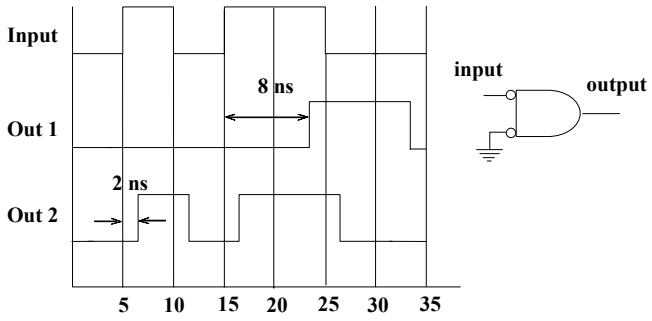
Delay Models in VHDL

Delay Models in VHDL

- Inertial delay
  - Default delay model
  - Suitable for modeling delays through devices such as gates
- Transport Delay
  - Model delays through devices with very small inertia, e.g., wires
  - All input events are propagated to output signals
- Delta delay
  - What about models where no propagation delays are specified?
  - Infinitesimally small delay is automatically inserted by the simulator to preserve correct ordering of events

Inertial Delays: Example



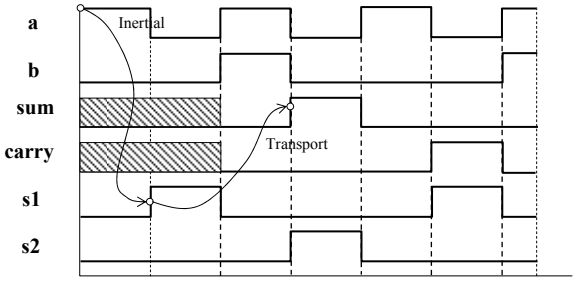
- *signal* <= **reject** *time-expression* **inertial** *value-expression* **after** *time-expression*;
- Most general form of a waveform element
- VHDL 1993 enables specification of pulse rejection width

Inertial Delays

- This model is designed for devices that have inertia and will reject spikes/pulses smaller than a specified width
- Default rejection width is propagation delay

Transport Delays: Example

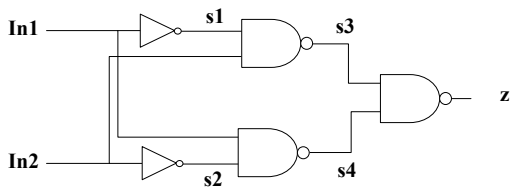
```
architecture transport_delay of half_adder is
signal s1, s2: std_logic:= '0';
begin
s1 <= (a xor b) after 2 ns;
s2 <= (a and b) after 2 ns;
sum <= transport s1 after 4 ns;
carry <= transport s2 after 4 ns;
end architecture transport_delay;
```



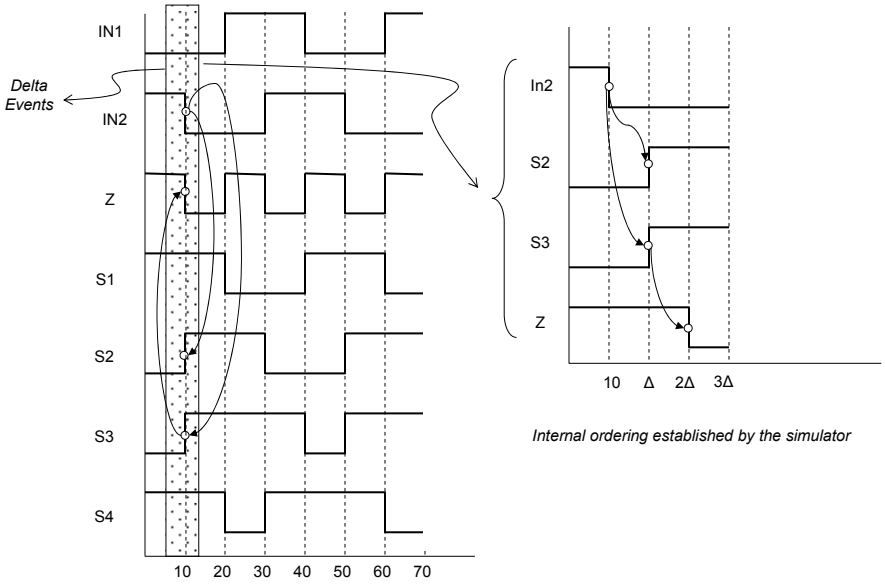
Delta Delays: Example

```
library IEEE;
use IEEE.std_logic_1164.all;
entity combinational is
port (In1, In2: in std_logic;
z : out std_logic);
end entity combinational;
```

```
architecture behavior of combinational
signal s1, s2, s3, s4: std_logic:= '0';
begin
s1 <= not In1;
s2 <= not In2;
s3 <= not (s1 and In2);
s4 <= not (s2 and In1);
z <= not (s3 and s4);
end architecture behavior;
```



Delta Delays: Behavior



Chapter Summary

- Delay models
  - Inertial
    - For devices with inertia such as gates
    - VHDL 1993 supports pulse rejection widths
  - Transport
    - Ensures propagation of all events
    - Typically used to model elements such as wires
  - Delta
    - Automatically inserted to ensure functional correctness of code blocks that do not specify timing
    - Enforces the data dependencies specified in the code