

Cell Characterization Concepts

Digital CAD

Introduction to Cell Characterization



SILVACO

Syllabus

- Overview
- Cell Characterization Attributes
- Delay Modeling
- Timing Arcs
- Lookup Table Templates
- Timing Constraints
- Power Modeling

Digital CAD

Overview

- Objective of Cell Characterization
- Digital Design Tools That Use Standard Cell Models
- Input Data Files Required by Digital Design Tools (Generated by AccuCell)
- Input Data Files Required by Digital Design Tools (Generated by Other Tools)
- Types of Standard Cell Libraries
- Digital Circuit Representation – Inverter
- Analog Circuit Description - Inverter
- Input Views of Circuits
Bridging Analog and Digital
- Static Timing Analysis Use of Liberty Format

Objective of Cell Characterization

- Create a set of high quality models of a standard cell library that accurately and efficiently model cell behavior
 - This set of models are used by several different digital design tools for different purposes

Digital Design Tools That Use Standard Cell Models

- Synthesis Tools
- Place and Routing Systems
- High level Design Language (HDL) Simulators (Verilog and VHDL)
- Floor planning Tools
- Physical Placement tools
- Static Timing Analysis (STA) tools
- Power Analysis tools
- Formal Verification tools
- Automatic Test Program Generation (ATPG) tools
- Library Compiler

Digital CAD

Input Data Files Required by Digital Design Tools (Generated by AccuCell)

- .lib Technology library source files
- .v Generated Verilog simulation libraries
- .tbench Verilog testbench to compare SPICE to Verilog with same stimulus
- .html HTML datasheet

Digital CAD

Input Data Files Required by Digital Design Tools (Generated by Other Tools)

- .db Compiled technology libraries in Synopsys internal database format
- Synopsys Milkyway Files - Abstracts or Bounding Boxes
- Cadence Encounter Files - Abstracts or Bounding Boxes
- LEF
- DEF
- GDS

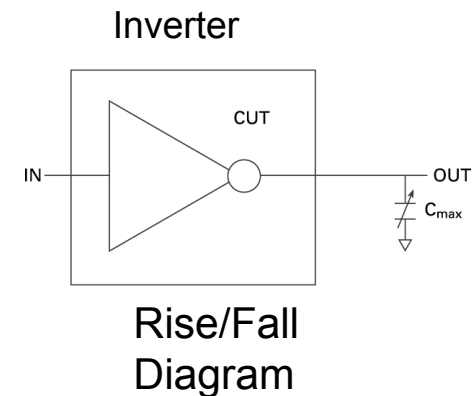
Types of Standard Cell Libraries

- There are often several cell libraries per semi process that typically contain 100 to 1,000 cells including:
 - Functions
 - Gates – inverter, AND, NAND, NOR, XOR, AOI, OAI
 - Flops – Flip flops (D, RS, JK), Latches, Scan Flops, Gated Flops
 - I/O Cells – Input pads, Output pads, Bidirectional Pads, Complex
 - Process Options
 - Mask layer options, gate shrinks, # of metals, special diffusions, thick metal, multiple oxides
 - Cell Options
 - Drive strengths, sets, resets, scans, substrate ties, antenna diodes
 - Optimized for Addressing Tradeoffs Between
 - High speed, high density, low power, low leakage, low voltage, low noise
- Cell Libraries are Produced by Foundries, IP Vendors, Fabless and IDMs

Digital Circuit Representation – Inverter

IEEE-1164 Verilog Logic States

| Strength | State | Value |
|-----------|-------|----------------|
| | U | Uninitialized |
| Driven | X | Unknown |
| Driven | 0 | Low |
| Driven | 1 | High |
| | Z | High impedance |
| Resistive | W | Weak X |
| Resistive | L | Weak 0 |
| Resistive | H | Weak 1 |
| | -- | Don't care |



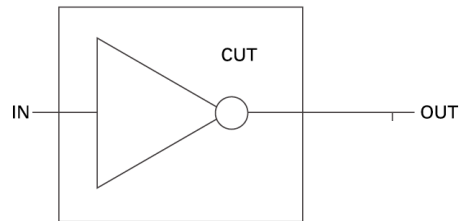
Verilog Language Description of Inverter

```
not i1 (out, in); // basic inverter
```

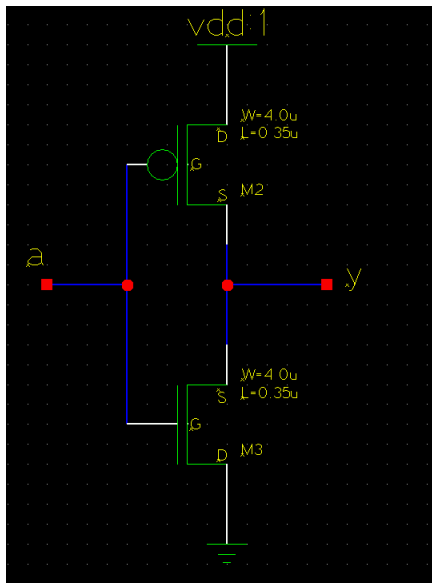
```
not #(5,3)i1 (out, in); // Rise=5ns, Fall=3ns
```

Digital CAD

Analog Circuit Description - Inverter



Transistor Inverter Schematic



Schematic Netlist

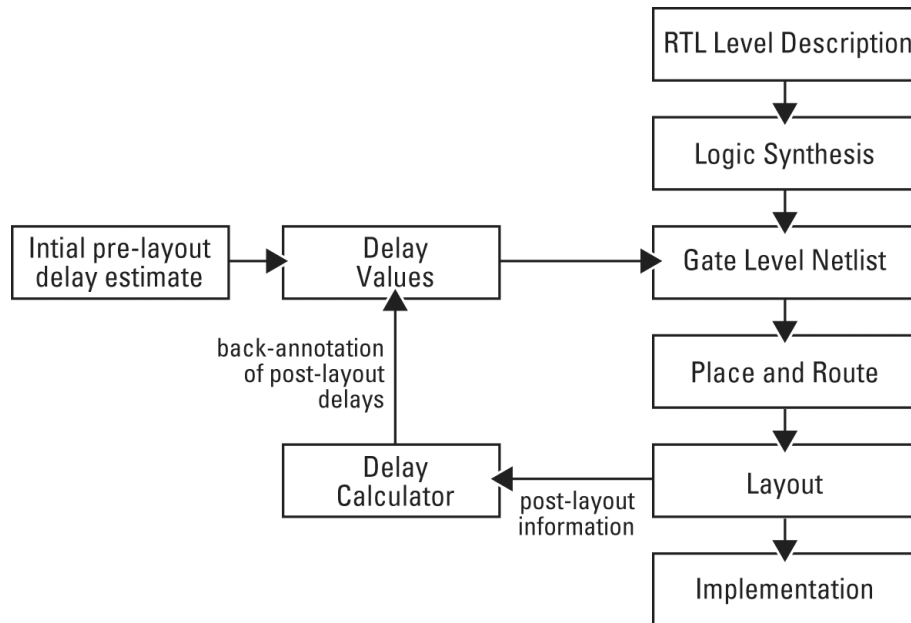
```
*svc_inv.sch
M3 y a gnd gnd nmos L=0.35u W=4.0u
M2 y a vdd vdd pmos L=0.35u W=4.0u
.END
```

Schematic Netlist with Parasitics

```
*svc_inv.sch
M3 y a gnd gnd nmos L=0.35u W=4.0u
M2 y a vdd vdd pmos L=0.35u W=4.0u
C1 ....
C2 ....
C3 ....
.END
```

Digital CAD

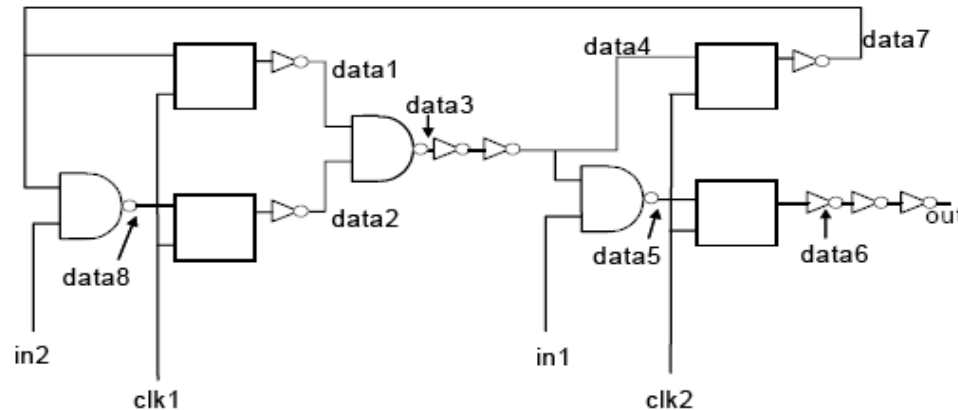
Input Views of Circuits – Bridging Analog and Digital



- Timing back annotation for Verilog simulator (gate, behavioral) Model must work in Verilog-XL, VCS, NCsim, Modelsim, SILOS
- Methodology has limitations on accuracy (load based only)
- STA is preferred methodology

Static Timing Analysis Use of Liberty Format

- In a standalone flow, STA operates independently of characterization reading both a Verilog netlist and multiple timing libraries in Liberty format
 - It can also read interconnect parasitic data in DSPF or SDF formats



Cell Characterization Attributes

- Cell Library Attributes
- Measurements
- Cell Library Model Quality
- Liberty .lib File Structure
- Liberty .lib File Library Level Attributes
- Operating Conditions
- Cell Attributes in .lib File
- Datasheet View of AND2
- Pin Attributes
- Setting Output Load Limits

Cell Library Attributes

- Pin Types →

- direction
- function

```
pin (A) {  
    direction : output ;  
    function : "X + Y" ;  
}
```

- Loads

- Capacitive
- Active
- Fanout and wire loads →

```
lu_table_template(wire_delay_table_template) {  
    variable_1 : fanout_number;  
    variable_2 : fanout_pin_capacitance;  
    variable_3 : driver_slew;  
    index_1 ("1.0 , 3.0");  
    index_2 ("0.12, 4.24");  
    index_3 ("0.1, 2.7, 3.12");  
}  
lu_table_template(trans_template) {  
    variable_1 : total_output_net_capacitance;  
    index_1 ("0.0, 1.5, 2.0, 2.5");  
}  
wire_load("05x05") {  
    resistance : 0 ;  
    capacitance : 1 ;  
    area : 0 ;  
    slope : 0.186 ;  
    fanout_length(1,0.39) ;  
    interconnect_delay(wire_delay_table_template)  
    values("0.00,0.21,0.3", "0.11,0.23,0.41", \  
    "0.00,0.44,0.57", "0.10 0.3, 0.41");  
}
```

- Stimulus

- PWL for slope
- Active drivers

- Indexes

- Load
- Input slope

Measurements

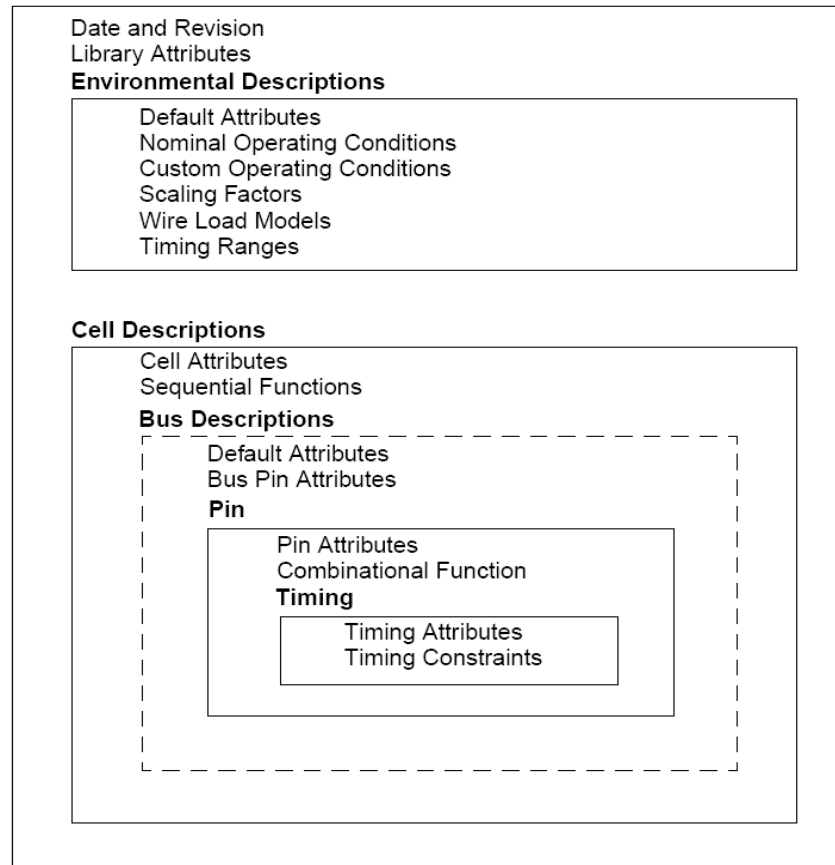
- Capacitance
- Thresholds/switching points
- Rise Time
- Fall Time
- Delay (propagation + transition = cell) (i.e. timing arcs)
- Power (static state dependent leakage, dynamic, short-circuit, hidden, internal) (i.e. power arcs)

Cell Library Model Quality

- Accuracy to silicon over the required power supply voltage, load range, input signal slope range
- Completeness of characterization (state, types[rise/fall], indexes, pins) – all timing arcs are included
- Conformance with digital tool format requirements (syntax, units, thresholds)
- Conformance with digital tool value constraints (monotonicity) and multi-tool timing engine correlation
- Model Efficiency - speed of execution of model in digital tool that runs many times on large circuits using generated models
- Characterization time efficiency – runs once but characterizing a single flop can take hours
- Minimum size of model file - .lib files can become huge, especially with noise data

Liberty .lib File Structure

Technology Library



- **Structural information**
 - Describes each cell's connectivity to the outside world, including cell, bus, and pin descriptions.
- **Functional information**
 - Describes the logical function of every output pin of every cell so that the digital design tools can map the logic of a design to the actual technology.
- **Timing information**
 - Describes the parameters for pin-to-pin timing relationships and delay calculation for each cell in the library.
- **Environmental information**
 - Describes the manufacturing process, operating temperature, supply voltage variations, and design layout, all of which directly affect the efficiency of every design.

Liberty .lib File Library Level Attributes

```
library (name) {  
    technology (name) ;/* library-level attributes */  
    delay_model : generic_cmos | table_lookup |  
                 cmos2 | piecewise_cmos | dcm |  
                 polynomial ;  
    bus_naming_style : string ;  
    routing_layers(string);  
    time_unit : unit ;  
    voltage_unit : unit ;  
    current_unit : unit ;  
    pulling_resistance_unit : unit ;  
    capacitive_load_unit(value,unit);  
    leakage_power_unit : unit ;  
}
```

Defines units for entire library

Default Units

| Component | Unit of measure |
|-------------|-----------------------------------|
| Area | Equivalent gates |
| | Square microns for standard cells |
| | Cell-based custom designs |
| Capacitance | Picofarads |
| | Standardized loads |
| Delay | Nanoseconds |
| Resistance | Kilohms |
| Temperature | Degrees centigrade |
| Voltage | Volts |

Operating Conditions

```
library (example) {  
  operating_conditions (WCCOM) {  
    process : 1.5 ;  
    temperature : 70 ;  
    voltage : 4.75 ;  
    tree_type : worst_case_tree ;  
    power_rail(v1, 2.5);  
    power_rail(v2, 3.9) ;  
  }  
}
```

name

- The name (WCCOM in the example) identifies the set of operating conditions

process

- The scaling factor accounts for variations in the outcome of the actual semiconductor manufacturing steps. This factor is typically 1.0 for normal operating conditions

temperature

- The ambient temperature in which the design is to operate

voltage

- The operating voltage of the design

tree_type

- The definition for the environment interconnect model.

power_rail

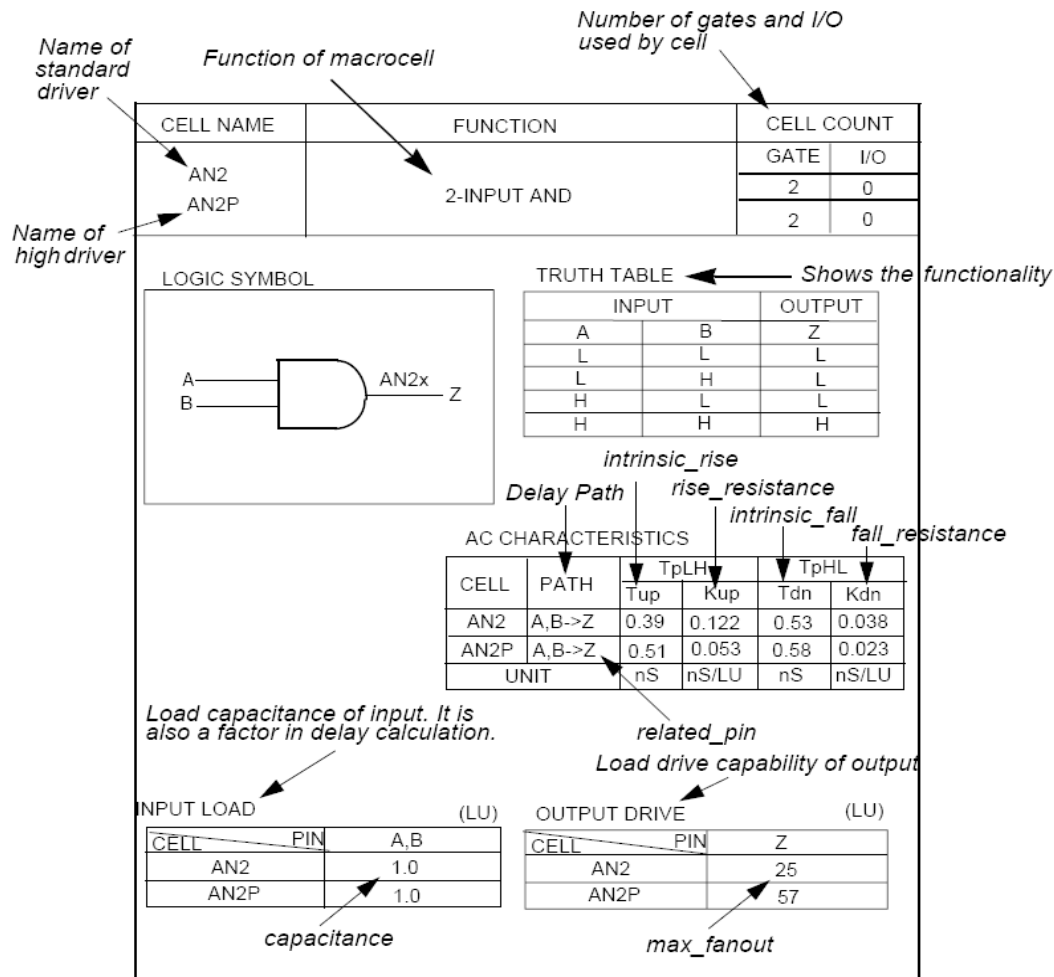
- The voltage value for a power supply

Cell Attributes in .lib File

- Structure
 - The cell, bus, and pin structure that describes each cell's connection to the outside world.
- Function
 - The logical function of every output pin of each cell that digital design tools use to map the logic of a design to the actual technology.
- Timing
 - Timing analysis and design optimization information, such as the parameters for pin-to-pin timing relationships, delay calculations, and timing constraints for sequential cells.
- Power
 - Modeling for state-dependent and path-dependent power
- Other parameters
 - These parameters describe area and design rules.

Digital CAD

Datasheet View of AND2



- Correlation between datasheet and .lib representation of a 2 input AND gate

```

pin(A,B) {
    direction : input ;
    capacitance : 1 ;
}
pin(Z) {
    direction : output ;
    function : "A * B" ;
    timing() {
        intrinsic_rise : 0.39 ;
        intrinsic_fall : 0.53 ;
        rise_resistance : 0.122 ;
        fall_resistance : 0.038 ;
        related_pin : "A B" ;
    }
}
    
```

Pin Attributes

- direction
 - Defines the direction of each pin. In the example on the previous page, A and B are defined as input pins and Z as an output pin
- capacitance
 - Defines the input pin load (input capacitance) placed on the network. Load units should be consistent with other capacitance specifications throughout the library
 - Typical units of measure for capacitance are picofarads and standardized loads
- function
 - Defines the logic function of an output pin in terms of the cell's input or inout pins. In the example, the function of pin Z is defined as the logical AND of pins A and B
- timing
 - Describes timing groups. The timing groups describe the following:
 - - A pin-to-pin delay
 - - A timing constraint such as setup and hold
 - In the example, the timing group for pin Z describes the delays between pin Z and pins A and B

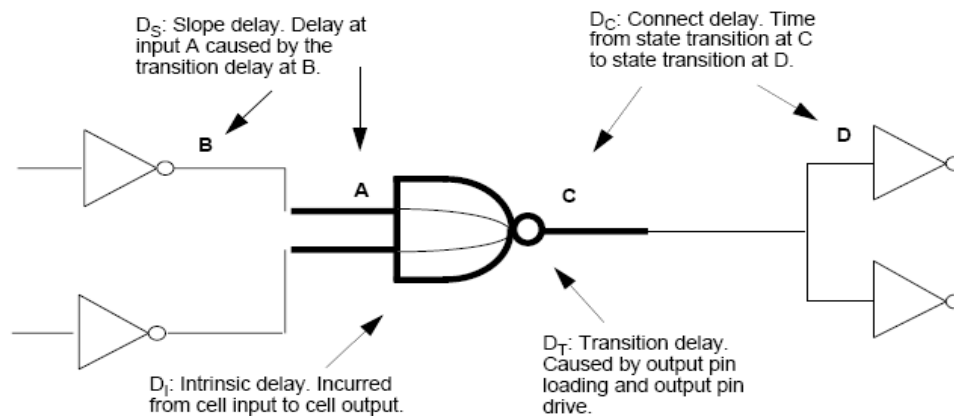
Setting Output Load Limits

- **fanout_load**
 - Specifies how much to add to the fanout on the net
- **max_fanout**
 - Specifies the maximum number of loads a pin can drive
- **max_transition**
 - Specifies the maximum rise or fall transition time on an output due to total capacitive load
- **max_capacitance**
 - Specifies the maximum total capacitive load that an output pin can drive
- **min_fanout**
 - Specifies the minimum number of loads that a pin can drive
- **min_capacitance**
 - Specifies the minimum total capacitive load that an output pin can drive

Delay Modeling

- Total Delay Equation
- Total Delay Scaling
- Slope Delay
- Slew Modeling
- Intrinsic and Transition Delays
- Connect Delay
- Interconnect Delay

Total Delay Equation

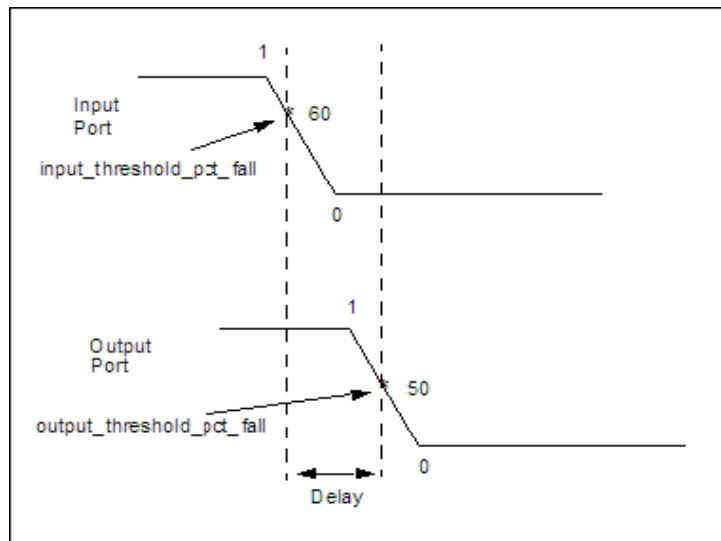


- $D_{total} = D_I + D_S + D_C + D_T$
- D_I
 - Intrinsic delay inherent in the gate and independent of particular instantiation
- D_S
 - Slope delay caused by the ramp time of the input signal
- D_C
 - Connect media delay to an input pin (wire delay)
- D_T
 - Transition delay caused by loading of the output pin

Digital CAD

Total Delay Scaling

- When calculating total delay, the digital tool scales each parameter of D_{total} individually.
- Each component of the total delay has its own global parameters to model the effects on the nominal case of variations in process, temperature, and voltage.



*Total Delay is typically measured from 50% to 50%, regardless of where transition thresholds are set

Slope Delay

- The slope delay of an element (D_S) is the incremental time delay caused by slowly changing input signals. This is not used by AccuCell
- In some technologies, this delay is a strong function of the ramp time
- D is calculated with the transition delay at the previous output pin, plus a slope sensitivity factor, as shown here: $D_S = D_{T(\text{prevstage})}$
 - This equation calculates both the rise and fall delays. Where applicable, use the “rise” parameter to calculate the rise delay and the “fall” parameter to calculate the fall delay
- DS
 - Transition delay is calculated at the previous stage of logic. Therefore, the calculation of DS enforces a global order on local analysis
- SS
 - Slope sensitivity factor. This factor accounts for the time during which the input voltage begins to rise but has not reached the threshold level at which channel conduction begins. The attributes that define it in the timing group of the driving pin are `slope_rise` and `slope_fall`
- $DT(\text{prevstage})$
 - The transition delay calculated at the previous output pin

Digital CAD

Slew Modeling

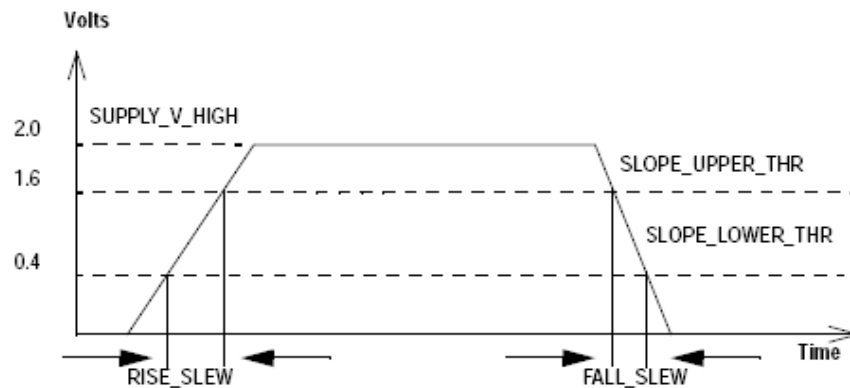


Figure 4-4: Slew measurement within appropriate thresholds

- Slew is the time it takes for the voltage value to fall or rise between two designated threshold points on an input, an output, or a bidirectional port
- The designated threshold points must fall within a voltage falling from 1 to 0 or rising from 0 to 1

Intrinsic and Transition Delays

- Intrinsic Delay

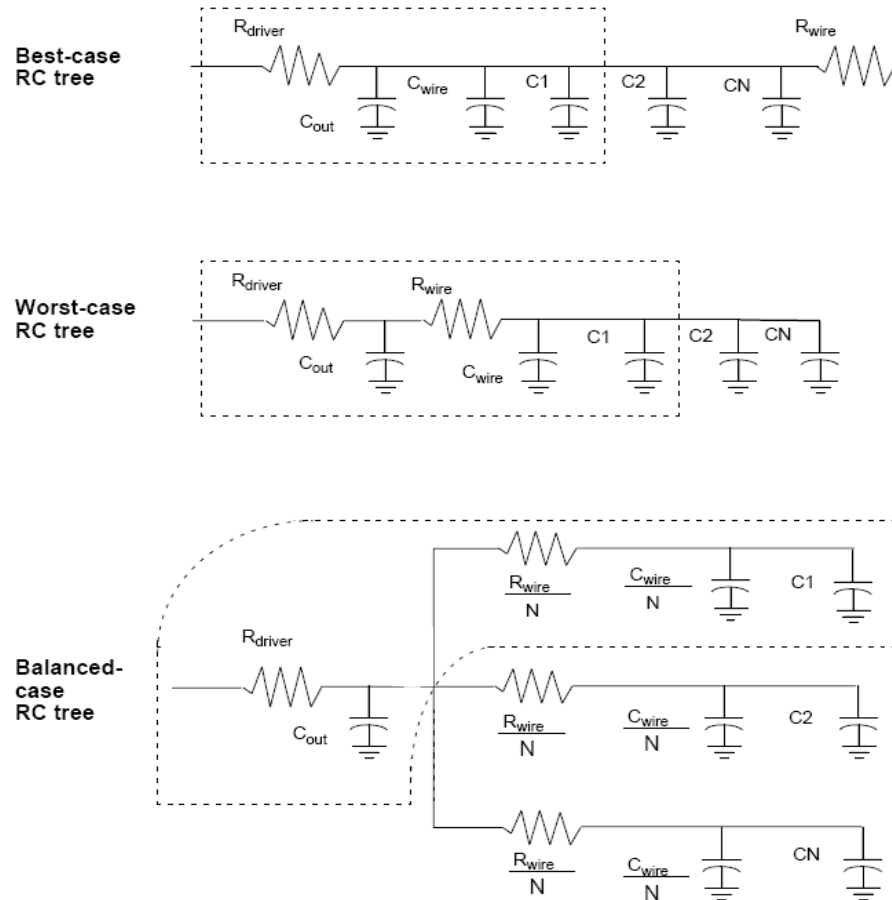
- The intrinsic delay of a circuit element (DI) is the portion of the total delay that is independent of the circuit element's usage. This portion is the fixed (or zero load) delay from the input pin to the output pin of a circuit element.

- Transition Delay

- The transition delay of a circuit element is the time it takes the driving pin to change state. The transition time of the output pin on a net is a function of the capacitance of all pins on the net and the capacitance of the interconnect network that ties the pins together.
 - This equation calculates the rise and fall delays.

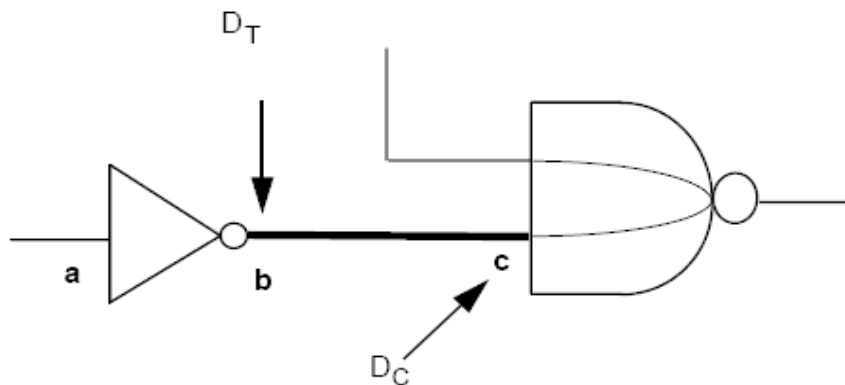
Digital CAD

Connect Delay



- The connect delay of an element (DC) is the time it takes the voltage at an input pin to charge after the driving output pin has made a transition
- This delay is also known as time-of-flight delay, which is the time it takes a waveform to travel along a wire

Interconnect Delay



$$D_{\text{interconnect}} = D_T + D_C$$

- Interconnect delay is defined as the delay caused by connect delay and fanout
 - It is calculated as the sum of D_T and D_C
- Include the capacitance attribute in the pin group of the input pin
- Give zero capacitance to the pin group of the output pin
- Resistance is attributed entirely to the output pin

Timing Arcs

- Timing Arc Concepts
- Combinational Timing Arcs
- Sequential Timing Arcs
- Timing Arcs Between Single and Multiple Pins
- Three-State Timing Arcs
- Edge-Sensitive Timing Arcs
- Preset Arcs
- Clear Arcs
- Defining Delay Arcs With Lookup Tables

Timing Arc Concepts

- Timing arcs can be delay arcs or constraint arcs
- Each timing arc has a startpoint and an endpoint
- The startpoint can be an input, output, or inout pin
- The endpoint is always an output pin or an inout pin
- The only exception is a constraint timing arc, such as a setup, hold, recovery or removal constraint between two input pins
- `related_pin`
 - This attribute defines the pin or pins representing the startpoint of a timing arc

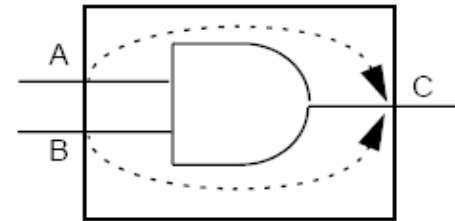
Timing Arc Concepts (cont'd)

- All delay information in a library refers to an input-to-output pin pair or an output-to-output pin pair defined as:
 - intrinsic delay
 - The fixed delay from input to output pins
 - transition delay
 - The time it takes the driving pin to change state. Transition delay attributes represent the resistance encountered in making logic transitions
 - slope sensitivity
 - The incremental time delay due to slow change of input signals

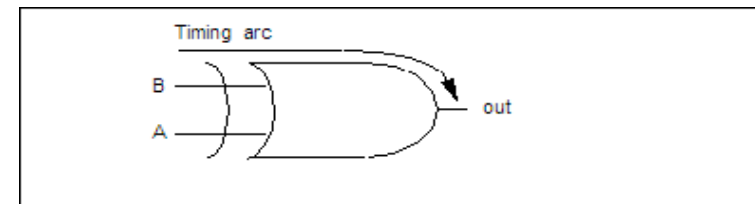
Combinational Timing Arcs

- A combinational timing arc describes the timing characteristics of a combinational element
 - The timing arc is attached to an output pin, and the related pin is either an input or an output
 - AccuCell does not use these
- A combinational timing arc is of one of the following types:
 - combinational
 - combinational_rise
 - combinational_fall
 - three_state_disable
 - three_state_disable_rise
 - three_state_disable_fall
 - three_state_enable
 - three_state_enable_rise
 - three_state_enable_fall

AND Gate With Timing Arc



XOR Gate With State-Dependent Timing Arc



Sequential Timing Arcs

- A sequential timing arc is of one of the following types:
 - Edge-sensitive (rising_edge or falling_edge)
 - Preset or clear
 - Setup or hold (setup_rising, setup_falling, hold_rising, or hold_falling)
 - Nonsequential setup or hold (non_seq_setup_rising, non_seq_setup_falling, non_seq_hold_rising, non_seq_hold_falling)
 - Recovery or removal (recovery_rising, recovery_falling, removal_rising, or removal_falling)
 - No change (nochange_high_high, nochange_high_low, nochange_low_high, nochange_low_low)

Timing Arcs Between Single and Multiple Pins

```
cell (my_inverter) {  
  ...  
  pin (A) {  
    direction : input;  
    capacitance : 1;  
  }  
  pin (B) {  
    direction : output;  
    function : "A'";  
    timing (A_B) {  
      related_pin : "A";  
      ...  
    }/* end timing() */  
  }/* end pin B */  
}/* end cell */
```

Pin and a Single Related Pin

```
cell (my_and) {  
  ...  
  pin (A) {  
    direction : input;  
    capacitance : 1;  
  }  
  pin (B) {  
    direction : input;  
    capacitance : 2;  
  }  
  pin (C) {  
    direction : output;  
    function : "A B";  
    timing (A_C, B_C) {  
      related_pin : "A B";  
      ...  
    }/* end timing() */  
  }/* end pin B */  
}/* end cell */
```

Pin and Multiple Related Pins

*Timing Arcs can also be between pins, groups, and busses

Three-State Timing Arcs

- Assign related_pin to the enable pin of the three-state function
- Define the Z-to-1 propagation time with the intrinsic_rise statement
- Define the Z-to-0 propagation time with the intrinsic_fall statement
- Include the timing_type : three_state_enable statement

```
timing () {  
    related_pin : "OE" ;  
    timing_type : three_state_enable ;  
    intrinsic_rise : 1.0 ; /* Z-to-1 time */  
    intrinsic_fall : 1.0 ; /* Z-to-0 time */  
}
```

Edge-Sensitive Timing Arcs

- Edge-sensitive timing arcs, such as the arc from the clock on a flipflop, are identified by the following values of the `timing_type` attribute in the timing group
- `rising_edge`
 - Identifies a timing arc whose output pin is sensitive to a rising signal at the input pin
- `falling_edge`
 - Identifies a timing arc whose output pin is sensitive to a falling signal at the input pin
- These arcs are path-traced; the path tracer propagates only the active edge (rise or fall) path values along the timing arc

Preset Arcs

- Select
 - timing_type : preset;
 - timing_sense :
- positive_unate
 - Indicates that the rise arrival time of the arc's source pin is used to calculate the arc's delay
 - This calculation produces the rise arrival time on the arc's endpoint pin
 - In the case of slope delays, the source pin's rise transition time is added to the arc's delay
 - The source pin is active-high

Preset Arcs (cont'd)

- **negative_unate**
 - Indicates that the fall arrival time of the arc's source pin is used to calculate the arc's delay
 - This calculation produces the rise arrival time on the arc's endpoint pin
 - In the case of slope delays, the source pin's fall transition time is added to the arc's delay
 - The source pin is active-low
- **non_unate**
 - Indicates that the maximum of the rise and fall arrival times of the arc's source pin is used to calculate the arc's delay
 - This calculation produces the maximum arrival time on the arc's endpoint pin
 - In the case of slope delays, the maximum of the source pin's rise and fall transition times is added to the arc's delay

Clear Arcs

- Clear arcs affect only the fall arrival time of the arc's endpoint pin
 - A clear arc means that you are asserting a logic 0 on the output pin when the designated related_pin is asserted
- Select
 - timing_type : clear;
 - timing_sense :
- positive_unate
 - Indicates that the fall arrival time of the arc's source pin is used to calculate the arc's delay
 - This calculation produces the fall arrival time on the arc's endpoint pin
 - In the case of slope delays, the source pin's fall transition time is added to the arc's delay
 - The source pin is active-low

Clear Arcs (cont'd)

- `negative_unate`
 - Indicates that the rise arrival time of the arc's source pin is used to calculate the arc's delay
 - This calculation produces the fall arrival time on the arc's endpoint pin
 - In the case of slope delays, the source pin's rise transition time is added to the arc's delay
 - The source pin is active-high
- `non_unate`
 - Indicates that the maximum of the rise and fall arrival times of the arc's source pin is used in calculating the arc's delay
 - This calculation produces the maximum fall arrival time on the arc's endpoint pin
 - In the case of slope delays, the maximum of the source pin's rise and fall transition times is added to the arc's delay

Defining Delay Arcs With Lookup Tables

- Transition time is the time it takes for an output signal to make a transition between the high and low logic states. With nonlinear delay models, it is computed by table lookup and interpolation. Transition delay is a function of capacitance at the output pin and input transition time.
- Group attributes:
 - cell_rise
 - cell_fall
 - rise_propagation
 - fall_propagation
 - retaining_rise
 - retaining_fall
 - retain_rise_slew
 - retain_fall_slew

To specify cell delay independently of transition delay, use one of these timing group attributes as your lookup table:

To specify transition delay as a term in the total cell delay, use one of these timing group attributes as your lookup table

Lookup Table Templates

- Defining Lookup Table Templates
- Assigning Values to Lookup Tables

Defining Lookup Table Templates

- CMOS Nonlinear Delay Model is specified by a one or two dimensional table of delay values dependent on input net transition and output capacitance

```
/* Define template of size 4 x 4*/

lu_table_template(basic_template) {
    variable_1 : input_net_transition;
    variable_2 : total_output_net_capacitance;
    index_1 ("0.0, 0.5, 1.5, 2.0");
    index_2 ("0.0, 2.0, 4.0, 6.0");
}

/* Define library-level one-dimensional lu_table of size 4 */

lu_table_template(one_dimensional) {
    variable_1 : input_net_transition;
    index_1 ("0.0, 0.5, 1.5, 2.0");
}

. . .
```

Assigning Values to Lookup Tables

```
cell (general) {
    . . .
    pin(a) {
        direction: output;
        timing() {
            . . .

            /* Inherit the 'basic_template' template */

            cell_rise(basic_template) {

                /* Specify all the values */

                values ("0.0, 0.13, 0.17, 0.19", "0.21, 0.23, 0.30, \
                    0.41", "0.22, 0.31, 0.35, 0.47", "0.33, \
                        0.37, 0.45, 0.50");

            }
        }
        . . .
    }
}
pin(b) {
    direction: output;
    timing() {
        . . .
        /* Inherit the 'one-dimensional' template */

        cell_rise(one_dimensional) {

            /*Specify all the values within a pair of ""*/

            values ("0.1, 0.15, 0.20, 0.29");

        }
    }
}
```

- Referring to tables defined in previous slide
- Pin a is two dimensional 4X4
- Pin b is one dimensional X4
- These timing values are the results of SmartSpice .MEASURE statements within AccuCell

Timing Constraints

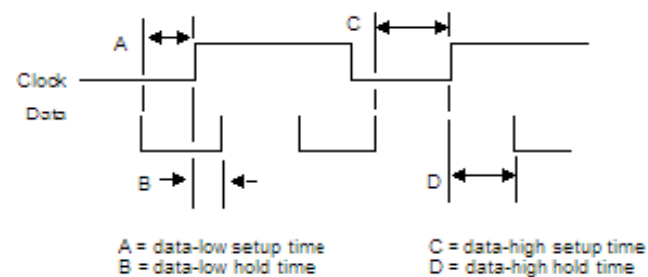
- Timing Constraint Concepts
- Setup and Hold Constraints
- Non Sequential Setup and Hold Constraints
- Recovery Timing Constraints
- Removal Timing Constraints
- .lib of State Table Flip Flop
- .lib of Type ff D Flip Flop

Timing Constraint Concepts

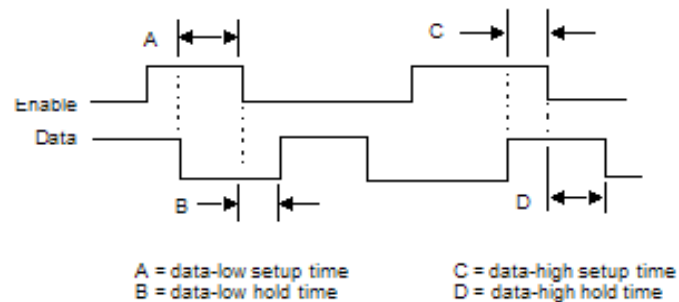
- setup and hold arcs
 - Set these constraints to ensure that a data signal has stabilized, before latching its value
- recovery and removal arcs
 - Use the recovery timing arc and the removal timing arc for asynchronous control pins such as clear and preset
- skew
 - This is another constraint that the VHDL library generator uses for simulation.
- You can also set state-dependent and conditional constraints

Setup and Hold Constraints

Setup and Hold Constraints for Rising-Edge-Triggered Flip-Flop

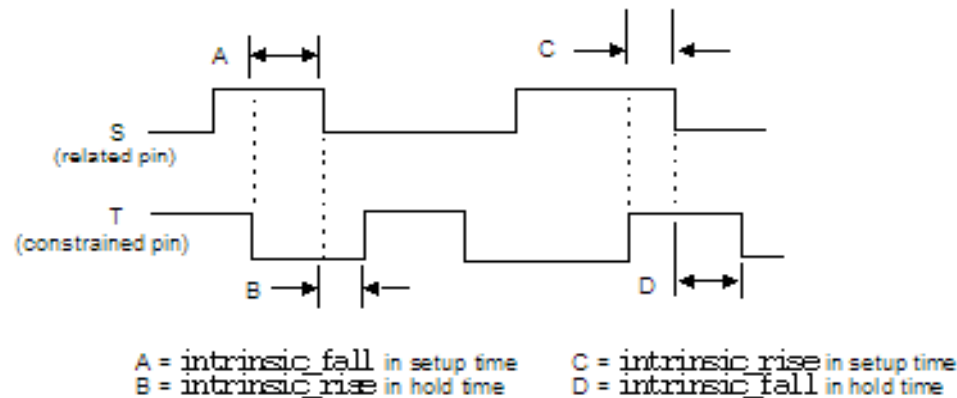


Setup and Hold Constraints for High-Enable Latch



Non Sequential Setup and Hold Constraints

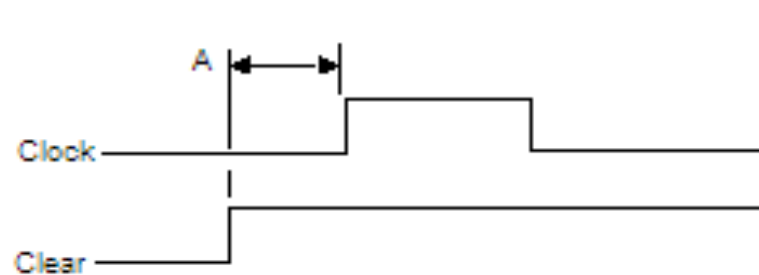
- In some nonsequential cells, the setup and hold timing constraints are specified on the data pin with a nonclock pin as the related pin
- The signal of a pin must be stable for a specified period of time before and after another pin of the same cell range state for the cell to function as expected



Nonsequential Setup and Hold Constraints

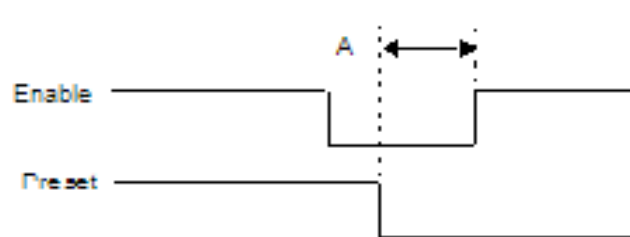
Recovery Timing Constraints

Recovery Timing Constraint for a Rising-Edge-Triggered Flip-Flop



A = clear to clock recovery time

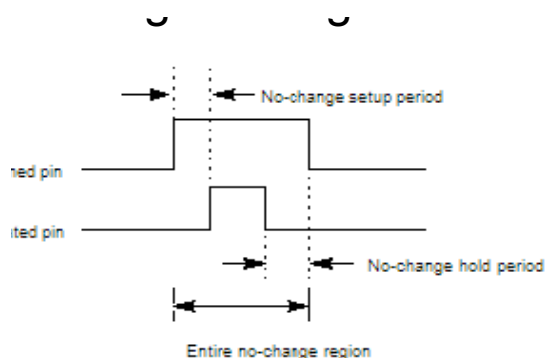
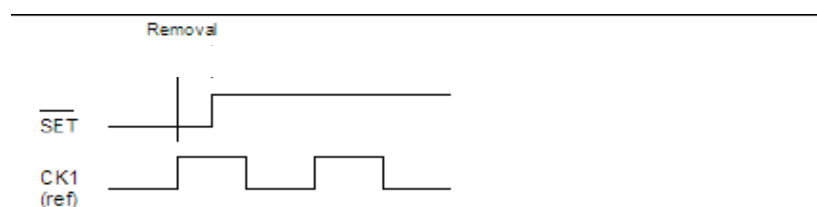
Recovery Timing Constraint for a Low-Enable Latch



A = preset to enable recovery time

Removal Timing Constraints

Timing Diagram for Removal Constraint



- Removal Constraint
 - This constraint is also known as the asynchronous control signal hold time.
 - The removal constraint describes the minimum allowable time between the active edge of the clock pin while the asynchronous pin is active and the inactive edge of the same asynchronous control pin
- No-Change Timing Constraints
 - You can model no-change timing checks to use in static timing verification during synthesis
 - A no-change timing check checks a constrained signal against a level-sensitive related signal
 - The constrained signal must remain stable during an established setup period, for the width of the related pulse, and during an established hold period

Digital CAD

.lib of State Table Flip Flop

```
cell(flipflop1) {
  area : 7 ;
  pin(D) {
    direction : input ;
    capacitance : 1.3 ;
    timing() {
      timing_type : setup_rising ;
      intrinsic_rise : 0.9 ;
      intrinsic_fall : 0.9 ;
      related_pin : "CP" ;
    }
    timing() {
      timing_type : hold_rising ;
      intrinsic_rise : 0.5 ;
      intrinsic_fall : 0.5 ;
      related_pin : "CP" ;
    }
  }
  pin(CP) {
    direction : input ;
    capacitance : 1.3 ;
    min_pulse_width_high : 1.5 ;
    min_pulse_width_low : 1.5 ;
  }
  statetable (" D    CP", "IQ  IQN") {
    table :    L/H  R  : - - : L/H  H/L, \
              -   ~R : - - : N    N" ;
  }
  . . . . .
}
```

```
pin(Q) {
  direction : output ;
  internal_node : "IQ" ;
  timing() {
    timing_type : rising_edge ;
    intrinsic_rise : 1.11 ;
    intrinsic_fall : 1.43 ;
    rise_resistance : 0.1513 ;
    fall_resistance : 0.0544 ;
    related_pin : "CP" ;
  }
}
pin(QN) {
  direction : output ;

  internal_node : "IQN"
  timing() {
    timing_type : rising_edge ;
    intrinsic_rise : 1.58 ;
    intrinsic_fall : 1.56 ;
    rise_resistance : 0.1513 ;
    fall_resistance : 0.0544 ;
    related_pin : "CP" ;
  }
}
}
```

SILVACO

.lib of Type ff D Flip Flop

```
cell(flipflop1) {  
    area : 7 ;  
    pin(D) {  
        direction : input;  
        ...  
    }  
    pin(CP) {  
        direction : input;  
        ...  
    }  
    ff(IQ,IQN) {  
        next_state : "D" ;  
        clocked_on : "CP" ;  
    }  
    pin(Q) {  
        direction : output ;  
        function : "IQ" ;  
        ...  
    }  
    pin(QN) {  
        direction : output ;  
        function : "IQN";  
        ...  
    }  
}
```

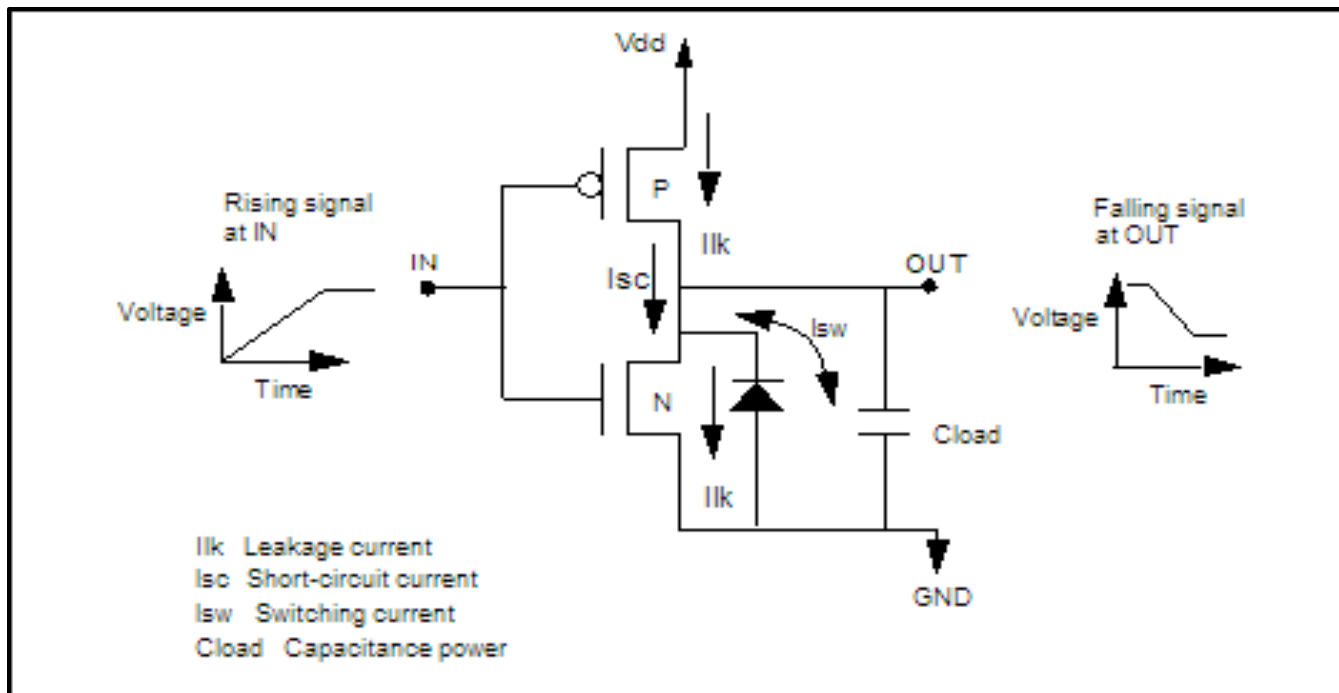
- The ff group statement replaces the statetable group statement
- The function attribute, rather than the internal_node attribute, defines the output pin's function
- The D flip-flop defines two variables, IQ and IQN
- The next_state equation determines the value of IQ after the next clocked_on transition
- In this example IQ is assigned the value of the D input

Power Modeling

- Components of Power Dissipation
- Power Modeling Concepts
- State Dependent Leakage Power
- Modeling Internal Power Lookup Tables
- Internal Power Calculations
- Clock Pin Power
- Output Pin Power
- Power Lookup Tables Descriptions 1D, 2D, 3D
- Internal Power Table for Cell Output
- Calculating Switching Power
- Switching Power Calculations
- Syllabus for Advanced Cell Characterization

Digital CAD

Components of Power Dissipation



$$P_{\text{Dynamic}} = \underbrace{\sum_{\forall \text{cells}(i)} \left(P_{\text{CellInternal}_i} \times E_i \right)}_{\text{Internal power}} + \frac{V_{\text{dd}}^2}{2} \underbrace{\sum_{\forall \text{nets}(i)} \left(C_{\text{Load}_i} \times E_i \right)}_{\text{Switching power}}$$

Power Modeling Concepts

- Leakage Power
 - Leakage power is the static (or quiescent) power dissipated when a gate is not switching
- Short-Circuit Power
 - Short-circuit or internal power is the power dissipated whenever a pin makes a transition
 - This can be handled in two ways:
 - Include the effect of the output capacitance in the internal_power group (defined in a pin group within a cell group), which gives the output pins zero capacitance
 - Give the output pins a real capacitance, which causes them to be included in the switching power, and model only the short-circuit power as the cell's internal power (in the internal_power group)

Power Modeling Concepts (cont'd)

- Switching Power
 - Switching (or interconnect) power is the power dissipated in the circuit as a result of a logical transition of the capacitive load
 - Switching power (along with internal power) is used to compute the design's total dynamic power dissipation

State Dependent Leakage Power

$$P_{\text{LeakageTotal}} = \sum_{\forall \text{cells}(i)} P_{\text{CellLeakage}_i}$$

$$P_{\text{leakage}} = VDD \times I_{\text{supply}}$$

```
cell (my cell) {  
    ...  
    leakage_power () {  
        when : "! A";  
        value : 2.0;  
    }  
    cell_leakage_power : 3.0;  
}
```

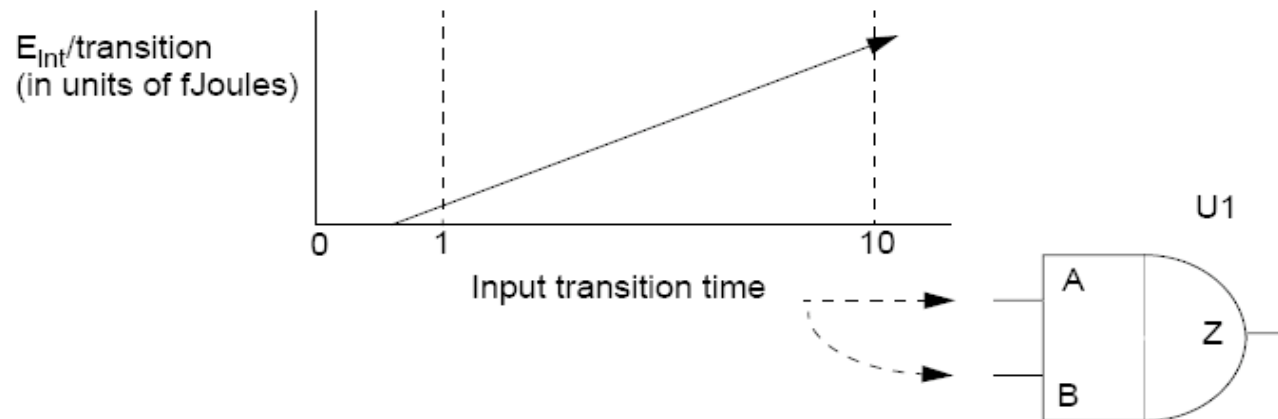
- Leakage power is state dependent based on input pin state values

Modeling Internal Power Lookup Tables

- You should measure the energy dissipated by varying either input voltage transition or output load while holding the other constant
- Because a table indexed by T input transition times and C output load capacitances has $T \times C$ entries, the cell's internal power must be characterized $T \times C$ times, once for each input transition time and output load capacitance combination
- For example, if internal power will be modeled by use of a 3×3 table at the output of the cell, the design will have 9 input voltage transitions—output load combinations where energy dissipation must be measured
- The library group supports a one-, two-, or three-dimensional internal power lookup table indexed by the total output load capacitances (best model), the input transition time, or both
- NOTE: The input pin power is added to the output pin power
 - When you model the library, avoid double counting

Modeling Internal Power Lookup Tables

Power is calculated by integrating energy



Internal Power Calculations

- To calculate the internal power for cell U1, use the following equation:
 - P_{Int}
 - Total internal power for the cell.
 - E
 - Internal energy for the pin.
 - AF
 - Activity factor.
- Accurate sequential modeling requires a separate table for the clock and for the output pin the clock controls
 - The two tables are used to ensure that clock pin power and output power are accounted for separately, because a clock pin often toggles without causing any observable state change on the output pin

Clock Pin Power

- This energy is characterized by simulation of a single full cycle (one rise transition and one fall transition) of the clock, with no transition at the output and input pins
 - A one-dimensional internal power table indexed by input transition time should be attached to the clock pin
 - Total energy dissipated in the cell during this simulation is measured. If separate rise and fall power modeling is not used, the energy measured must be divided by 2 to get the energy dissipated by the clock pin transition, because the measurement is done for two transitions of the clock
- $\text{Clk_Pin_Energy} = \text{Clk_Total} / 2$
- Add Clk_Pin_Energy as an entry indexed by input transition time in the one-dimensional internal power table attached to the clock pin

Output Pin Power

- This power is characterized by simulation of two full cycles of the clock, with two rise and fall transitions at the output
 - A two-dimensional internal power table should be attached to the output pin
 - Total energy dissipated in the cell during the two-full-cycle simulation (Out_total) is measured
 - If separate rise and fall power modeling is not used, the energy measured must be divided by 2, because the measurement is done for two transitions.
- $\text{Output_Pin_Energy} = (\text{Out_total})/2 - 2 * (\text{Clk_Pin_Energy})$

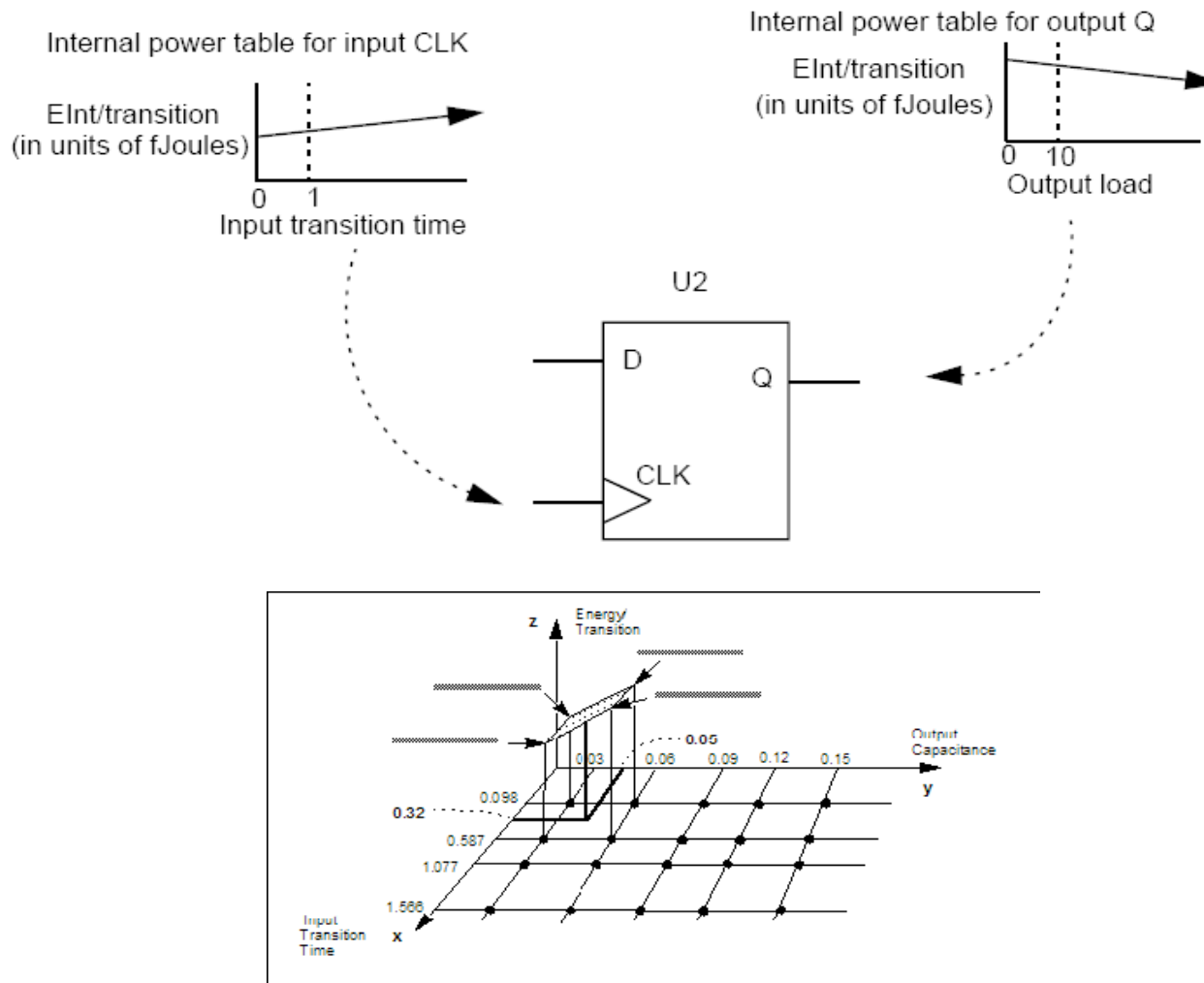
Power Lookup Tables Descriptions 1D, 2D, 3D

```
power_lut_template (output_by_cap) {  
    variable_1 : total_output_net_capacitance ;  
    index_1 ("0.0, 5.0, 20.0") ;  
}  
  
power_lut_template (output_by_cap_and_trans) {  
    variable_1 : total_output_net_capacitance ;  
    variable_2 : input_transition_time ;  
    index_1 ("0.0, 5.0, 20.0") ;  
    index_2 ("0.1, 1.0, 5.0") ;  
}  
  
power_lut_template (input_by_trans) {  
    variable_1 : input_transition_time ;  
    index_1 ("0.0, 1.0, 5.0") ;  
}  
  
power_lut_template (output_by_cap2_and_trans) {  
    variable_1 : total_output_net_capacitance ;  
    variable_2 : input_transition_time ;  
    variable_3 : equal_or_output_net_capacitance ;  
    index_1 ("0.0, 5.0, 20.0") ;  
    index_2 ("0.1, 1.0, 5.0") ;  
    index_3 ("0.1, 0.5, 1.0") ;  
}
```

- The example at left shows shows four power_lut_template groups that have one-, two-, or three-dimensional templates
- The index values are lists of floating-point numbers greater than or equal to 0.0
- The values in the list must be in increasing order
- The number of floating-point numbers in the indexes determines the size of each dimension

Digital CAD

Internal Power Table for Cell Output



Calculating Switching Power

- Switching (or interconnect) power is the power dissipated in the circuit as a result of a logical transition of the capacitive load
- With internal power, switching power is used to compute the design's total dynamic power dissipation
- Switching power information is a function of a net's capacitive loading, associated clock frequency, and the supply voltage level of the design
- An explicit units attribute is not required for switching power, because the units are implicitly determined by the units of the voltage, time, and capacitance attributes

$$\text{Power_Units} = \frac{(1 \text{ V}^2) \times 0.1 \text{ ff}}{1 \text{ ns}} = 0.1 \text{ } \mu\text{W}$$

Switching Power Calculations

- For a single net with a total load of 100 femtofarad, a toggle rate of two transitions every 100 ns, and a supply voltage of 5 volts, the calculation of the net's power dissipation is:

$$\begin{aligned}\text{Net_Power} &= \frac{V_{dd}^2}{2} \sum_{\forall \text{nets}(i)} (C_{\text{Load}_i} \times TR_i) \\ &= \frac{5^2}{2} \times 100 \times \frac{2}{100} \times (0.1 \mu\text{W}) \\ &= 25 \times (0.1 \mu\text{W}) \\ &= 2.5 \mu\text{W}\end{aligned}$$

- TR
 - Toggle rate (number of toggles per unit of time)
- CLoad
 - Capacitive load of each net

Syllabus for Advanced Cell Characterization

- Review of Introduction to Cell Characterization
- Latches
- Scan Flop
- Gated Clocks
- Definition of I/O cell terms
- Example of I/O Cell
- Active Loads
- Active Drivers
- Derating factors, K factors
- Verilog Timing Checks
- Noise Considerations (CCS, ECSM)