ABCD: Booleanizing Continuous Systems for Analog/Mixed-Signal Design, Simulation, and Verification

Aadithya V. Karthik,

Sayak Ray, Pierluigi Nuzzo, Alan Mishchenko,

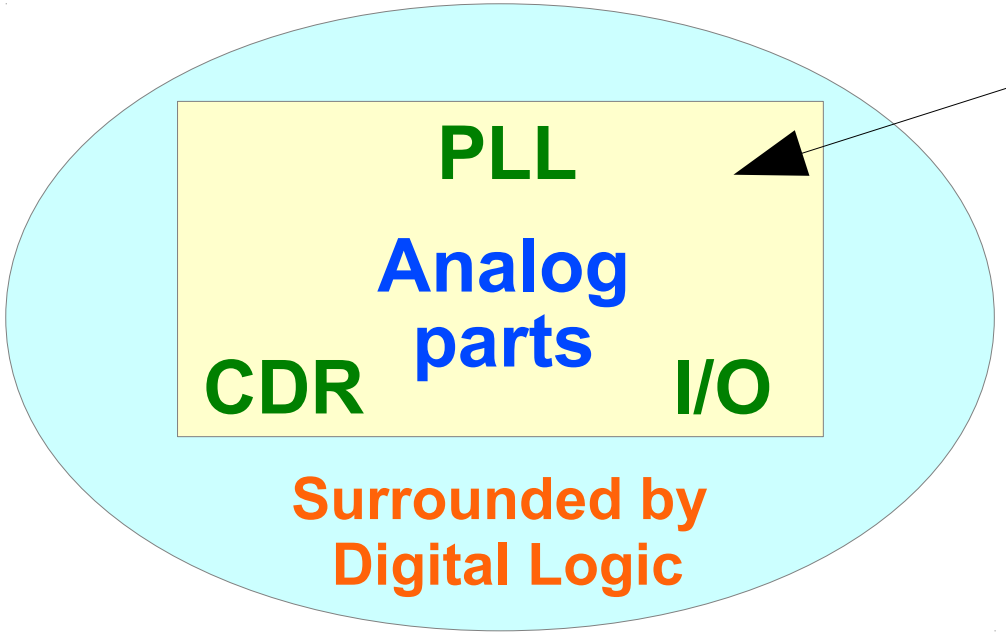
Robert Brayton, and Jaijeet Roychowdhury

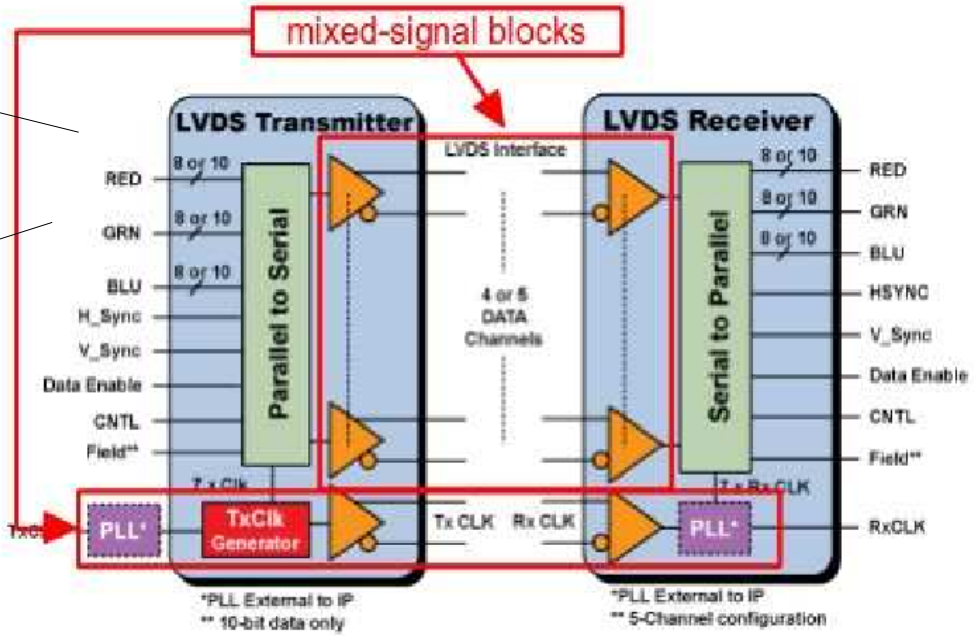
EECS Dept., The University of California, Berkeley

TAU 2014, Santa Cruz

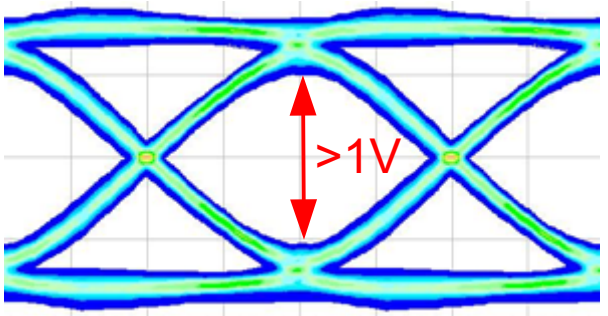
The Problem: AMS Verification

Example: SERDES





- Want to verify complete system
  - e.g., eye opening height > 1V?
- Proof or counter-example needed

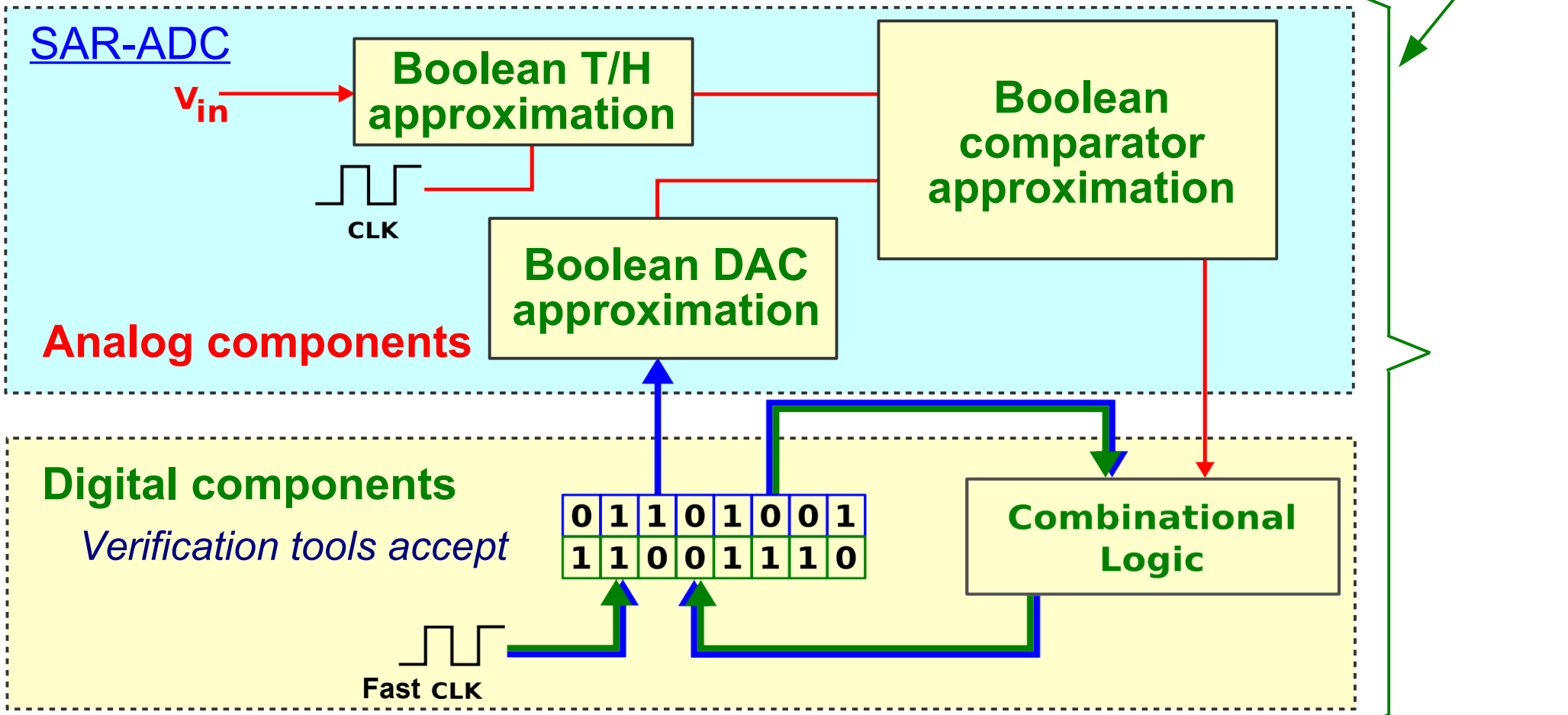


Our approach: "Booleanize" the analog parts

~~Challenge:~~ **ABCD: Boolean approximation** + Digital models *Boolean* **(don't mix)**

Best verification tools = all Boolean, no continuous

ALL BOOLEAN

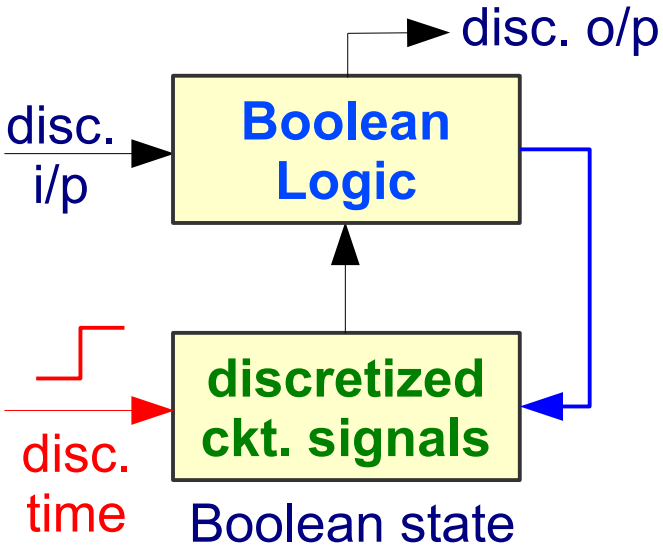


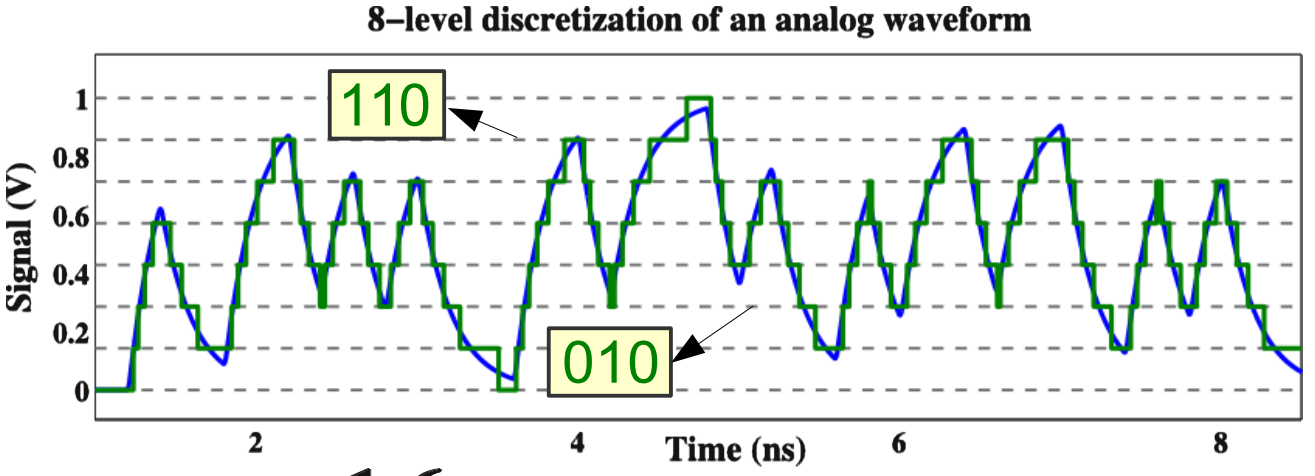
Formal verification, high-speed simulation, test pattern generation, ...

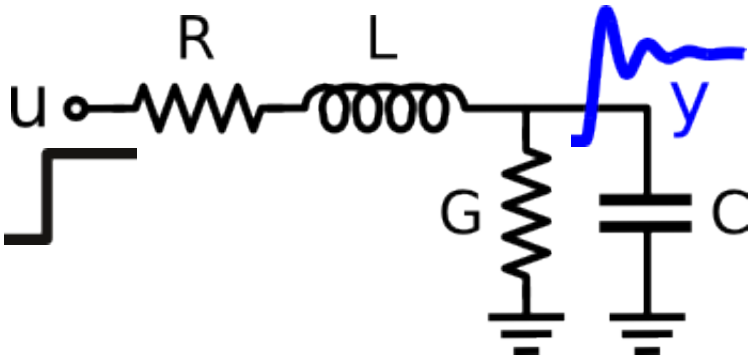
… for the full combined system!

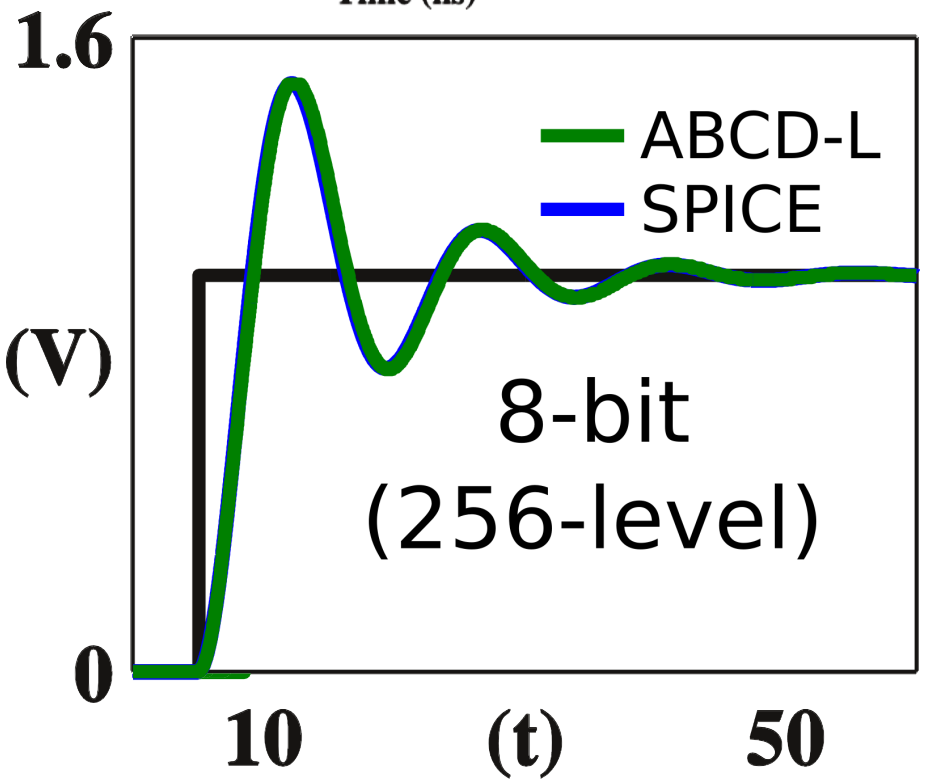
ABCD: Boolean, but Accurate

- What is “accurate Booleanization”?









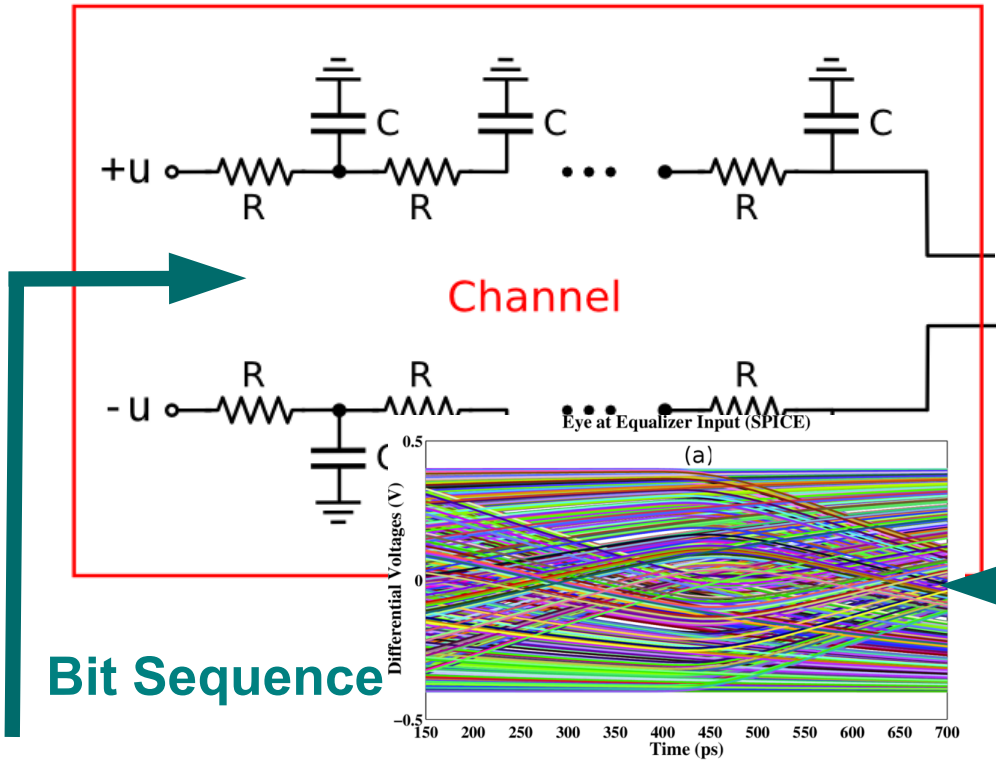
More bits, better accuracy

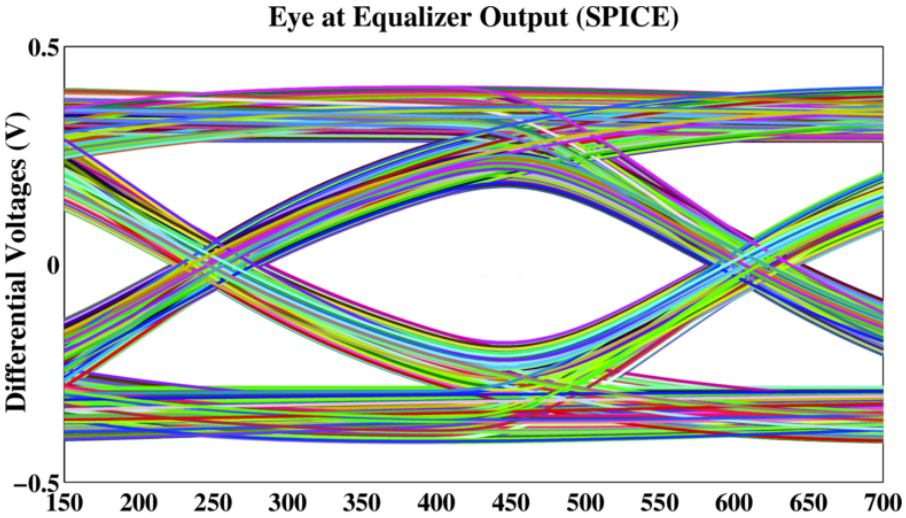
Prior work: ABCD-L (DAC 2013)

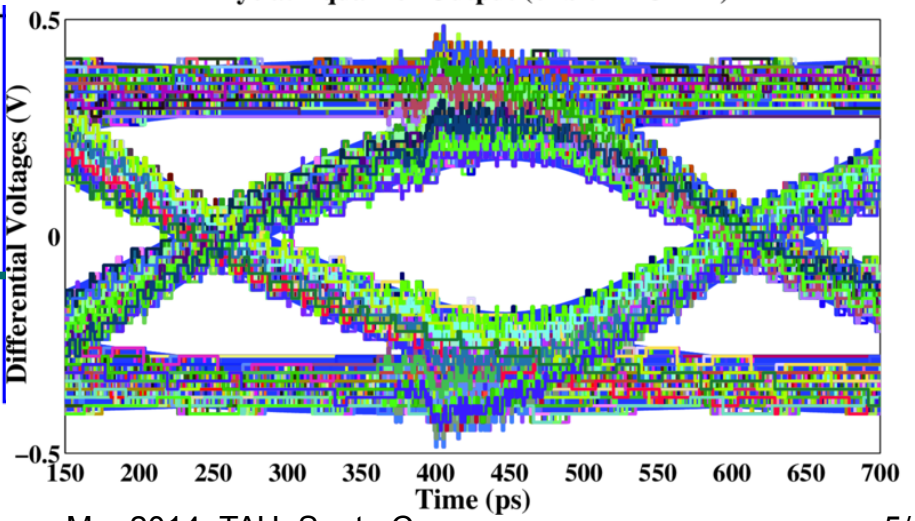
Linear Analog Circuit → **ABCD-L** → **Purely Boolean Model**

Works only for linear systems!

Example: Channel + Equalizer







Introducing ABCD-NL

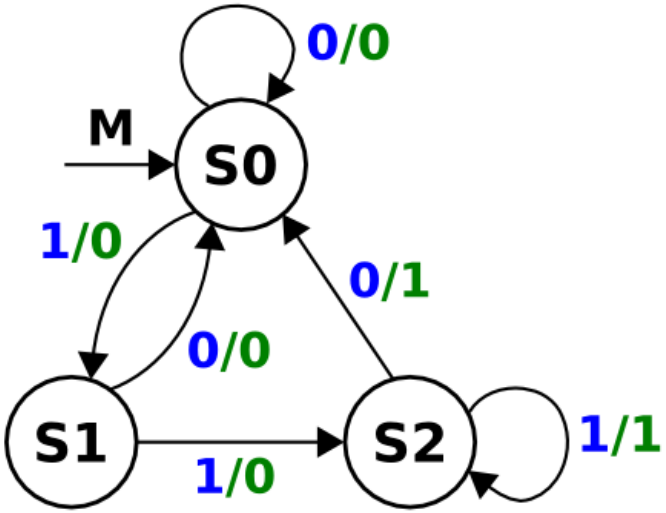
Rest of this talk

1. How ABCD-NL works
2. Results: Charge pump, ADC, DAC, etc.
3. End user applications

   High-speed simulation, formal verification, test pattern generation, etc. for non-linear AMS ckts.

ABCD-NL Models are FSMs

- Finite number of states
- Arcs denoting state transitions
  - Each arc: ip/op pair
- **Purely Boolean form**

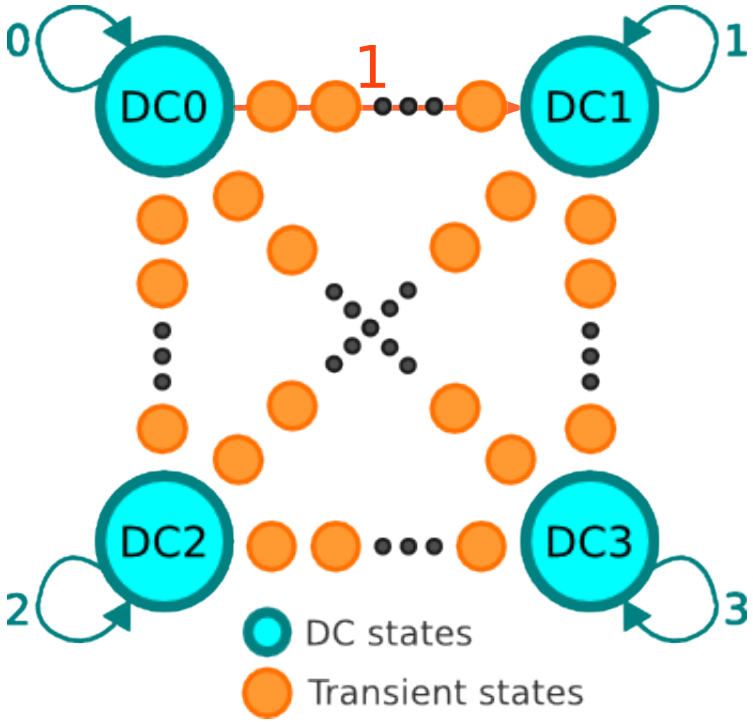


1100 → 0010

Key idea: DC, TRAN FSM states

(DC input → DC output)

Starts at DC0 and
eventually settles at DC1

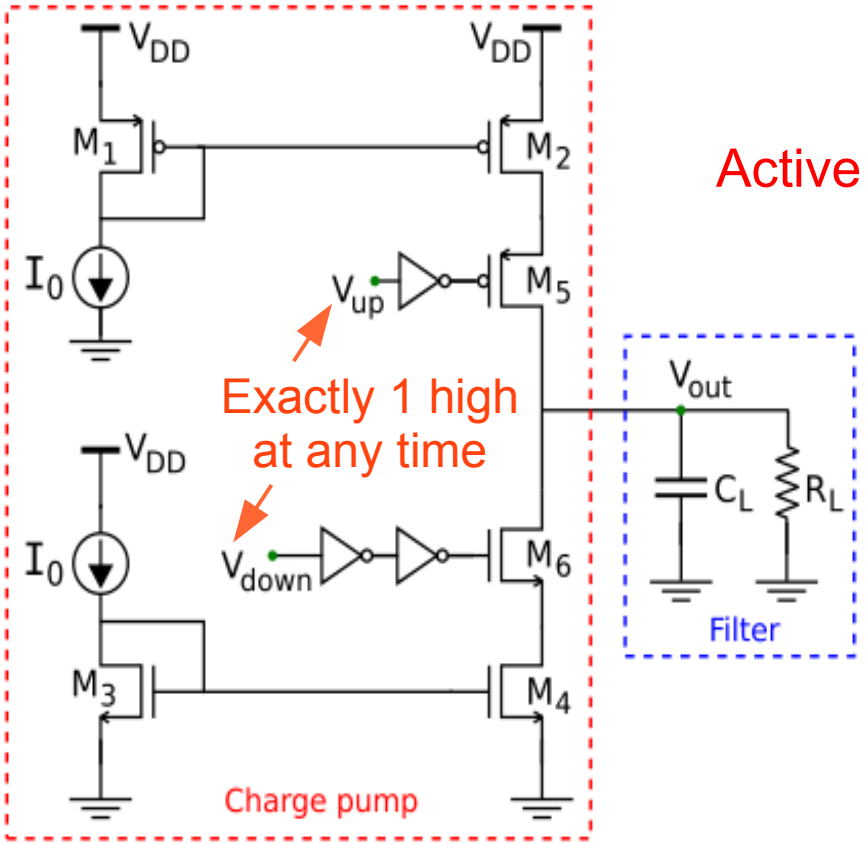


- DC FSM states w/ loops
- Multiple such DC states
  - Capture different DCOPs
- DCOPs don't change instantly
- TRAN FSM states
  - Between each pair of DC states
- **Purely Boolean Model**

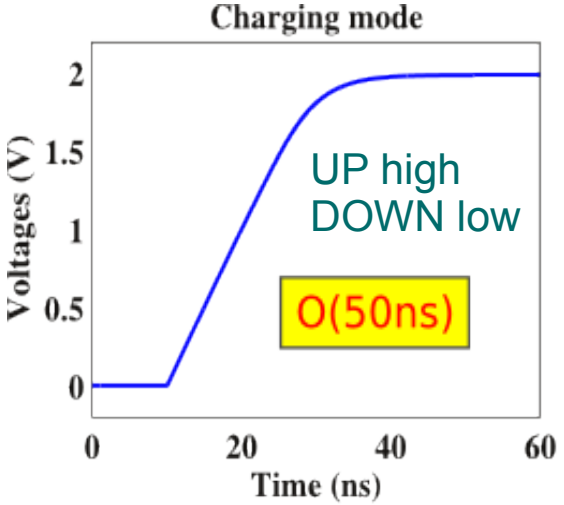
Booleanizing a Charge Pump (1/3)

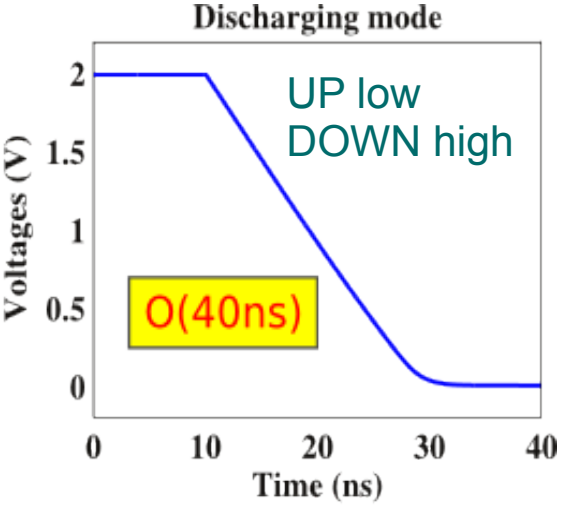
1 Do SPICE simulations

Discretize Vup, Vdown using 1 bit each, Vout using 5 bits



Active





Booleanizing a Charge Pump (2/3)

2

Analyze SPICE waveforms

Build “Analog Transition Table”

```
(1) charge-pump-B
1 1 0 0 2
1 2 5953 0 2 531.47 1 2103.6 0
1 3 27370 0 2 308.59 3 946.86 4 1585.7 5 2225.2 6 2865.3 7 3506 8 4147.7 9 4790.6 10 5435 11 6081.4 12 6730.2 13 7381.7 14 8036.3 15 8694.6 16 9356.7 17
1 4 2.7687e+05 0 2 1296.7 3 6674.3 4 15132 5 26788 6 42588 7 65017 8 1.0136e+05 9 1.9873e+05 10
2 1 1.6072e+06 0 0 76170 1 4.2257e+05 2
2 2 0 0 0
2 3 28414 0 0 318.37 1 955.6 2 1593.4 3 2231.6 4 2870.5 5 3510 6 4150 7 4790.8 8 5432.5 9 6075.3 10 6719.8 11 7366.2 12 8014.9 13 8666.5 14 9321.1 15 99
2 4 2.6574e+05 0 0 316.2 1 1511.7 2 4131.5 3 9507.9 4 17962 5 29612 6 45400 7 67805 8 1.0408e+05 9 2.0088e+05 10
3 1 2.9139e+06 0 31 8442.7 30 33342 29 62433 28 94928 27 1.301e+05 26 1.6746e+05 25 2.0677e+05 24 2.4798e+05 23 2.9114e+05 22 3.3637e+05 21 3.8384e+05 2
3 2 20630 0 31 213.69 30 804.46 29 1395.6 28 1987.2 27 2579.1 26 3171.5 25 3764.3 24 4357.5 23 4951.3 22 5545.6 21 6141 20 6738.6 19 7339.1 18 7943.4 17
3 3 0 0 31
3 4 2.6294e+05 0 31 184.79 30 887.44 29 1730.8 28 2760.5 27 4042.2 26 5669.4 25 7769.9 24 10495 23 13960 22 18162 21 23007 20 28457 19 34579 18 41523 17
4 1 2.3022e+06 0 10 24257 9 1.356e+05 8 2.6091e+05 7 4.0429e+05 6 5.7234e+05 5 7.7767e+05 4 1.0524e+06 3 1.5462e+06 2
4 2 9741.9 0 10 196.27 9 844.33 8 1498.7 7 2160.2 6 2829.7 5 3509.3 4 4204.6 3 4936.9 2 5809.4 1 7381.5 0
4 3 23603 0 10 512.85 11 1159.3 12 1808 13 2459.5 14 3114.2 15 3772.4 16 4434.6 17 5101.1 18 5772.6 19 6449.8 20 7133.9 21 7827.5 22 8536 23 9270.7 24 1
4 4 0 0 10
~
                                                                                                                                    1,1           All
```

TRAN path DC3 to DC2: up (down) goes 1→0 (0→1)

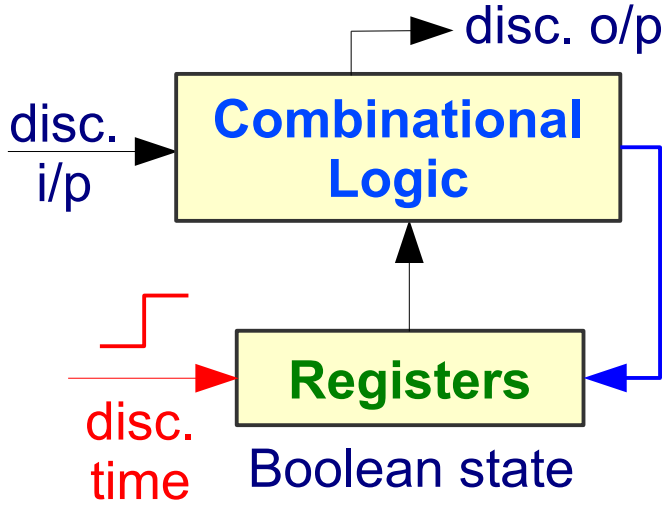
Total time 20.63ns, o/p starts @ 31, becomes 30 @ 213.6ps, ….

Booleanizing a Charge Pump (3/3)

3 Analog Transition Table → Boolean Model

```
(1) charge-pump-s
.i 21
.o 24

000000000000000000000 000100000000000000000000
010000000000000000000 000100001000000000000001
010000100000000000001 000100001000000000000010
<ipbv> <cstbv>  <opbv>     <nstbv>
010001000000000000011 000100001000000000000100
010001000000000000100 000010001000000000000101
010001000000000000101 000010001000000000000110
010001000000000000110 000010001000000000000111
010001000000000000111 000010001000000000001000
010001000000000001000 000010001000000000001001
010001000000000001001 000010001000000000001010
010001000000000001010 000010001000000000001011
010001000000000001011 000010001000000000001100
010001000000000001100 000010001000000000001101
```



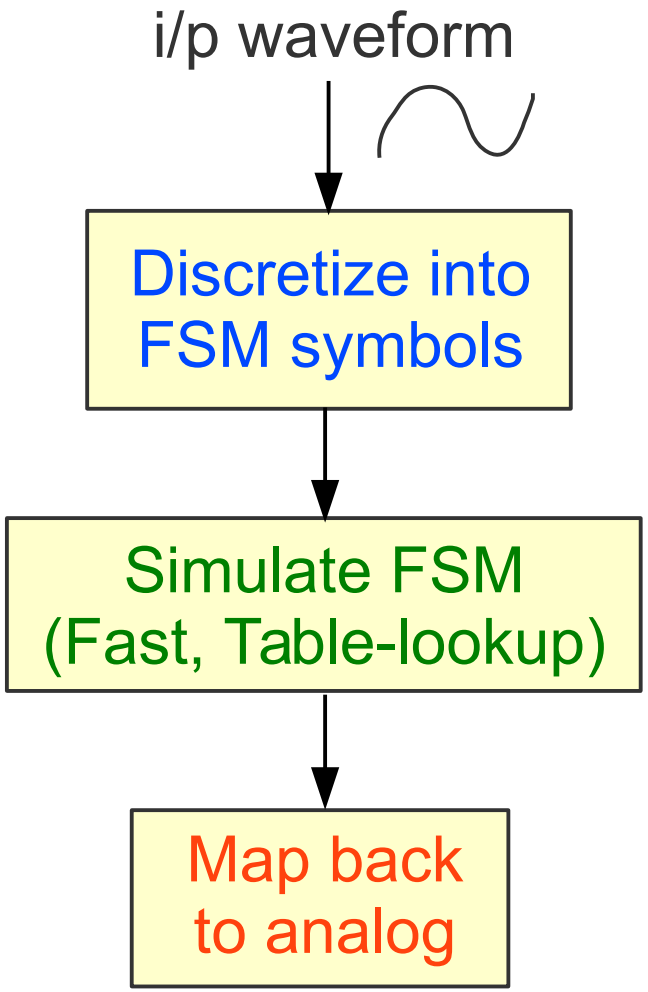
- i/p: 2-bit encoding
- o/p: 5-bit encoding
  - 32 levels in [0, Vdd]
- FSM state: 19 bits
  - Lots of don't cares (75%)

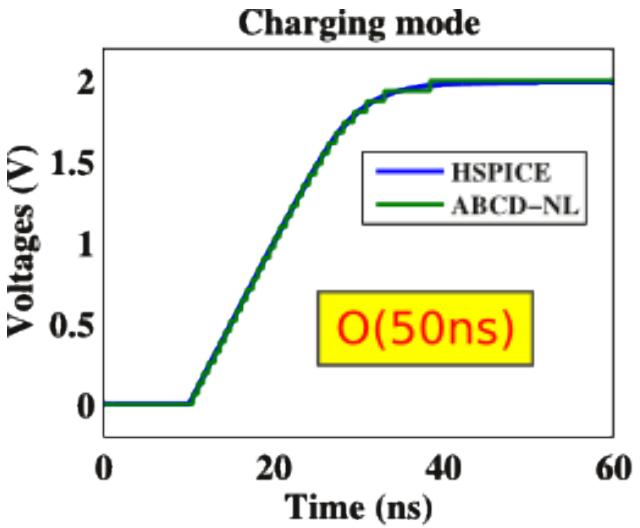
ABCD-NL: The 3-Step Algorithm

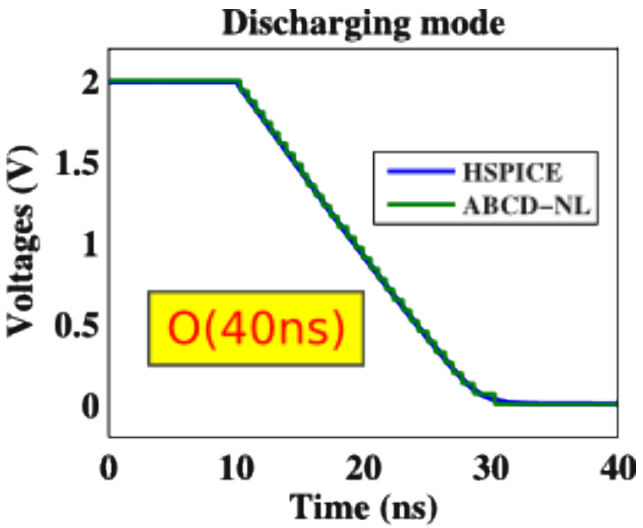
1. Do SPICE simulations
2. Analyze SPICE waveforms

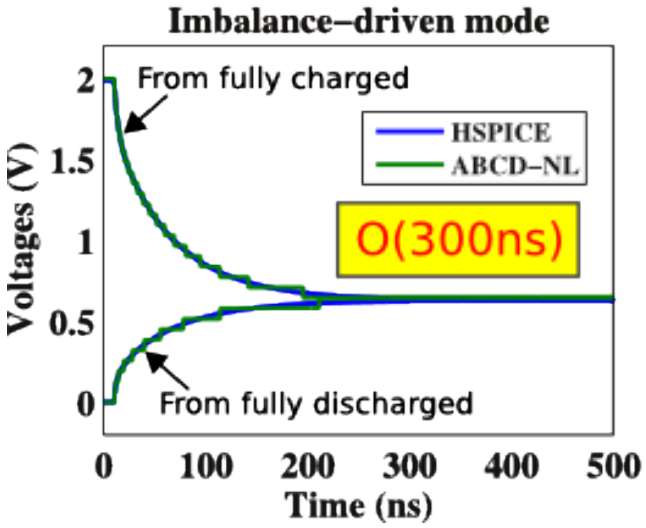
   Build “Analog Transition Table”
3. Analog Transition Table → Boolean Model

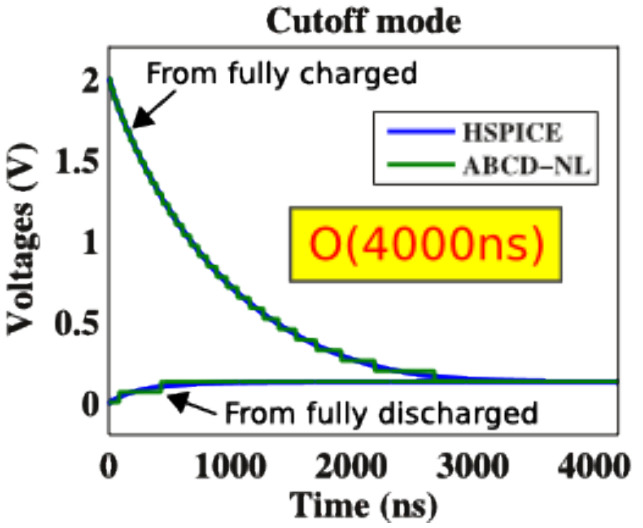
Simulating the Boolean Model



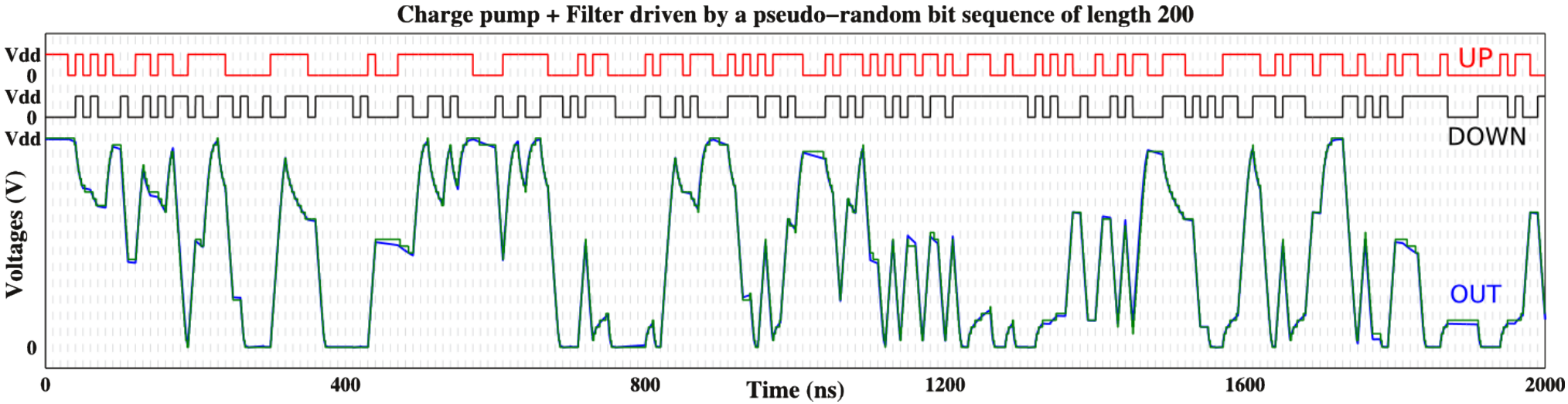








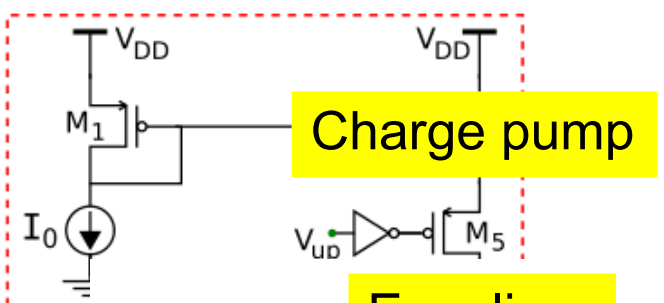
Charge Pump: Long PRBS



Non-linear analog dynamics accurately captured over long time-frame

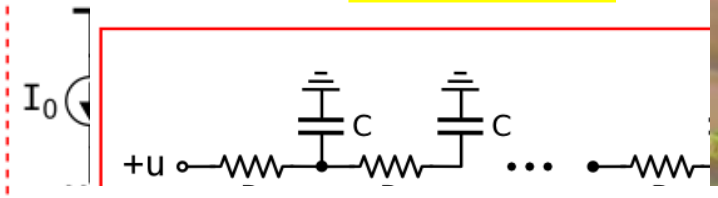
~10x Speedup, even in Python

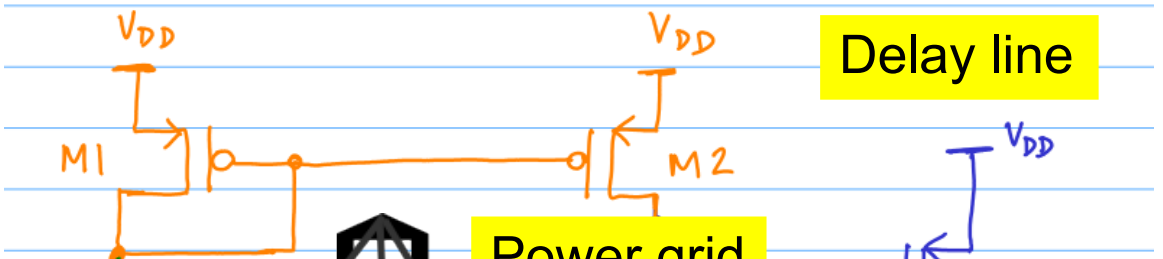
Circuits Successfully Booleanized



Charge pump

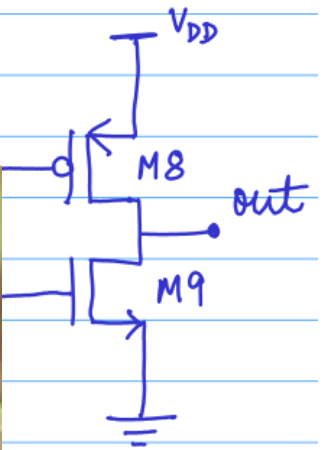
Equalizer

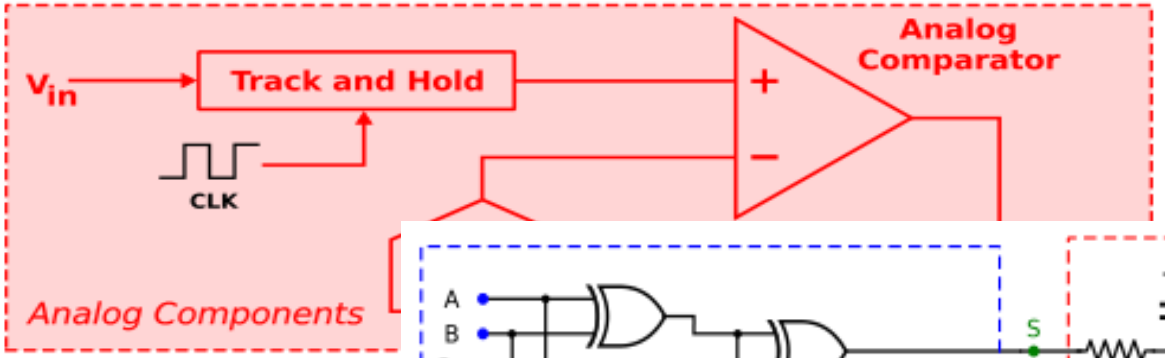




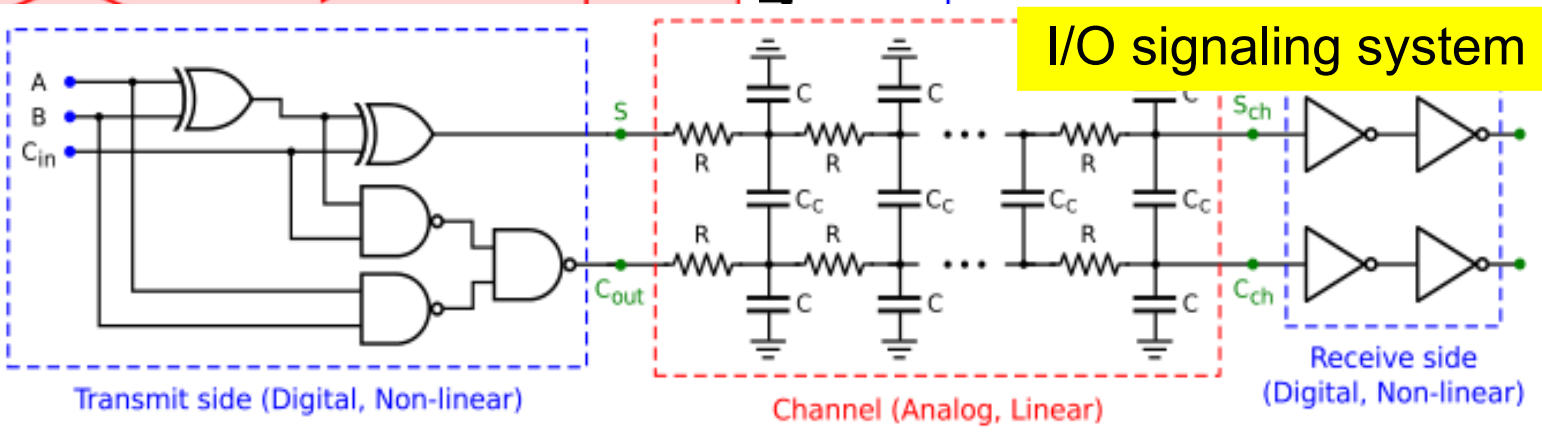
Delay line

Power grid



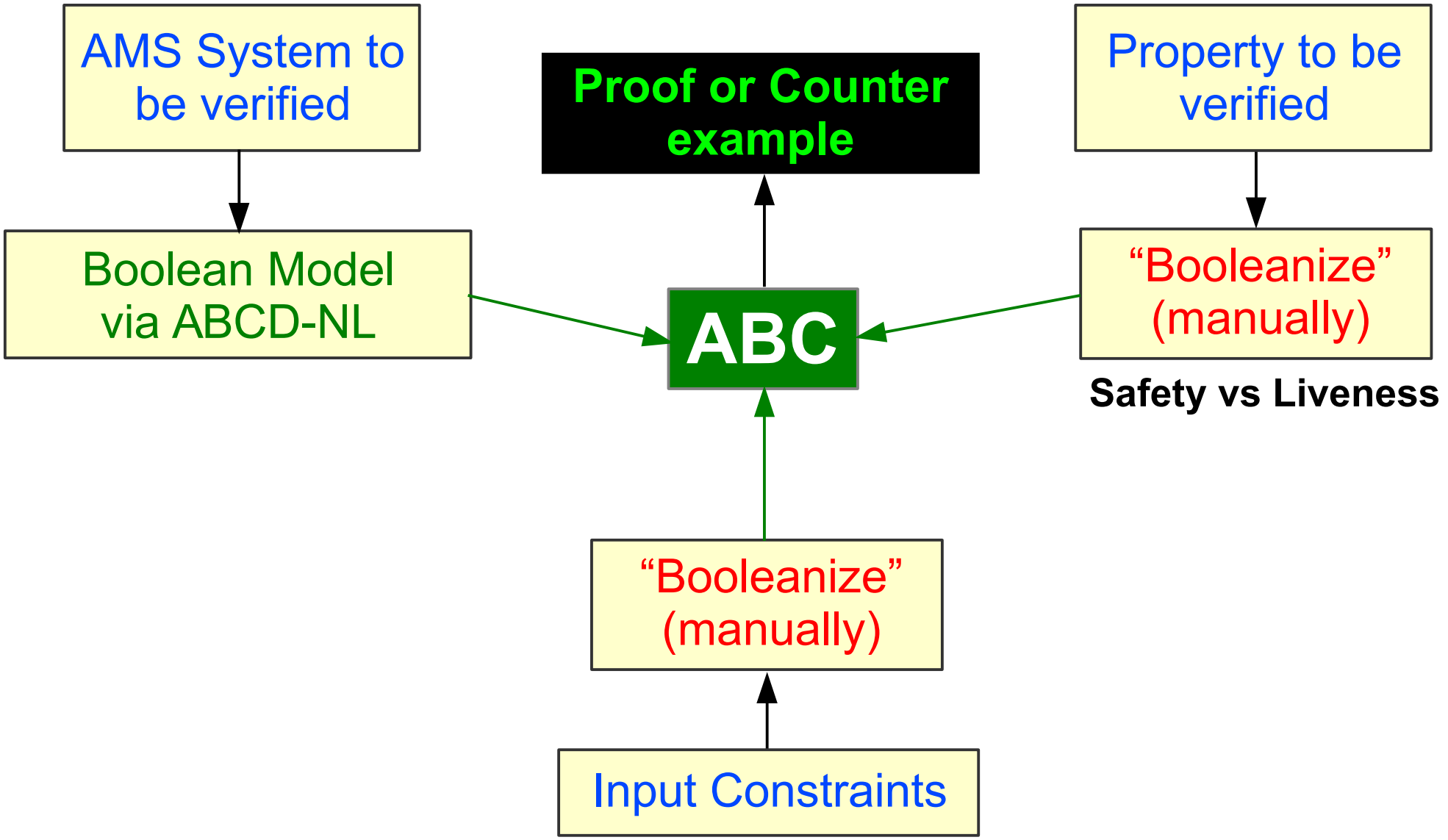


SAR-ADC

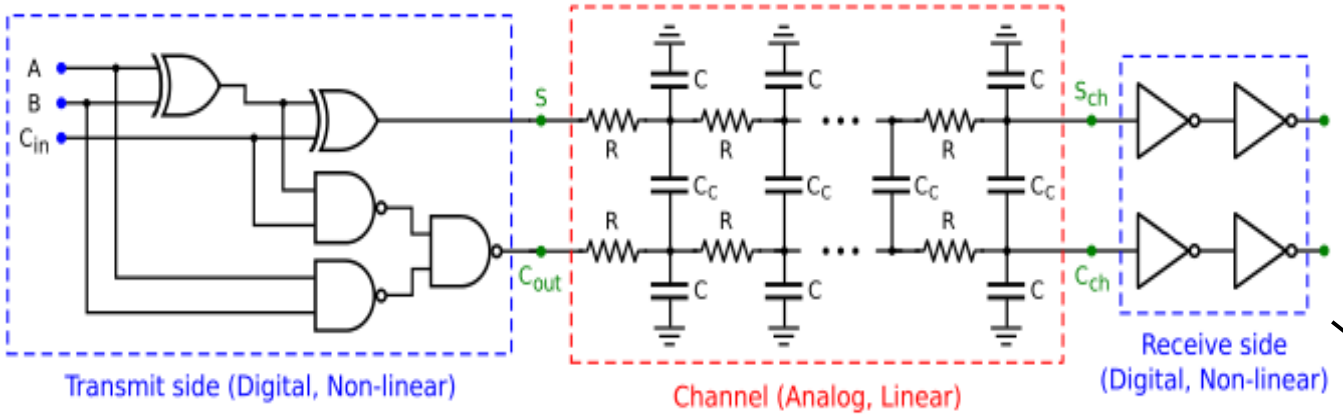


I/O signaling system

AMS Verification: The Flow



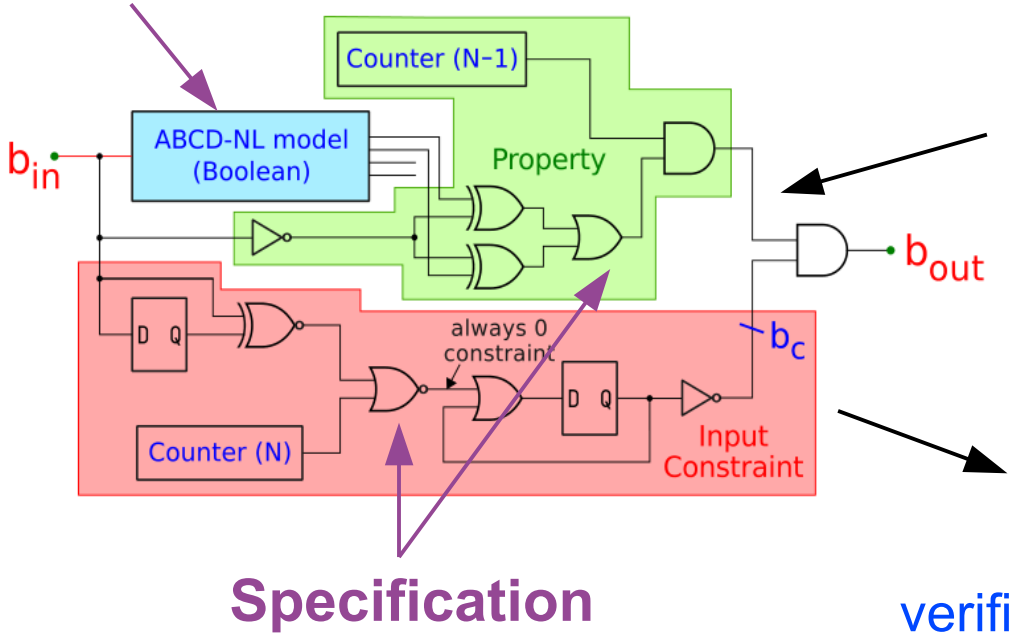
Example: Verifying a Signaling System



What is the max bitrate that can be sustained?

Ans: 2.56Gbps

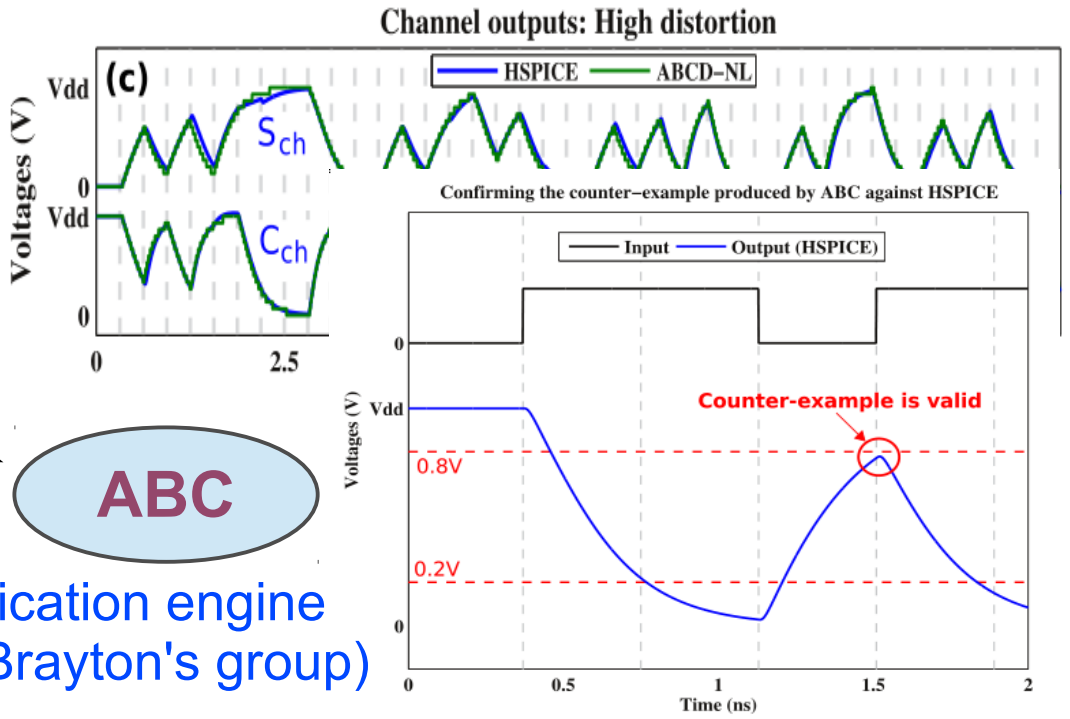
ABCD model



Specification

ABC

verification engine
(Bob Brayton's group)



Summary

- AMS modelling, verification a challenge: 20% bugs
- Our approach: Booleanize AMS Components
  - ABCD-L: for linear AMS systems
  - ABCD-NL: for non-linear systems
- Applied to A/D and D/A converters, delay lines, charge pumps, equalizers, filters, on-chip power grids, etc.
- Accurate and Scalable
- Applications
  - High-speed simulation
  - Formal verification (in conjunction with ABC)

Questions