Liberty Timing File

The .lib file is an ASCII representation of the timing and power parameters associated with any cell in a particular semiconductor technology

The timing and power parameters are obtained by simulating the cells under a variety of conditions and the data is represented in the .lib format

The .lib file contains timing models and data to calculate

- I/O delay paths
- Timing check values
- Interconnect delays

I/O path delays and timing check values are computed on a per-instance basis

Path delays in a circuit depend upon the electrical behavior of interconnects between cells
This parasitic information can be based on the layout of the design, but must be estimated when no layout information is available

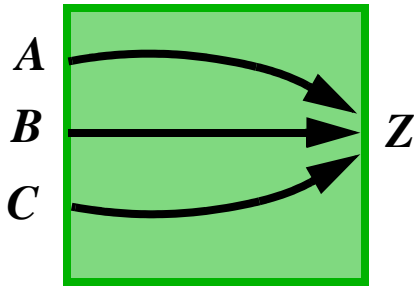
Also it is not possible to predict the process, voltage and temperature variations and derating factors can be included to compensate for these variations

Cell-Based Delay Calculation

Cell-based delay calculation is modeled by characterizing ***cell delay*** and ***output transition time (output slew)*** as a function of ***input transition time (input slew)*** and the ***capacitive load*** on the output of the cell.

Timing checks are also functions of input slew and output capacitive load

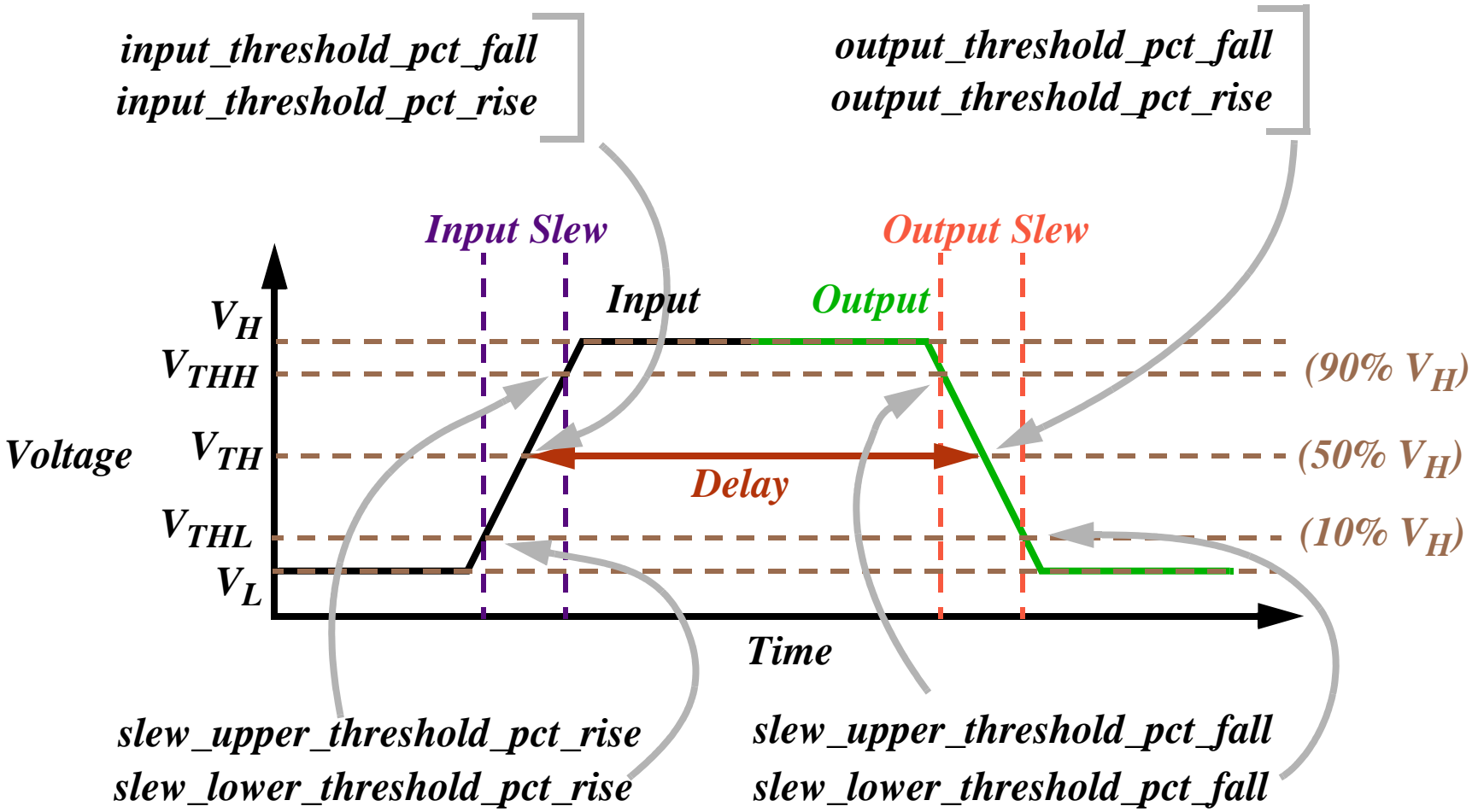
Each cell has a specific number of input-to-output paths



- Path delays can be described for each input signal transition that affects an output signal
- The path delay can also depend on signals at other inputs (state dependencies)
- In many sequential cells, the path delay from an input pin to an output pin can depend on the path delay from another output pin to this output pin

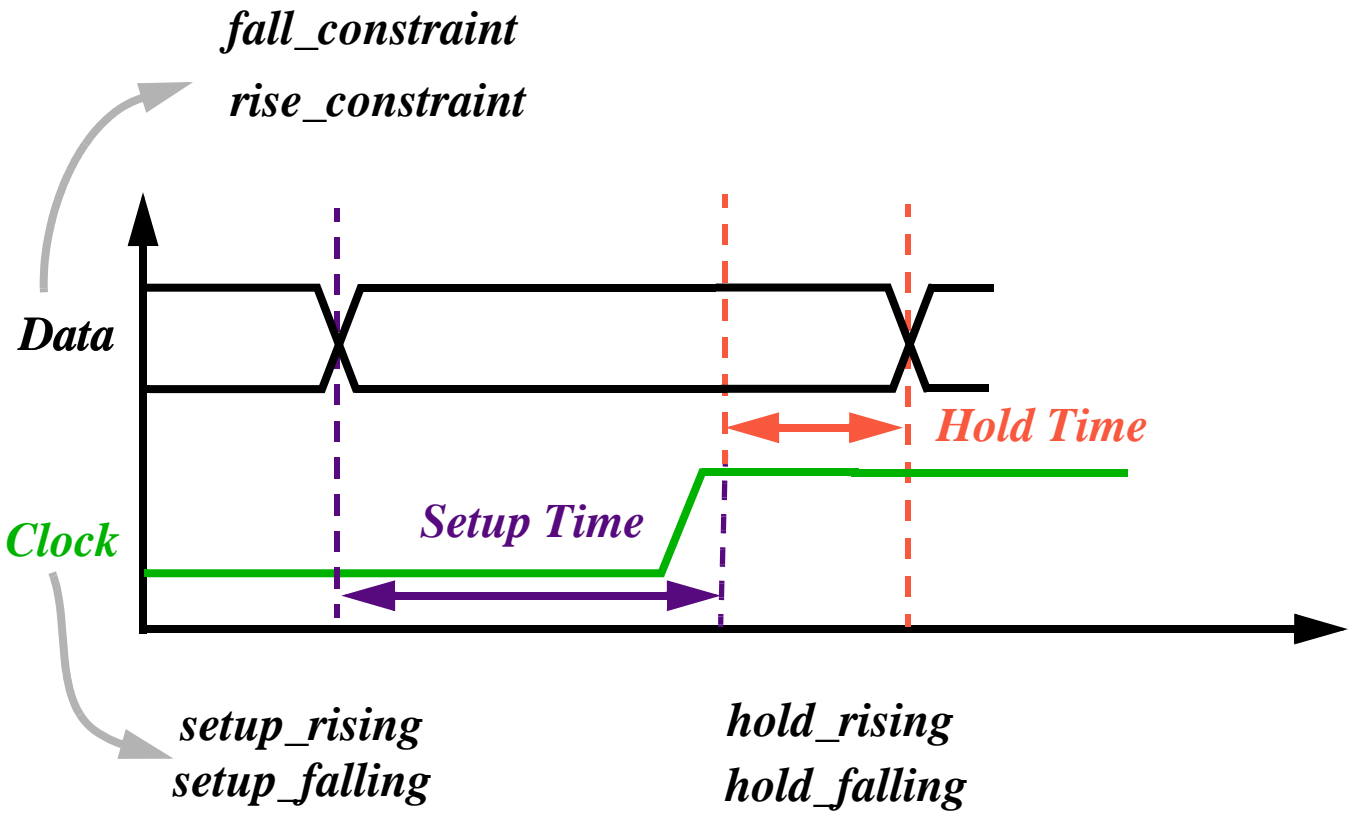
Delay Calculation and Timing Checks

Input-Slew, Output-Slew and Cell Delay



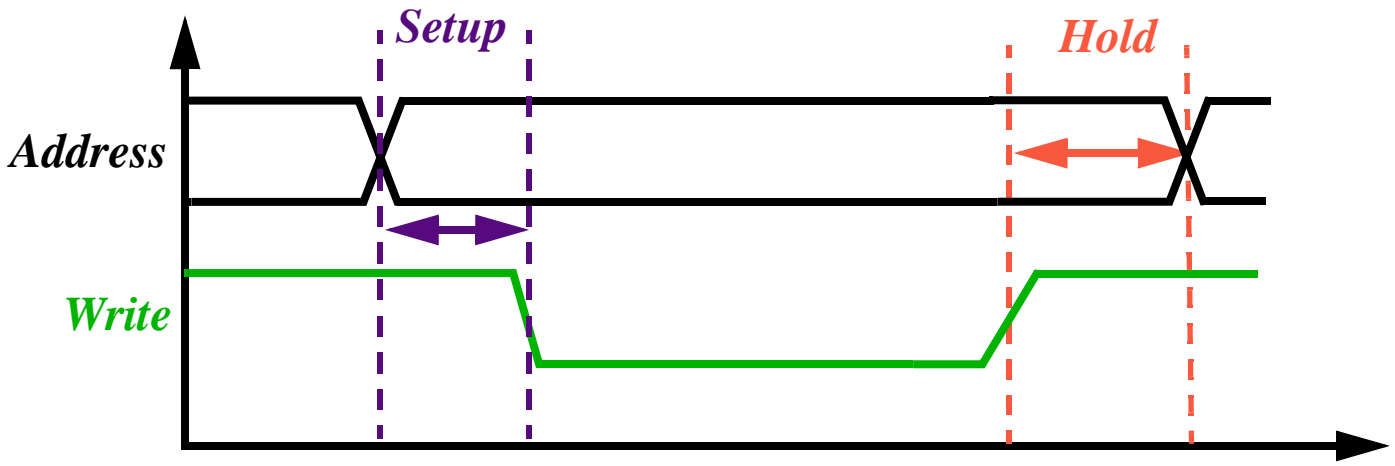
Timing Checks

Setup and Hold Time (Timing Checks)



Timing Checks

No-Change



nochange_low_low
nochange_high_low

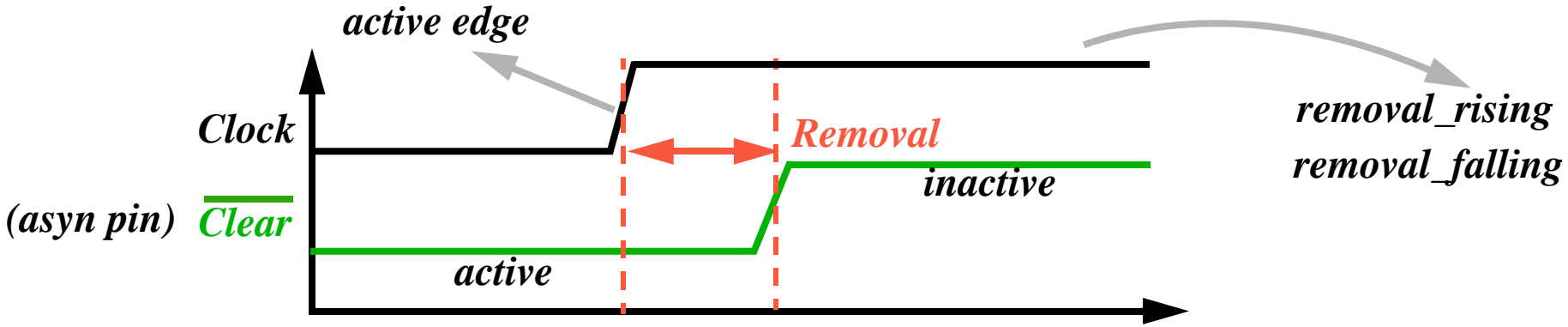
nochange_high_high
nochange_low_high

constrained pin (eg. address)

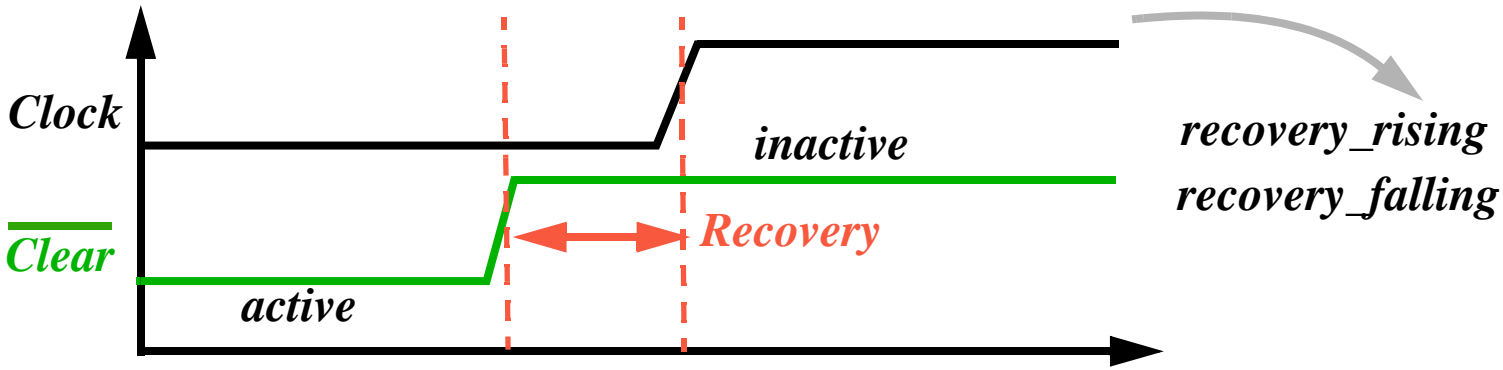
related pin (eg. write)

Timing Checks

Removal:



Recovery



Timing Library

What we will have and not have in our library?

Library Information

- Header information
- No wireload models
  - Prior design data is required to accurately generate these models
  - We will rather use tools use physical information during synthesis
- Operation conditions, derating factors, limits and units
  - Three different values are usually required: typical, worst and best case
  - However, to accurately get these three values process parameters and transistor models for the entire process spread are required
  - This information is only available to the foundry
  - We can perform simulations only with MOSIS provided models
  - Average extraction parameters and spice models will be used for the simulations
  - We can still run simulations at various temperatures and voltages
  - We can use +/- 5% or +/- 10% variations as best and worst case values
  - When using the library, keep in mind that you need to guard band for these variations

Timing Library

- Operation conditions, derating factors, limits and units (contd.)

library and delay_model

Provide a library name and the delay model to use. We will be using the table_lookup (non-linear delay) model

nom_process property

Specifies the reference points for process scaling used for the characterization of the cells. Our file will contain values for only one process point and so a 1.0 will be used However, we can create three different files for typical, worst and best.

nom_temperature and nom_voltage

Specifies the tempreatue and voltage reference points

operating_conditions

Defines the process, temperature and voltage values at the library level along with the default_operating_conditions

Timing Library

- Operation conditions, derating factors, limits and units (contd.)

slew and delay threshold points (previously discussed)

Low and high threshold values for slew calculation (10% - 90% points) and the threshold for delay calculations (50% points)

default values for fanout, capacitance, slew

Specifies the limits on maximum input slew on an input pin, input/inout pin capacitance and the maximum output capacitance on any output pin

units

Specifies the units used for time, capacitance, power, voltage, current etc.

derating factors and wire-load models

As previously discussed we need detailed process information for this as well as extracted parasitics from previous designs.

Timing Library

Lookup table templates

Define templates of common information to us in lookup tables. These are defined for timing arcs, power and timing checks that will be included in the cell definitions

Cell Definitions

*Cell(**cell_name**)*

The cell name

Area

Specifies the cell area, used during logic synthesis and timing analysis, no units

*<lookup tables>(**lookup_table_template_name**)*

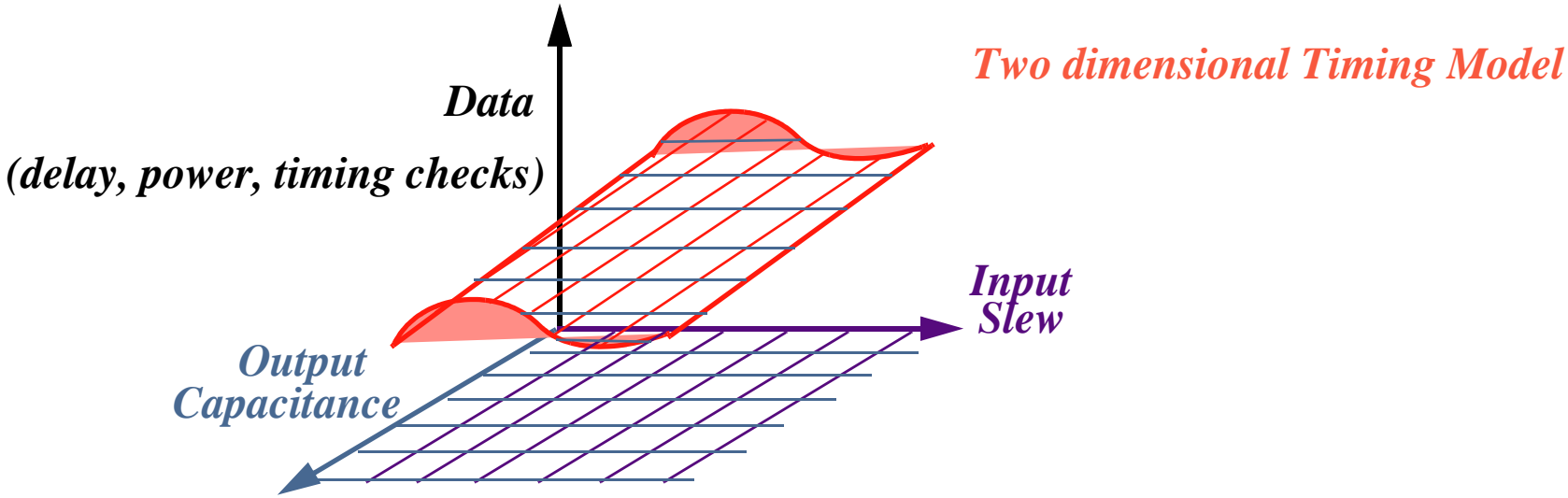
Specifies the timing models, power, timing checks to use for the particular path in the circuit

One, two or three dimensional models are used depending on the lookup-table being created

Timing Library

Two dimensional model

The two independent axis variables are input slew and output load capacitance



cell_fall(*fall_template_name* n x m)

index_1 (*value1, value2, ... , value n*)

index_2 (*value1, value2, ... , value m*)

values (\

$data\_max_{11}$:$data\_typ_{11}$:$data\_min_{11}$, ..., $data\_max_{1m}$:$data\_typ_{1m}$:$data\_min_{1m}$ \

..... \

$data\_max_{n1}$:$data\_typ_{n1}$:$data\_min_{n1}$, ..., $data\_max_{nm}$:$data\_typ_{nm}$:$data\_min_{nm}$);

Timing Library

pin(pin_name)

direction : *input, output, inout, internal*

clock_pin

function(*expression*)

Used for output or bidirectional pins. The expression defines the value of the output pin as a function of input pins

max_capacitance

The maximum output capacitive load that an output pin can drive

capacitance

The capacitive load of an input, inout, output or internal pin. Usually defined as 0 for output pins

internal_power()

Output pins in combinational cells, define the rise_power and fall_power to a related input pin. Input and clock pins also define this in sequential cells

timing()

Output pins in combinational cells, define the rise_delay, fall_delay, rise_transition and fall_transition to a related input pin

Timing Library

timing() contd.

Timing checks are also defined for sequential cells

pulse width definitions, recovery, removal

Required for clocks, asynchronous set and reset pins

Flip-flops and latches

Flip-flops and latches need to be defined using *ff* and *latch* groups

ff (<state of noninverting output>, <state of inverting output>) {

clocked_on: <clock pin name>

next_state: <input combination that produces the next state>

clear: <active value of the clear input>

preset: <active value of the preset input>

clear_preset_var1: <value of noninverting output when both active>

clear_preset_var2: <value of inverting output when both active> }

latch()

latch is similar but requires enable and data in, instead of clock and next state

Timing Library

Scan Cells

Requires a *test_cell* group to be defined along with the *ff* or *latch* group

Two *ff* groups need to be defined, one in the cell (function defined with testing features) and one inside the *test_cell* group (without the testing features)

test_cell(){

Inside the *test_cell* group all pins are defined and test related pins are given a *signal_type* or *test_output_only* attribute

signal_type can be:

test_scan_in: scan input pin

test_scan_in_inverted: inverted scan input

test_scan_out: scan output pin

test_scan_out_inverted: inverted scan output

test_scan_enable: high on this pin puts it in test mode (scan and shift)

test_scan_enable_inverted: same as above but inverted

test_scan_clock: test scan clock for clocked-scan

other clocks defined for LSSD scan