Timing & Concurrency III

Delay Model foundations for simulation and VHDL simulation paridigm.

Overview – Timing & Concurrency

- Simulation Techniques
- The VHDL Simulation Cycle
- Waveform updating

Delay Models

- Rise/Fall Delay Model – allows the specification of different delays for when the output signal rises (0 to 1) and falls (1 to 0) due to electrical properties, and thus enables the designer to more accurately analyze the timing performance of a curcuit.

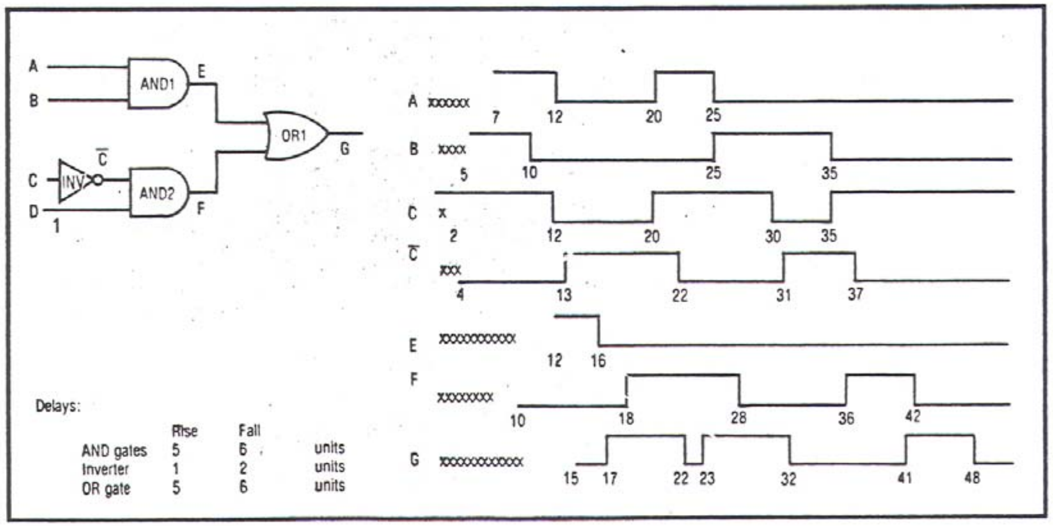
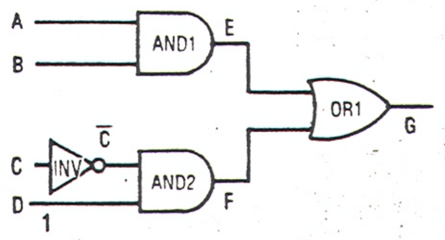
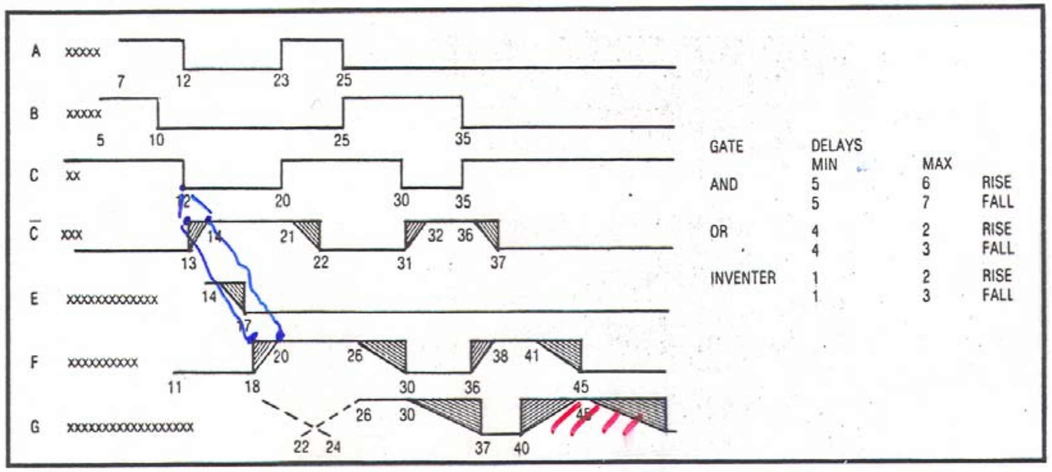


Figure 2. Rise/fall delay model.

Delay Models

- Ambiguity Delay Model – takes into account the minimum and maximum values for the rise and fall delays. This model deals with a range rather than a specific value for the delays. The range, know as the ambiguity region, specifies the limits within which the signal changes. This region will tend to accumulate or "stretch" as it propagates through the circuit. **Good for worst case analysis.**





GATE	DELAYS MIN	MAX	
AND	5	6	RISE
	5	7	FALL
OR	4	2	RISE
	4	3	FALL
INVENTER	1	2	RISE
	1	3	FALL

Figure 3. Ambiguity delay model.

Delays Models

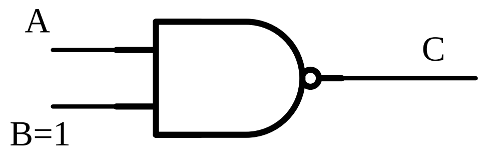
- Inertial Delay Model – In order to switch state, logic gates require that the input pulse cross a certain threshold and remain unchanged for a certain period of time (hold time). If the pulse is small, the gate will not change state. The minimum pulse width for an input pulse to cause a change in state for a gate or other device is called the *inertial delay* of the gate or device.

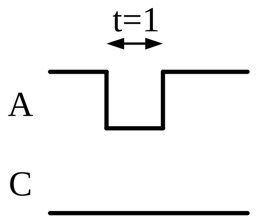
Inertial delay

- Consider the following gate.
    - The inertial delay of the gate is 2 time units.
        - WHAT IF THE PULSE IS 1 TIME UNIT?

Inertial delay

- Consider the following gate.
  - The inertial delay of the gate is 2 time units.
    - WHAT IF THE PULSE IS 1 TIME UNIT?
    - Output does not change
    - High Frequency signal rejection a property that digital gates have



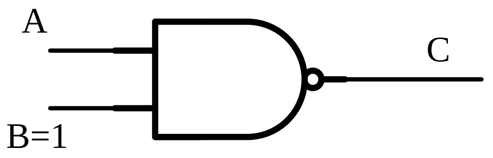


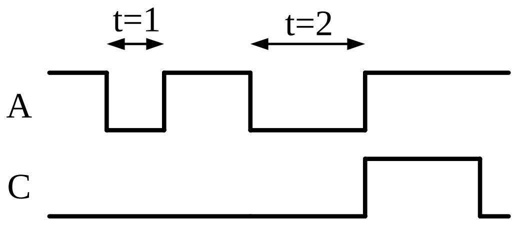
Inertial delay

- Consider the following gate.
  - The inertial delay of the gate is 2 time units.
    - WHAT IF THE PULSE IS 2 TIME UNITs?

Inertial delay

- Consider the following gate.
  - The inertial delay of the gate is 2 time units.
    - WHAT IF THE PULSE IS 2 TIME UNITs?
    - Gate responds with a 2 TIME UNIT delay from the input.



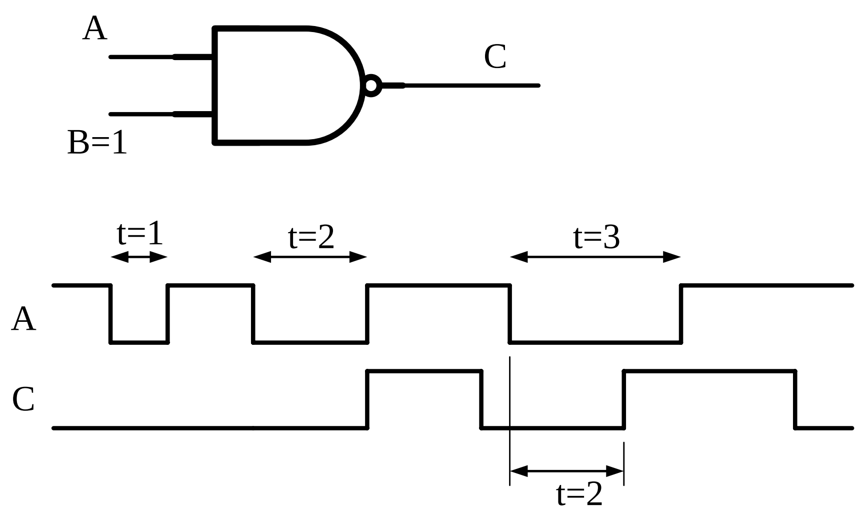


Inertial delay

- Consider the following gate.
  - The inertial delay of the gate is 2 time units.
    - WHAT IF THE PULSE IS 3 TIME UNITs?
    - Gate responds with a delay of 2 TIME UNITs.

Inertial delay

- Consider the following gate.
  - The inertial delay of the gate is 2 time units.
    - WHAT IF THE PULSE IS 3 TIME UNITs?
    - Gate responds with a 2 TIME UNIT delay from the input.

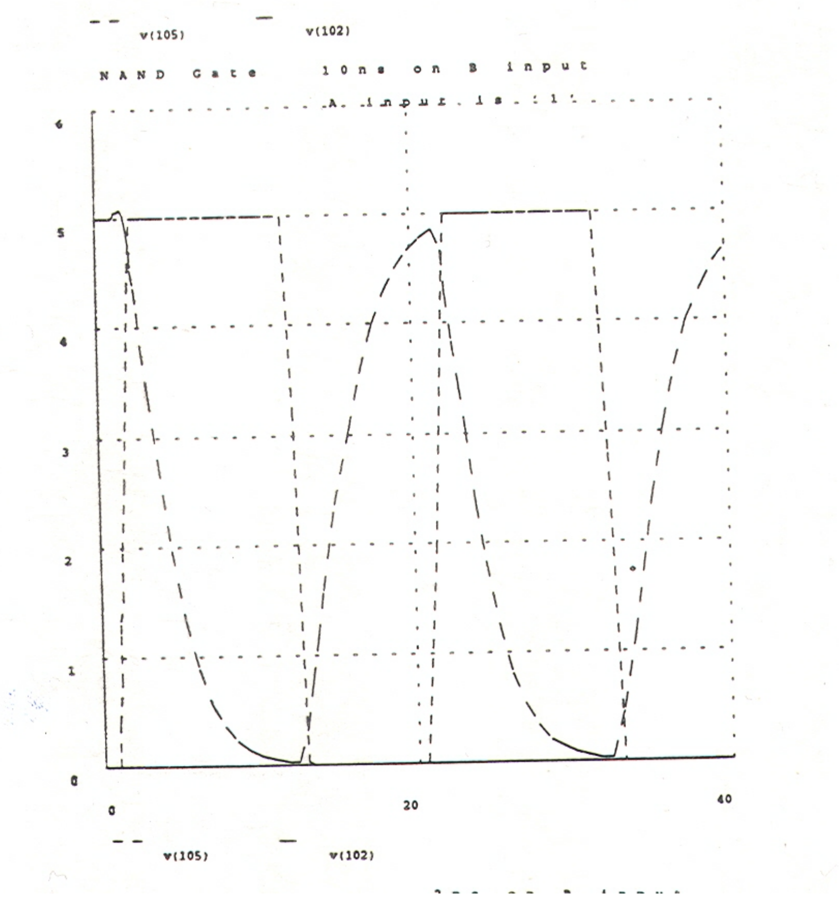


A final word on Inertial Delay

- *Inertial Delay* allows for high-frequency signal rejection – filtering out short spikes or pulses at the input that have a width shorter than the inertial delay of the gate.
- And we can see this in circuit simulation of real gates.

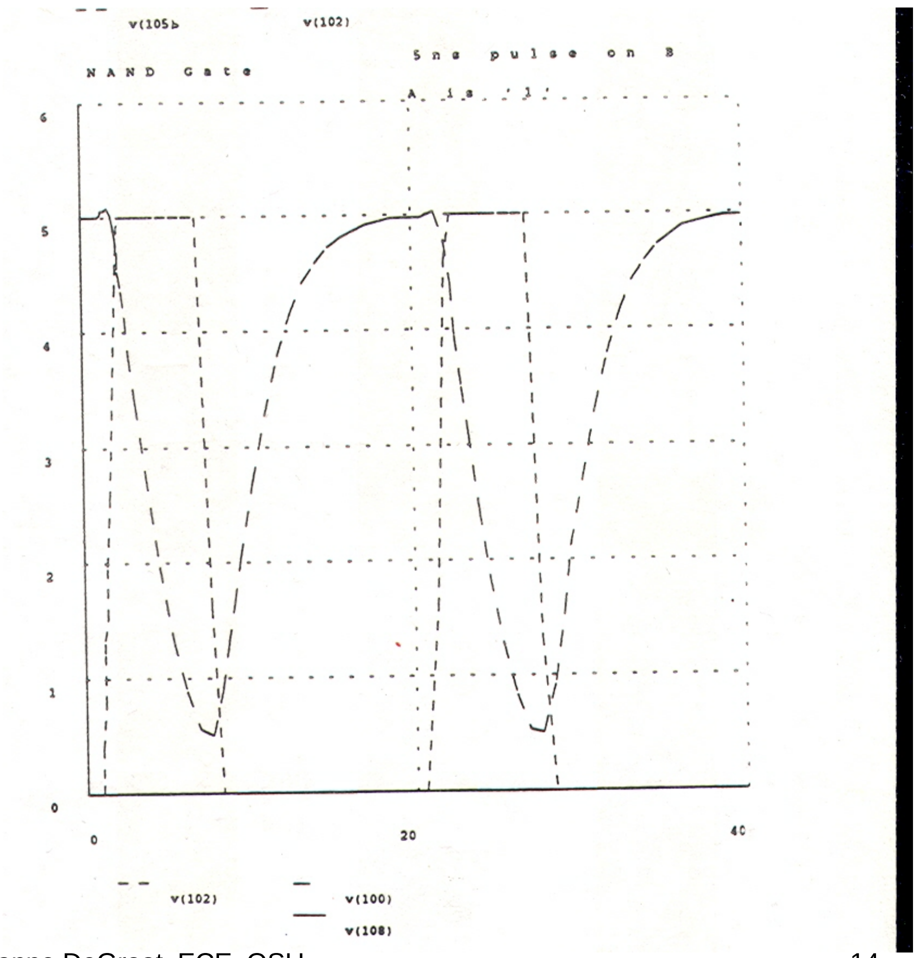
Spice simulation

- Spice circuit simulator
- NAND GATE
- 200 pF load
- B input held at 1
- Input pulse width of 10 ns.
- Input has a 1 ns rise and fall time
- Note output shape



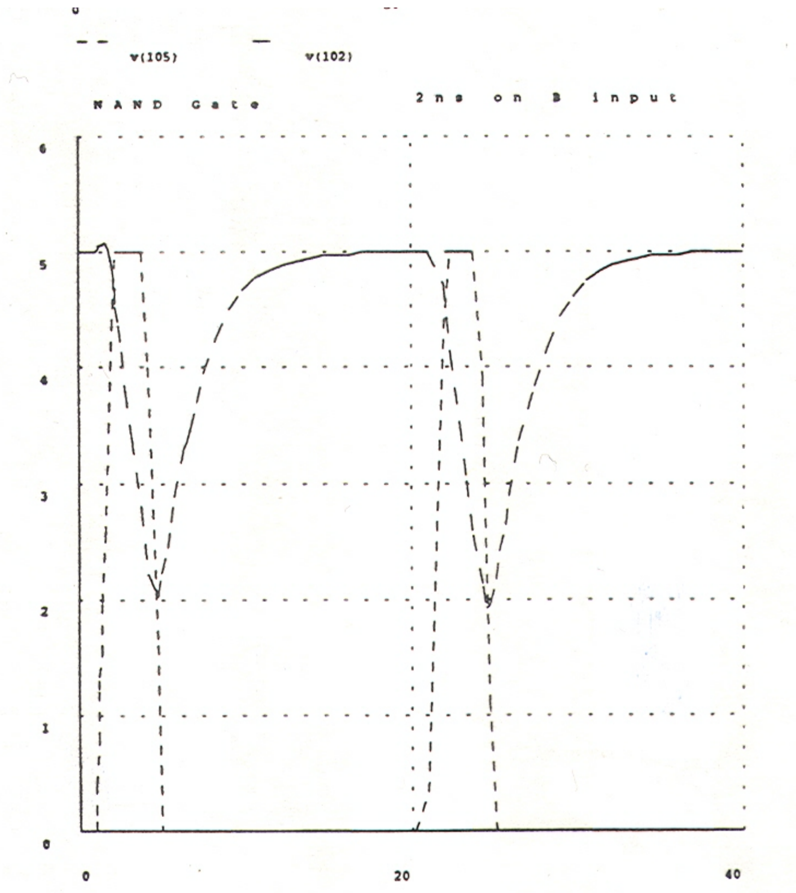
Spice Simulation 2

- Input pulse shortened to 5 ns
- Note that output never quite falls fully to 0.



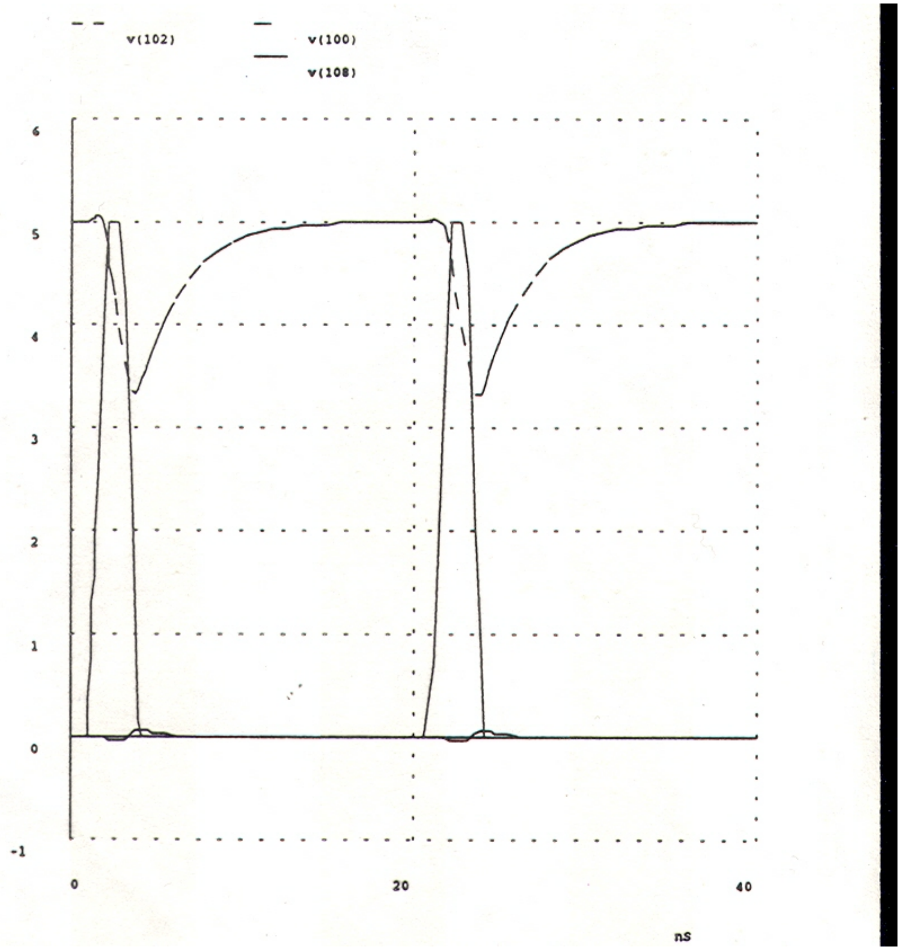
Spice Simulation 3

- Now shorten the input to a 2 ns pulse with a 1 ns rise time and a 1 ns fall time
- Note that output would barely (if even) reach the treshold need to trigger the next gate.



Spice Simulation 4

- Now shorten the pulse to 1 ns
- Change on output is not sufficient to even come close to causing a change on the subsequent gate as you only drop to ~3.5V on the output of this gate.



Delay Models

- Transport Delay Model – The transport delay for an element can be modeled as shown and is a delay simply added to the output of the element. If all delays were set to 1 this is a *unit delay simulator*. If the delay for all elements are set to 0, it is a *zero delay simulator*. It will not show an accurate measure of the circuits timing performance. <u>Glitches will get through that would not appear in a real circuit.</u>

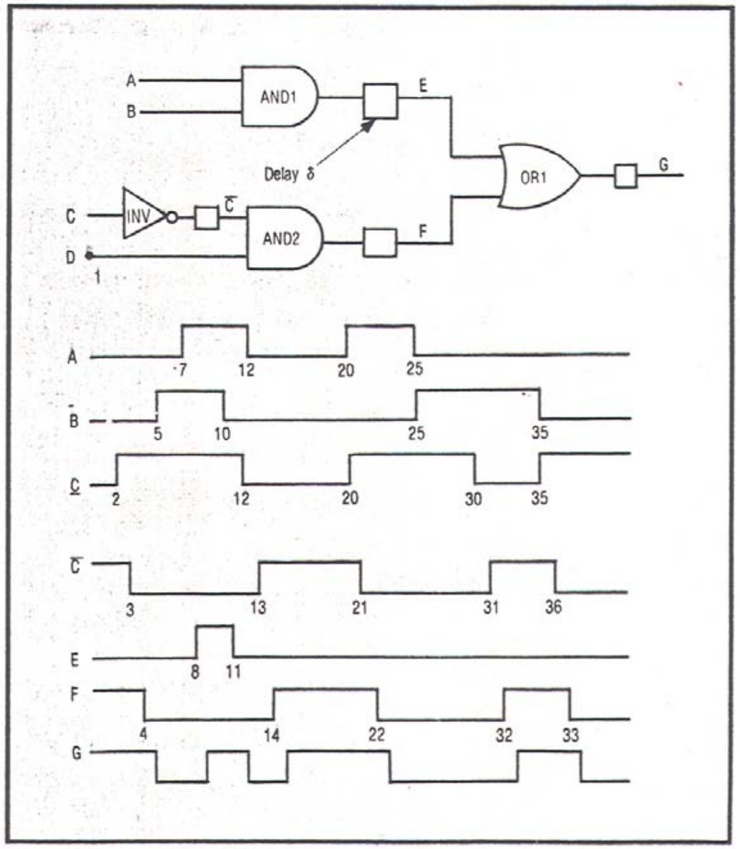


Figure 1. Transport delay model.

Inertial and transport delay in VHDL

- VHDL 1987
  - **Inertial Delay**
  - Example: X <= A and B   AFTER 5 NS;
  - Here 5 NS is both the gate delay and the inertial delay of the gate.
  - **Transport Delay**
  - Example: X<=**transport** A and B AFTER 5 NS;
  - Here 5 NS if the transport gate delay and there is no high-frequency pulse rejection.

Inertial and transport delay in VHDL

- VHDL 1993 and on
  - signal_assignment_statement:
  - [label:]target<=[delay_mechanism] waveform;
  - delay_mechanism::= **transport** |
  - [**reject** time_expression] **inertial**
  - target::= name | aggregate
  - waveform::= waveform_element
  - {,waveform_element}

Inertial and transport delay in VHDL

- VHDL 1993 and on example
  - The following 3 are equivalent to each other
  - X <= A and B after 4 ns; --inertial(same as 87)
  - X <= inertial A and B after 4 ns;
  - X <= reject 4 ns inertial A and B after 4 ns;
  - Assignments with pulse rejection limit < device delay
  - X <= reject 5 ns inertial A and B after 10 ns;
  - X <= reject 3 ns inertial A and B after 10 ns,
  - A nand B after 20ns;
  - Note: Reject time must be < or = smallet delay in after clause.
  - Transport Delay is the same at in the ’87 standard.
  - This does affect posting of transactions algorithm slightly but is not in the algorithm presented

The VHDL Simulation Cycle

- Initialization Phase
  - 1. The effective value of each declared signal is computed, and the current value of the signal is set to this effective value. The value is assumed to have been the value of the signal for an infinite length of time prior to the start of simulation.
  - 2. The value of each implicit signal of the form S'Stable(T) or S'Quiet(T) is set to TRUE.
  - 3. The value of each implicit GUARD is set to the result of evaluating the corresponding guard expression.
  - Each process in the model is executed until it suspends. At the beginning of simulation the current time is assumed to be 0 ns.

The VHDL Simulation Cycle

- Simulation Phase (repeats till quiescent)
  - A simulation cycle consists of the following steps:
  - 1. If no driver is active, then simulation time advances to the next time at which a driver becomes active or a process resumes. Simulation is complete when time advances to TIME'HIGH.
  - 2. Each active explicit signal in the model is updated. (Events may occur as a result.)
  - 3. Each implicit signal in the model is updated. (Events may occur on signals as a result.)
  - 4. For each process P, if P is currently sensitive to a signal S, and an event has occurred on S in this simulation cycle, then P resumes.
  - 5. Each process that has just resumed is executed until it suspends.
- **→Advance Time → Update Signal Values → Mark Sensitive Processes → Run Resumed Processes →**

Updating the Projected Output Waveform

- When a signal assignment statement is executed a transaction is generated
- Transactions consist of a value time pair

- Updating a projected output waveform consists of the deletion of zero or more previously computed transactions (called *old* transactions) from the projected output waveform, and the addition of new transactions, as follows:

Updating the Projected Output Waveform I

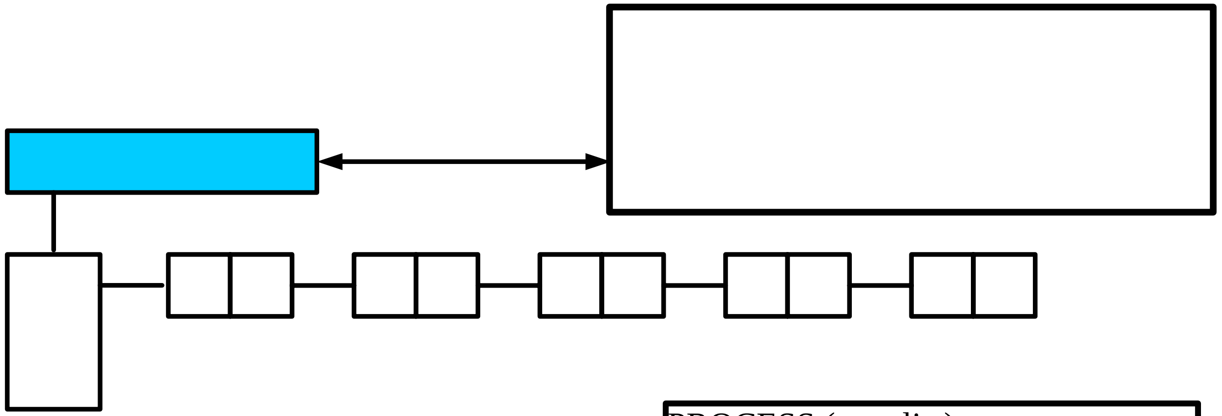
  - 1. All old transactions that are projected to occur at or after the time at which the earliest new transaction is projected to occur are deleted from the projected output waveform.
  - 2. The new transactions are then appended to the projected output waveform in the order of their projected occurrence.
- If the reserved word **transport** does not appear in the corresponding signal assignment statement then the delay is considered to be inertial delay, and the projected waveform is further modified as follows:

Updating the Projected Output Waveform II

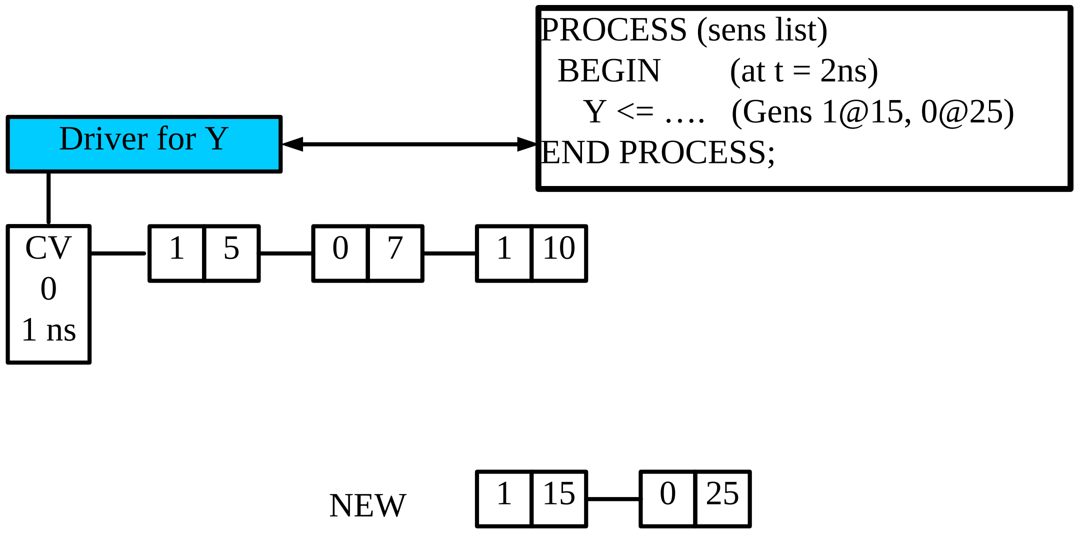
- For inertial delay:
  - 1. All of the new transactions are marked.
  - 2. An old transaction is marked if it immediately precedes a marked transactions and its value component is the same as that of the marked transaction.
  - 3. The transaction that determines the current value of the driver is marked.
  - 4. All unmarked transactions (all of which are old transactions) are deleted from the projected output waveform.
  - 93 addition of the reject limit slightly modifies this

Example – 1st part (87 on)

- Current state

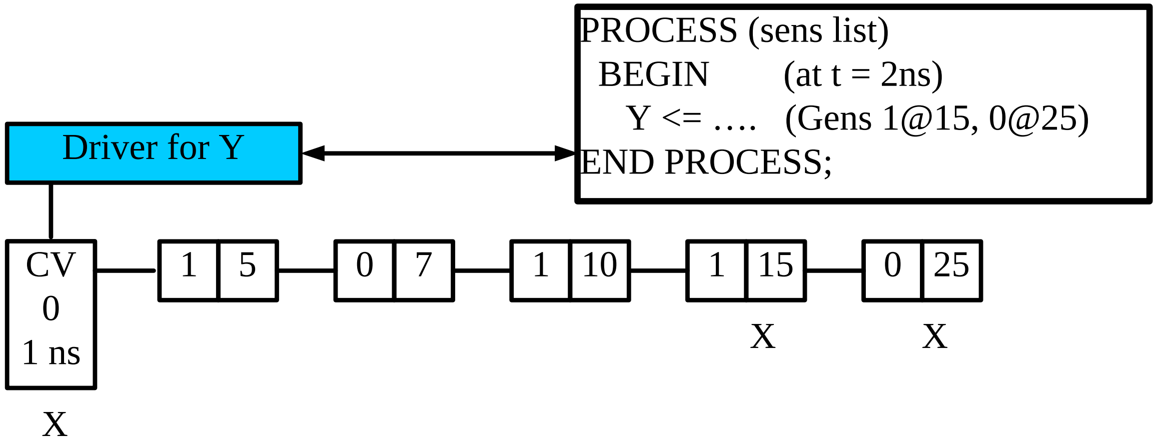


- Delete at or after



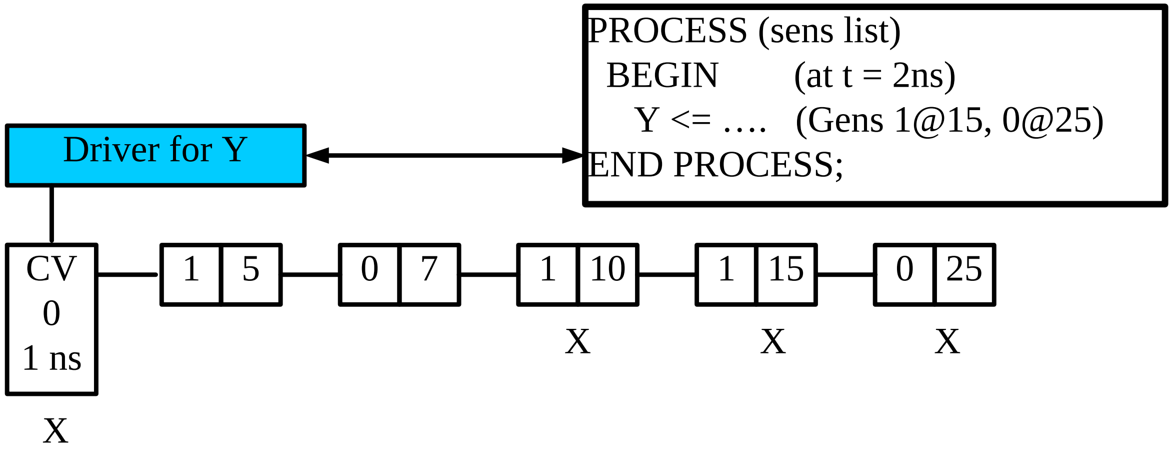
Example (cont)

- After appending



PROCESS (sens list)
  BEGIN   (at t = 2ns)
    Y <= ….   (Gens 1@15, 0@25)
END PROCESS;

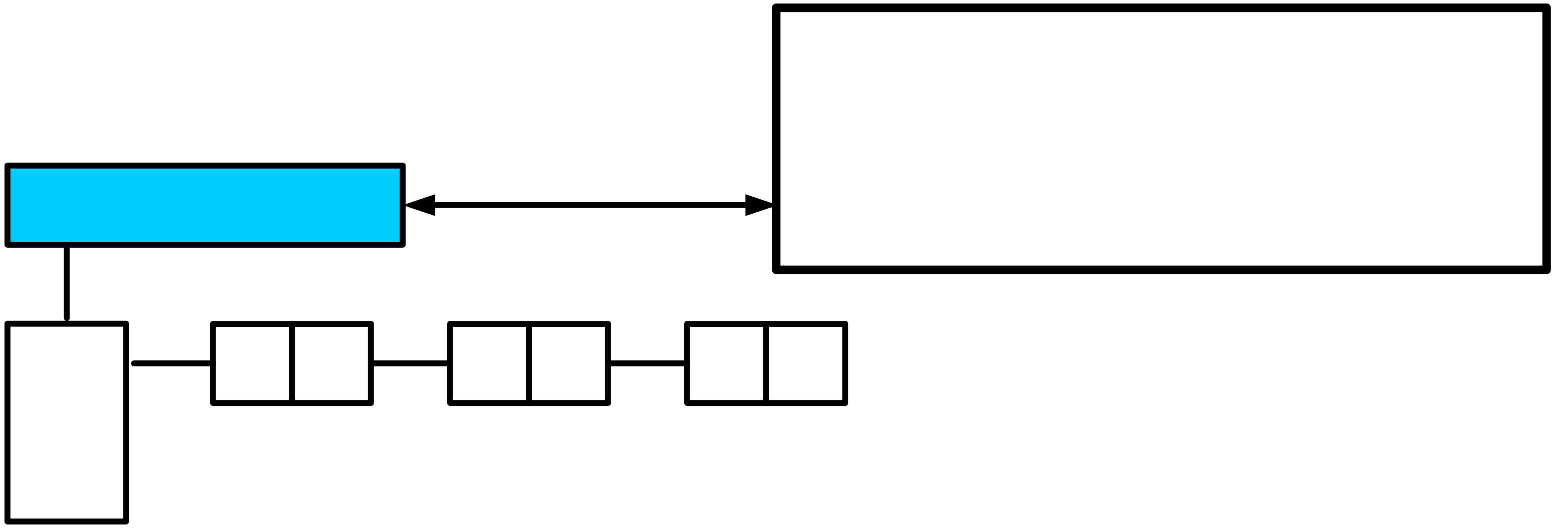
- And Marking



PROCESS (sens list)
  BEGIN   (at t = 2ns)
    Y <= ….   (Gens 1@15, 0@25)
END PROCESS;

Example

- And now delete the unmarked

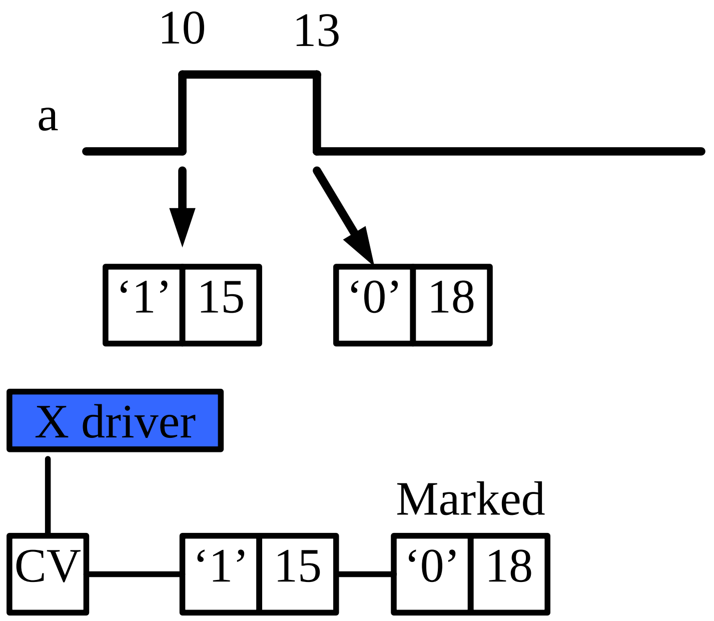


Updating the Projected Output Waveform II – 93 mod

- For inertial delay:
  - 1. All of the new transactions are marked.
  - 2. An old transaction is marked if it immediately precedes a marked transactions and its value component is the same as that of the marked transaction **or it precedes a marked transaction by more than the reject time such that the reject time for the preceding transaction has elapsed.**
  - 3. The transaction that determines the current value of the driver is marked.
  - 4. All unmarked transactions (all of which are old transactions) are deleted from the projected output waveform.

Example of no reject time

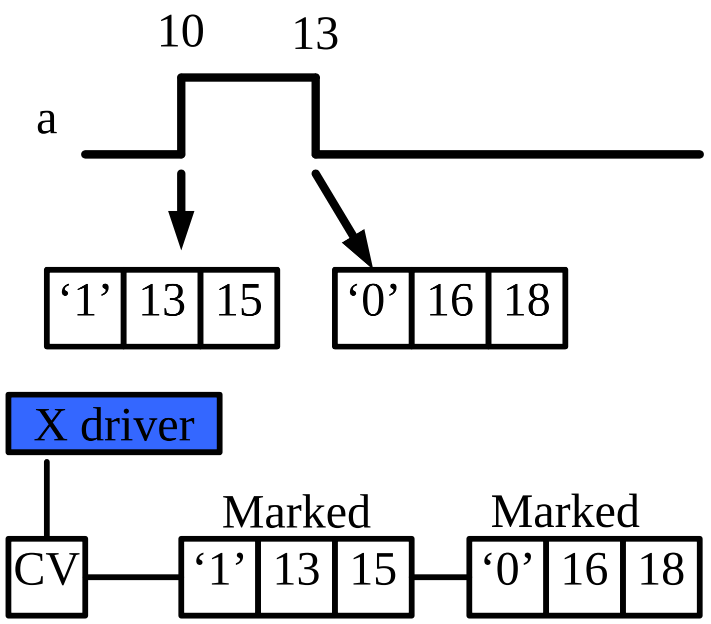
- Have the concurrent statement
  - x <= A after 5 ns;
- At 10 ns A has an event. The transaction just gets appended to the CV
- At 13 ns the transaction a new transaction is generated and is appended
- The inertial algorithm causes deletion of the transaction at 15.



Example (93 and on with reject)

- x <= reject 3 ns inertial
- A after 5 ns;

- Transactions has reject term to show reject time
- At 10 it is just appended like before
- At 13 ns the transaction is appended and then marked as it is a new transaction. The existing transaction is also Marked as the reject time has elapsed.


10
13
a
'1' 13 15
'0' 16 18
X driver
Marked
Marked
CV
'1' 13 15
'0' 16 18