Washington University in St. Louis — Engineering

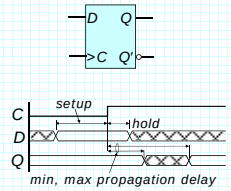
Timing Issues in Digital Circuits

- Setup and hold time constraints
- Input timing constraints
- Clock period analysis
- Metastability and synchronizer reliability

Washington University in St. Louis — Engineering

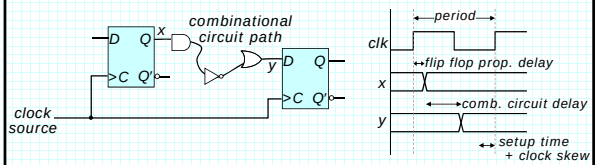
Edge-Triggered D Flip Flop

- *D* flip flop stores value at *D* input when clock rises
- Most widely used storage element for sequential circuits
- *Propagation time* is time from rising clock to output change
- If input changes when clock rises, new value is uncertain
  - » output may oscillate or may remain at intermediate voltage (*metastability*)
- Timing rules to avoid metastability
  - » *D* input must be stable for *setup time* before rising clock edge
  - » must remain stable for *hold time* following rising clock edge



Washington University in St. Louis — Engineering

Implications of Setup Time Constraints



- To avoid setup time violations, require
  period ≥ (max FF prop. delay) + (max comb. circuit delay) + (FF setup time) + (max clock skew)
- CAD tools can check all FF-to-FF paths to verify
  - » both component delays and wiring delays matter
  - » accurate estimate requires component locations and information about routing of wires

Washington University in St. Louis — Engineering

Clock Period Analysis from Synthesis

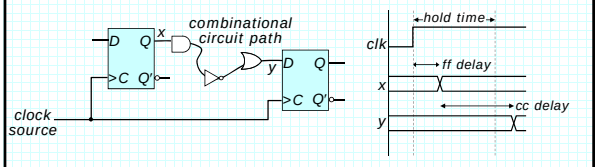
```
================================================================================
Timing constraint: Default period analysis for Clock 'clk'
  Clock period: 4.227ns (frequency: 236.560MHz)
  Total number of paths / destination ports: 45 / 5
--------------------------------------------------------------------------------
Delay:               4.227ns (Levels of Logic = 3)
  Source:            state_FFd1 (FF)
  Destination:       cnt_2 (FF)
  Source Clock:      clk rising
  Destination Clock: clk rising

  Data Path: state_FFd1 to cnt_2
                                Gate     Net
    Cell:in->out      fanout   Delay    Delay   Logical Name (Net Name)
    ----------------------------------------  ------------
     FDR:C->Q             9    0.626    1.125   state_FFd1 (state_FFd1)
     LUT2:I1->O           1    0.479    0.740   _mux0001<2>20_SW0 (N123)
     LUT4_L:I2->LO        1    0.479    0.123   _mux0001<2>24_SW0 (N119)
     LUT4:I3->O           1    0.479    0.000   _mux0001<2>43 (_mux0001<2>)
     FDS:D                     0.176            cnt_2
    ----------------------------------------  ------------
    Total                      4.227ns (2.239ns logic, 1.988ns route)
                               (53.0% logic, 47.0% route)

================================================================================
```

synthesis estimates only; placement and routing information needed for accurate analysis

FF setup time

Washington University in St. Louis — Engineering

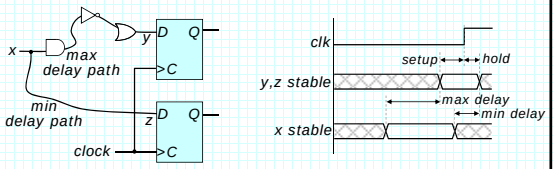
Implications of Hold Time Constraints



- To avoid hold time violations, require
  hold time ≤ (*min* FF prop. delay) + (*min* comb. circuit delay) – (max clock skew)
- CAD tools can check all FF-to-FF paths to verify
- In FGPAs, it is often the case that
  hold time < (min FF prop. delay) – (max clock skew)
  so, hold time violations cannot occur

Washington University in St. Louis — Engineering

Input Timing



- Setup and hold times constrain when inputs to a circuit can change
  - » stable period starts at clock – (setup + max delay)
  - » and lasts until clock + hold – (min delay)
- Common simplification is to hold input stable from clock – (setup + max delay) until clock

Input Delay Analysis from Synthesis

```
================================================================================
Timing constraint: Default OFFSET IN BEFORE for Clock 'clk'
  Total number of paths / destination ports: 17 / 12
--------------------------------------------------------------------------------
Offset:              4.356ns (Levels of Logic = 4)
  Source:            dIn (PAD)
  Destination:       cnt_2 (FF)
  Destination Clock: clk rising

  Data Path: dIn to cnt_2
                                Gate     Net
    Cell:in->out      fanout   Delay    Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     IBUF:I->O             7   0.715    1.201  dIn_IBUF (dIn_IBUF)
     LUT4:I0->O            1   0.479    0.704  _mux0001<2>33 (_mux0001<2>_map1
     LUT4_L:I3->LO         1   0.479    0.123  _mux0001<2>24_SW0 (N119)
     LUT4:I3->O            1   0.479    0.000  _mux0001<2>43 (_mux0001<2>)
     FDS:D                     0.176           cnt_2
    ----------------------------------------
    Total                     4.356ns (2.328ns logic, 2.028ns route)
                                      (53.4% logic, 46.6% route)
================================================================================
```

dIn input should be stable, from 4.356 ns before clock edge, until clock edge

Review Questions

1. Consider a flip flop with a setup time of 3 ns and a hold time of 1 ns. If the clock input rises at time t, is it ok for the data input to change at time $t-2$? What about $t-4$? What about $t+1/2$? What about $t+2$? Explain why it is not acceptable for the data input to change at certain times.
2. Consider the clock period analysis on page 4. How would the clock period change if the flip flop propagation delay was 1 ns instead of .626 and the gate delay for the LUTs was 0.6 ns, instead of 0.479? (You may assume that the net delays don't change.)
3. Consider a circuit in which there is a path from an input x to a flip flop that has a maximum possible delay of 7 ns, and there is also a path from x to another flip flop with a minimum delay of 3 ns. If the setup and hold times are 2 ns and 1 ns respectively and the clock input rises at time t, is it ok for x to change at time $t-5$? What about $t-1$? What about t? What about $t-10$?

Question

1. Consider a flip flop with a setup time of 5 ns and a hold time of 3 ns. The clock input rises at time 20 ns.

   What it the latest time that the D input can change prior to the clock edge to ensure proper timing?

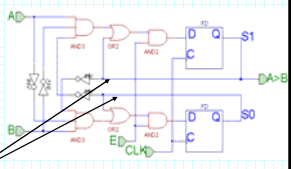
   A. 15 ns B. 17 ns C. 19 ns D. 21 ns E. 23 ns

Timing Analysis Procedure

1. Check for internal hold time violations
   - » for every ff-to-ff path, check
     (***minimum*** ff prop. delay) + (***minimum*** comb. circuit delay)
     ≥ (hold time) + (clock skew)
     omit skew for paths from output to input of same ff
   - » fix violations by adding delay
   - » no violations possible if hold-time<(min-ff-prop-delay)–skew
2. Determine minimum clock period
   - » find ff-to-ff path with largest value of
     (***maximum*** ff prop. delay) + (***maximum*** comb. circuit delay)
     + (setup time) + (clock skew)
     omit skew for paths from output to input of same ff
3. Input timing analysis
   - » each input must be stable from
     (clock_edge) – ((***maximum*** input-to-ff delay) + (setup time))
     to ((clock_edge) + (hold time)) – (***minimum*** input-to-ff delay)
4. Timing analysis for synchronous outputs
   - » synchronous outputs have potential to change any time from
     (clock_edge) + (***minimum*** clock-to-output delay)
     to (clock_edge) + (***maximum*** clock-to-output delay)

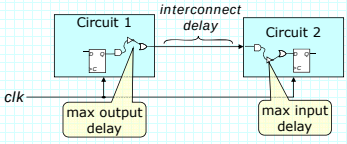
Timing Analysis of Sequential Comparator

- Timing parameters
  - » gate delay: 0.25 to 1 ns
  - » ff setup time: 2 ns
  - » ff hold time: 1 ns
  - » ff prop. delay: 0.5-2 ns
  - » clock skew: 1 ns
- Internal hold time violation?
  - » yes - .5 + 4(.25) < 1 + 1
  - » add inverter pair to feedback paths from ffs



- Minimum clock period - 2 + 6×1 + 2 + 1 = 11 ns or 90 MHz
- Input timing requirements
  - » *A* and *B* must be stable from (*clock_edge* – 2) – 4×1 until (*clock_edge* +1) – 3 ×.25, so from -6 ns to +.25
- Output timing - outputs can change .5 to 2 ns after clock

Combining Circuits



- When combining two components, check setup constraints manually

  clock period ≥ (max output delay) + (max input delay)
  + (max inter-connect delay) + skew
- Note, skew much larger across different components than within a single component
- Hold time violations unlikely across components
  - » inter-chip delays much larger than (hold time)+skew

Dealing with Timing Failures

- To fix hold time errors, add delay
  - » rarely issue in FPGAs, but can be in ASIC designs
- To correct setup time failures
  - » if you can't increase clock period, must reduce delay
  - » find long delay paths and modify circuit to reduce
  - » adjust synthesis/implementation properties
    - focus on speed optimization, increase effort level
  - » study synthesis report to identify worst-case paths
    - rewrite VHDL to produce faster circuit
    - e.g. replace ripple-carry circuits with carry lookahead
    - if need be (and feasible), insert pipeline registers to divide long combinational paths into smaller parts

Question

1. Consider a flip flop with a setup time of 5 ns and a hold time of 3 ns. The clock input rises at time 20 ns.

   What it the earliest time that the D input can change after the clock edge to ensure proper timing?

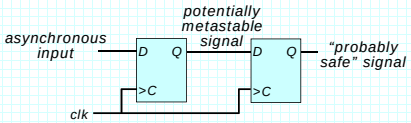
   A. 15 ns B. 17 ns C. 19 ns D. 21 ns E. 23 ns

Metastability

- Most digital systems have asynchronous inputs
  - » keyboard input on a computer,
  - » sensor on a traffic light controller,
  - » card insertion on an ATM, etc.
- Asynchronous inputs change at unpredictable times
  - » so, can change during clock transition, causing *metastability*
- Output of a metastable flip flop can oscillate or remain at intermediate value
  - » leads to unpredictable behavior in other flip flops
  - » metastability usually ends quickly, but *no definite time limit*
  - » so, circuit failures due to metastability *are unavoidable*
  - » however, systems can be designed to make failures rare

Synchronizers

- *Synchronizers* are used to isolate metastable signals until they are "probably safe"

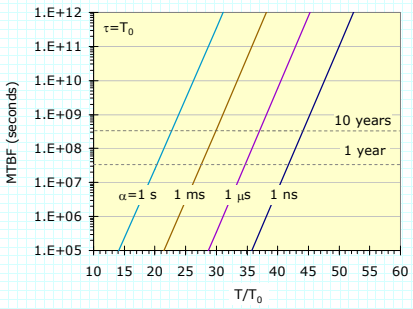


- If the clock period is long enough, failure probability is small and expected time between failures is large

  MTBF = Mean Time Between Failures $\approx (\alpha T/T_0)e^{T/\tau}$

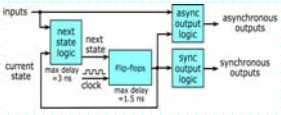
  where T is the clock period, α is the average time between asynchronous input changes, τ and T_0 are parameters of the flip flop being used
- If T = 50 ns, α = 1 ms, τ = 1 ns, T_0 = 1 ns, MTBF ≈ 8 trillion years, if T = 10 ns, MTBF becomes 220 seconds!

MTBF Chart

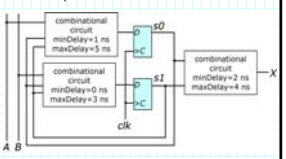


Exercises

1. Consider the generic state machine shown below with the indicated propagation delays. If the setup time for the flip flops is 1.5 ns and the maximum clock skew is .5 ns, what is the smallest clock period for which the circuit is guaranteed to work correctly?



2. For the state machine shown below, assume that the flip flop setup time is 2 ns, the hold time is 0.5 ns and the flip flop propagation delay is between 1 and 3 ns. Also, that the clock skew is 0.3 ns.



Is this circuit subject to internal hold time violations? Justify your answer. What is the smallest clock period for which the circuit is not subject to setup time violations? Be sure to take into account any modifications from the previous step. What is the latest time relative to the clock, when it is safe for input *B* to change? What is the latest time after the clock when output *X* can be changing?

3. Consider a synchronizer used to synchronize an asynchronous input signal. Let the average time between changes of the input signal be 50 microseconds. Let the flip flop parameters be T_0 = 3 ns and τ = 2 ns. If the clock period for the synchronizer is 10 ns, what is the mean time between synchronizer failures? What is the smallest clock period (to the nearest ns) for which the mean time between failures is 10 years? What is the smallest clock period for which the mean time between failures is 10,000 years?

Solutions

1. The minimum clock period is 1.5+3+1.5+0.5=6.5 ns.
2. There are no hold time violations because the minimum flip flop propagation delay is larger than the hold time plus the skew.
   The maximum delay for the next state logic is 5 ns. This gives a minimum clock period of 3+5+2+0.3=10.3 ns.
   Input *B* has a maximum delay path of 5 ns, so it must be stable by 3+2=5 ns before the clock rises.
   The latest time after the clock when output *X* can be change is 3+4=7 ns.
3. $\alpha = 50 \times 10^{-6}$, $T_0 = 3 \times 10^{-9}$ and $\tau = 2 \times 10^{-9}$, so for $T = 10 \times 10^{-9}$, MTBF = $167 \times 10^{-6} e^5 = 24{,}735 \times 10^{-6}$ seconds or about 25 milliseconds. By trial and error, one finds that for T=53 ns the MTBF is about 9.2 years and for T=54 ns, it is about 15 years, so we need a target clock period of about 54 ns to get a 10 year MTBF. Also, by trial and error, one finds that for T=66 ns the MTBF is about 7,600 years and for T=67 ns, it is about 12,800 years, so we need a target clock period of about 67 ns to get a 10,000 year MTBF. So, adding 13 ns has improved the reliability by a factor of 1,000.