De-emphasis Buffer Modeling Issues with IBIS

Nanditha Rao
Intel Corporation
nanditha.p.rao@intel.com

Asian IBIS Summit
Tokyo, Japan
November 14, 2008

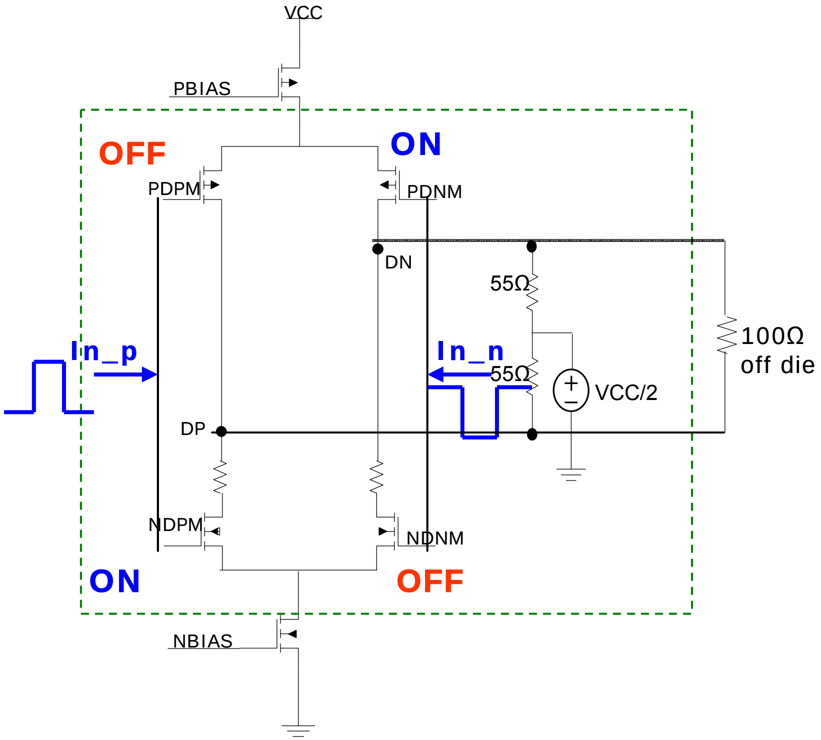
Legal Disclaimer

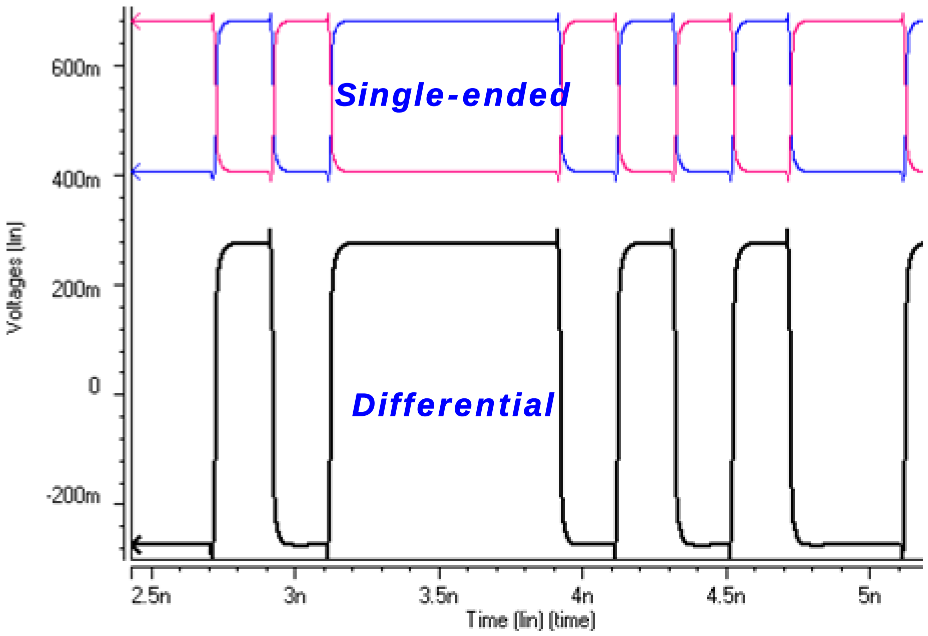
Agenda

- Brief background
  - Differential buffer with de-emphasis
  - De-emphasis implementation in IBIS
- Different de-emphasis implementation
  - A different de-emphasis approach
  - Boost stimulus derivation
  - Example boost stimulus in SPICE
- Modeling the “new” boost
  - Issues with defining the boost
  - Single-ended or differential?
  - Verilog* AMS approach
- Comments and future work

Differential Buffer

- PCI Express* differential buffer example

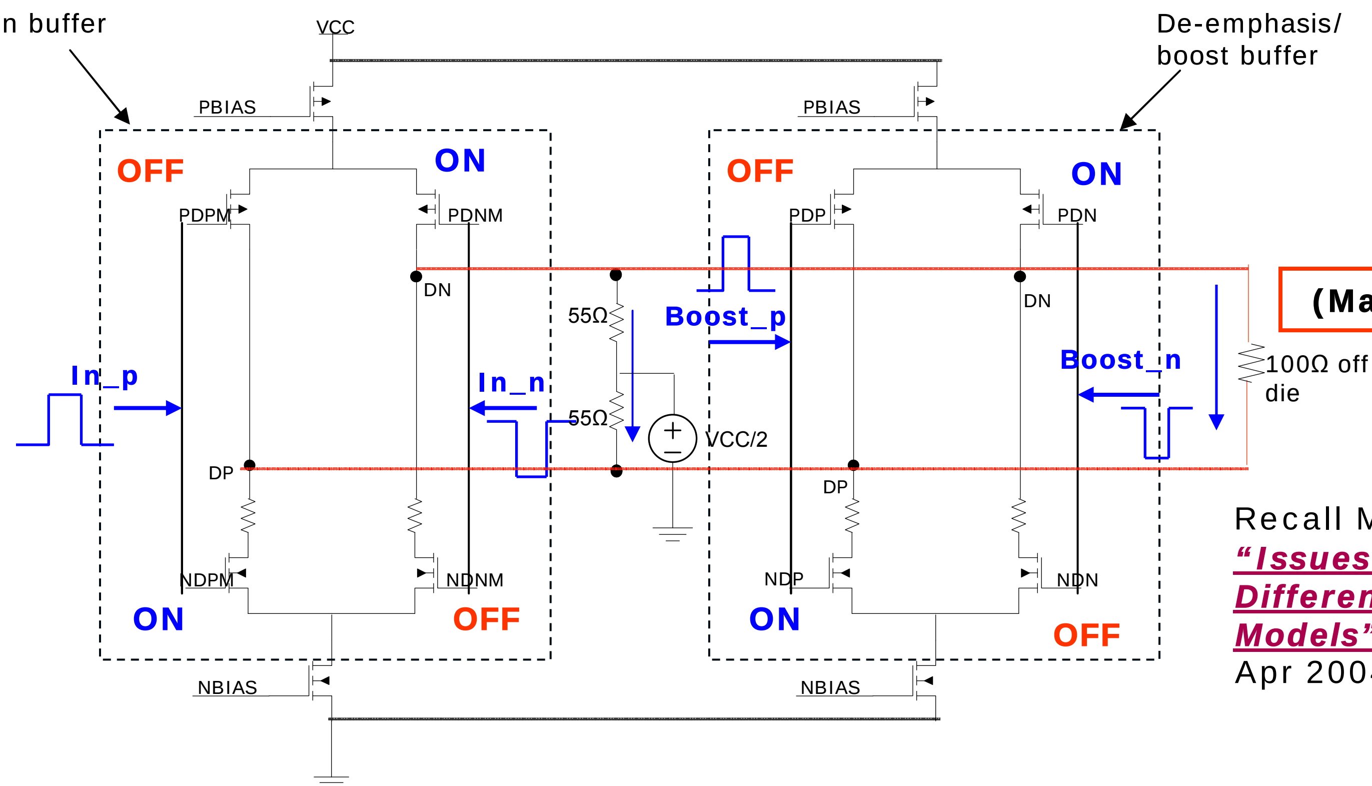




Note: NBIAS and PBIAS are set to get 10mA from the driver

Differential Buffer with De-emphasis

- Adding a second buffer (de-emphasis/boost)- WIRED-OR
- Boost adds to the main during full-swing (101010) operation



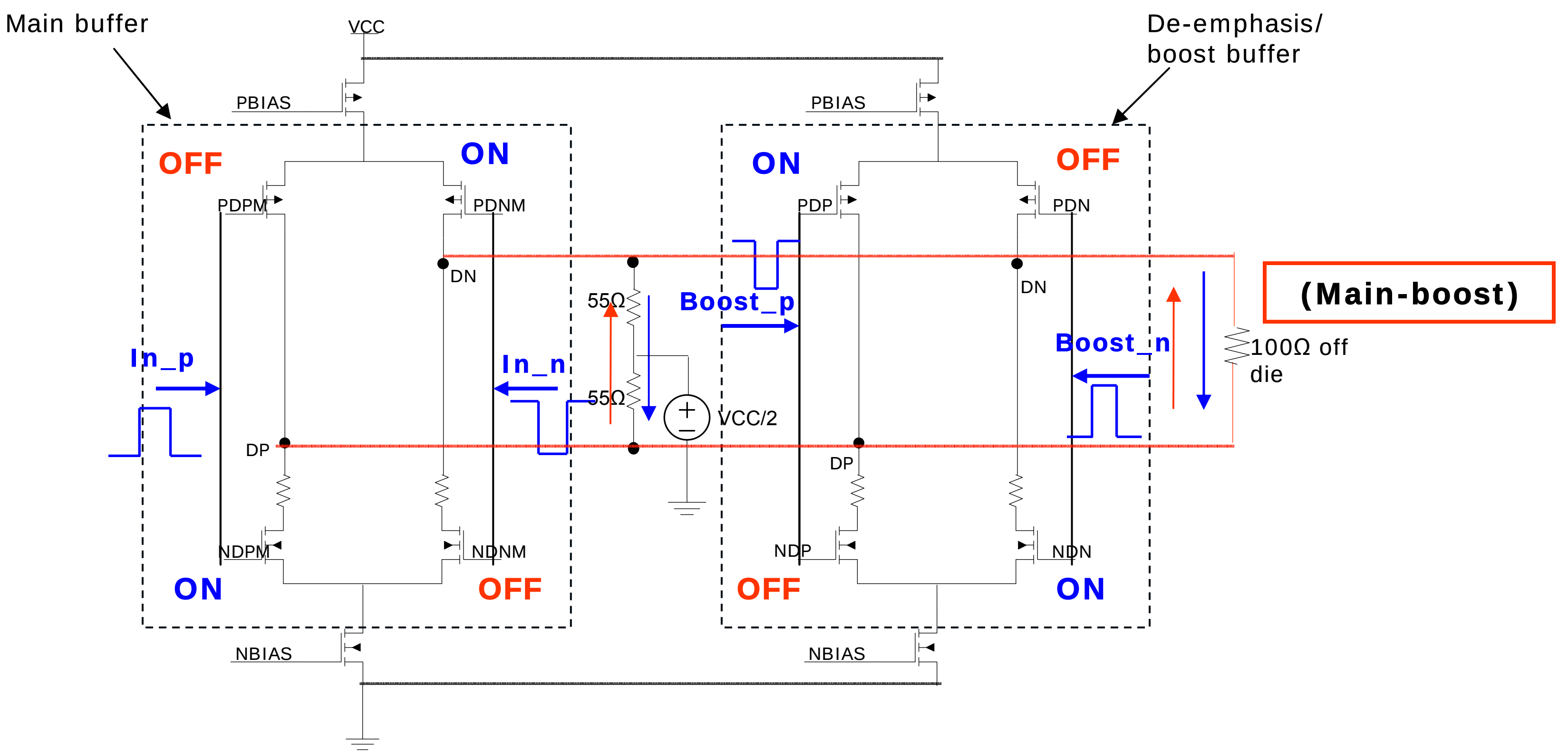
Recall M. Mirmak's
"Issues with C_comp and Differential Multi-stage IBIS Models"
Apr 2004

In_p and In_n are main stimulus patterns
Boost_p = In_p
Boost_n = In_n

*Other names and brands may be claimed as the property of others

Differential Buffer with De-emphasis

During de-emphasized operation (continuous ones/zeroes), opposite legs of the buffer are switched on (one of the methodologies)

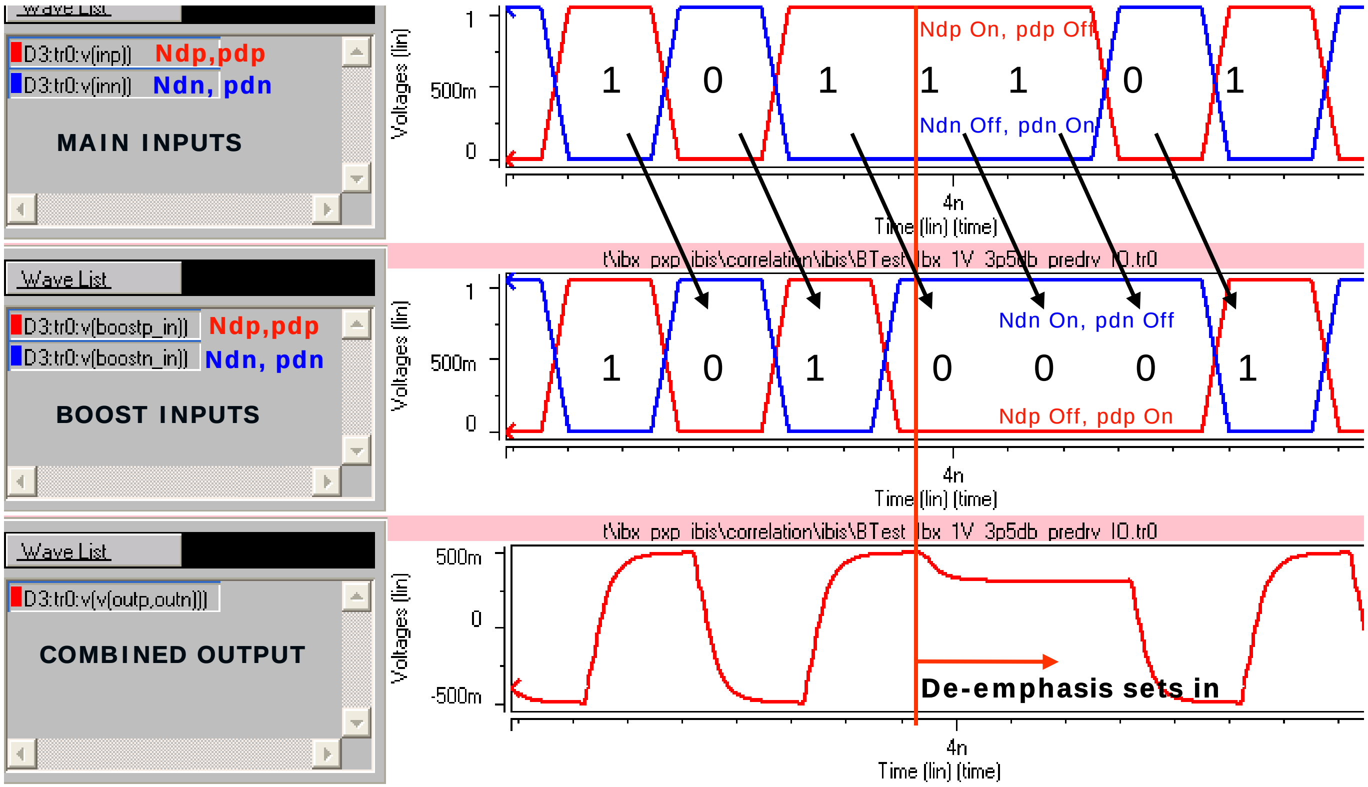


$\text{Boost\_p} = \overline{\text{In\_p}}$

$\text{Boost\_n} = \overline{\text{In\_n}}$

De-emphasis Implementation in IBIS

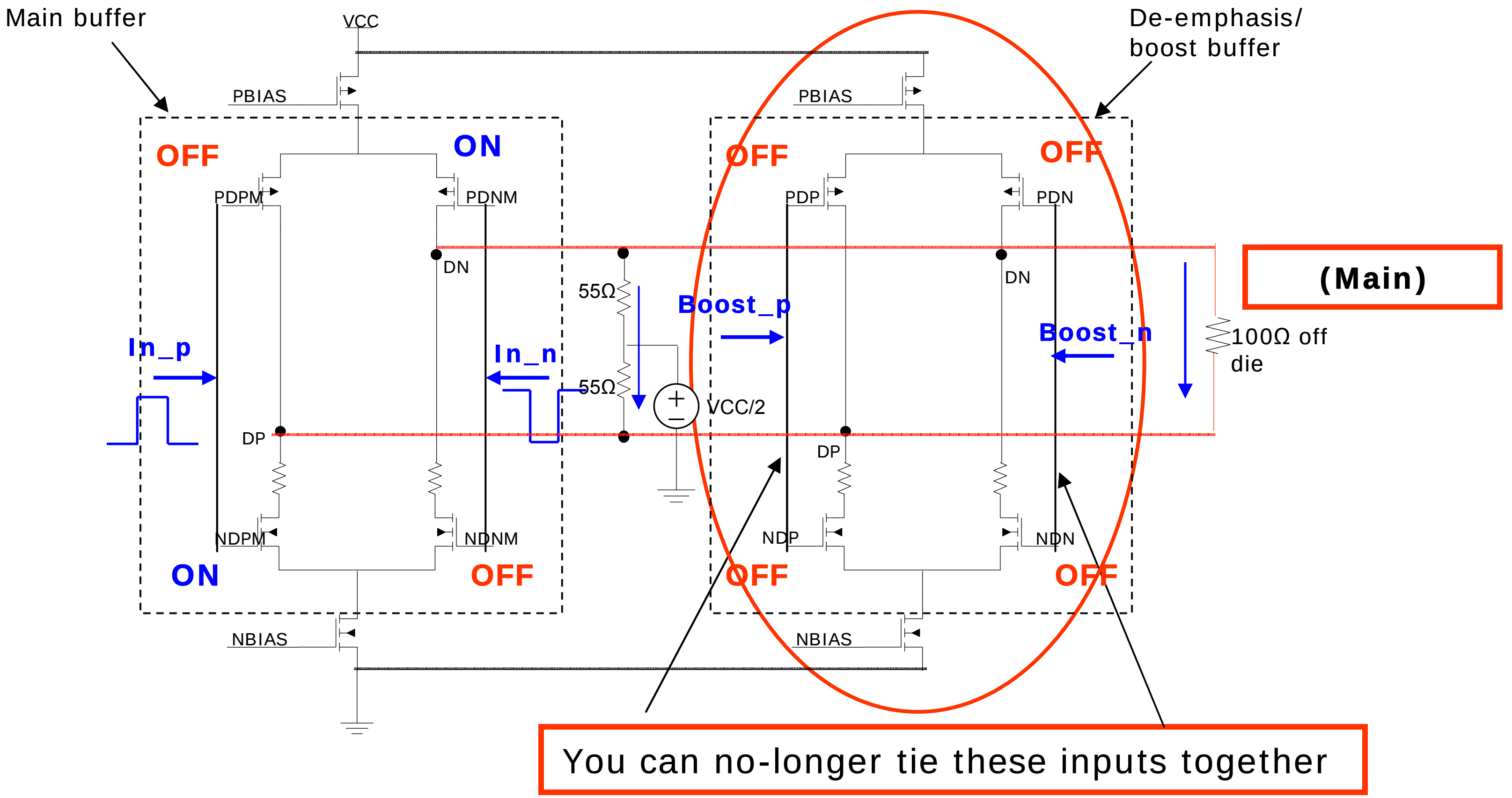
- Two differential buffers– main and boost are WIRED-OR
- Terminations and parasitics shouldn't be duplicated
- Boost is driven by a stimulus derived from the main pattern depending on the de-emphasis logic
- C_comp is split between main and boost in the ratio of number of driver slices (one of the ways)



$\boxed{Boost\ (t) = \overline{Inp\ (t-1)}}$

De-emphasis - Different Approach

- During de-emphasized operation, the boost buffer is completely **shut off**. Load sees only main



Boost Stimulus… Some Derivations

Boost input stimulus will need to be different

Inp	1	0	1	0	1	1	1	1	0	0
						OFF	OFF	OFF		OFF
Boost_NDP	1	0	1	0	1	0	0	0	0	0
Boost_PDP	1	0	1	0	1	1	1	1	0	1
						OFF	OFF	OFF		OFF

Inp(t-1)	Inp(t)	Boost_NDP	Boost_PDP
0	0	0	1
0	1	1	1
1	0	0	0
1	1	0	1

$$\text{Boost\_NDP} = \overline{\text{Inp(t-1)}} \,.\, \text{Inp(t)}$$

$$\text{Boost\_PDP} = \overline{\text{Inp(t-1)} \,.\, \overline{\text{Inp(t)}}}$$

$$\text{Boost\_PDP} = \overline{\overline{\text{Boost\_PDP}}}$$

Similar logic applies for ndn and pdn, except that they will follow Inn

$$\text{Boost\_NDN} = \overline{\text{Inn(t-1)}} \,.\, \text{Inn(t)}$$

$$\text{Boost\_PDN} = \overline{\text{Inn(t-1)} \,.\, \overline{\text{Inn(t)}}}$$

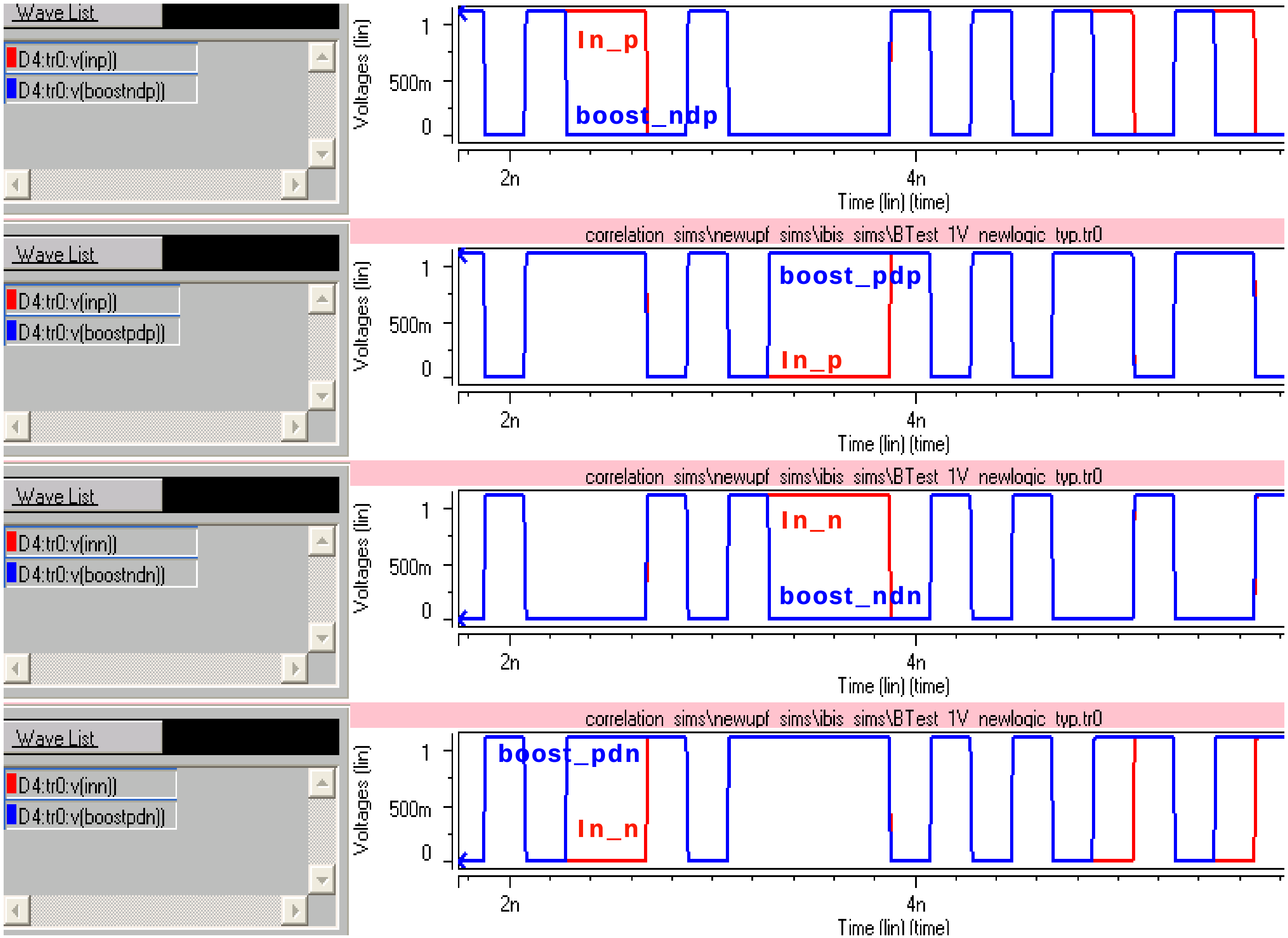
$$\text{Boost\_PDN} = \overline{\overline{\text{Boost\_PDN}}}$$

Example Boost Stimulus in SPICE

```
*********************** Logic for Boost_NDP************************
E_inx     inpdelay  0 DELAY inp        0 TD='period/2'    ****Delay
E_intn    in2       0        vol='inV-V(inpdelay)'   **Inversion of the delay
**AND inp and in2
Gswitch Boost_NDP inp VCR PWL(1) inp 0 0v,1e-6 vcc,10m
Gswitch1 Boost_NDP in2 VCR PWL(1) in2 0 0v,1e-6 vcc,10m

Rxxx Boost_NDP 0 1E6   *** OUTPUT LOAD

************************Logic for Boost_PDP*********************
E_inx1    inpdelay1 0 DELAY   inp 0 TD='period/2‘  **Delay
E_inn1    inpbar    0 vol='inV-V(inp)‘  **Inversion

Gswitch2 bpbar inpbar VCR PWL(1) inpbar 0 0v,1e-6 vcc,10m
Gswitch3 bpbar inpdelay1 VCR PWL(1) inpdelay1 0 0v,1e-6 vcc,10m

E_boostp  Boost_PDP 0  vol='inV-V(bpbar)‘  **Inversion

Rxxx1 Boost_PDP 0 1E6  *** OUTPUT LOAD
```

Similarly for NDN and PDN

The ‘Working’ Stimulus



NDP=0 (OFF) during de-emphasis

N follows i/p during normal mode

PDP=1 (OFF) during de-emphasis

P follows i/p during normal mode

NDN=0 (OFF) during de-emphasis

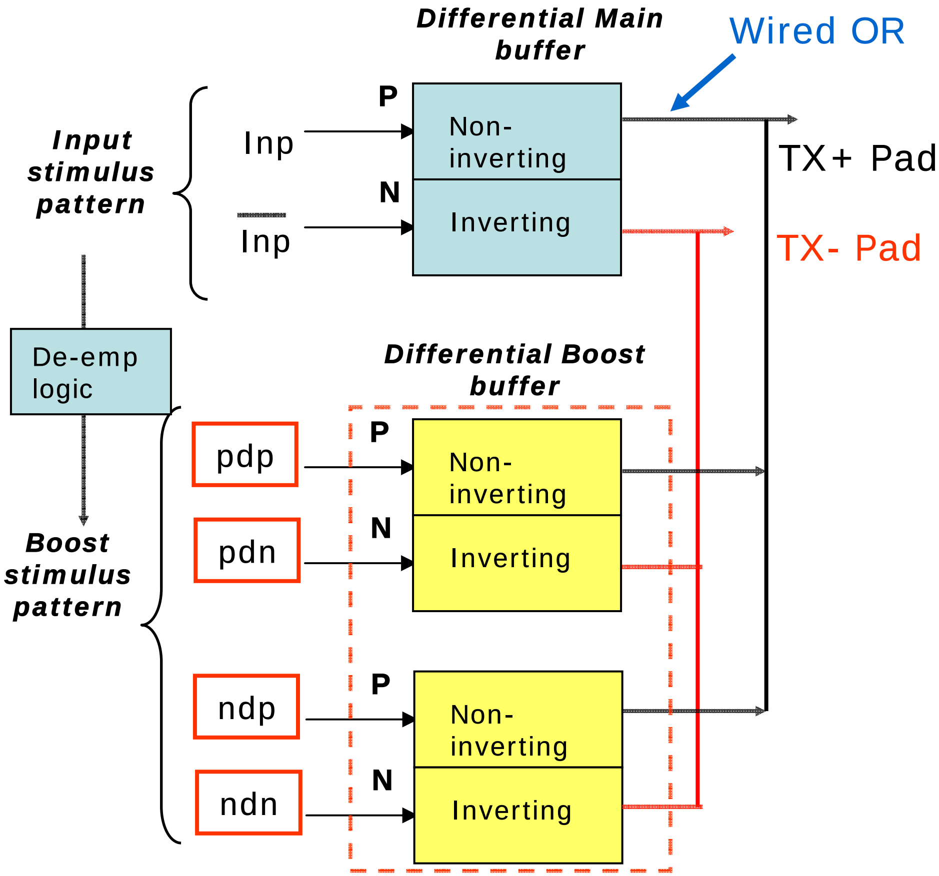
N follows i/p during normal mode

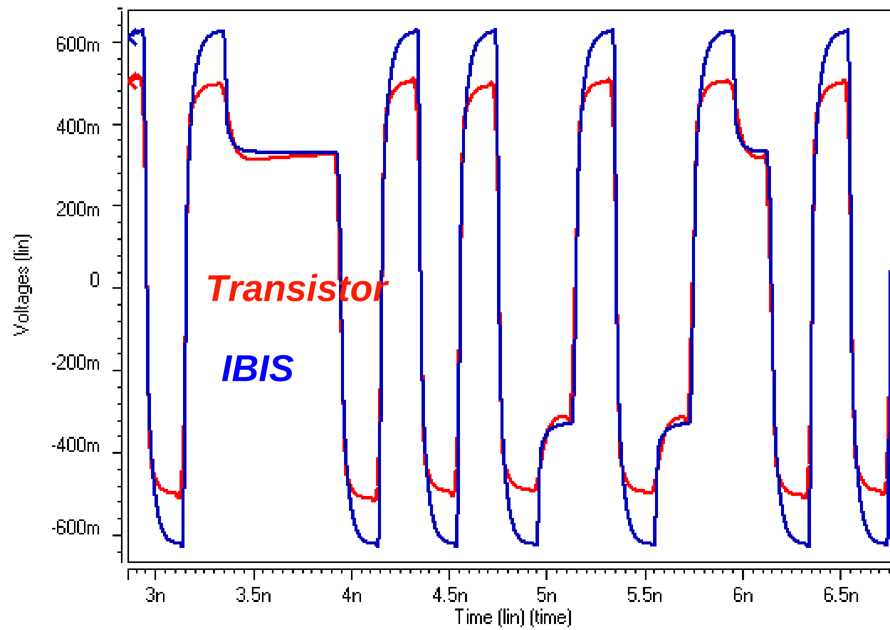
PDN=1 (OFF) during de-emphasis

P follows i/p during normal mode

Modeling the 'New' Boost

- Approach 1: Define 2 B-elements for main and 1 B-element for each of pdn, pdp,ndp and ndn (modeled as differential buffer)



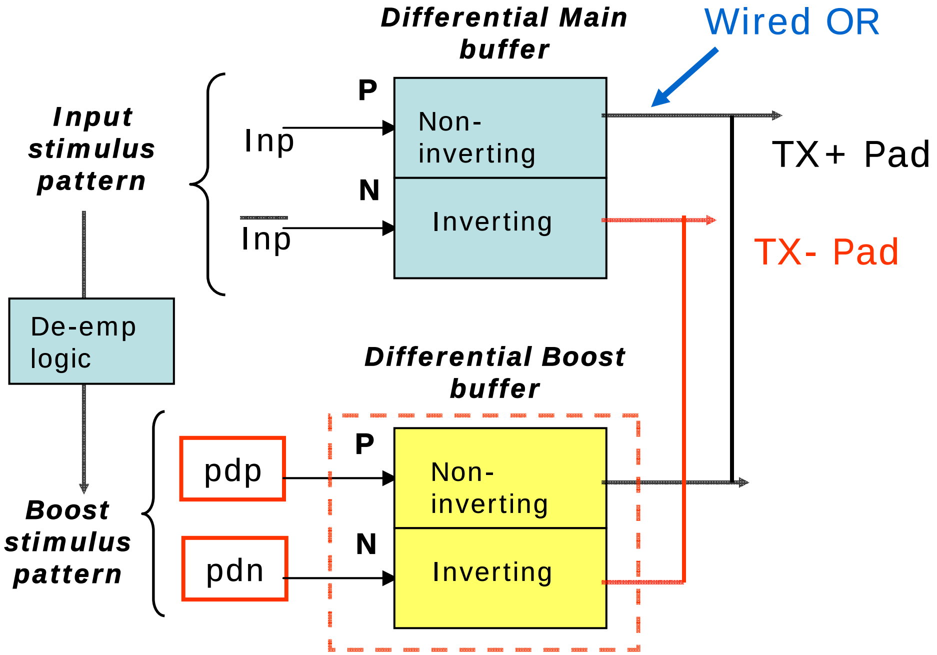


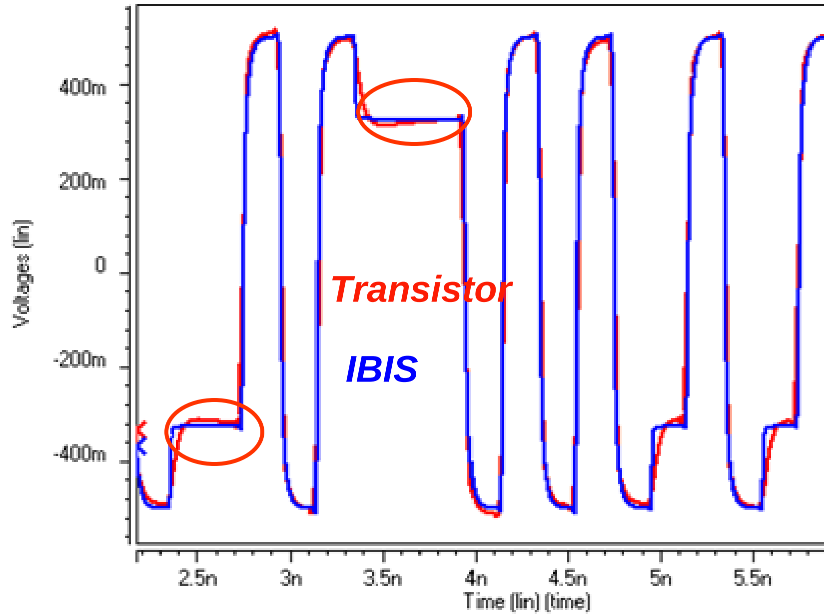
Doesn't work as

Boost == 2 differential buffers

Modeling the 'New' Boost

- Approach 2: Define either pdp and pdn *or* ndp and ndn (modeled as differential buffer)

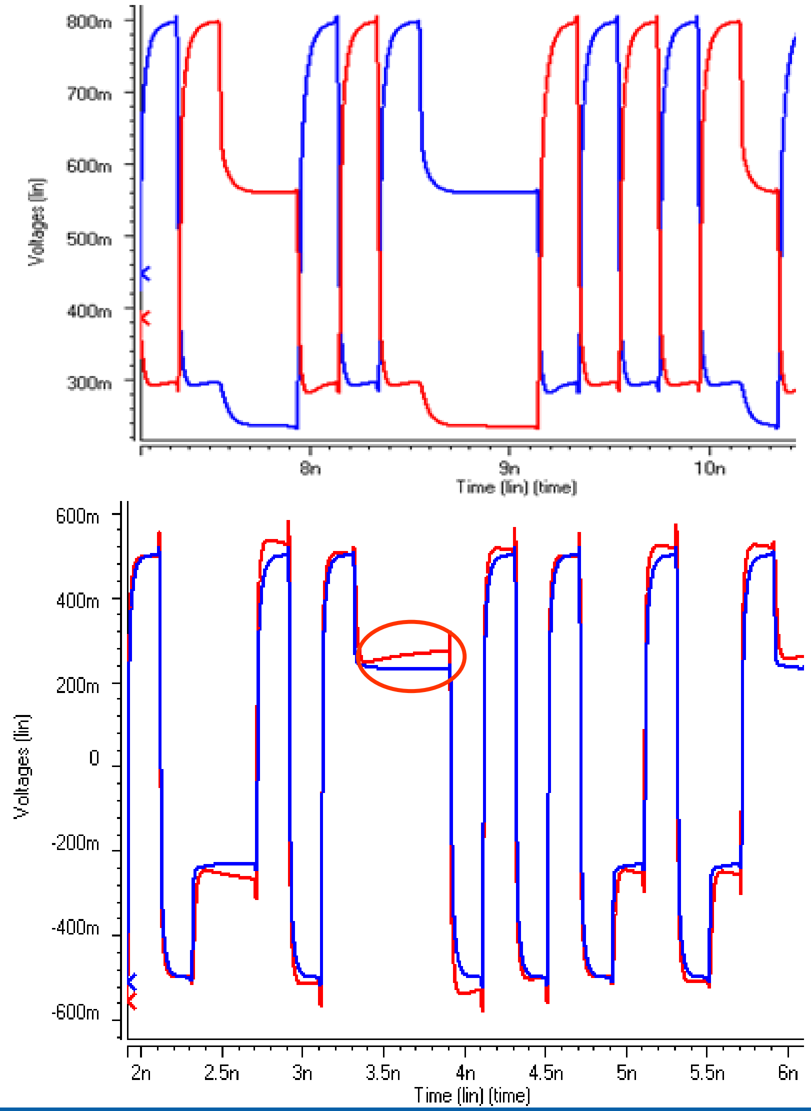




- Differential swing correlates well
- Mismatch on de-emphasis transition
- C_comp distribution gets tricky

Recall A.Muranyi's ***"Data Dependent Buffer Characteristics"*** Jan 2003

A Further Issue



However, single ended signals are incorrect as we ignored ndn and ndp

Transistor

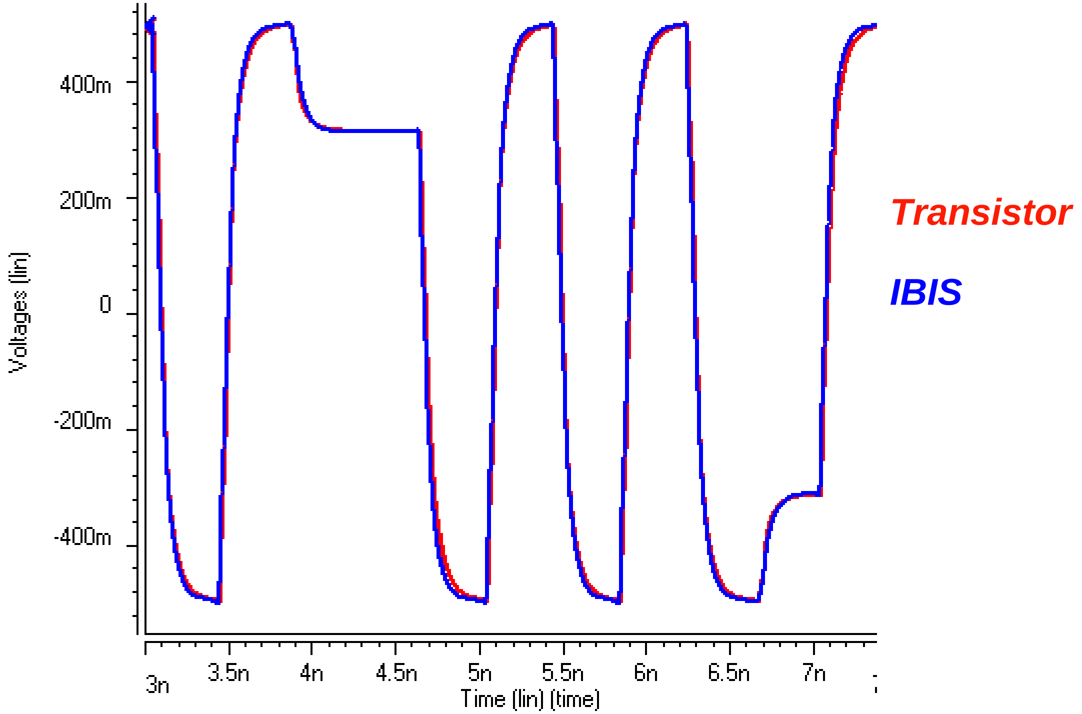
IBIS

6-dB De-emphasis

- Half the driver is switched off during switch from transition to non-transition ==>Odd behavior in transistor
- Table-based IBIS cannot mimic this behavior

Earlier De-emp Approach

Current reverses through boost buffer during de-emphasis

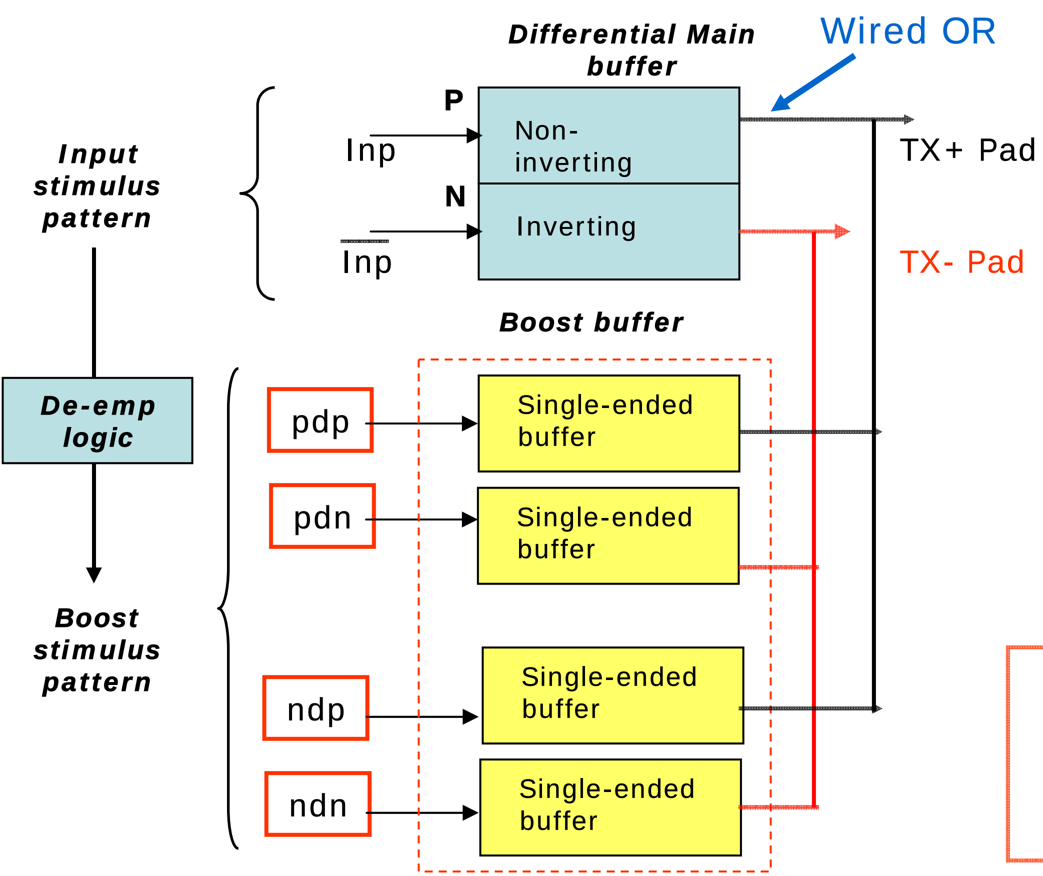


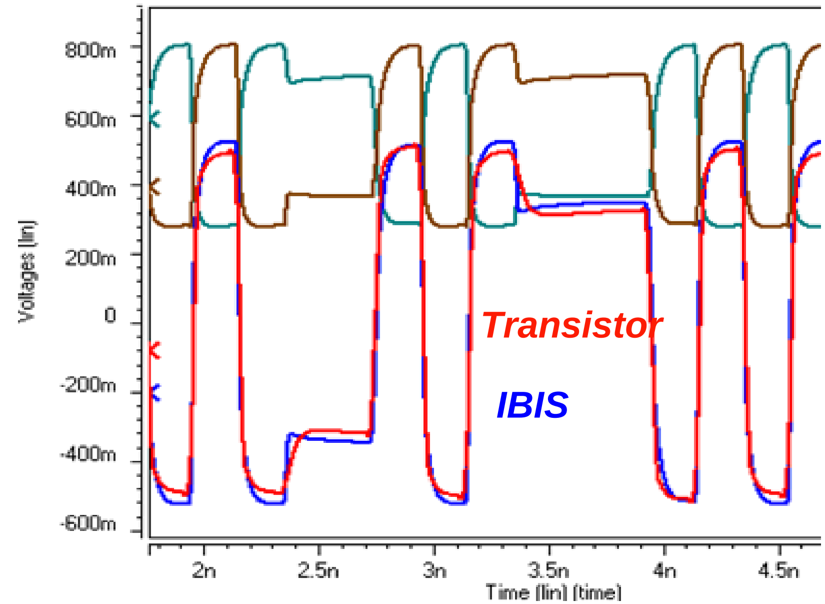
Miscorrelation was not observed with the previous de-emp methodology

Fairly good C_comp distribution

Modeling the 'New' Boost

- Approach 3:
  - Model boost as single-ended buffer instead of differential
  - Define 4 B-elements- with pdp, pdn, ndp and ndn stimulus



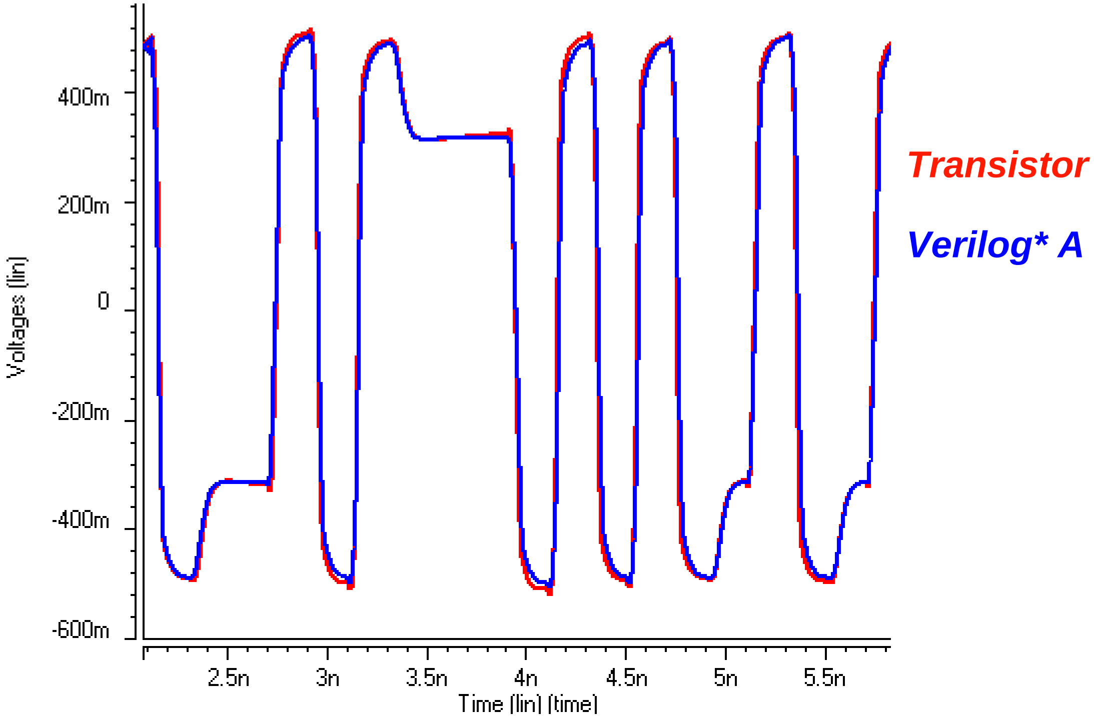


Issue with weird single ended signals resolved

But miscorrelation in differential swing!

Verilog* AMS Model

- Approach 4:
  - Create equation based Verilog*-AMS model for the differential buffer
  - Shows good correlation



Comments and Future Work

- Different de-emphasis implementations will need different boost stimulus logic
- Correlation issues observed while modeling the new de-emphasis buffer using table-based ibis
  - Manual fine tuning could fix this, but tedious and not the best approach
- Verilog* AMS approach provides good correlation
- Alternatives for investigation:
  - Driver schedule
  - AMS approach