MII to SMII Proposal

William Lo

March 11, 2024

MII Pin Count Reduction

- Do we want this?

- SMII (Serial MII) is a physical interface and not a logical one
  - Goal is to reduce real pin count
  - Has actual pin to pin timing specifications
  - No voltage level specification – do not want to limit process technology
- SMII does not replace MII
- Standard defines how to convert MII signals to lower pin count SMII

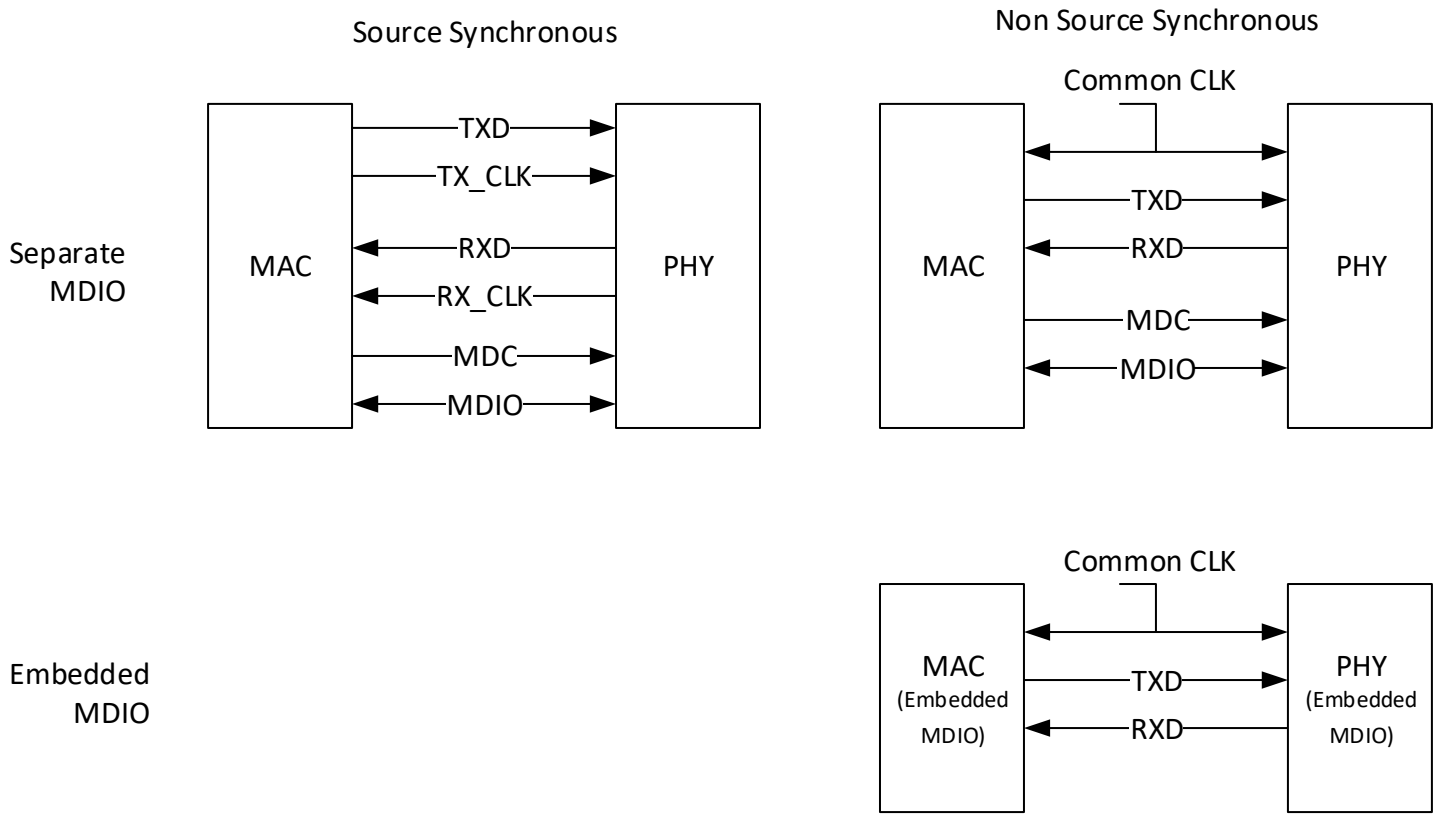
Based On Presentations

- Core Operations
  https://grouper.ieee.org/groups/802/3/dg/public/May_2022/Lo_3dg_01_1115202 3.pdf

- Single Port Configuration (Slide 27 reference below)
- This proposal does not consider time division multiplexed multiport configurations
  https://grouper.ieee.org/groups/802/3/dg/public/May_2022/potterf_3dg_01_012 524.pdf

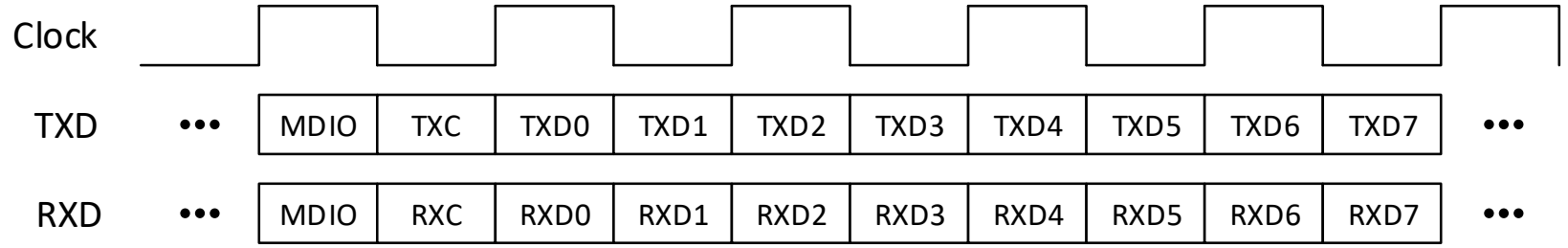
Three Possible Configurations

- Non-Source Synchronous allows multiple ports to be added without adding more clock pins
- Embedded MDIO only permitted in non-source synchronous mode
- If we want embedded MDIO in source synchronous need modifications to proposed protocol



Components To Baseline

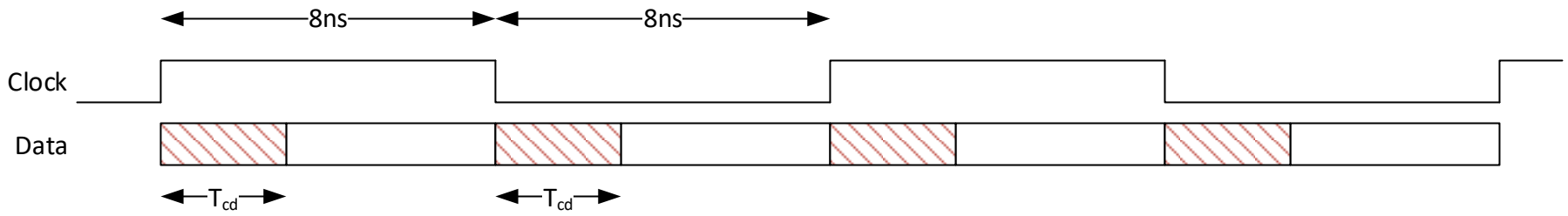
- TXD and RXD 10-bit frames (no SYNC) (Lo_3dg_01_11152023.pdf slide 5)
- DDR Clock – High level aligns to MDIO bit



- Optional capability – embed MDIO (Lo_3dg_01_11152023.pdf slide 6 to 10)
  - If not supported MDIO bit set to 1

Clock / Data Timing

- Keep it simple. One timing mode regardless of configuration
- DDR Clocking – high and low times nominally 8ns
- T_{cd} Clock edge to data stable
  - min = 0ns, max = 3ns.
  - Measured from the clock to data pins at each device



9-bit coding

- MII to SMII mapping
- Code choice for Idle and LPI reduces pin toggling power
- Invalid codes shall never be used as they interfere with alignment
- Sync SMII order set introduced – see next slide

MDIO	TXC/RXC	TXD/RXD[7:0]	Description
X	0	Data[7:0]	Data
X	1	0xFF	Idle
X	1	0x0F	Low Power Idle
X	1	0x01	Symbol Error
X	1	0x02	PLCA BEACON request
X	1	0x03	PLCA COMMIT request
X	1	0x0E	False Carrier (RXC/RXD only)
X	1	0xAA	Sequence
X	1	0x54	Invalid
X	1	0x55	Invalid
X	1	0x52	Invalid
X	1	0x56	Invalid
X	1	0x4A	Invalid
X	1	0x5A	Invalid
X	1	0x2A	Invalid
X	1	else	Reserved

Symbol 0	Symbol 1	Symbol 2	Symbol 3	Description
Sequence	0x00	0x00	0x00	Reserved
Sequence	0x00	0x00	0x01	Local Fault
Sequence	0x00	0x00	0x02	Remote Fault
Sequence	0x00	0x00	0x03	Link Interruption
Sequence	0x55	0x55	0x55	Sync SMII
Sequence	other combinations			Reserved

SYNC MII Order Set

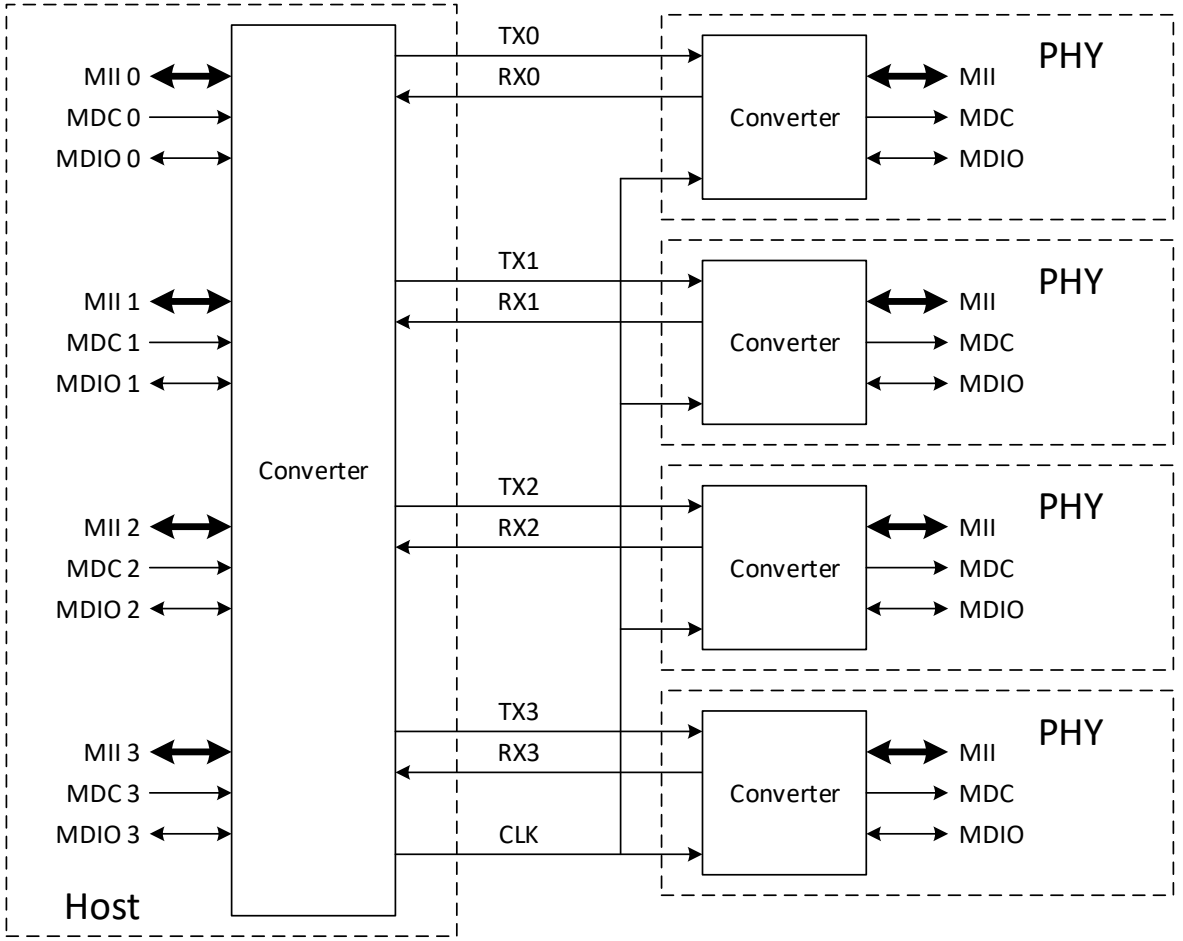
- Needed to find 10-bit alignment
- At least one SYNC MII Order Set sent to replace Idle, LPI, Local Fault, Remote Fault every X octets exchanged.
  - Recommend X be 10,000 octets (0.8ms)
  - Sufficient time to acquire lock prior to link up
  - Not so often to consume SMII toggling power during long periods of idles and LPI
  - Allows for reacquiring lock in case something goes out of sync in the SMII
- On receive side interpret SYNC MII Order Set as Idle, LPI, Local Fault, or Remote Fault depending on what was last seen

THANK YOU

BACKUP

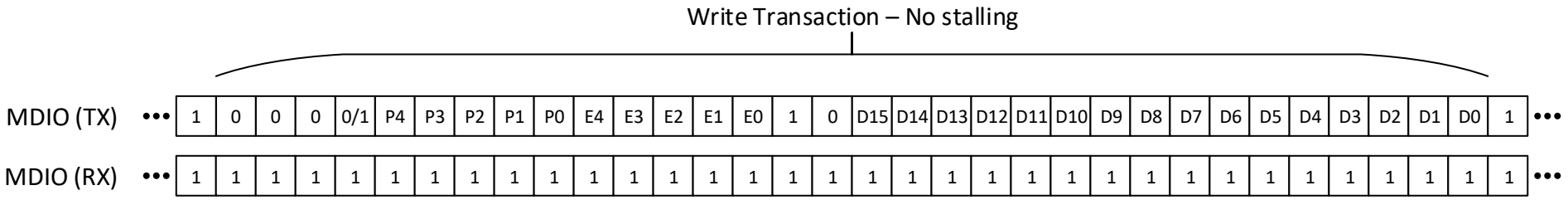
Non-Source Synchronous Multiport

- If all enhancements implemented then 2N + 1 pins for N ports
- Embedded MDIO
- Sequence Ordered Set
- Half rate clock
- No Sync pin



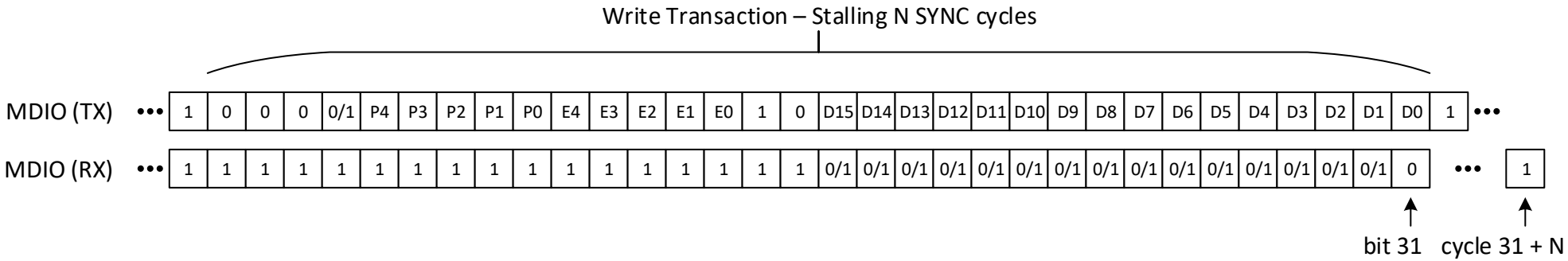
Write Operation

- MDIO bit set to 1 when inactive – just like the MDIO bus being pulled high when device Hi-Z
- Write protocol exactly the same as on a physical MDIO bus.
- MDIO bit on RX always set to 1 if no stalling
- Clause 45 example below



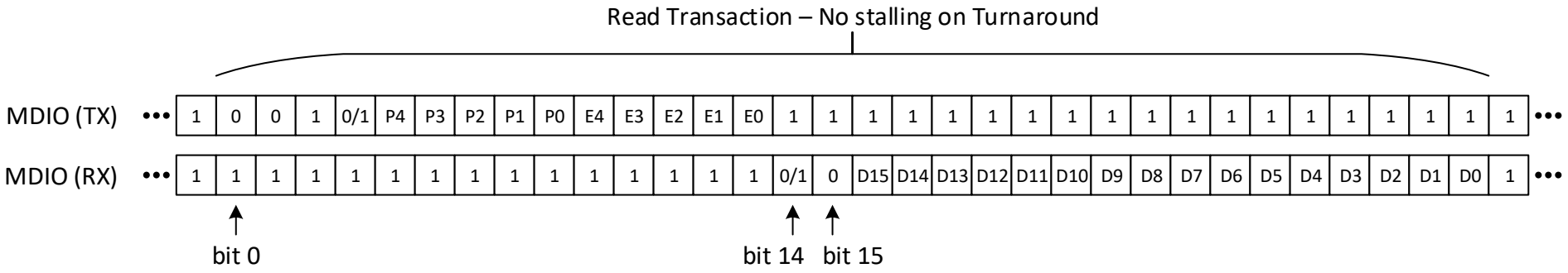
Write Operation with stalling

- In case PHY cannot process write transaction fast enough set bit 31 of the transaction to 0 on the RX to indicate to STA not to start another transaction
- Set bit to 1 once ready for another transaction
- Can optionally set other bits to 0 for earlier indication of a stall



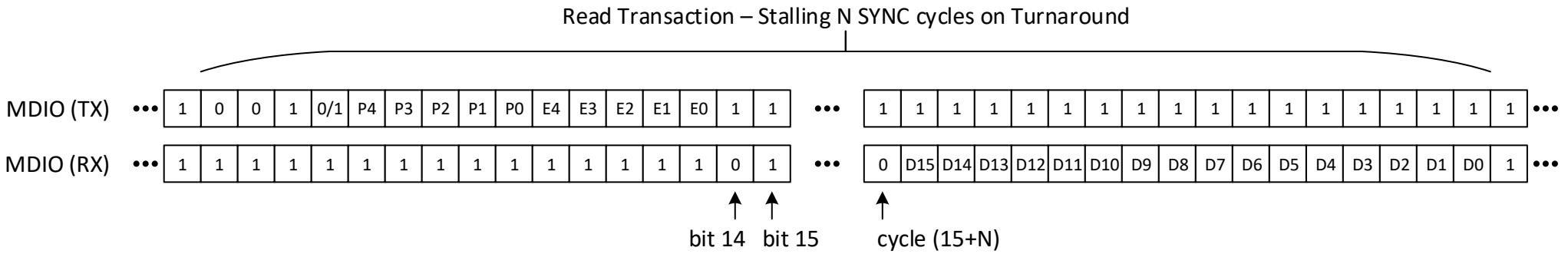
Read Operation

- Read protocol exactly the same as on a physical MDIO bus except handling during turn around.
- Bit 15 set to 0 on RX if data is available starting in bit 16
- Bit 14 can be 0 or 1 for easier implementation (next slide)



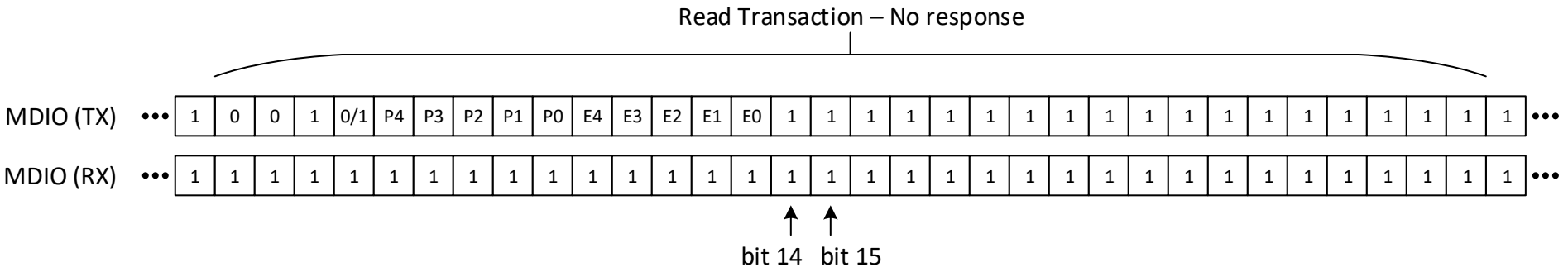
Read Operation with stalling

- PHY on physical MDC/MDIO has no options for clock stretching during turnaround
- Bit 14 set to 0 on RX if data is not going to be available starting in bit 16
- Set MDIO bit to 1 starting in bit 15 and hold to 1 until read data is ready. Set next bit to 0 and then continue returning read data.



No response to Read Operation

- Bits 14 and 15 never set to 0 indicating no response



Code value chosen to minimize toggling

- Idle – Assuming MDIO is not active
  - TXC = 1, TXD[7:0] = 0xFF - no toggling on TX.
- Low power idle
  - TXC = 1, TXD [7:0] = 0x0F – One low to high, one high to low per SYNC cycle
- Symbol error
  - TXC = 1, TXD [7:0] = 0x01 – One low to high, one high to low per SYNC cycle
- Sequence symbol
  - TXC = 1, TXD[7:0] = 0xAA – reason discussed later

Enhancement – Eliminate SYNC pin

- When combined with half rate clock, the SYNC boundary can be found using the Sync SMII sequence ordered set pattern.
- TXC = 1, TXD[7:0] = 1x1x1x1x pattern cannot be generated by skewing any bitstream with TXC = 0, TXD[7:0] = xxxxxxxx on the falling edges of CLK
- Hence the only combinations of control symbols with other symbols can create a pattern that looks like the Sync SMII sequence ordered set.
- Chart below in gray shows the control symbols to exclude from use (invalid Sync) to prevent false alignments

MDIO	TXC	TXD0	TXD1	TXD2	TXD3	TXD4	TXD5	TXD6	TXD7	MDIO	TXC	TXD0	TXD1	TXD2	TXD3	TXD4	TXD5	TXD6	TXD7	MDIO	TXC	TXD0	TXD1	TXD2	TXD3	TXD4	TXD5	Symbol causing issues
x	1	0	1	0	1	0	1	0	1	x	0	1	0	1	0	1	0	1	0									OK
		x	1	0	1	0	1	0	1	0	1	x	0	1	0	1	0	1	0	1	0							0x54, 0x55
				x	1	0	1	0	1	0	1	0	1	x	0	1	0	1	0	1	0	1	0					0x52, 0x56
						x	1	0	1	0	1	0	1	0	1	x	0	1	0	1	0	1	0	1	0			0x4A, 0x5A
								x	1	0	1	0	1	0	1	0	1	x	0	1	0	1	0	1	0	1	0	0x2A, 0x6A

AXONNE