Advanced Digital Design

Unit-5

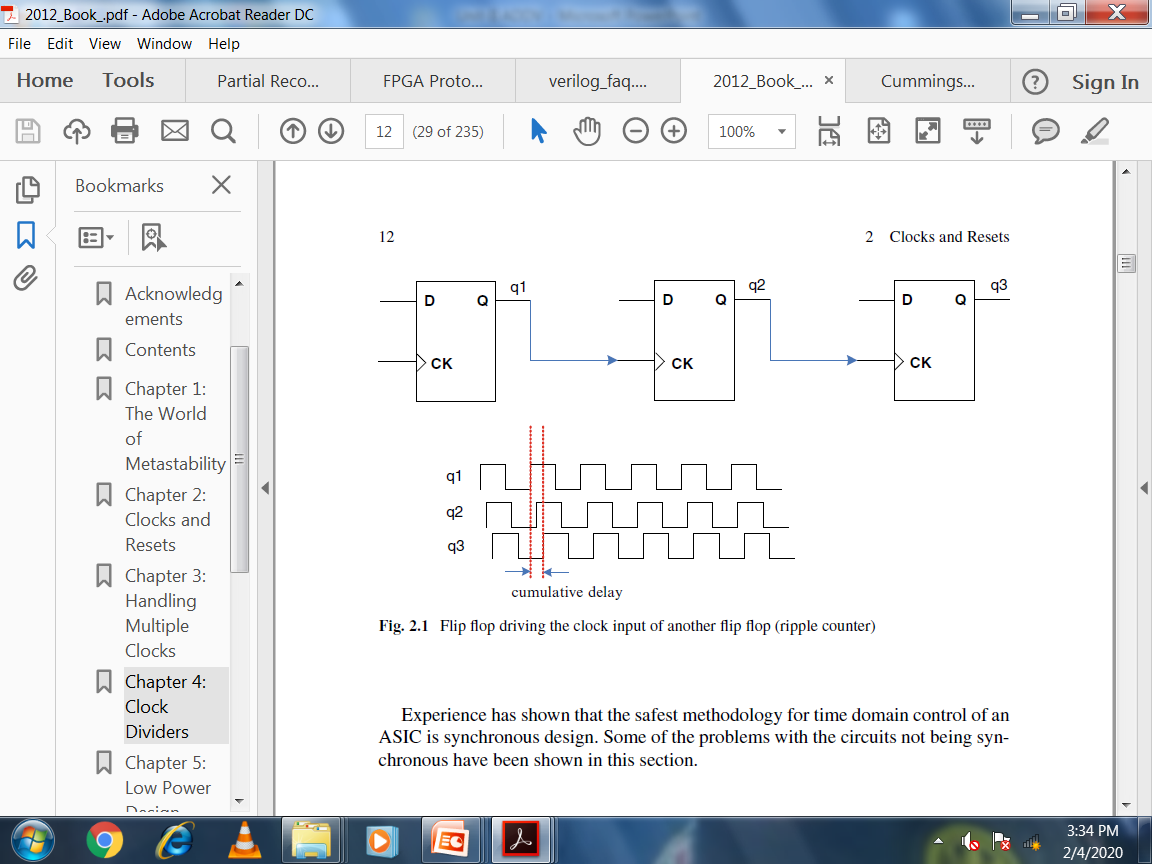
Question Answers

**Lecture -1: Clock Generation**

1. Describe why ripple counters should be avoided for clock generation/clock divider.

Answer:

* Flip Flops driving the clock input of other flip flops is somewhat problematic. The clock input of the second flip-flop is skewed by the clock-to-q delay of the first flip-flop, and is not activated on every clock edge. This cumulative effect with more than two Flip Flops connected in a similar manner forms a Ripple counter.



1. List the techniques applied to minimize the clock skew.

Answer:

* Adding Delay in Data Path
* Clock Reversing.
* Alternate Phase Clocking.
* Balancing Trace Length

**Lecture -2: Resets and Metastability**

1. Describe the recommendations for avoiding Metastability.

Answer:

* Use Synchronizers.
* Use Faster Flip Flops (narrower metastable window TW).
* Use metastable hardened Flip Flops (designed for very high bandwidth and reduced sampling times that are optimized for clock domain input circuitry).
* Cascade flip-flops as Synchronizers (two or more)
* Reduce Sampling rate.
* Avoid input signals with low dV/dt.

**Lecture -3: Introduction to Pipelining**

1. Describe the importance of Fan out of four (FO4)

Answer: In [digital electronics](https://en.wikipedia.org/wiki/Digital_electronics), Fan-out of 4 is a measure of time used in digital [CMOS](https://en.wikipedia.org/wiki/CMOS) technologies: the [gate delay](https://en.wikipedia.org/wiki/Propagation_delay#Electronics) of a component with a [fan-out](https://en.wikipedia.org/wiki/Fan-out) of 4.

Fan out = Cload / Cin, where

Cload = total MOS [gate capacitance](https://en.wikipedia.org/wiki/Gate_capacitance) driven by the logic gate under consideration

Cin = the MOS gate capacitance of the logic gate under consideration

As a delay metric, one FO4 is the delay of an [inverter](https://en.wikipedia.org/wiki/Inverter_(logic_gate)), driven by an inverter 4x smaller than itself, and driving an inverter 4x larger than itself. Both conditions are necessary since input signal rise/fall time affects the delay as well as output loading.

FO4 is generally used as a delay metric because such a load is generally seen in case of tapered buffers driving large loads, and approximately in any logic gate of a logic path sized for minimum delay. Also, for most technologies the optimum fanout for such buffers generally varies from 2.7 to 5.3.

Advanced Digital Design

Unit-4

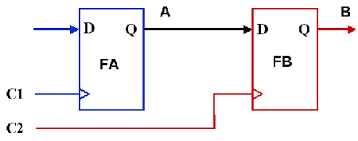
Question Answers

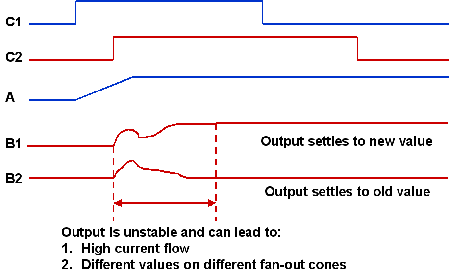
**Lecture -1: - Synchronization Fundamentals**

1. Explain the problem of Clock Domain Crossing with neat waveforms.

Answer:

* If the transition on signal A happens very close to the active edge of clock C2, it could lead to setup or hold violation at the destination flop “FB”. As a result, the output signal B may oscillate for an indefinite amount of time. Thus the output is unstable and may or may not settle down to some stable value before the next clock edge of C2 arrives. This phenomenon is known as **metastability** and the flop “FB” is said to have entered a metastable state.

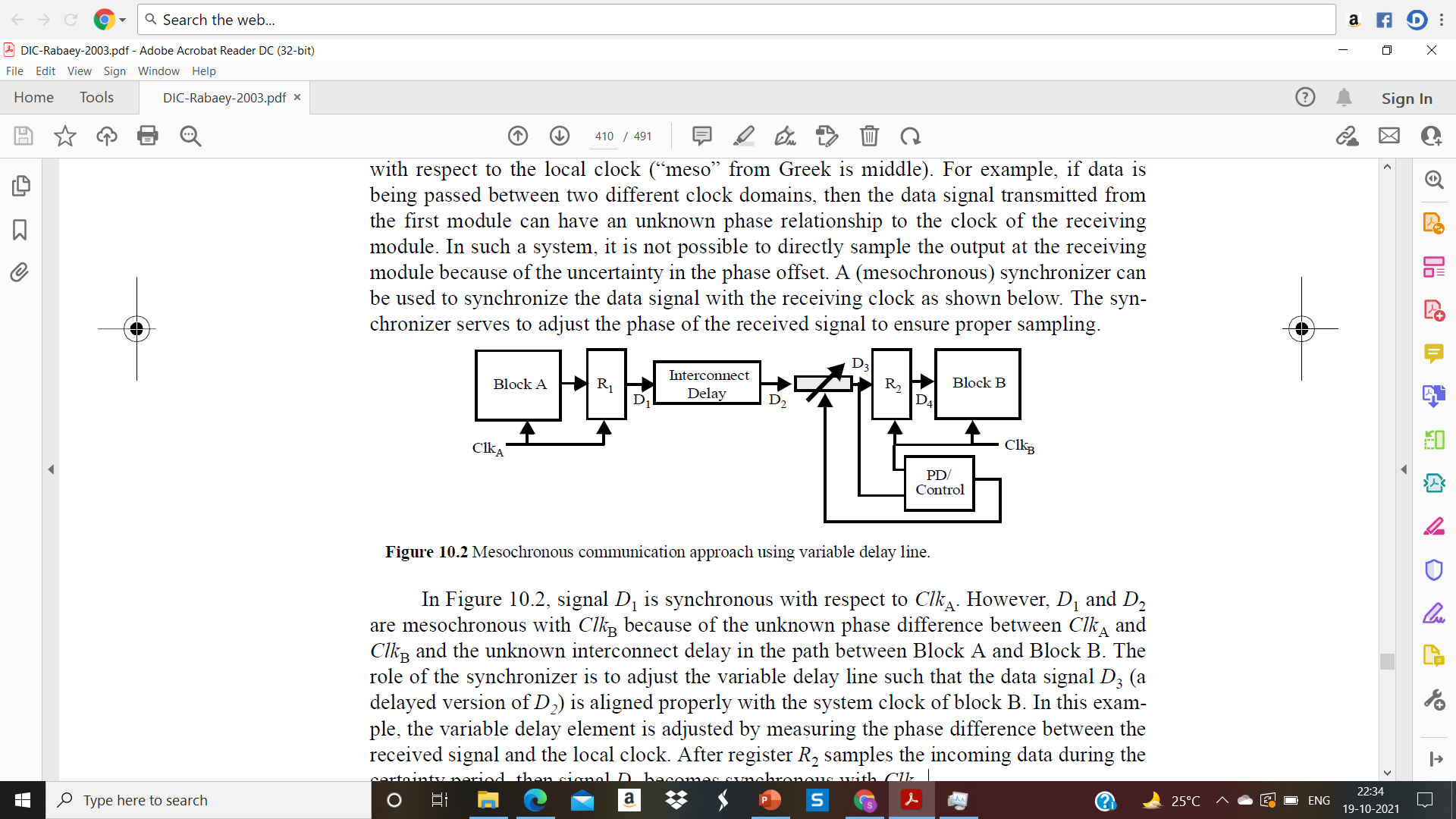




1. Explain the following: -
   1. Mesochronous Interconnect

Answer:

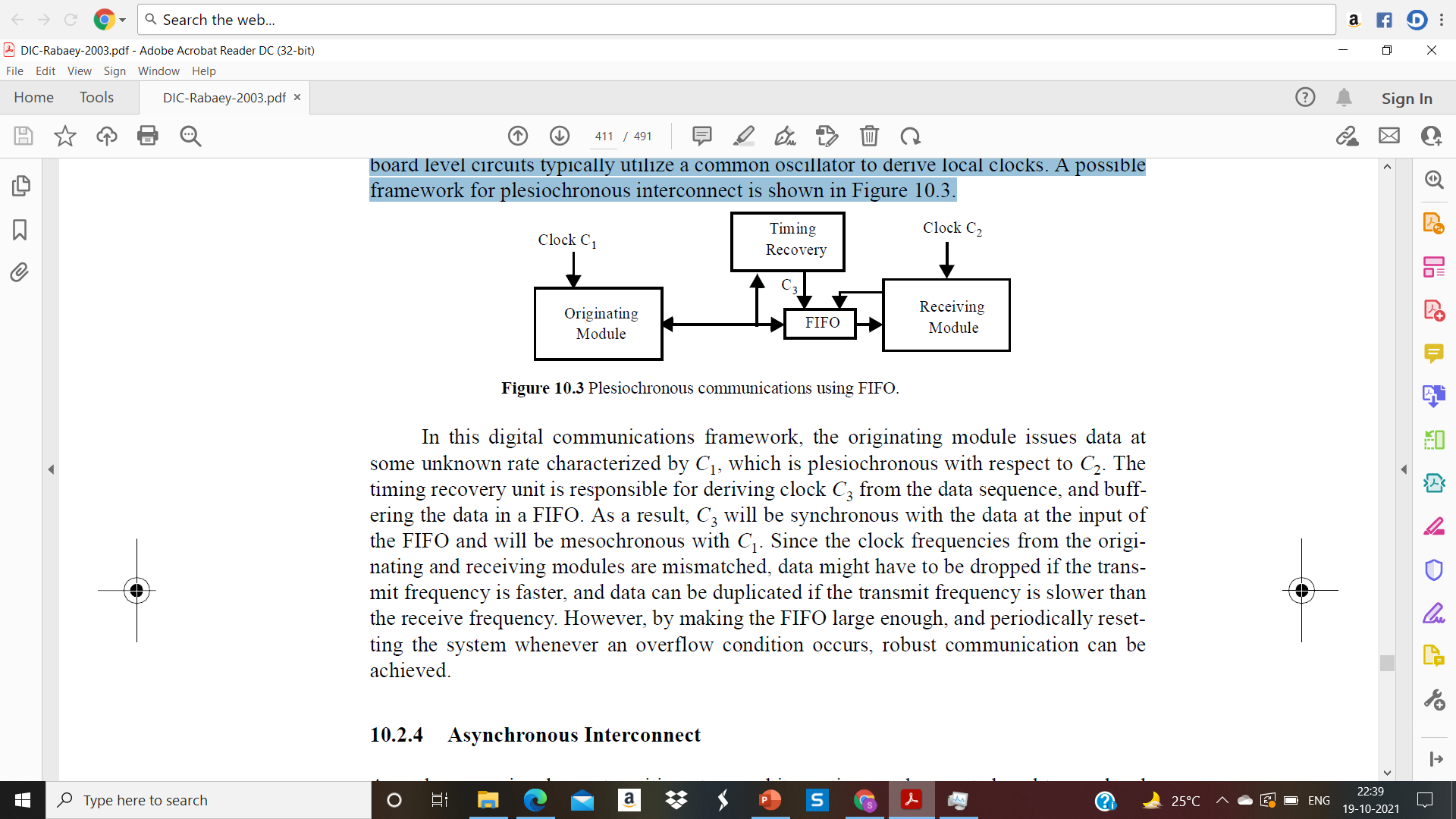
* A mesochronous signal is one that has the same frequency but an unknown phase offset with respect to the local clock (“meso” from Greek is middle).
* For example, if data is being passed between two different clock domains, then the data signal transmitted from the first module can have an unknown phase relationship to the clock of the receiving module.
* In such a system, it is not possible to directly sample the output at the receiving module because of the uncertainty in the phase offset.
* A (mesochronous) synchronizer can be used to synchronize the data signal with the receiving clock. The synchronizer serves to adjust the phase of the received signal to ensure proper sampling.



* In signal D1 is synchronous with respect to ClkA. However, D1 and D2 are mesochronous with ClkB because of the unknown phase difference between ClkA and ClkB and the unknown interconnect delay in the path between Block A and Block B.
* The role of the synchronizer is to adjust the variable delay line such that the data signal D3 (a delayed version of D2) is aligned properly with the system clock of block B.
* In this example, the variable delay element is adjusted by measuring the phase difference between the received signal and the local clock. After register R2 samples the incoming data during the certainty period, then signal D4 becomes synchronous with ClkB.
  1. Plesiochronous Interconnect

Answer:

* A plesiochronous signal is one that has nominally the same, but slightly different frequency as the local clock (“plesio” from Greek is near). In effect, the phase difference drifts in time.
* This scenario can easily arise when two interacting modules have independent clocks generated from separate crystal oscillators. Since the transmitted signal can arrive at the receiving module at a different rate than the local clock, one needs to utilize a buffering scheme to ensure all data is received.

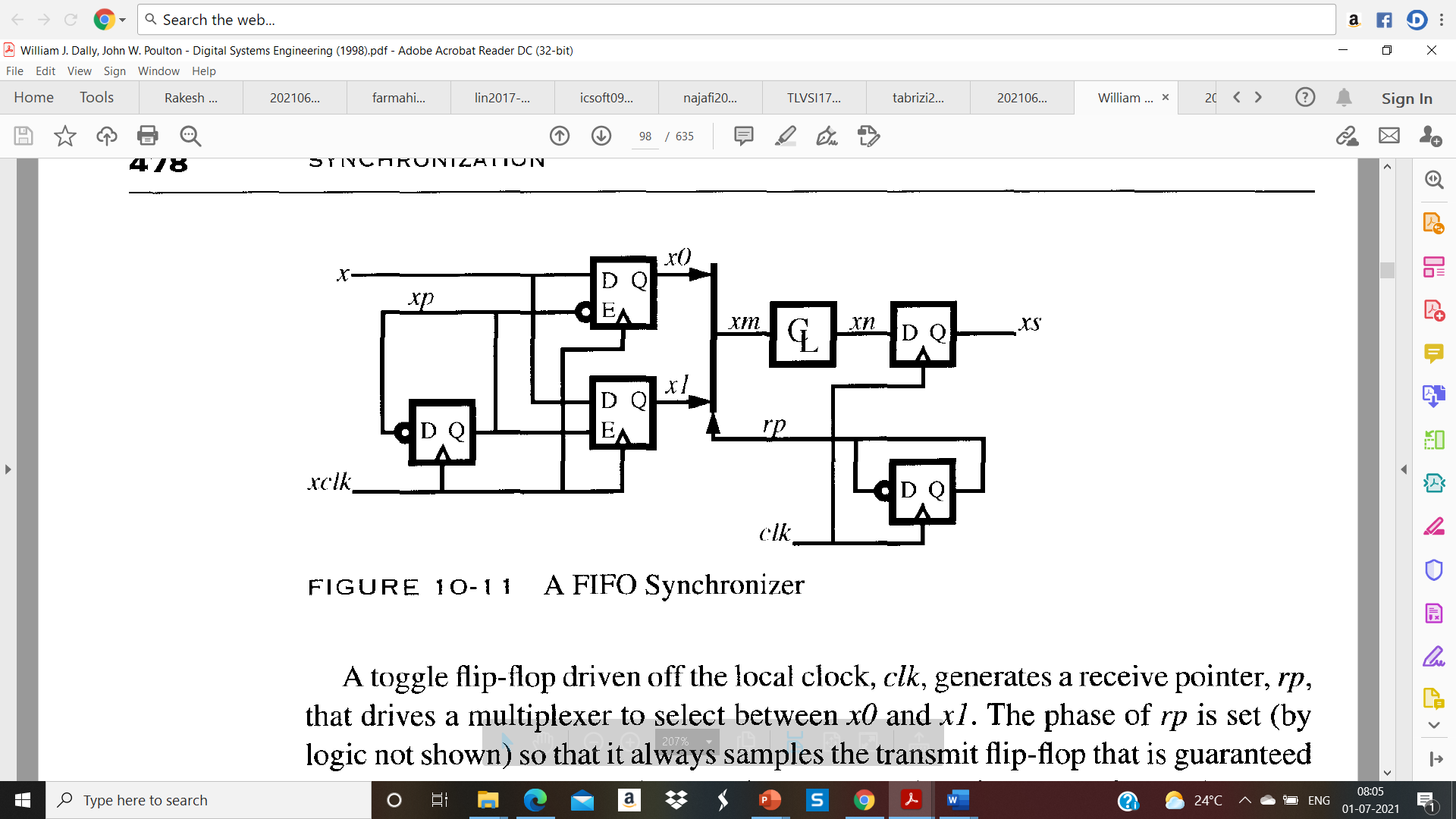


* In this digital communications framework, the originating module issues data at some unknown rate characterized by C1, which is plesiochronous with respect to C2.
* The timing recovery unit is responsible for deriving clock C3 from the data sequence, and buffering the data in a FIFO. As a result, C3 will be synchronous with the data at the input of the FIFO and will be mesochronous with C1.
* Since the clock frequencies from the originating and receiving modules are mismatched, data might have to be dropped if the transmit frequency is faster, and data can be duplicated if the transmit frequency is slower than the receive frequency.
* However, by making the FIFO large enough, and periodically resetting the system whenever an overflow condition occurs, robust communication can be achieved.

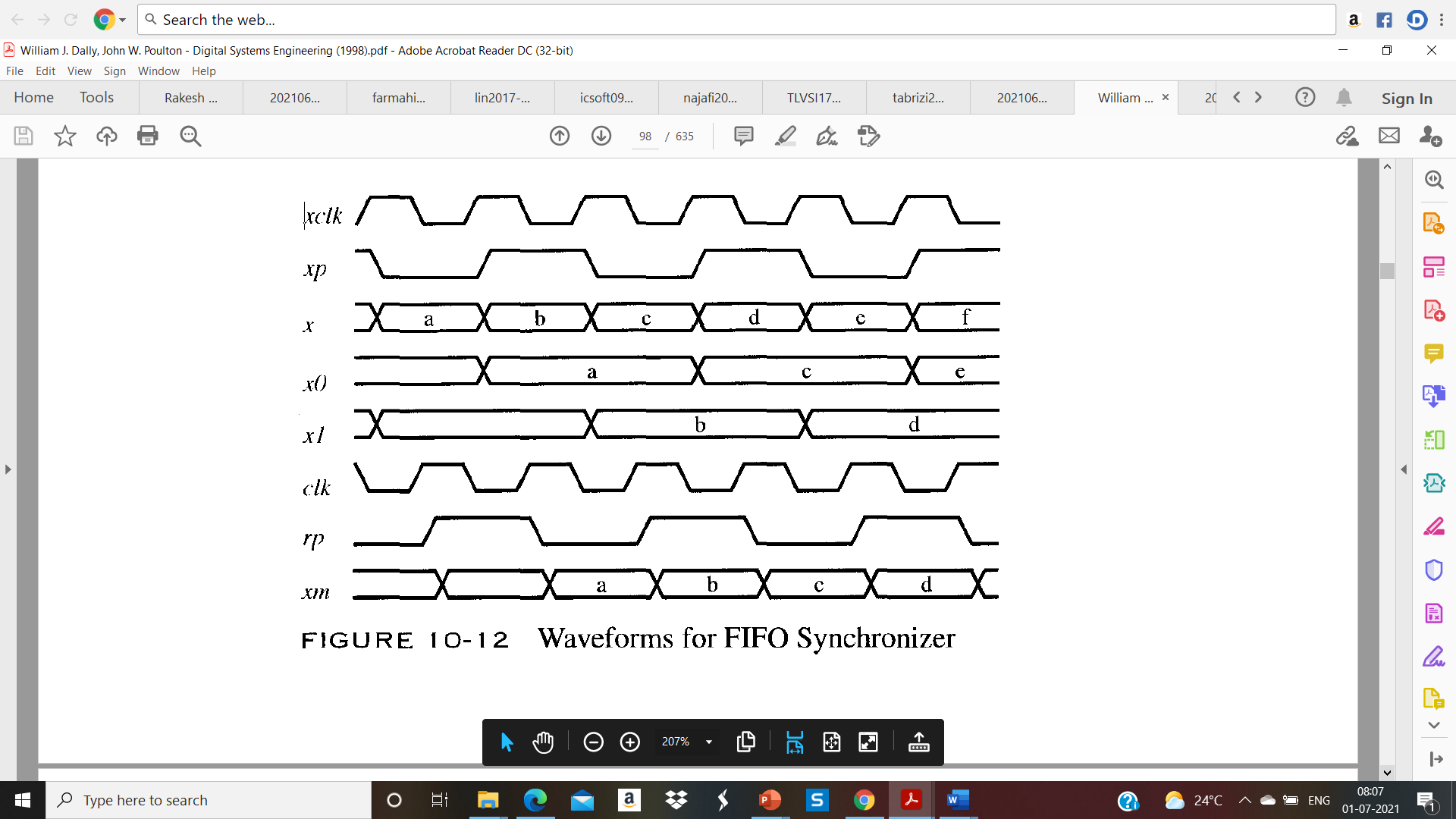
**Lecture -2: Synchronization Design**

1. With neat circuit diagram and waveforms, explain the FIFO synchronization for mesochronous interconnect.

Answer:



* A FIFO synchronizer uses a small ring-buffer to decouple the transmitter and receiver timing. The transmit clock, **xclk**, is used to alternately sample the input signal, **x**, into a pair of flip-flops. A toggle flip-flop generates a transmit pointer, **xp**, that selects which flip-flop samples each symbol on input **x** via the flip-flop's clock-enable input.
* The outputs of the transmit flip-flops are a pair of signals, **x0 and x1**, that are still synchronized to the transmit clock but which change only every other cycle.



* A toggle flip-flop driven off the local clock, **clk**, generates a receive pointer, **rp**, that drives a multiplexer to select between **x0 and x1**. The phase of **rp** is set (by logic not shown) so that it always samples the transmit flip-flop that is guaranteed to be stable for the next clock cycle.
* The resulting signal, **xm**, is synchronized with the receive clock, changing in response to **rp**, and thus can be used in combinational logic and sampled by the local clock.
* Figure shows waveforms for a FIFO synchronizer operating with transmit and local clocks 180 degree out of phase. The first five rows illustrate how the transmit side of the synchronizer splits the input stream on signal **x** into interleaved streams on **x0 and x1**.
* The last three rows show how **rp** selects the stable transmit flip-flop output. The transmit flip-flops only change state when they are not selected by **rp.**
* For correct sampling, **rp** must lag **xp** by at least **tcy**; otherwise, the changing output will be sampled. For a two-element ring-buffer, this lag in **rp** relative to **xp** by at least a clock cycle is equivalent to a lead by at most a clock cycle.
* Stated differently, **rp** must be high during the rising edge of **xp** so that **x1** is sampled when **x0** changes and vice versa. At startup time, **rp** is initialized by sampling **xp** with **clk**, which waits an odd number of cycles for metastable states to decay and then uses the resulting value to initialize **rp.**

**Lecture -3: Clock Domain Crossing**

1. Describe the advantages and disadvantages of open loop solutions for clock domain crossing.

Answer:

* **Advantage: -** The Open-loop solution is the fastest way to pass signals across CDC boundaries that does not require acknowledgement of the received signal.
* **Disadvantage: -** The largest potential problem related to an open-loop solution is that another engineer might mistake the solution for a general-purpose solution, or the design requirements might change and an engineer might fail to re-analyze the original open loop solution.
* This problem can be minimized by adding a SystemVerilog Assertion to the model to detect if the input pulse ever fails to exceed the "three edges" design requirement.

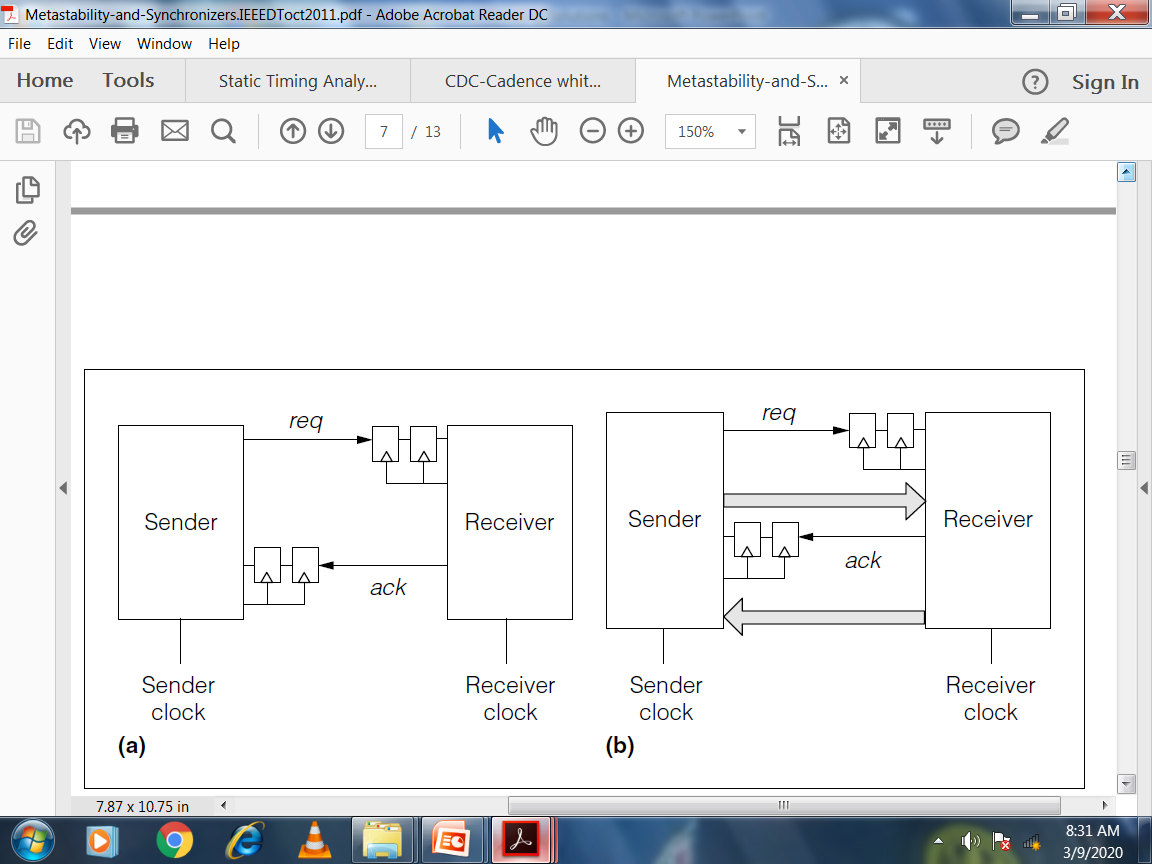
1. Explain the advantages of Multi Cycle path (MCP) technique for CDC

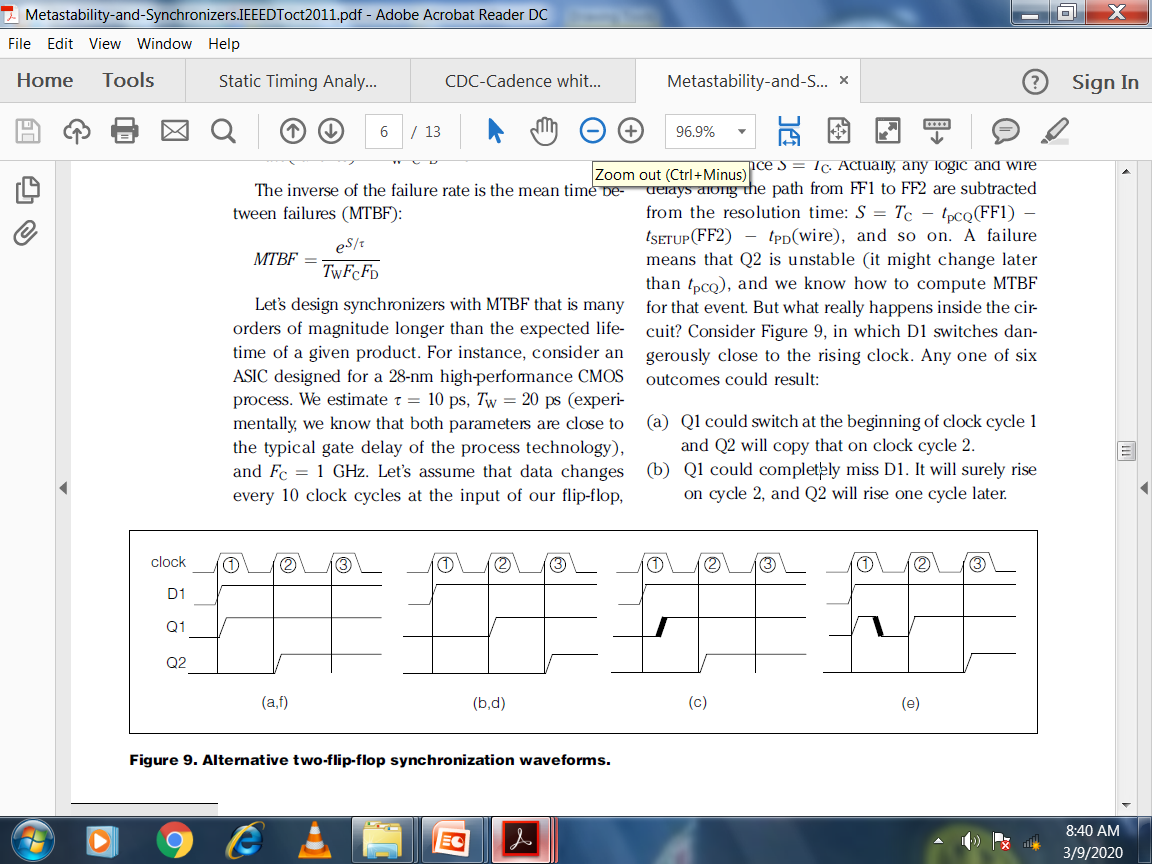
Answer:

* The sending clock domain is not required to calculate the appropriate pulse width to send between clock domains.
* The sending clock domain is only required to toggle an enable into the receiving clock domain to indicate that data has been passed and is ready to be loaded. The enable signal is not required to return to its initial logic level.
* The receiving clock domain is not allowed to sample the multi-bit CDC signals until the synchronized enable passes through synchronization and arrives at the receiving register.

1. Describe the Two Flop Synchronizer with neat circuit diagram.

Answer:





1. Q1 could switch at the beginning of clock cycle 1 and Q2 will copy that on clock cycle 2.
2. Q1 could completely miss D1. It will surely rise on cycle 2, and Q2 will rise one cycle later.
3. FF1 could become metastable, but its output stays low. It later resolves so that Q1 rises (the bold rising edge). This will happen before the end of the cycle (except, maybe, once every MTBF years). Then Q2 rises in cycle 2.
4. (d) FF1 could become metastable, its output stays low, and when it resolves, the output still stays low. This appears the same as case (b). Q1 is forced to rise in cycle 2, and Q2 rises in cycle 3.
5. (e) FF1 goes metastable, and its output goes high. Later, it resolves to low (we see a glitch on Q1). By the end of cycle 1, Q1 is low. It rises in cycle 2, and Q2 rises in cycle 3.
6. (f) FF1 goes metastable, its output goes high, and it later resolves to high. Q1 appears the same as case (a). Q2 rises in cycle 2.

