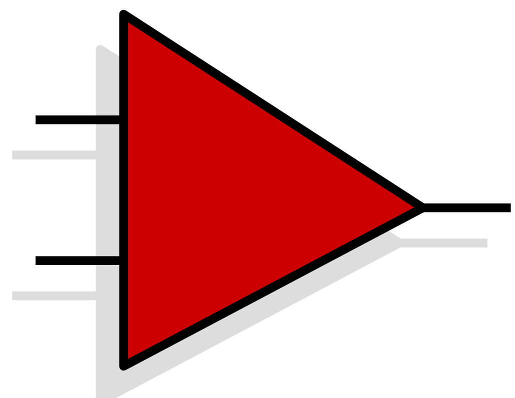
Behavioral Receiver Modeling

Donald Telian

Cadence, PCB Systems Division



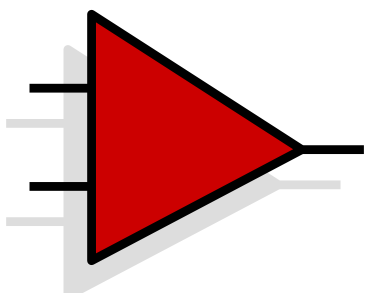
IBIS Summit: January 31, 2000

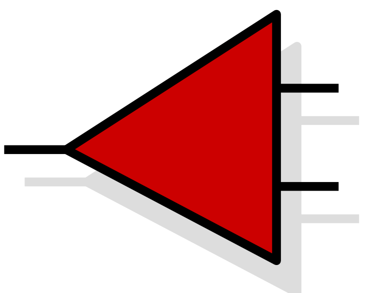
rev. 1.0

Cadence Design Systems, Inc.

AGENDA: Behavioral Receiver Modeling

- **Who?** The Digital Receiver Steps Forward
- **Why?** Reasons to Use Behavioral Receivers
- **What?** Comparing Various Receiver Modeling Methods
- **How?** The Macro-Structure of a Behavioral Receiver
- **Where?** That is the BIG Question

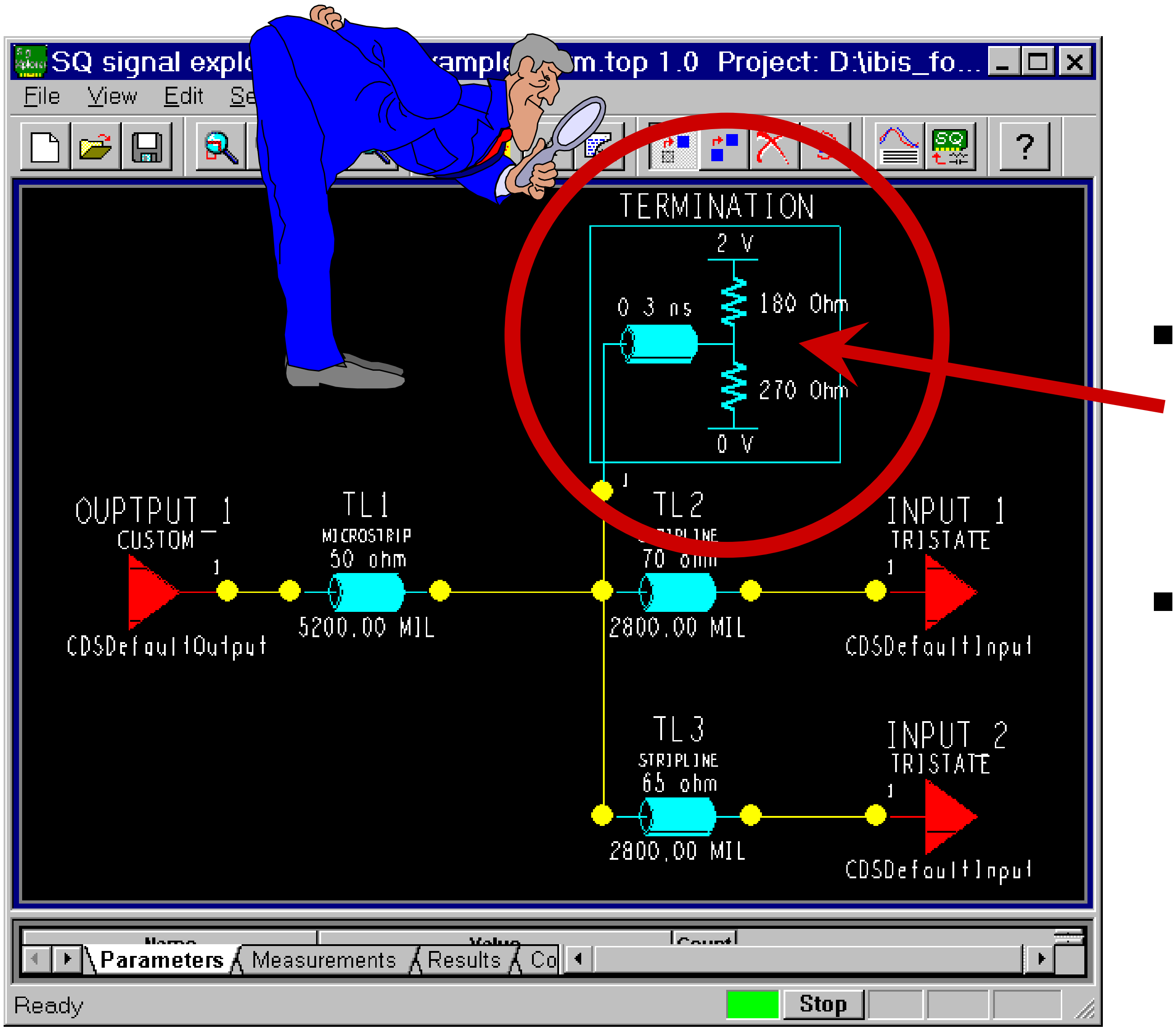




AGENDA: Behavioral Receiver Modeling

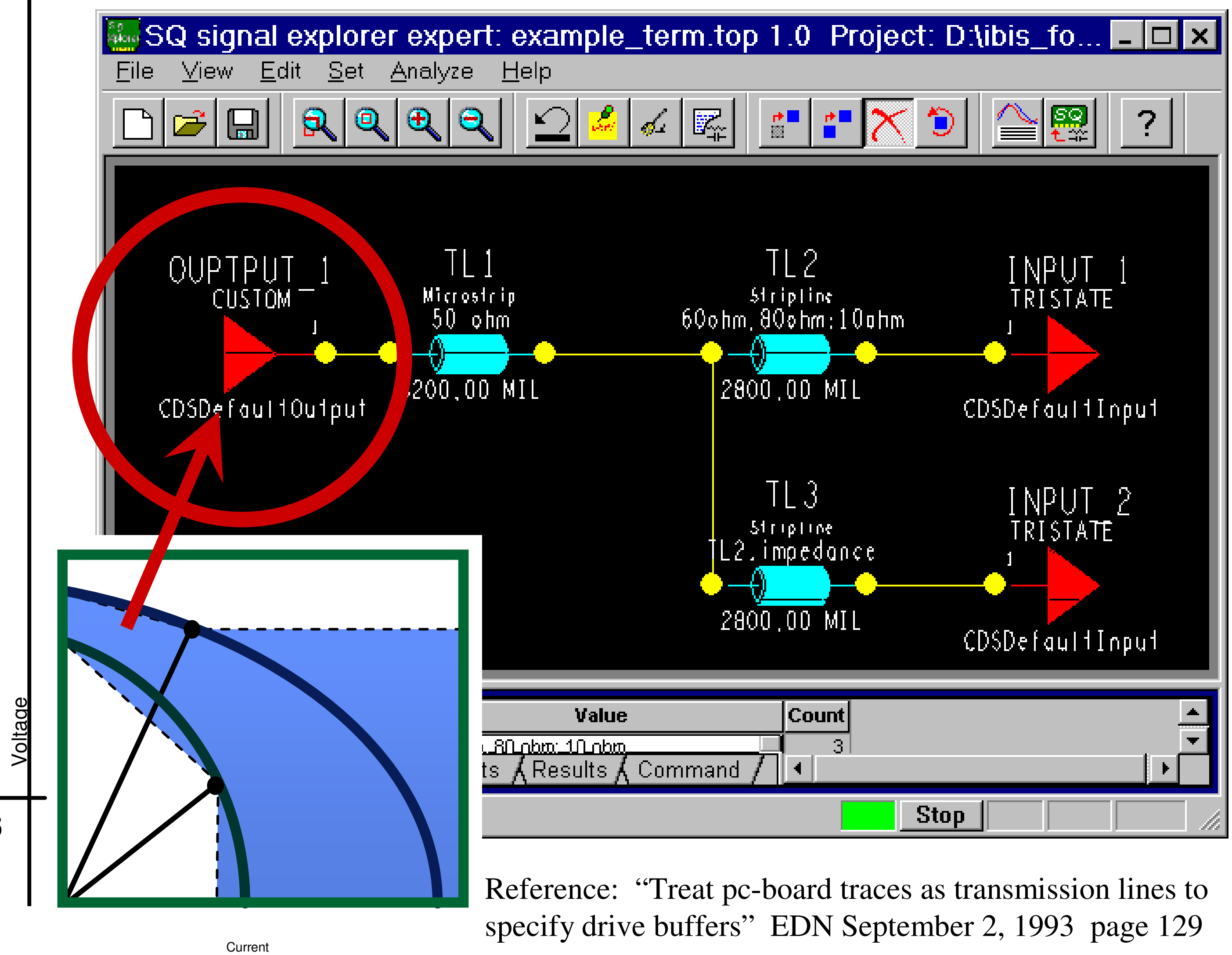
- **Who?** **The Digital Receiver Steps Forward**
  - termination modeling
  - driver sizing & scaling
  - interconnect optimization
  - who's left ???
- **Why?** Reasons to Use Behavioral Receivers
- **What?** Comparing Various Receiver Modeling Methods
- **How?** The Macro-Structure of a Behavioral Receiver
- **Where?** That is the BIG Question

The Study & Optimization of Digital Interfacing



- Mid - ‘80s = Terminations !
- Most of the rest was “off-the-shelf” to the digital engineer

Next Came Driver Scaling & Tuning

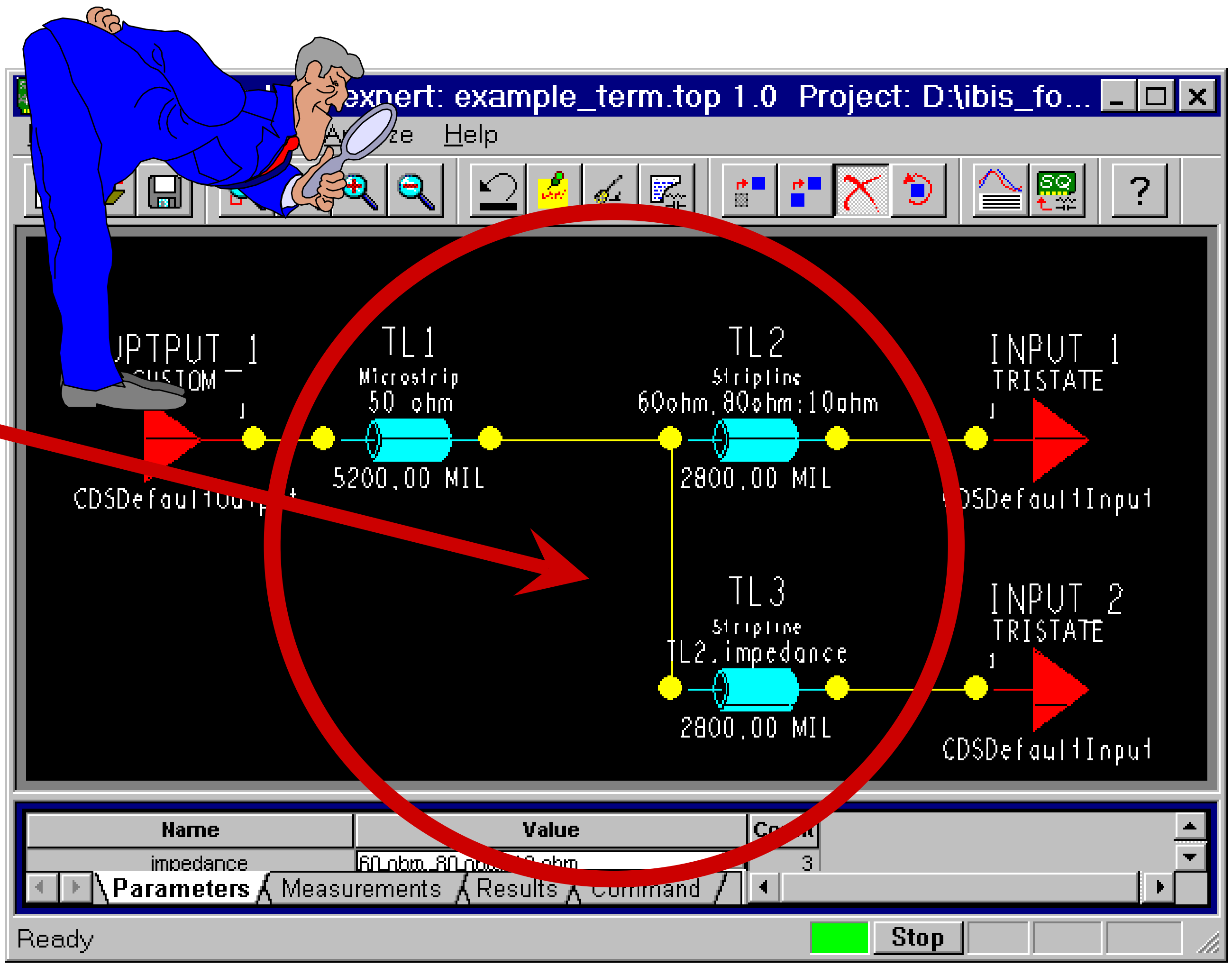


Reference: "Treat pc-board traces as transmission lines to specify drive buffers" EDN September 2, 1993 page 129

- Custom IC & ASIC Design brought the ability to sweep and tune Driver Characteristics
- Set the stage for "IBIS" representation in early '90s

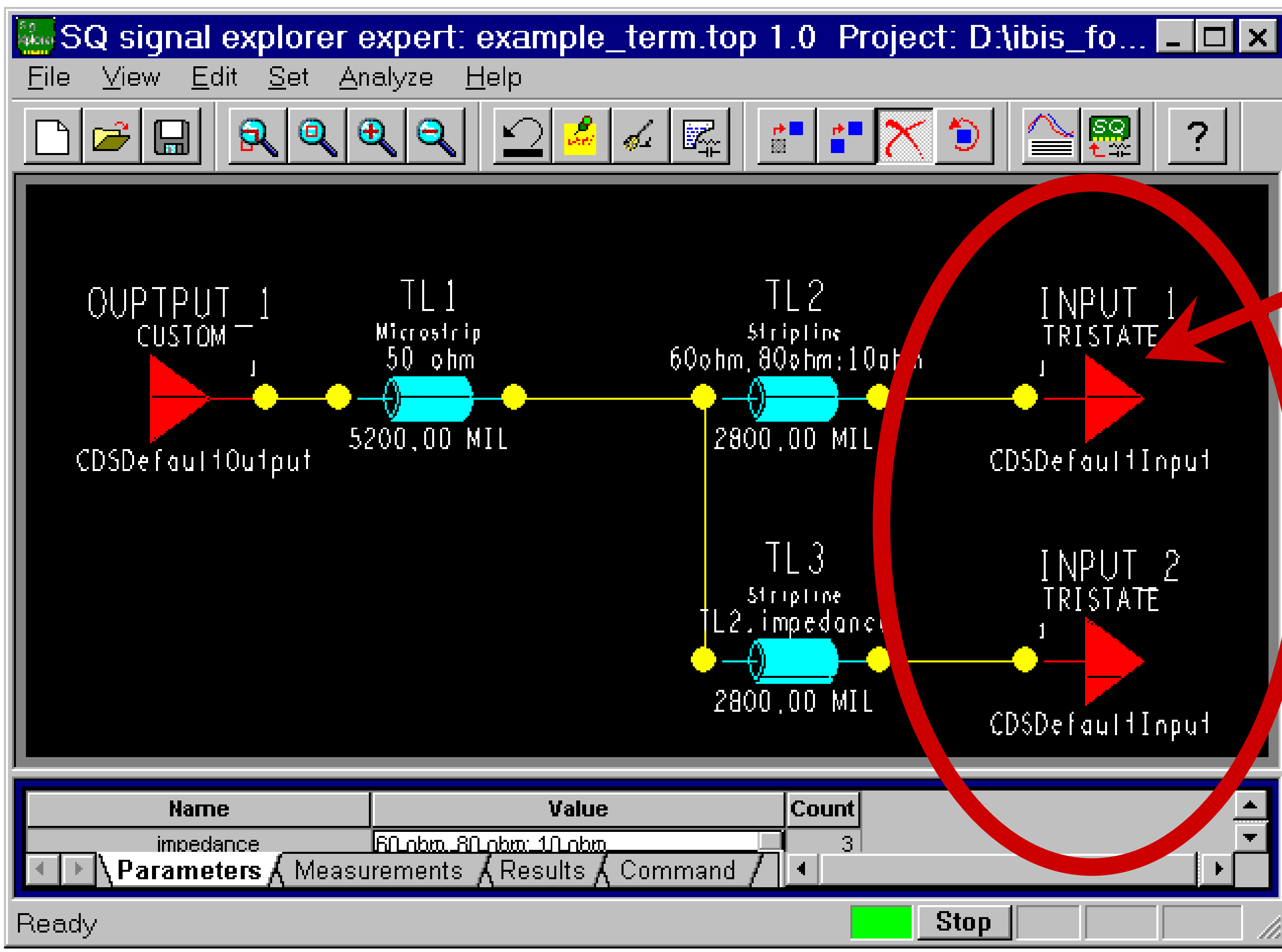
IBIS & Tools Allowed Topology Optimization

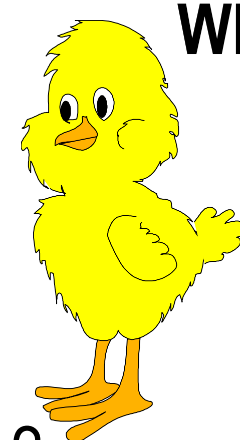
- Given well-tuned driver technology, constraining “Topology” factors brought more speed advances in the mid ‘90s



- Buffers acquired new dynamic behaviors

But What About the Lowly Receiver?





What?

Me?

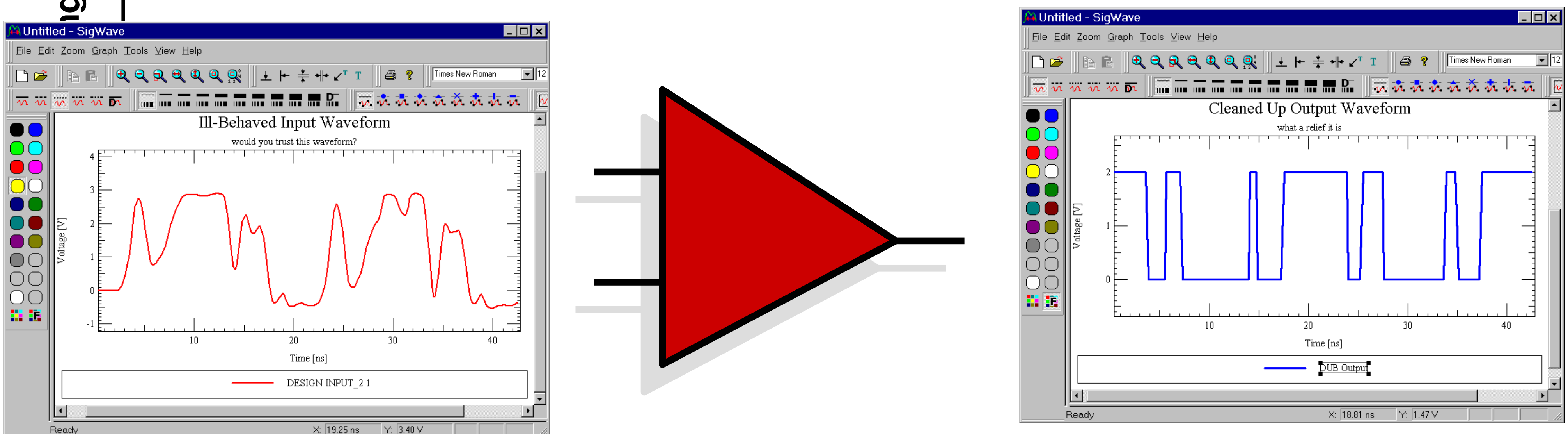
…a b-b-BIRD?

I, uh, I guess I hoped no one would notice!

AGENDA: Behavioral Receiver Modeling

- **Who?** The Digital Receiver Steps Forward
- **Why?** **Reasons to Use Behavioral Receivers**
  - Simplification
  - Performance
  - Design Optimization
- **What?** Comparing Various Receiver Modeling Methods
- **How?** The Macro-Structure of a Behavioral Receiver
- **Where?** That is the BIG Question

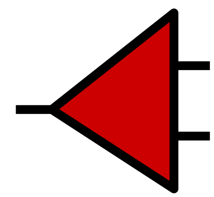
A Receiver is Basically an A - D Converter



- Consequently, it **simplifies** the question of how the (not-so-digital) digital signal is responded to by the device.
  - It's a "digital" problem once again!

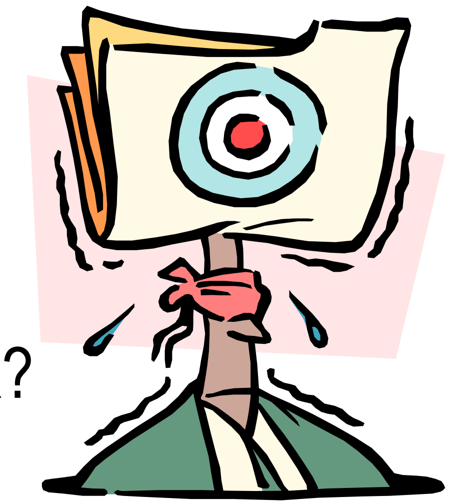
Receiver Modeling Allows Greater *Performance*

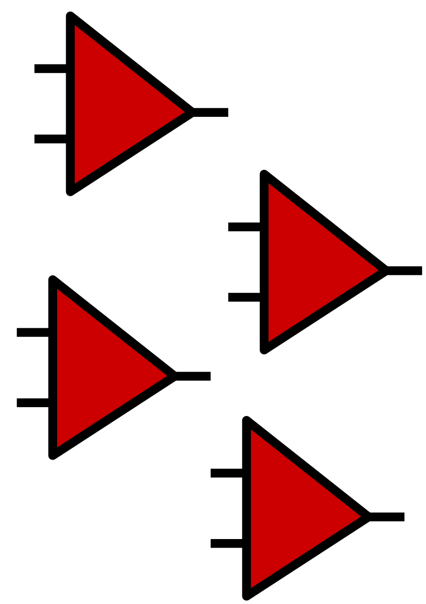
- Puts the whole digital transmission into one succinct transaction
- Eliminates awkward signaling hand-off at receiver node
  - double counting
  - extrapolations
- Allows the high-speed problem to be studied at one time



Receiver Modeling Allows *Optimization*

- What switching function do I need?
- What type of pulses will need to be rejected?
- What sort of propagation characteristics will work?
- What voltages should this all operate at?





In short, the Receiver can be "tuned" just like we're used to doing with the driver and the interconnect !

Behavioral Modeling

AGENDA: Behavioral Receiver Modeling

- **Who?** The Digital Receiver Steps Forward
- **Why?** Reasons to Use Behavioral Receivers
- **What?** **Comparing Various Receiver Modeling Methods**
  - Methods Available
  - Methods Contrasted
  - Cadence's Solution
- **How?** The Macro-Structure of a Behavioral Receiver
- **Where?** That is the BIG Question

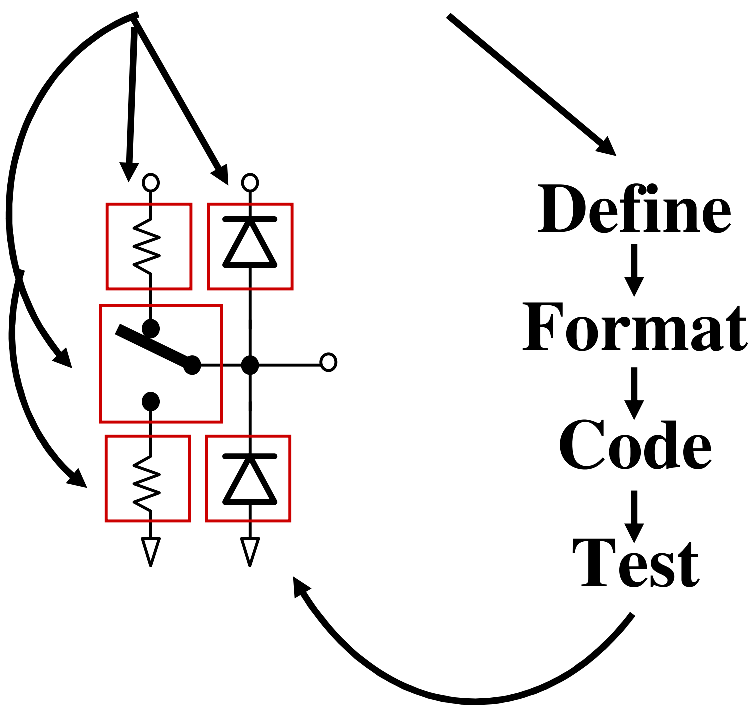
Methods to Model a Complete Receiver

❶ Silicon-level SPICE Description
- too slow, or is it ??
- too proprietary ?

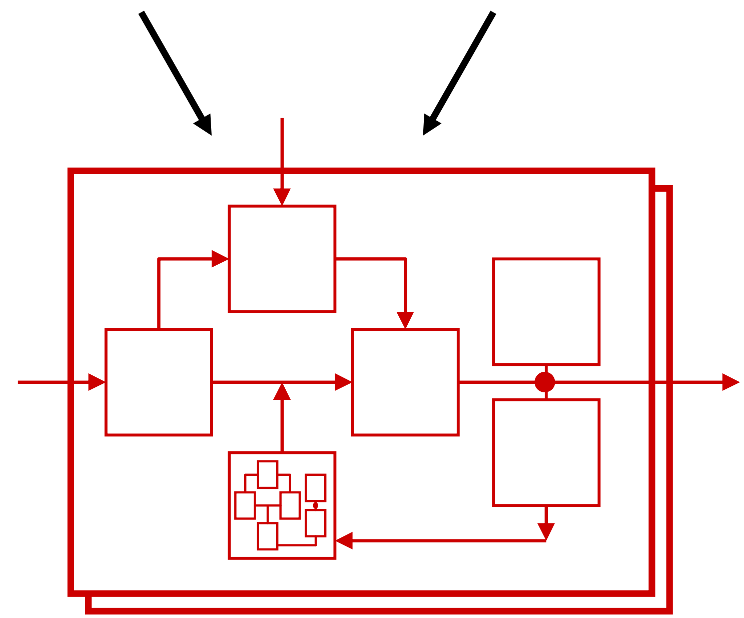
❷ Pre-Configured Behavioral Description
- model has defined structure / construction
- imitates one thing only, not a design tool
- too limiting (leads to IBIS' current roadblocks)

❸ Nodal Behavioral Description
- allows modeling a receiver's *characteristics*
- easily adapts to arbitrary behaviors
- this is what Cadence implemented

Pre-Configured vs Nodal Behavioral Models

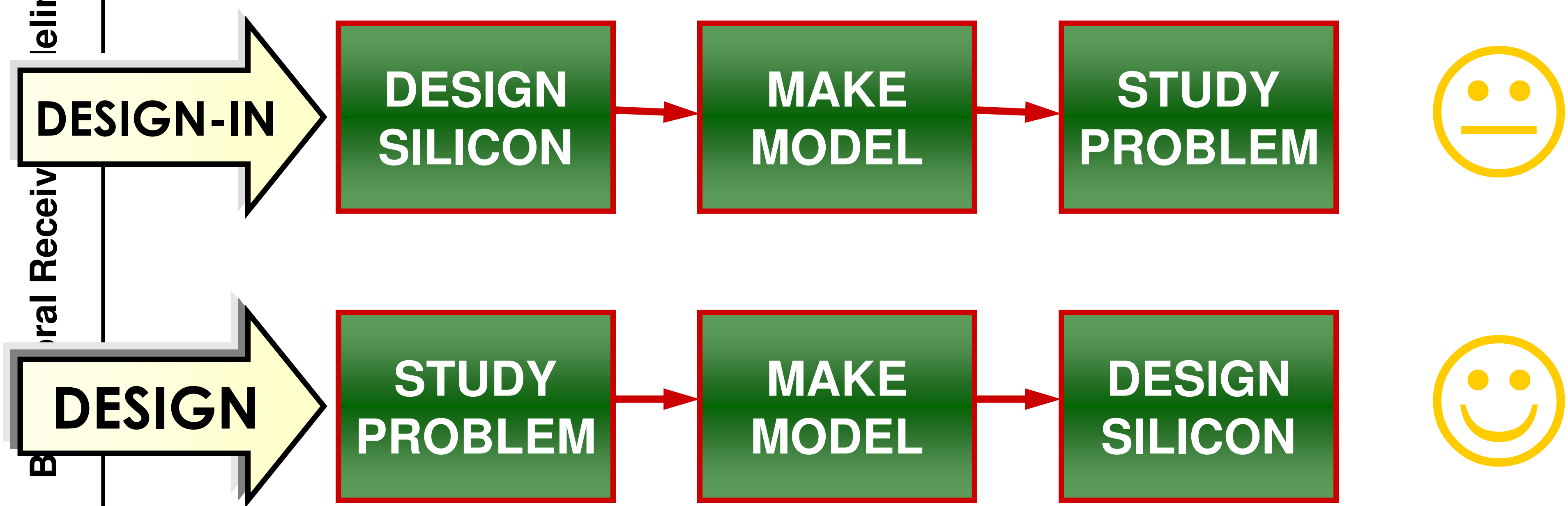


PRE-CONFIGURED



NODAL

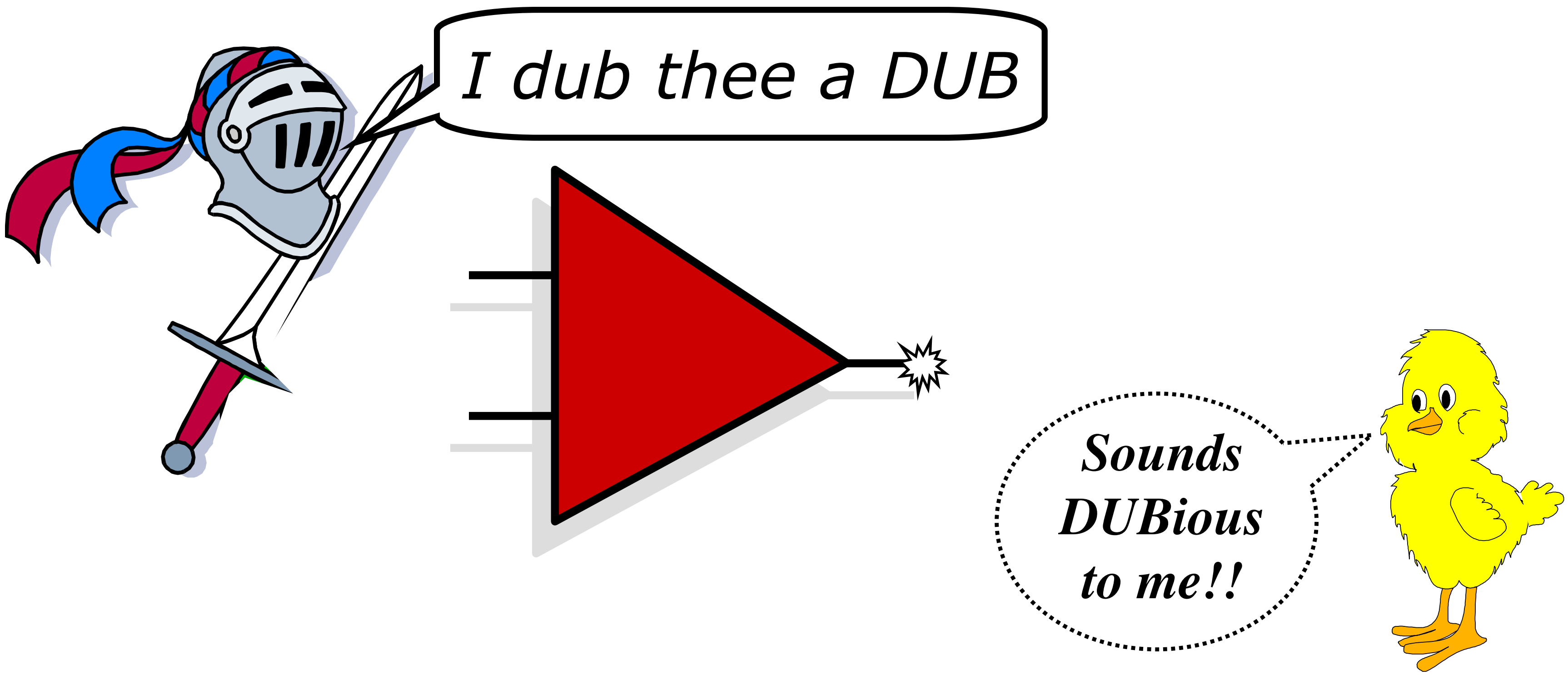
Nodal Better for “Design” not Just “Design In”



- More and more “Driver” design problems are solved this way
  - Sure is better to work this way, when you can!

1998: The Digital Universal Behavioral Receiver

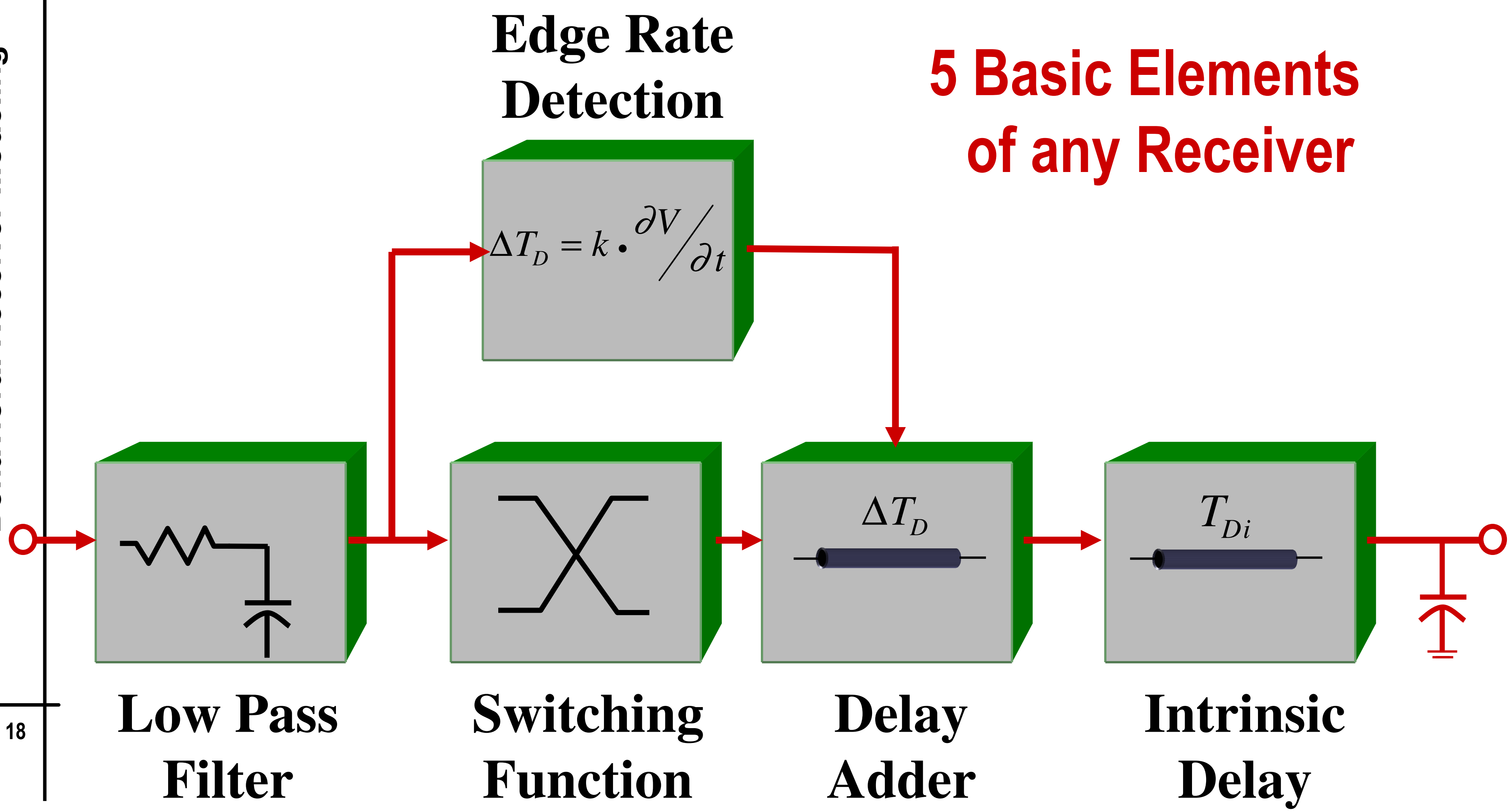
Working with Intel, on a certain unnamed project, in 1998 Cadence used Nodal Behavioral Modeling to develop the first Digital Universal Behavioral (DUB) Receiver Model



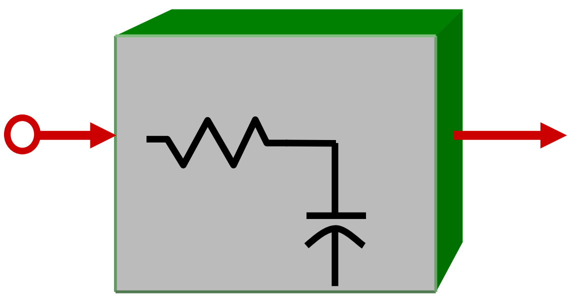
AGENDA: Behavioral Receiver Modeling

- **Who?** The Digital Receiver Steps Forward
- **Why?** Reasons to Use Behavioral Receivers
- **What?** Comparing Various Receiver Modeling Methods
- **How?** **The Macro-Structure of a Behavioral Receiver**
  - Block Diagram
  - Block - by - Block Explanation
  - Elements Required
  - How Fast? How Accurate?
- **Where?** That is the BIG Question

DUB Block Diagram



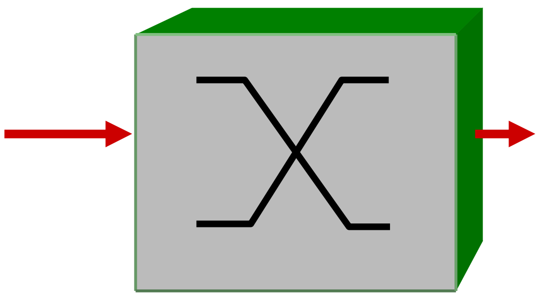
DUB Block #1: Low Pass Filter



Low Pass Filter

- Every Receiver Rejects Certain Pulse Spikes
- Simple to Characterize and Implement

DUB Block #2: Switching Function

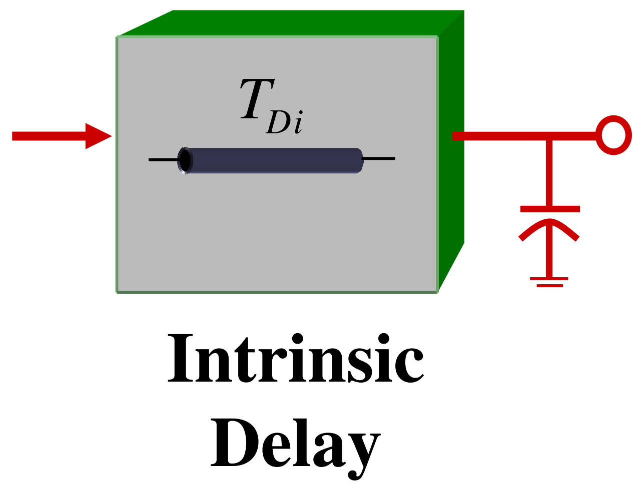


Switching Function

- Receiver's Basic Transfer Function
- Even Simpler for non-Differential Receiver

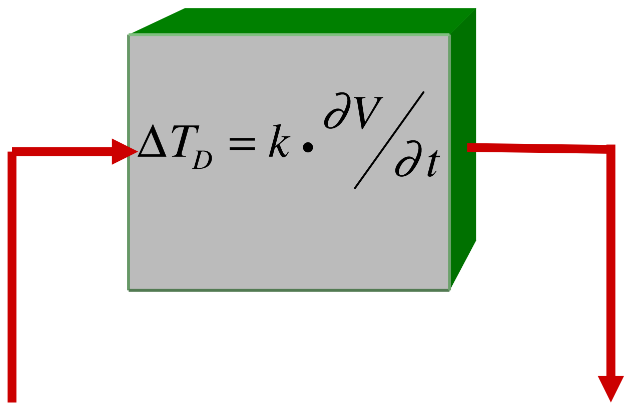
DUB Block #5: Intrinsic Delay

- Every Receiver has an Intrinsic (minimum) Delay
- Easily Determined by Measuring Against the Fastest Edge Expected



DUB Block #3: Edge Rate Detection

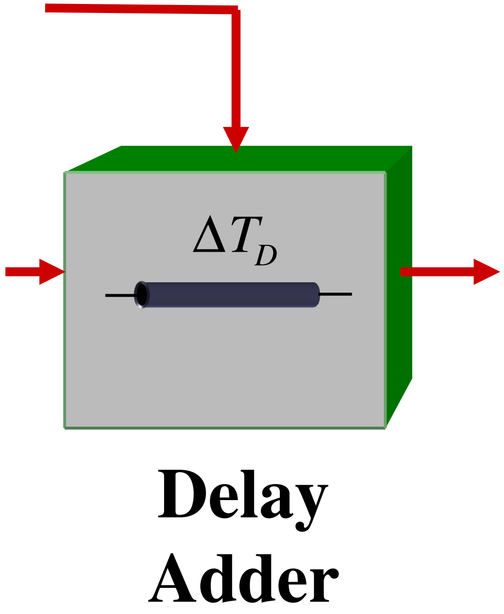
Edge Rate Detection



- Receiver Delay Changes with Input Edge Rate
- Mathematically Determine Derivative of Input Signal
- Use Real-time Derivative to Adapt Propagation Delay

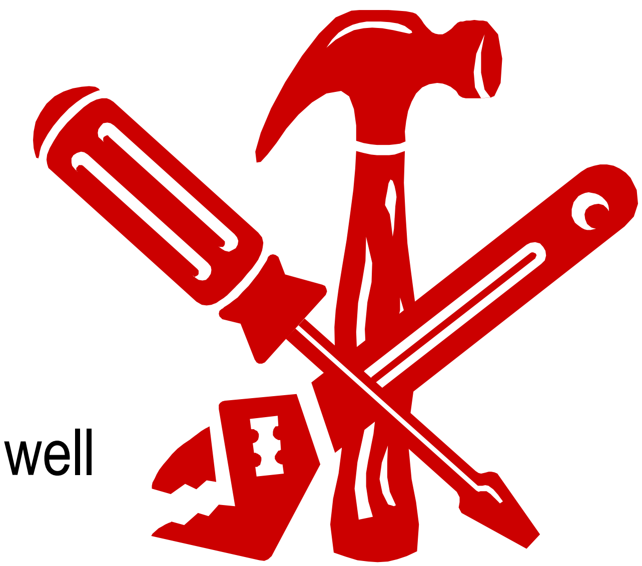
DUB Block #4: Delay Adder

- Circuit to Adjust Propagation Delay Based on Input Slope
- Real-time Adaptive Delay Line Construction



What is Needed for Nodal Behavioral DUB?

- Nodal Language
- G (VCCS) and E (VCVS) sources
  - pwl table driven
  - equation driven, with derivatives
  - event driven and time-controlled as well
- Real-time Adaptive Delay Line
- Subcircuit Nesting and Random Node Printing



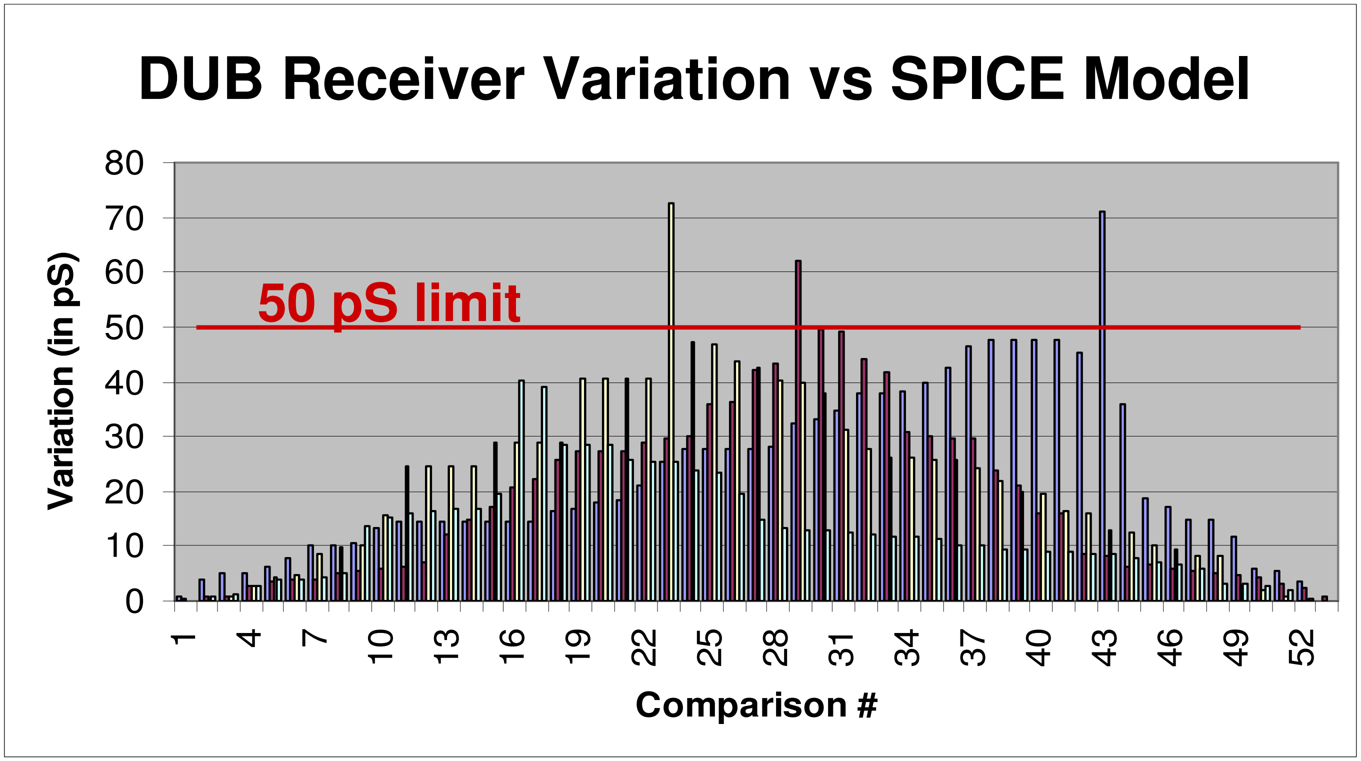
These are all available in SPECCTRAQuest's MacroModeling language

How Fast is the DUB Method?

Model Type	*Sim Time*	*Notes*
SPICE	1	Normalized
DUB	1.4	Standard Applications
Intel DUB	2.1	Complex Application

- Values Derived from Receiver Benchmark, not Application
- “Intel DUB” needed additional elements to work right
- For “small” SPICE circuits, “Behavioral” can be slower !!
  - that was the big surprise to this engineer
- However, in the Intel application, the pure receiver delta above was insignificant because the elaborate package model contributed to ~90% of the run-time hit

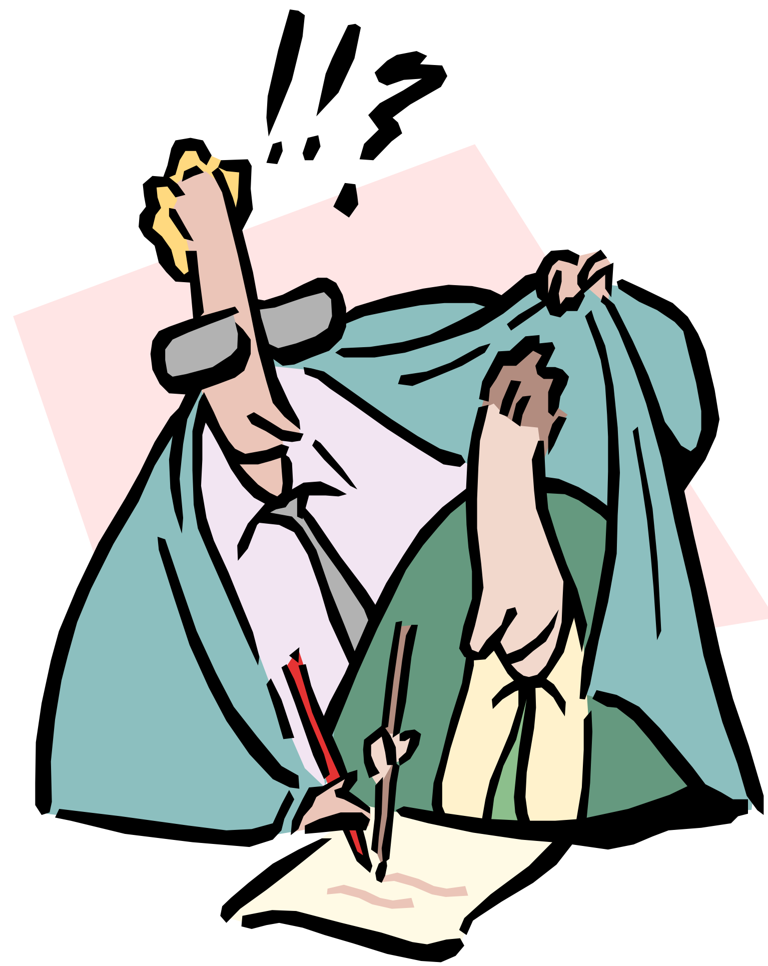
How Accurate? As Accurate as You Want !



- Picosecond variations for 242 **extremely difficult** test waveforms
- Average difference was 19 pS (std. dev. of 14 pS)
- *Could have matched even closer with more work, if desired*

And Here's the Rest of the Story...

- For the Original Intel DUB, other Elements Were Added to Accommodate Unique (Proprietary) Effects
  - this is what caused the extra 50% simulation slow down
- Cadence **HIGHLY** Recommends that if IBIS Implements Behavioral Receivers that a Nodal Syntax Language Be Used
  - any approach needs to be adaptable to unique situations



AGENDA: Behavioral Receiver Modeling

- **Who?** The Digital Receiver Steps Forward
- **Why?** Reasons to Use Behavioral Receivers
- **What?** Comparing Various Receiver Modeling Methods
- **How?** The Macro-Structure of a Behavioral Receiver
- **Where?** **That is the BIG Question**
  - why do answers always raise more questions?

The DUB is All Dressed Up...

...and Ready to Go!

The Big Question:

Will the Compelling Application Please Step Forward ????

IBIS Drivers had PCI and Pentium, but who will Rub-A-DUB?