Further Analyses of Reducing Complexity for Data Alignment

Zhongfeng Wang
Broadcom Corp., USA

IEEE P802.3bs, Logic Group meeting, June 29, 2015

MOTIVATION

- This presentation aims to give more detailed explanation and analyses about the improved data alignment scheme presented in May 2015 interim meeting [1].

[1] http://www.ieee802.org/3/bs/public/15_05/wang_z_3bs_01_0515.pdf

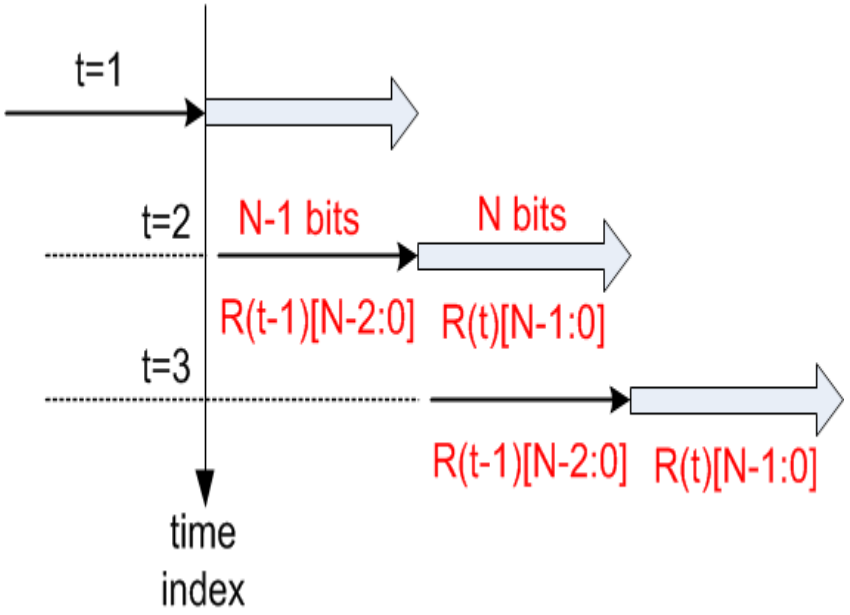
CONVENTIONAL DATA ALIGNMENT SCHEME

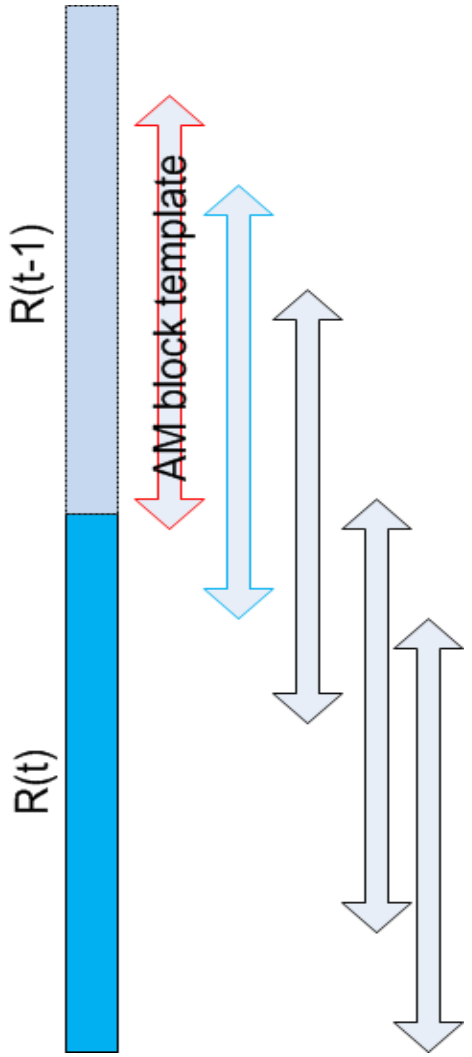
- Consider 8x50G for 400GE. Assume using a data bus of 80bits per physical lane. The data received at time instant k are denoted as x[k][79:0].

- To check whether the received data vector match the 64-b AM block, we need check a total of 80 cases per cycle, where each case involves up to 64-b data matching.
- Additionally, considering error tolerance in data matching such as 100G-KR4/KP4 system, we need counters to count total number of unmatched (or matched) symbols for each of 80 cases per cycle.

- A rough estimation for HW complexity is given below:
  - 8 (lanes) x 80 (cases) x 64 (bits) x (2 gates + 7 gates) ~= 370K gates
    where 1 XOR ~= 2 NAND gates,
    1-b adder ~= 7 NAND gates.
  - The complexity for FPGA implementation may be doubled.

PARALLEL DATA MATCH ON-THE-FLY





- For N-b data bus, form an extended vector with received vector at the current cycle and the one received in last cycle.
- Perform N-parallel data matching for a given AM data pattern

NEW DATA PATTERN MATCHING SCHEME

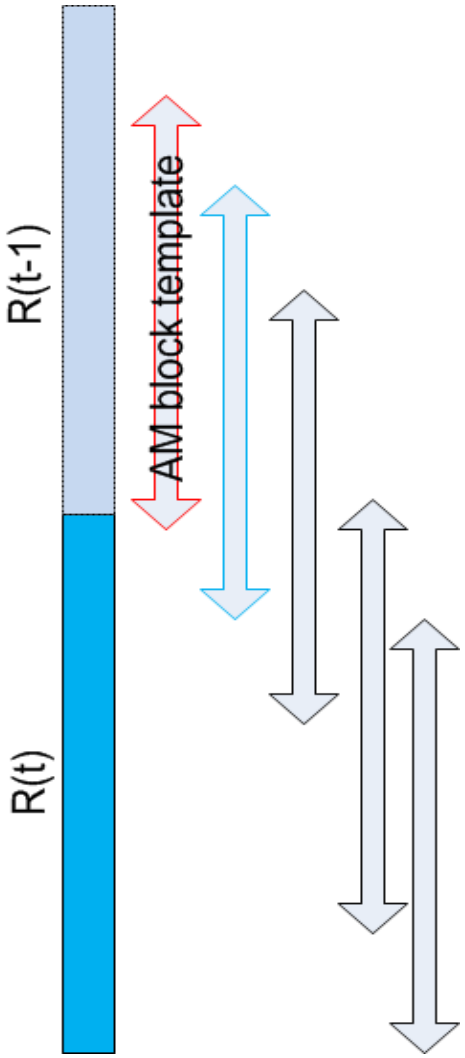
- In [1], we proposed an low-complexity approach for data matching. We check if the current received data (with partial history) matches with another data vector received in the past with a fixed distance (e.g., 2 cycles). For buswidth of 80 bits, we **only need check one case**: {x[k-1][78:0], x[k][79:0]) with {x[k-3][78:0], x[k-2][79:0]}.

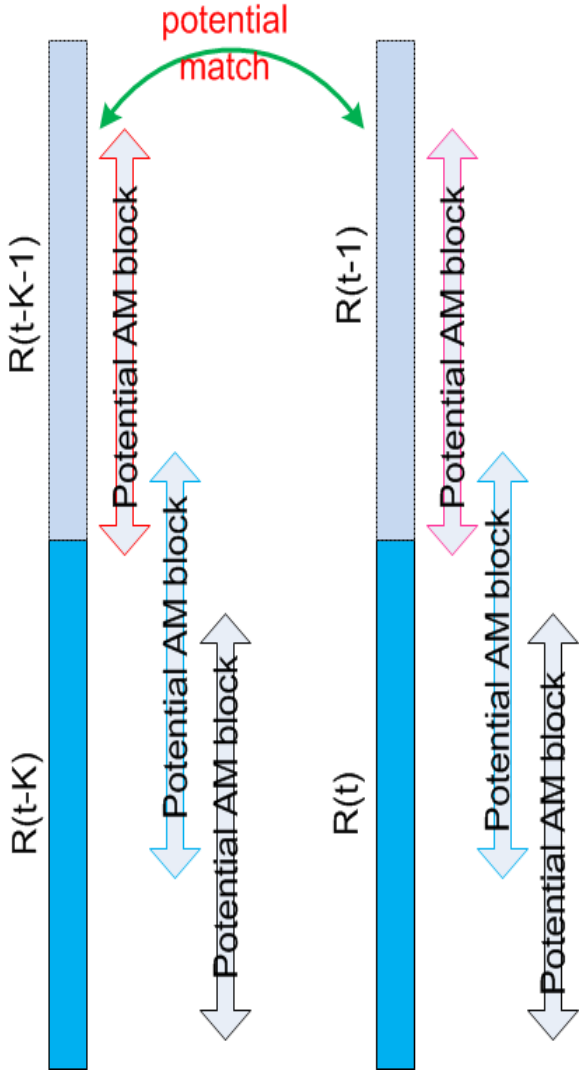
- The complexity reduction ratio estimation:
  ~= 80 (cases) *64 (bits) /160 (bits) =32.
  - Real reduction may be smaller considering HW sharing in former case. However, even 5X reduction is very significant.

- To facilitate this efficient data matching, we need set two related data patterns on a physical lane to be the same or bit-reversed.

[1] http://www.ieee802.org/3/bs/public/15_05/wang_z_3bs_01_0515.pdf

TWO DIFFERENT DATA MATCHING SCHEMES

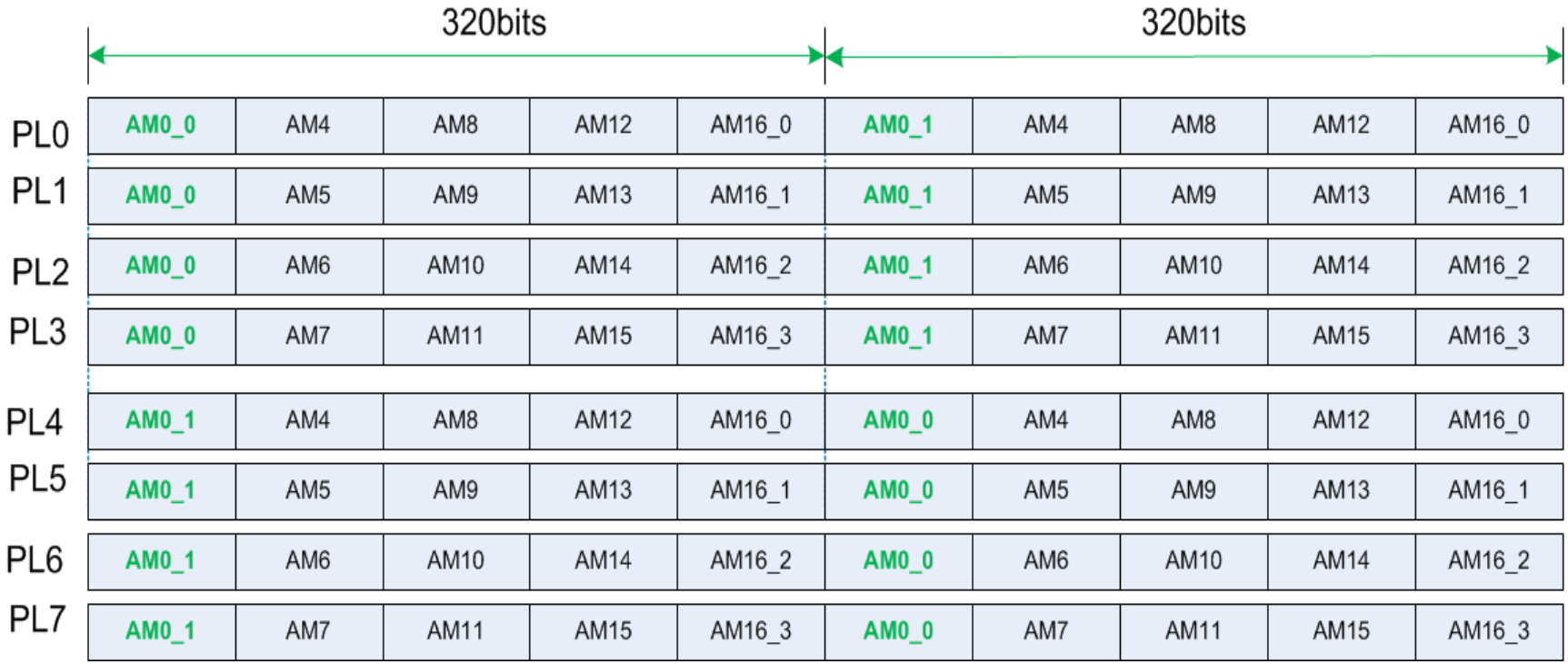




NEW DATA MATCHING SCHEME (II)

- As shown in the right diagram on the previous page, with the new scheme, we only need perform data matching for one case.
- To facilitate this efficient data matching, we need set two related data patterns transmitted over a physical lane to be the same or bit-reversed.
- Considering necessity of error-tolerance in data matching, we can allow a few number of bit errors or symbol (e.g., 4-b per symbol) errors for a data matching.
- To reduce false alarming probability, the control logic should check if the data matched streams are mainly over 1 long segment or over 2 long segments (depending on AM block patterns).

EXAMPLE ALIGNMENT MARKERS FOR 8 PL'S



- Assume AM0_0 is bit-reversed version of AM0_1 for the portions of data matching.
- Assume AM16_0, AM16_1, AM16_2, and AM16_13 are partially bit-reversed version of each other, e.g., AM16_0={u, v}, AM16_1={~u, v}, AM16_2={~u, ~v}, AM16_3={u, ~v}.

EXAMPLE ALIGNMENT MARKERS FOR 16 PL'S

PL0	AM0_0	AM4	AM0_2	AM12	AM16_0
PL1	AM0_0	AM5	AM0_2	AM13	AM16_1
PL2	AM0_0	AM6	AM0_2	AM14	AM16_2
PL3	AM0_0	AM7	AM0_2	AM15	AM16_3
PL4	AM0_1	AM4	AM0_3	AM12	AM16_0
PL5	AM0_1	AM5	AM0_3	AM13	AM16_1
PL6	AM0_1	AM6	AM0_3	AM14	AM16_2
PL7	AM0_1	AM7	AM0_3	AM15	AM16_3
PL8	AM0_2	AM4	AM0_0	AM15	AM16_3
.....					
PL14	AM0_3	AM6	AM0_1	AM14	AM16_2
PL15	AM0_3	AM7	AM0_1	AM15	AM16_3

FURTHER ANALYSES ABOUT NEW SCHEME

- AM0_0, AM0_1, AM0_2, and AM0_3 can be all the same as AM0. In this case, we need more variants of AM16 to differentiate 16 or 8 different PL's. The minimum distance between any of two variants of AM16 should be maximized.

- If people are not comfortable with the new data matching scheme, they can still use conventional data matching method.

- Since we have two (kind of) identical AM blocks in a group, we can speed-up data matching if necessary.

- It should be noted that some consecutive operations in data matching can be done in a pipelined way. For instance, in one cycle, we check total matched bits or symbols. In the next cycle, we check if these matched bits or symbols are mainly from 1 long data segment or 2 segments.

FINAL REMARKS

- We have presented more technical details for the new data alignment scheme.
- We have shown its big potential in reducing HW complexity (> 5X) while still keeping option of conventional data matching.
- There’re many ways to arrange AM blocks to facilitate the proposed data matching scheme.