EEE) has served as a reference point for these discussions
- Ethernet and Fibre Channel share a similar architecture at the lower layers
 - 16GFC employs 64B/66B encoding, FEC, transmitter training
 - Opportunity for additional re-use?
- This presentation provides information on the motivation, objectives, and architecture for Energy Efficient Ethernet
- It is intended to help guide the discussion
- It is not a proposal for an FC-EE architecture

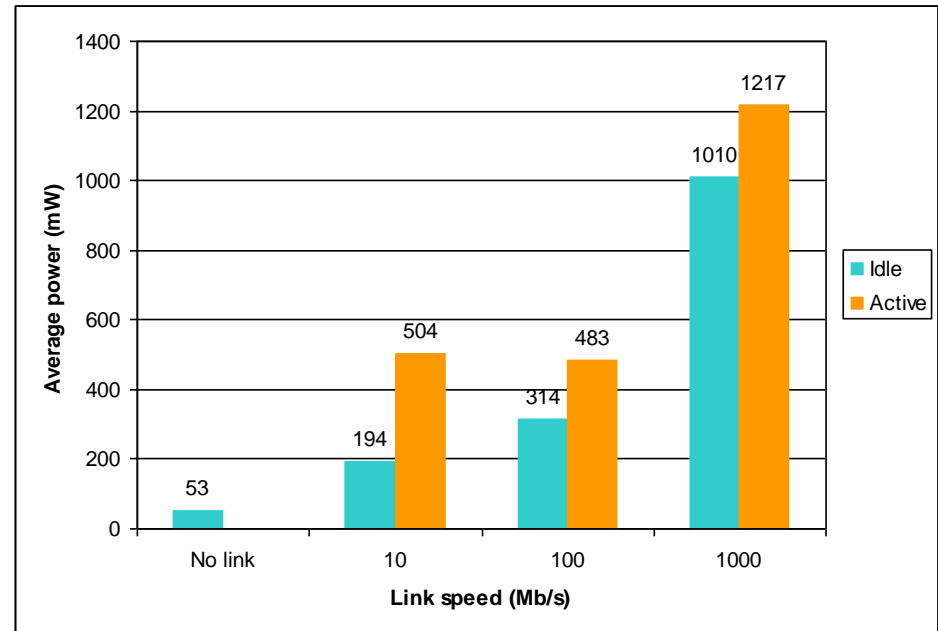
Agenda

- **Overview of Energy Efficient Ethernet**
- Low power idle architecture
- Low power idle walk-through
- Special considerations for FEC
- Considerations for FC-EE (Energy Efficient)

Key objectives for Energy Efficient Ethernet

- Define a mechanism to reduce power consumption during periods of low link utilization
- Define a protocol to coordinate transitions to and from a lower level of power consumption
- No frames in transit shall be dropped or corrupted during the transition to and from the lower level of power consumption
- The transition time to and from the lower level of power consumption should be transparent to upper layer protocols and applications

Single-port PCIe 10/100/1000 Mb/s controller
(MAC plus PHY)



Source: Intel, Intel® 82573L Gigabit Ethernet Controller, 130 nm

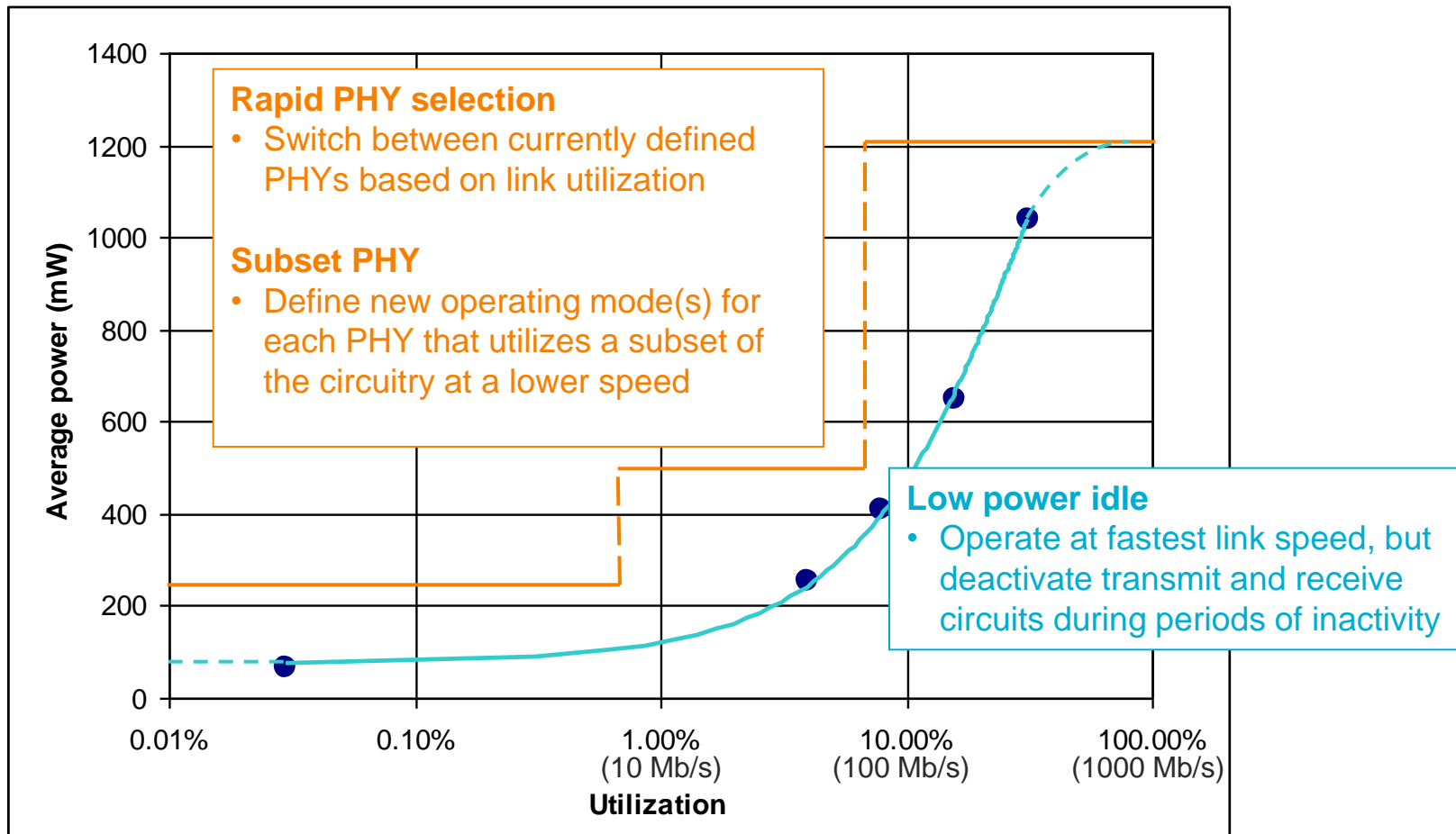
“Idle” = no traffic

“Active” = bi-directional, line-rate traffic

IEEE P802.3az project documents may be found at <http://ieee802.org/3/az/>



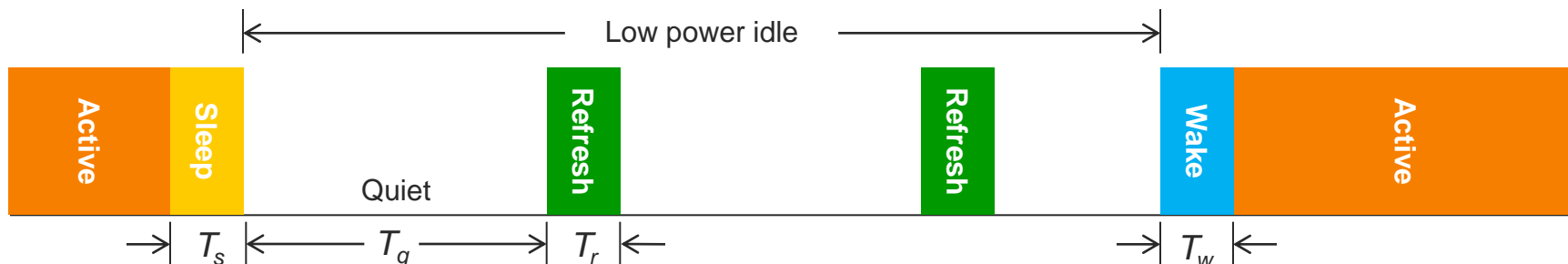
Architectures considered



Source: Intel, Configuration: Traffic profile = "Trace_VOIP_*.txt", low-power idle initialization wait = 10 ms, sleep time = 1 ms, wake time = 10 ms

Simulation C-code and traffic profiles available at <http://www.ieee802.org/3/az/public/tools/index.html>

Key principles of low power idle



- Transmit data at the fastest link speed available for the most energy efficient transmission (in terms of Joules/bit)
- When there is no data to send, enter the low power idle state where power may be reduced by turning off unused circuits
- Periodically transmit a signal during low power idle to refresh the receiver (e.g. update timing recovery, adaptive filter coefficients)
 - Facilitate fast transition from low power idle to normal operation (active)
 - As refresh duty cycle decreases, low power idle power approaches quiet power
 - Also serves as link heartbeat to detect link disconnects or other faults
- Transmitter initiates transition to (and from) low power idle and the receiver acquiesces

Key principles of low power idle (continued)

- Low-power idle not only enables power savings in the PHY, but also guarantees that data will not be transmitted
 - With the assurance that a packet will not be received, system may power down portions of the MAC, memory, CPU, etc.
- After initiating a transition from low-power idle to normal operation, the transmitter will defer transmission for a pre-defined time
- The PHY must be ready to receive data when this time expires
 - System may voluntarily increase the deferral time for deep sleep modes
- Ethernet Auto-Negotiation is used to advertise and enable EEE capability
- After link initialization, IEEE 802.1ab™ Link Layer Discovery Protocol (LLDP) may be used to adjust the wake time

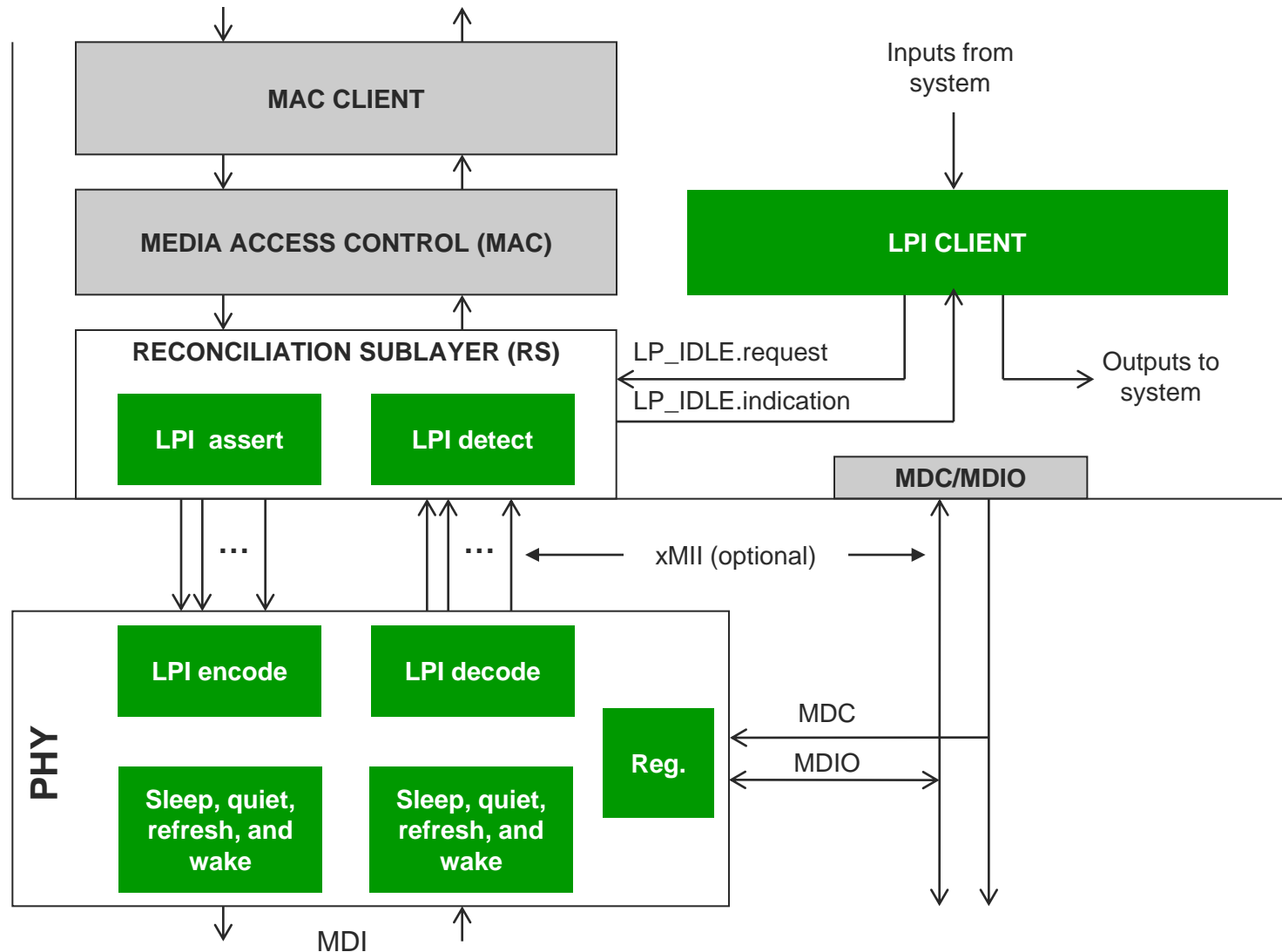
Scope of work for IEEE P802.3az

- Low power idle architecture and management parameters
- Low power idle for operation over twisted-pair cabling systems
 - Fast Ethernet (100BASE-TX)
 - Gigabit Ethernet (1000BASE-T)
 - 10 Gigabit Ethernet (10GBASE-T)
- Low power idle for operation over passive electrical backplanes
 - Gigabit Ethernet (1000BASE-KX)
 - 10 Gigabit Ethernet (XAUI, 10GBASE-KX4, 10GBASE-KR)
- New LLDP type, length, and value (TLV) information elements for negotiating system-level energy efficiency parameters

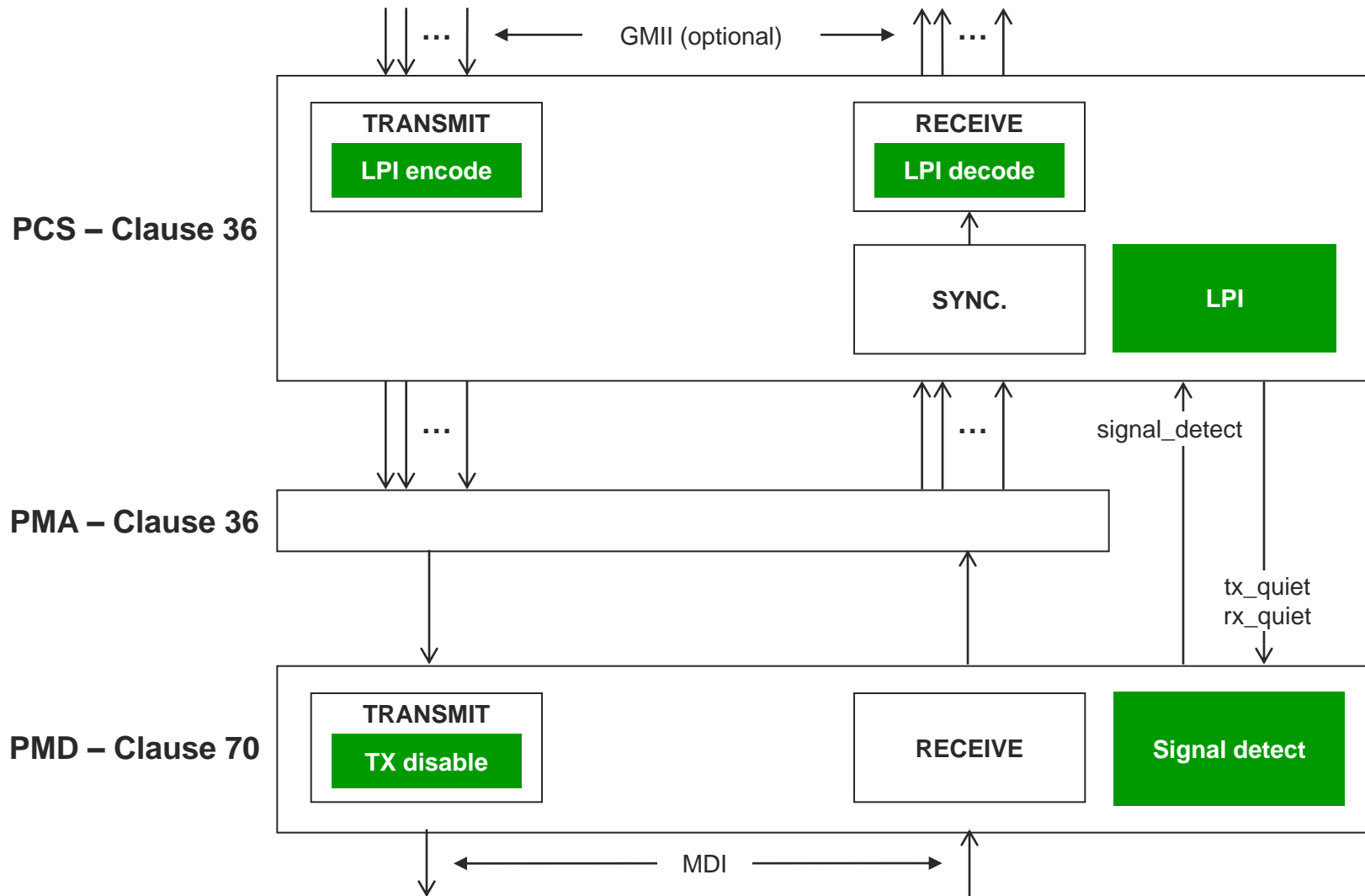
Agenda

- Overview of Energy Efficient Ethernet
- **Low power idle architecture**
- Low power idle walk-through
- Special considerations for FEC
- Considerations for FC-EE (Energy Efficient)

Low power idle architecture



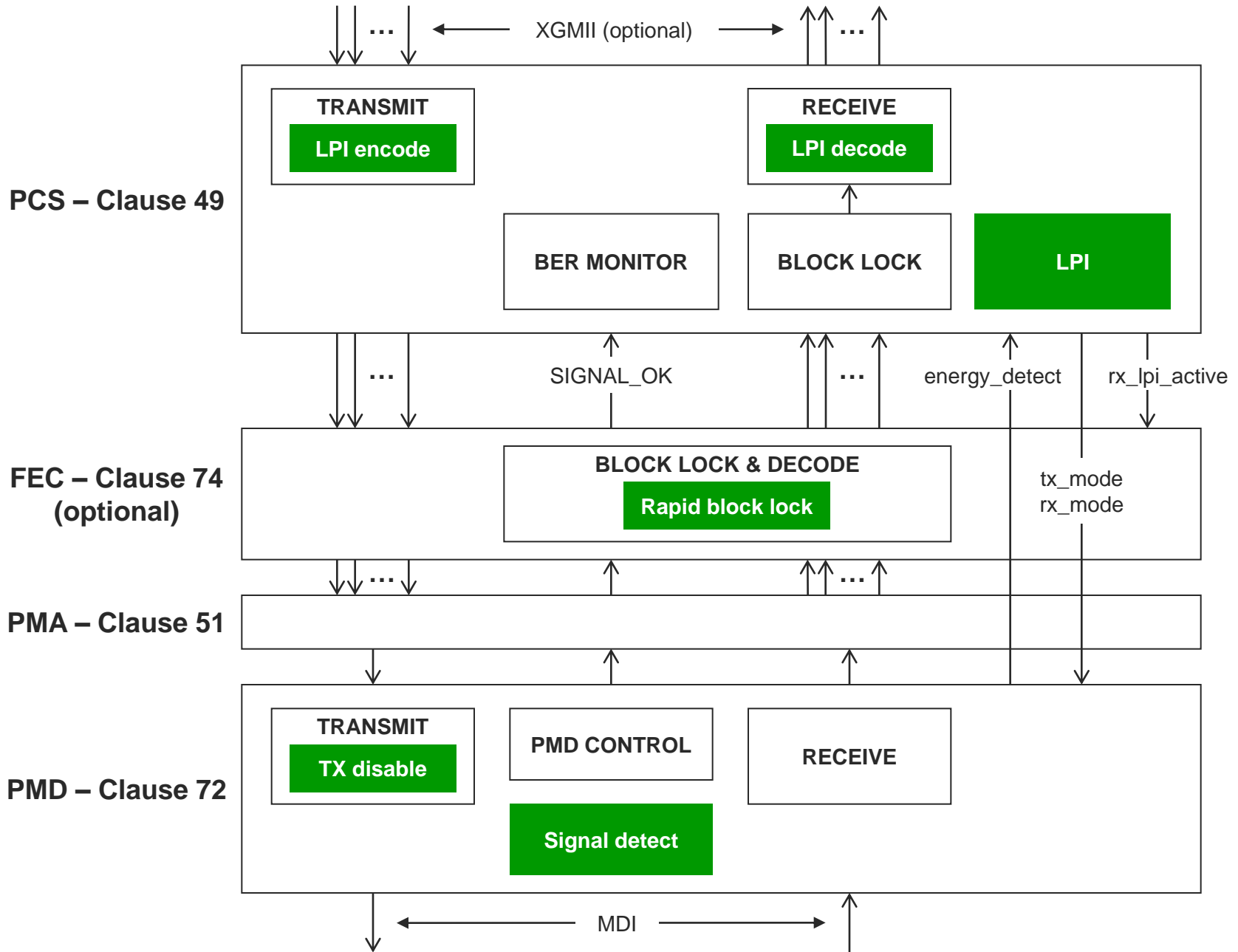
LPI architecture for 1000BASE-KX (8B/10B)



LPI encoding for 1000BASE-KX

Code	Ordered_set	Number of code-groups	Encoding
/I/	IDLE		Correcting /I1/, Preserving /I2/
/I1/	IDLE 1	2	/K28.5/D5.6/
/I2/	IDLE2	2	/K28.5/D16.2/
/LI/	LPI		Correcting /LI1/, preserving /LI2/
/LI1/	LPI 1	2	/K28.5/D6.5/
/LI2/	LPI 2	2	/K28.5/D26.4/

- Correcting and preserving idle ordered sets differentiate normal idle from low-power idle



LPI encoding for 10GBASE-KR (64B66B)

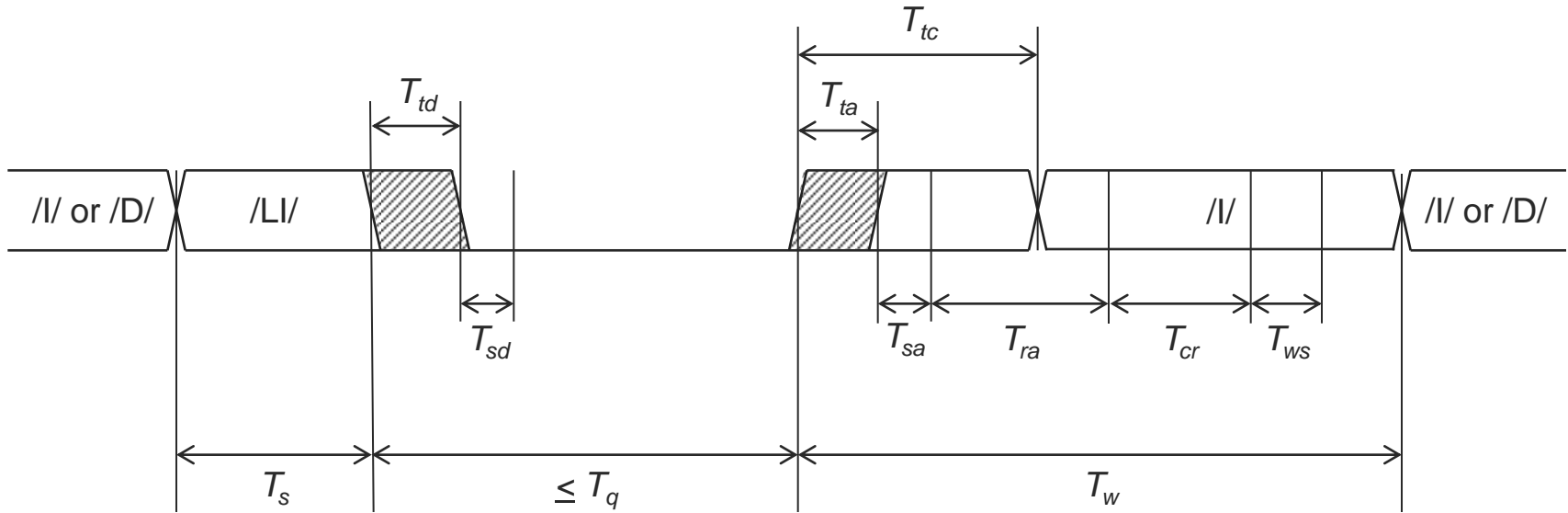
Control character	Notation	XGMII control code	10GBASE-R control code
idle	//	0x07	0x00
LPI	/LI/	0x06	0x07

- Low power idle encoding employs the existing 66-bit control block format (block type 0x1E) filled with the 7-bit control code defined above

Agenda

- Overview of Energy Efficient Ethernet
- Low power idle architecture
- **Low power idle walk-through**
- Special considerations for FEC
- Considerations for FC-EE (Energy Efficient)

LPI timing diagram



Legend

T_s	Sleep time	T_{sd}	Receiver signal_detect de-assertion time
T_q	Quiet time	T_{sa}	Receiver signal_detect assertion time
T_w	Wake time	T_{ra}	Receiver activation time
T_{td}	Transmitter de-assertion time	T_{cr}	Receiver timing acquisition time
T_{ta}	Transmitter partial activation time	T_{ws}	Receiver PCS synchronization time
T_{tc}	Transmitter full activation time		

Going to sleep

- Upon detection of “Assert LPI” signaling at the XGMII, the PCS transmit function will encode low power idle into the transmitted blocks
 - This notifies the receiver that the loss of signal event that follows is related to low power idle and not a link disconnect or some other fault
- The receiver’s PCS receive function will decode low power idle from the received blocks and present “Assert LPI” signaling at the XGMII
- After for T_s microseconds, the PCS will deactivate the transmitter
- The coefficients of the transmitter’s finite impulse response filter (FIR) are preserved
- Upon detection of loss of signal, the PCS will deactivate the receiver
 - With the exception of the PMD signal detect function

Waking the transmitter

- Transmit and receive functions may be deactivated during the quiet periods to conserve energy
 - With the exception of the PMD signal detect function
- Upon detection of “Normal inter-frame” signaling at the XGMII, or upon expiration of the quiet timer, the PCS will re-activate the transmitter
- The transmitter first transmits an alert signal that facilitates the correct operation of the receiver’s PMD signal detect function
 - Square wave pattern with a 16 unit interval period
 - Transmitter is preset for the duration of the alert signal and then the stored FIR coefficients are restored
- The transmit functions will take some time to achieve normal operation following activation
 - Let T_{ta} be the time it takes for the transmitter to deliver a signal capable of triggering the receiver’s signal detect function
 - Let T_{tc} be the time it take the transmitter to achieve compliant operation

Waking the receiver

- If, during the quiet period, signal_detect is asserted, then receiver functions are reactivated
 - Let T_{sa} be the signal_detect assertion time, assuming the transmitter is delivering a suitable signal
 - Let $T_{ra} + T_{cr}$ be the time it takes the receiver to return to normal operation and recover timing from the incoming signal, assuming the transmitter is delivering a compliant signal
 - Let T_{ws} be the time it takes for the 64B/66B decoder to establish the boundaries between consecutive blocks
- Once the receiver has block delineation, it can establish whether or not low power idle or normal idle is being received
 - Distinguish between a refresh and a wake
- The balance of the wake time is then used by the system to re-activate higher-level functions

Summary of 10GBASE-KR EEE timing parameters

Parameter	Symbol	Min.	Max.	Units
Sleep time	T_s	5.0 – 1%	5.0 + 1%	μ s
Quiet time	T_q	1.7 – 1%	1.7 + 1%	ms
Wake time	T_w	–	11	μs
Transmitter deactivation time	T_{td}	–	500 ¹	ns
Receiver signal_detect deactivation time	T_{sd}	–	500	ns
Transmitter partial activation time	T_{ta}	–	500 ²	ns
Receiver signal_detect assertion time	T_{sa}	–	500	ns

¹ For the transmitter peak-to-peak differential output amplitude to be less than or equal to 30 mV

² For the transmitter to reach 90% of its trained peak-to-peak differential output voltage

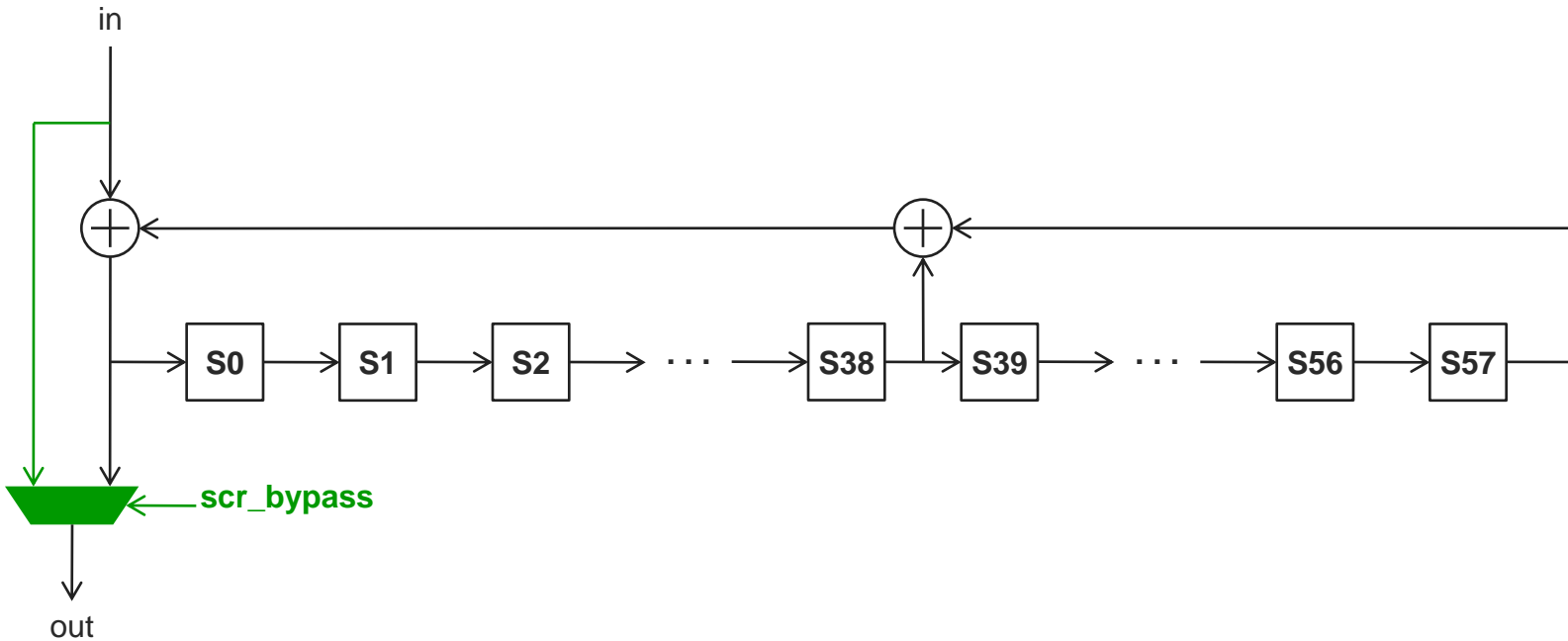
Agenda

- Overview of Energy Efficient Ethernet
- Low-power idle architecture
- Low-power idle walk-through
- **Special considerations for FEC**
- Considerations for FC-EE (Energy Efficient)

FEC synchronization time

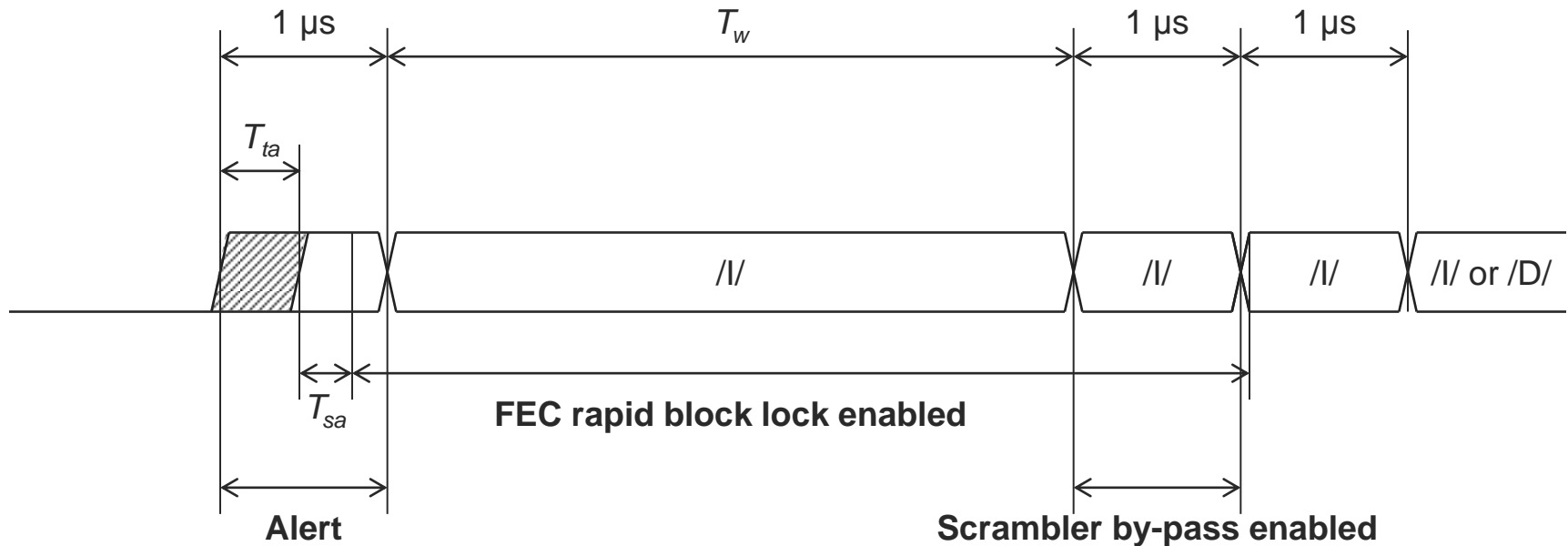
- The Clause 74 FEC block structure consists of 32 66-bit blocks, where the sync. header for each block is compressed to 1 bit (32 x 65 bits), followed by a 32-bit parity field (2112 bits total)
- FEC block lock is based on the 32-bit parity field of the candidate block matching the parity computed for that same block
 - 4 consecutive successful parity checks are required to achieve block lock
- Under error free operating conditions, no more than 2111 candidate frame positions would need to be checked to get the first valid parity, and then 3 additional frames would then be required to confirm block lock
 - In turn, this takes $(2111+3) \times 2112$ bits or approximately 430 μ s at 10.3125 Gb/s
- Some means to accelerate FEC block lock is required to satisfy the wake time constraints

Solution: Scrambler bypass



- When the 64B/66B scrambler is bypassed, the output of the FEC encoder is a deterministic pattern
- The receiver may use this deterministic pattern to quickly identify FEC block boundaries
- This process is referred to as FEC rapid block lock

10GBASE-KR with FEC timing diagram



- Following time T_w , the transmitter bypasses the 64B/66B scrambler for 1 μ s
 - The receiver may leverage the deterministic pattern to achieve FEC block lock
- The transmitter then inserts the 64B/66B scrambler and defers for an additional 1 μ s
 - The receiver can then confirm FEC block lock
- When the optional FEC sublayer is included, wake time is extended by 2 μ s

Agenda

- Overview of Energy Efficient Ethernet
- Low power idle architecture
- Low power idle walk-through
- Special considerations for FEC
- **Considerations for FC-EE (Energy Efficient)**

Considerations for FC-EE (Energy Efficient)

- Does a low power idle architecture make sense for FC-EE?
- Low power idle for (retimed) optical interfaces?
- Tolerance for medium access latency?
- Mechanism to advertise and enable FC-EE?
- Mechanism to modulate the wake time for various depths of sleep?
- Is FC-EE for 32GFC only or should it also consider 16GFC, 8GFC, etc.?