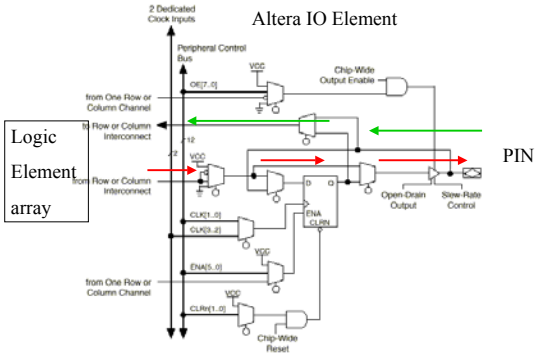
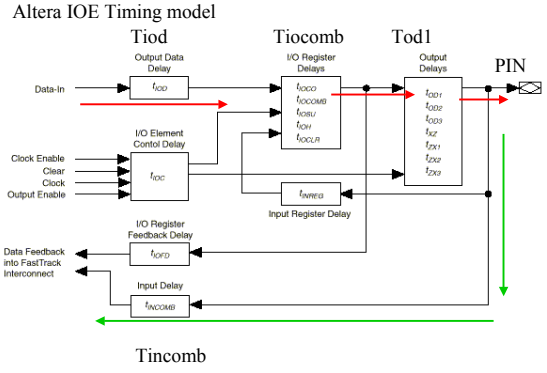
Altera IO Element



Altera IOE Timing model



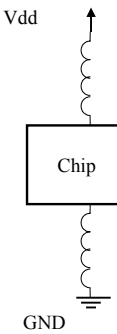
IOE Delays

- Input path
  - Tincomb - input pad and buffer to fasttrack interconnect delay
- Output path (combinatorial path with fast output slew)
  - Tiod - data delay
  - Tiocomb - combinatorial delay
  - Tod1 - slow rate = off, Vccio = Vccint (Vcc of IO pad is same as internal Vcc).

Aside: Why programmable Output slew?

- Slew rate is the measure of how fast an output can change value (measured in Volts/Sec).
- Most FPGA vendors offer the capability of programming the output to be either fast slew or slow slew ----- WHY?
  - Fast Slew rates cause more noise problems via ground bounce, especially when multiple outputs are switching
  - If you have room in your timing spec, should use slow slew rate if possible

GND Bounce



Large change of current on Vdd/Gnd pins (inrush current) due to multiple outputs changing simultaneously causes induced voltage on GND plane:

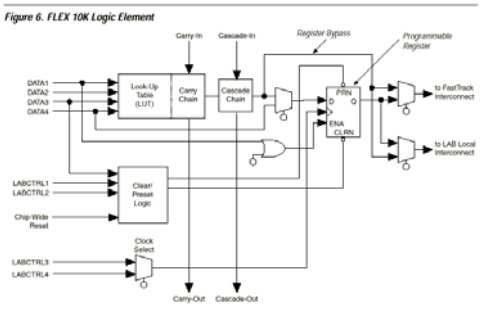
$v(t) = L * di/dt$

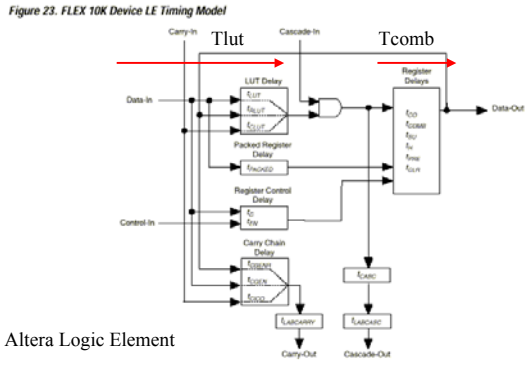
Larger the inductance, larger the change in current, larger the induced voltage.

Two ways to reduce Voltage:
Reduce Inductance : More vdd/gnd pins (inductance in parallel reduces total inductance), better packaging (different packages have more/less inductance than others).
Flex 10K20 240 pin package has 19 Vdd pins, 18 Gnd pins).

Reduce di/dt : slower slew rate!!!!

Altera Logic Element





Altera Logic Element

Minimum Pin To Pin Delay

[Input Pin delay] + [Logic Element Delay] + [Output Delay]

[Tincomb] + [Tlut + Tcomb] + [Tiod + Tiocomb + Tod1]

What about Routing Delays? Table 20 has routing delays.

Tdin2data - delay from dedicated input or clock to LE data

Tsamecolumn - delay from LE output to IOE in same column

Tsamerow - delay from LE output IOE in same row

Minimum Pin To Pin Delay

[Input Pin delay] + [Routing] + [Logic Element Delay] + [Routing] + [Output Delay]

[Tincomb] + [Tdin2data] + [Tlut + Tcomb] + [Minimum (same col,row)] + [Tiod + Tiocomb + Tod1]

[2.8] + [4.3] + [1.4 + 0.5] + min(1.4,3.7) + [1.3 +0.0 + 2.6]

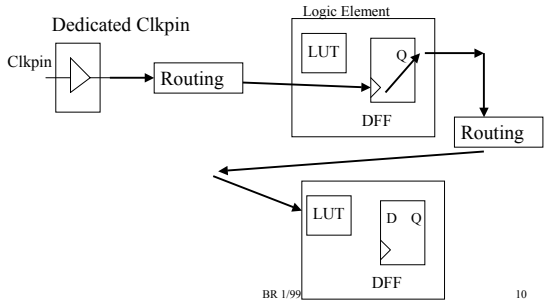
= 14.3 ns

if ignore routing, then 8.6 ns (this is what marketing will quote).

Note that same column routing much faster than row routing. (hence dedicated carry chains run in column routing).

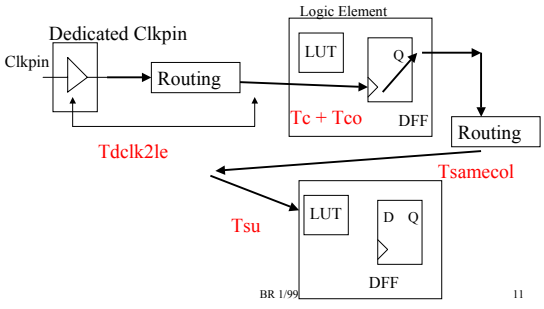
Minimum Register to Register

[Input Pin delay] + [Routing] + [Logic element clock-to-Q] + [Routing] + [Logic Element Delay] + [Routing] + [Logic Element Setup Time]



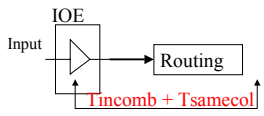
Minimum Register to Register

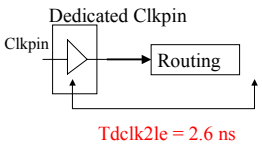
[Input Pin delay] + [Routing] + [Logic element clock-to-Q] + [Routing] + [Logic Element Setup Time]



Dedicated Inputs/Clock Pins vs IOE inputs

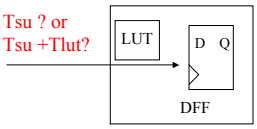
A dedicated input pin or dedicated clock pin does not have the IOE logic. The input timing is specified as routing delay only:





Use dedicated input pins to minimize input delay. Not many on device - 10K20 240 pin package only has 4 dedicated inputs and 2 dedicated clock pins.

Setup Time for Logic Element

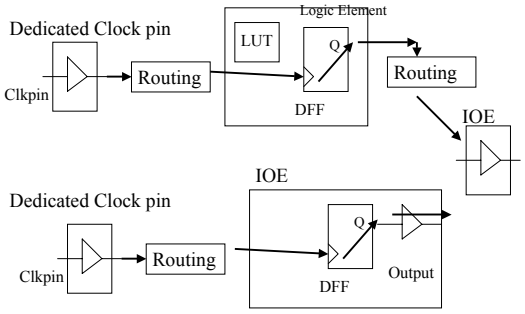


Typically, the setup time specification for an external data input already accounts for the LUT delay since the data input has to pass through the LUT on its way to the D input.

The altera spec is a bit confusing - my best guess is that Tsu includes the LUT delay. There is no doubt that the Xilinx Virtex Tsu spec includes the LUT delay.

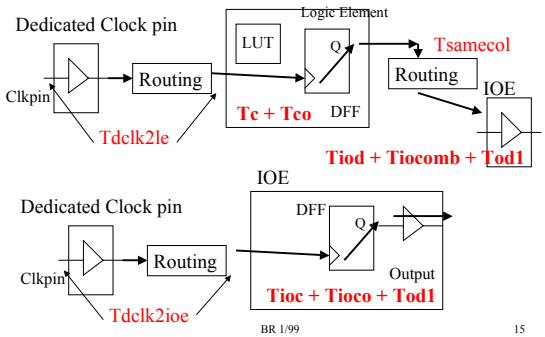
Clock To Out

Two different Choices here - is the Dff in the LUT or the IOE??



Clock To Out

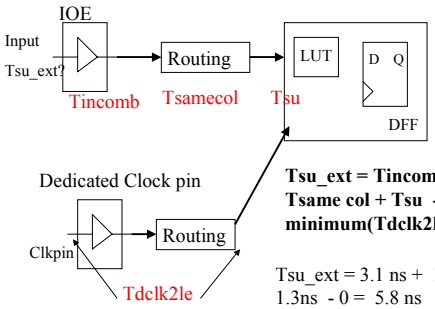
Two different Choices here - is the Dff in the LUT or the IOE??



Latching in IOE or LE?

- The DFF in the IOE can be configured to either latch incoming data or outgoing data
  - Can latch ingoing/outgoing data in either IOE or LE (logic element)
- Using the DFF in the IOE to latch outgoing data will usually reduce Clock-2-Out time
  - DFF is closer to the Pin!
- Using the DFF in the IOE to latch ingoing data will reduce external setup time.
  - DFF is closer to the Pin!

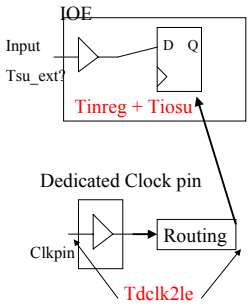
Minimum External Setup Time Data latched in LE



Tsu_ext = Tincomb + Tsame col + Tsu - minimum(Tdclk2le)

Tsu_ext = 3.1 ns + 1.4ns + 1.3ns - 0 = 5.8 ns

Minimum External Setup Time Data latched in IOE

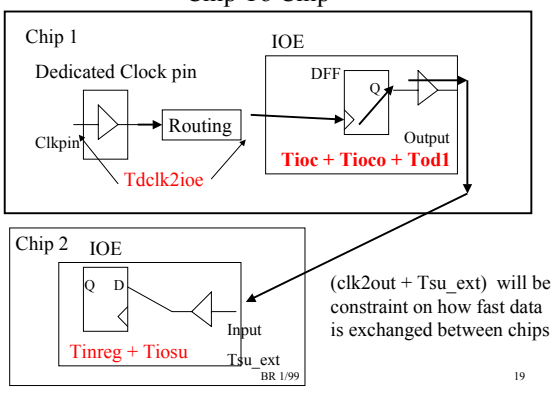


Tsu_ext = Tinreg + Tiosu - minimum(Tdclk2le)

Tsu_ext = 6.0 ns + 2.8 - 0 = 8.8 ns

!! Latching in IOE slower than in Logic Element? These are all worse case numbers in the datasheet which could account for this; also mentioned on page 28 that latching in LE element will sometimes give better setup time than an IOE. For other FPGA families this is usually not the case.

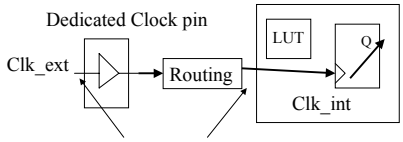
Chip To Chip



(clk2out + Tsu_ext) will be constraint on how fast data is exchanged between chips

PLL effects

PLL/DLL will synchronize internal clock to external clock. Aim is to have zero delay between clock edges at Logic elements and external clock edge



Want a 'zero-delay' clock, no difference in edge arrival times of clock edges at 'Clk_ext' and 'Clk_int'.