# BoW Power Management


**May. 11th, 2022**

# Quick Review

- **Initially want to focus on defining two additional power states (besides "active"):**
  - Clocks Active, Data Idle
    - For 16Gb/s terminated signaling, saves ~30% of TX+RX power
    - For 8Gb/s terminated signaling, saves ~43% of TX+RX power
    - For unterminated signaling, saves "nothing"
  - Clocks Gated, Data Idle
    - Can save an additional 40-50% of TX+RX power relative to the above
- **Both of the states above are intended to be relatively "fine grained"**
  - I.e., entry/exit latencies can be 0 (data idle) to ~1-2ns (clocks gated)
- **Key consideration is how to achieve this without significant penalties on peak power, bandwidth density (number of wires), or complexity**
  - Note that everything in the rest of this deck should be considered optional features; PHYs can be operated in active mode continuously

# Proposal #1: Clock Active, Data Idle

BLUE CHEETAH
ANALOG DESIGN

- **Define all 0's on all "D" lines (including AUX and FEC) as an "IDLE"**
  - PHY (mostly) does not even need to know about this – link layer would simply control the data. I.e., this is mostly a PHY-implied requirement on the link layer

- **Relative to BoW 1.0, the main implication of the above is that specifically choosing all 0's as the IDLE will preclude power savings in this state for PHY RXs implemented with VDDIO or VDDIO/2 termination**
  - Suggestion is to eliminate this termination optionality for RX's that support the power management features proposed here

# Proposal #2: Clock Gated, Data Idle

BLUE CHEETAH ANALOG DESIGN

- **In clock gated state, TX must send IDLE (all 0's), CLK+/CLK- set to differential low**
  - A new slice-level interface signal "PHY_Idle" should be defined to allow the link layer to indicate this state to the PHY
  - Clock domain of PHY_Idle should be the same as the parallel data
  - Timing of PHY_Idle relative to bursts will be shown in slide 8

- **Will still burn termination power due to CLK- driver sending a high**
  - For "default" implementation, limits power in this state to ~5-8mW for typical VDDIO values / worst-case termination resistance
    - As compared to ~125mW when active for a BoW-256 terminated link
    - Seems to be an acceptable floor in this kind of scenario; for unterminated no power floor associated with this state
    - And there are likely further tricks that could be played to eliminate this power if desired

# Proposal #2a: Preamble

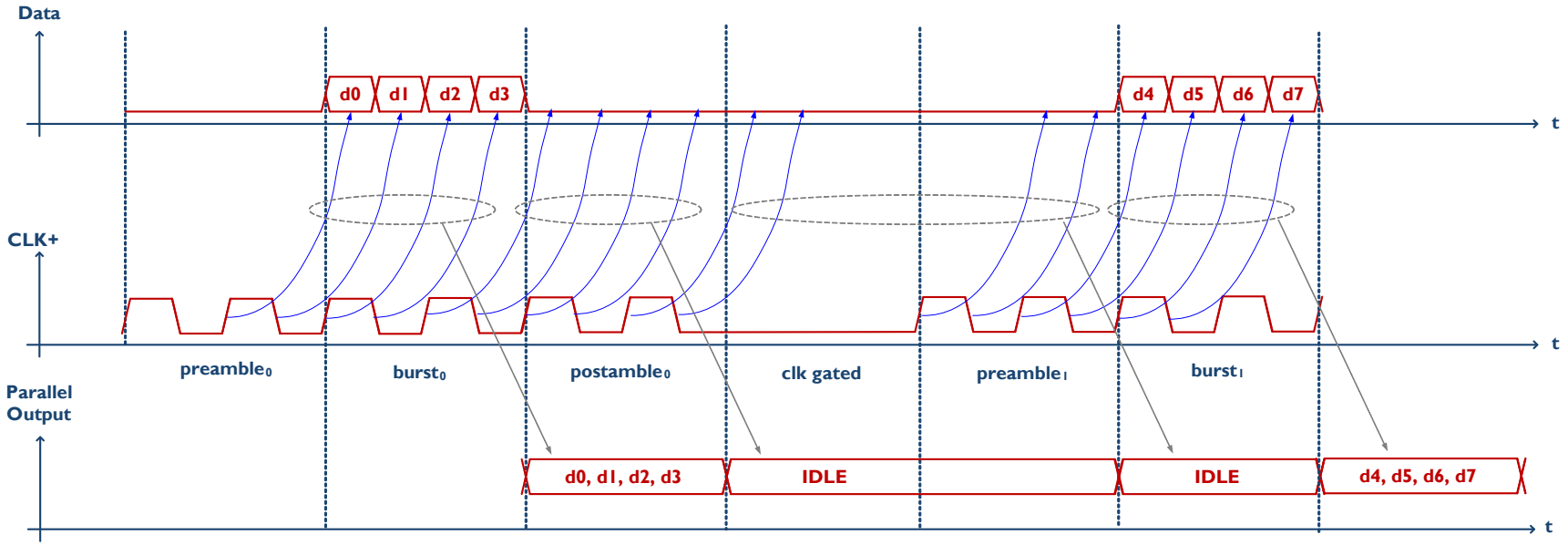
BLUE CHEETAH ANALOG DESIGN

- **In moving from clock gated to active state, TX link layer must cause the TX PHY slice to send (at least) one full parallel word* of IDLE data and clock edges as a preamble**
  - For example, if the link layer runs at 1GHz and the PHY runs at 8Gb/s, 8 edges of the 4GHz forwarded clock must be included in the preamble
  - This will allow the RX PHY slice clock circuitry to re-settle and "preload" the edges needed to receive the real data after the preamble
  - This does create a requirement that the RX PHY clock vs. data delay ($t_{RX\_clk}$) must be $<N_{ser}*T_{UI}$; the time available for RX PHY settling is $N_{ser}*TUI - t_{RX\_clk}$

*** Need to watch out for differing serialization / deserialization at TX/RX; preamble (and posamble) clock edge count likely needs to be max($N_{ser,TX}$, $N_{deser,RX}$). I.e., it is a full parallel word of whichever side is slower**

# Proposal #2b: Postamble

- **When moving from the active to clock gated state, the TX link layer must cause the TX PHY slice to send (at least) one full parallel word of IDLE data and clock edges as a postamble**
  - This is to ensure that we complete a full parallel IDLE word, as well as send what will become the beginning of another full parallel IDLE word
  - Example with $N_{ser}$ = 4, $t_{RX\_clk}$ = 2UI:



# Proposal #2c: Maximum Gated Duration

BLUE CHEETAH ANALOG DESIGN

- **A BoW TX cannot remain in the clock gated state for longer than 64 (128?) parallel words**
  - Even if there is no valid data to be sent by the TX link layer, it should send a full data idle burst (1 parallel word each of preamble + idle + postamble)

- **This represents a "minimum activity" of <5% idle bursts**
  - If no clock is forwarded to the RX over an extended period of time, its DLL (and any other "tracking" circuits) might lose lock (due to analog delay drifts/etc.)
  - The "forced" clock pulses can be used to keep these circuits / loops alive / tracking slow drifts
    - Note that similar considerations may exist on the TX, so this mechanism can assist with this as well

# Proposal #2d: PHY_Idle Timing

- **PHY_Idle should be asserted 1 parallel word in advance of the clock gating occurring**
  - I.e., PHY_Idle signal should rise on the same parallel word cycle as the postamble is being sent
- **PHY_Idle should be de-asserted 1 parallel word in advance of the preamble starting**
  - Both edges of PHY_Idle signal should occur concurrently with IDLE data being presented to PHY TX slice