EE273 Digital Systems Engineering
Final Exam

March 18, 2002
(version 1.0)
SOLUTIONS
(Total time = 120 minutes, Total Points = 100)

Name: (please print)_____

In recognition of and in the spirit of the Stanford University Honor Code, I certify that I will neither give nor receive unpermitted aid on this exam.

Signature:_____

This examination is open notes open book. You may not, however, collaborate in any manner on this exam. You have two hours to complete the exam. Please do all of your work on the exam itself. Attach any additional pages as necessary.

Before starting, please check to make sure that you have all 10 pages.

1		**52**
2		**15**
3		**15**
4		**18**
Total		**100**

If you are an SITN student, please complete the following information:

Circle one: **LOCAL** **ONLINE** **REMOTE**

Please PRINT your name_____

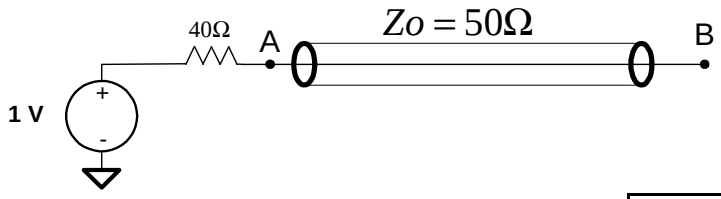
Phone (___)_____

COMPANY NAME _____ **Mailstop** _____

City _____ **State** _____

Problem 1: Short Answer (52 Points: 13 questions, 4 points each)

A. A voltage source drives a 1V step onto a 50-Ohm transmission line through a 40-Ohm series resistor. The far end of the line is open. What will the voltage be on the source end of the line (a) immediately after the step, and (b) in the steady state?



(a)The voltage at point A immediately after the 1 V step is: $1V \times \frac{50\Omega}{50\Omega + 40\Omega} = 0.556V$

(b)Since the line is left open, the voltage at point A in steady state is 1V.

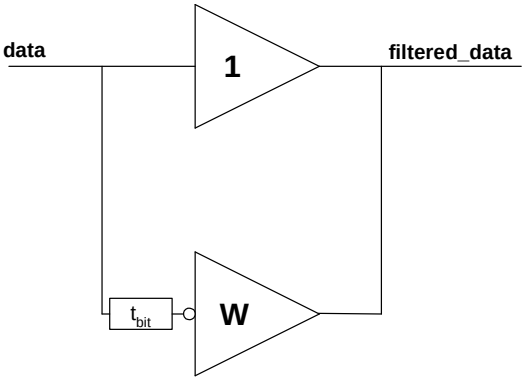
B. A signaling system has a bit-error rate (BER) of 10^{-15} with Gaussian noise of 10mV. If you reduce the Gaussian noise to 5mV, what will the new BER be? Using the following relationship

$BER = e^{-\frac{VSNR^2}{2}}$ *and* $VSNR = \frac{V_{NM}}{V_{rms}}$ *. By decreasing the Gaussian noise, V_{rms}, by 2 the VSNR is increased by 2. Thus, the new bit error rate is:*

$$BER = e^{-\frac{(2VSNR\_old)^2}{2}} = e^{-4\frac{VSNR\_old^2}{2}} = \left(e^{-\frac{VSNR\_old^2}{2}} \right)^4 = \left(10^{-15}\right)^4 = 10^{-60}$$

C. A signaling system has frequency-dependent attenuation of A=0.2 . That is, when a 1V step is put into the source end of the line, the far end of the line reaches 200mV after one bit time. There is no attenuation at DC. The system uses a two-tap FIR filter to correct this attenuation. If the first tap of the filter has a weight of 1.0, to first approximation, what should the weight of the second filter tap be?

In designing a two-tap filter,we want to boost the high frequencies and attenuate the low frequencies to counteract the effect of the line which attenuates the high frequencies (i.e. a lone 1) and passes the low frequencies (i.e. a long string of 1's). Thus,according to our design, for the lone 1 we get a filtered value of (1+w) sent on the line. When sending multiple ones we get a filtered value of (1-w) sent on the line. The lone 1 will be attenuated by the line by approximately 0.2 and the multiple ones ("long string of ones") will not be attenuated. We want to equalize these values so that value received for a lone 1 1 is the same as the value received for a 1 after a long string of ones. So we have: 0.2(1+w) = (1-w). Solving for w we get: w = 0.67.



D. A chip has 500nF of on-chip bypass capacitance and when operating in normal mode draws an average current of 10A. The bypass network must handle the transient when the chip switches to standby mode, and the average current switches from 10A to 0A, with a maximum change in voltage of 100mV. (a) What parameter of the first bank of off-chip capacitors determines whether this specification will be met? (b) What is the minimum value of this parameter that meets this specification?

(a) The inductance of the off-chip capacitors.
(b) The inductance of the first bank of off-chip capacitors is limited by the following constraint:

$$\boxed{L < C_B\left(\frac{\Delta V}{I_{ave}}\right)^2 = 500nF\left(\frac{100mV}{10A}\right)^2 = 0.05nH}$$ *(Note: the minimum value is actually 0, the* ***maximum*** *value for L is 0.05nH.)*

E. A signaling system has a rise/fall time of 25ps, an aperture time of 25ps, jitter of ±25ps (50ps peak-to-peak), and operates using ***synchronous timing*** over a line with 1ns of delay. (Assume you need the full voltage swing during the entire aperture time). (a) What is the fastest rate at which this system can operate? (b) Over what continuous frequency range including this rate can the system operate?

(a) The smallest bit time that can be operated due to timing constraints on the bit is:
$t_{bit} \geq t_r + t_a + t_j = 25ps + 25ps + 50ps =$ *100ps* *However, if we use this bit time, there would be an even number of bits on the line, i.e.* $\frac{1ns}{100\frac{ps}{bit}} = 10bits$. *With synchronous timing, we need to have N+1/2 bits nominally on the line at a time. Thus, since we decrease the number of bits on the line at a time to 9.5 and we get a bit time of:* $\frac{1ns}{9.5bits} = 105.3\frac{ps}{bit}$. *Thus, the fastest base bit rate that can operate on this system is* $\boxed{9.5Gb/s}$.

(b) The continuous frequency that can operate around the nominal frequency of 9.5Gb/s is determined by the following equation:

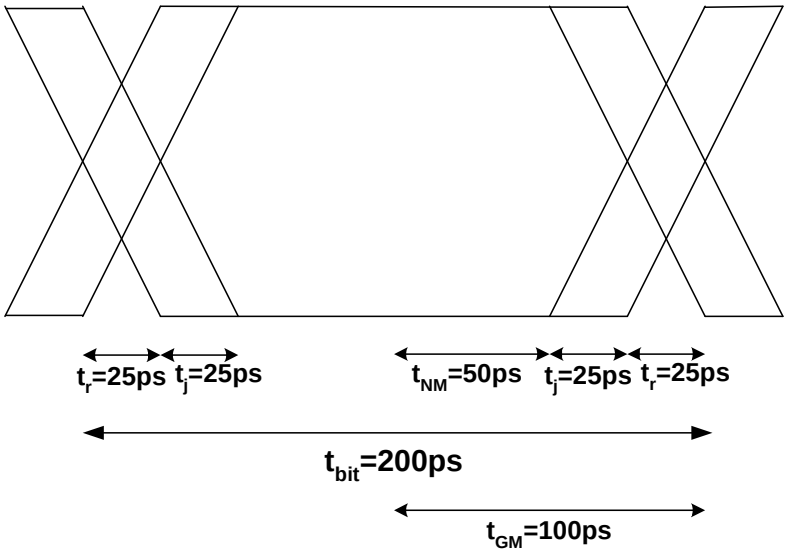
$$\frac{t_{wire} + [0.5(t_r + t_a) + t_u]}{N} \leq t_{bit} \leq \frac{t_{wire} - [0.5(t_r + t_a) + t_u]}{N-1}$$

Thus, $\frac{1ns + [0.5(50\,ps) + 25\,ps]}{N} = 105\,ps \leq t_{bit} \leq \frac{1ns - [0.5(50\,ps) + 25\,ps]}{N-1} = 105.6\,ps$

So the bitrate must fall in the following range: $\boxed{9.47Gb/s \leq bitrate \leq 9.52Gb/s}$

s

F. A signaling system using per-line closed-loop timing has a bit time of 200ps, a rise/fall time of 50ps, ±25ps (50ps peak-to-peak) of bounded jitter plus Gaussian jitter with an RMS (1σ) value of 5ps. Assuming there is no voltage noise in the system, what will the BER of this system be due to timing noise?



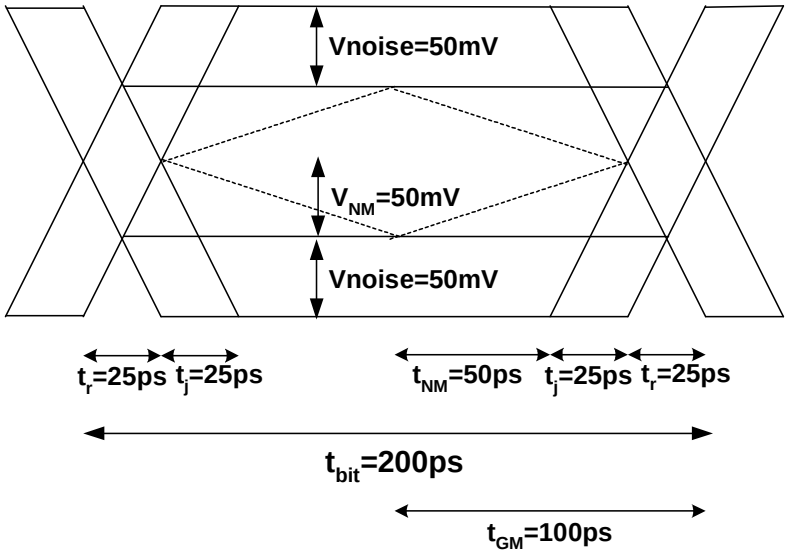
Using the same approach we used with voltage gaussian noise, we find the timing signal to noise ratio, TSNR. TSNR = $T_{NM}/T_{rms}a$=50ps/5ps=10. (Since the timing net margin is: T_{GM}-T_{noise} = T_{GM}-(25ps+25ps) = 200ps/2 – 50ps = 50ps.) Now we get a bit error rate of:

$$BER = e^{-\frac{TSNR^2}{2}} = e^{-\frac{10^2}{2}} = 1.93\times10^{-22}$$

(Note that in the figure to the left, only the worst-case jitter is shown – not the peak-to-peak jitter.)

G. Consider the system of problem F but now consider voltage noise in addition to the timing noise. The signal swing is 200mV, the bounded noise sources total 50mV, and there is 5mV RMS Gaussian noise. What is the BER of this system due to the combination of voltage noise and timing noise?

To be able to add the timing and voltage gaussian noise, we convert the Gaussian timing noise to Gaussian voltage noise.



The eye diagram, with the added voltage noise, looks as shown above. The dotted lines in the center of the eye approximate the relationship between the voltage and time. We note that these lines have slope

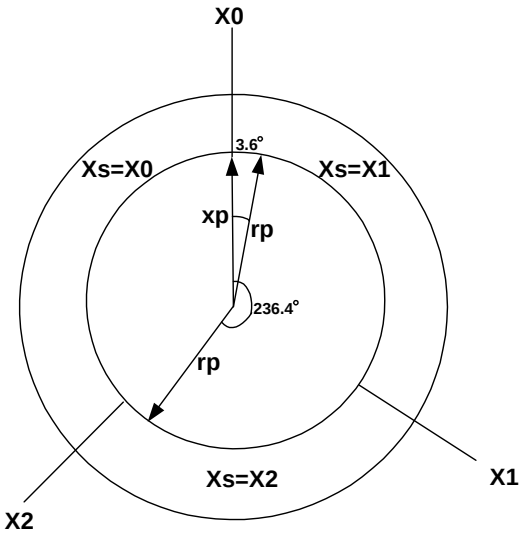
±(50mV/75ps)=0.667mV/ps. Since we have 5ps of rms timing noise this converts to 5ps x 0.667mV/ps = 3.33mV of additional Gaussian noise. So, the total Gaussian noise in our system is:

$Vgn = \sqrt{(5mV)^2 + (3.33mV)^2} = 6mV$ *The new VSNR is 50mV/6mV = 8.32 and the bit error rate is:* $\boxed{BER = e^{-\frac{VSNR^2}{2}} = 9.26 \times 10^{-16}}$

H. A per-line closed loop timing system cancels any mismatch between the length of two data lines true or false?

True *In per-line closed loop timing all the skew between data lines is cancelled.*

I. A plesiochronous three-register FIFO synchronizer moves a signal between two clock domains where the maximum frequency difference is 100ppm. By convention, the system puts a NULL symbol on the line at least once every 100 symbols. Ignoring flip-flop setup, hold, and delay times, at what relative phases must the receive pointer be adjusted when a NULL is present to ensure that a FIFO register never changes while it is selected by the output MUX?



*The maximum frequency difference between the 2 clocks is 100 parts per million. So, for example, if one clock is running at 10MHz, the other clock could be running at (10MHz + $100/10^6$ * 10MHz) = 10.001MHz . Or another way to look at is that with respect to the phase of one clock, the other clock will lose (or gain) $100/10^6 = 10^{-4}$ of a cycle each cycle. So, if there are 100 symbols per NULL, the maximum amount of a cycle lost between nulls is: $100 * 10^{-4} = 10^{-2}$ of a cycle. Since there are 360° in a cycle, the phase lost (or gained) by one clock relative to the other clock in 100 symbols is $10^{-2} * 360° = 3.6°$. Thus, the two possible cases where the receive pointer, rp, needs to be adjusted are: (1) when rp is leading xp, the transmit pointer, by* $\boxed{\leq 3.6°}$ *, and (2) when rp is lagging xp by* $\leq (120° + 3.6°) = 123.6°$*, that is the phase difference is* $\boxed{\geq 236.4°}$

Note that the first case will occur when rp is slow with respect to xp. The second case will occur when rp is fast with respect to xp. These constraints will ensure that the data that is duplicated or dropped is a NULL and not a valid data bit.

J. In a four-level signaling system the gross margin is what fraction of the maximum signal swing?

The gross margin is $\boxed{1/6}$*. There are 4 levels and thus 3 areas of eye opening. Each of these eye openings is divided by 2 to obtain the gross margin. So we get a gross margin of (1/3)/2 = 1/6.*
Taking another approach we can deduce the same result: there are 2 data bits per symbol (N=2). Thus,

$$\frac{1}{2(2^N - 1)} = \frac{1}{2(2^2 - 1)} = \frac{1}{6}$$

K. A brute-force synchronizer has a waiting time of t_w = 2ns, and a failure probability of $P_F = 10^{-30}$. Regeneration time constant τ_s is 100ps. If the waiting time is doubled to 4ns with no other changes, what will the new failure probability be?

The probability of failure is: $P_F = t_a f_{cy} e^{-\left(\frac{t_w}{\tau_s}\right)}$ *Plugging in all the other values for the case where* t_w = *2ns, we can solve for* $t_a f_{cy}$. *We get:* $t_a f_{cy} = 4.85 \times 10^{-22}$

After doubling the waiting time, the probability of failure is:

$$P_F = 4.85\times10^{-22} e^{-\left(\frac{4ns}{100\,ps}\right)} = 2.06\times10^{-39}$$

L. A system is built using flip-flops with a setup time of t_s = 100ps, a hold time of t_h = 50ps, a contamination delay of t_{cCQ} = 120ps, and a propagation delay of t_{dCQ} = 160ps. A mesochronous synchronizer is used to move a signal between two clock domains in this system with the minimum possible delay. What is the minimum delay you can guarantee for this synchronizer. (Do not include the phase difference between the transmit and receive clocks (sampling delay) in your answer – only include the additional delay of the synchronizer.)

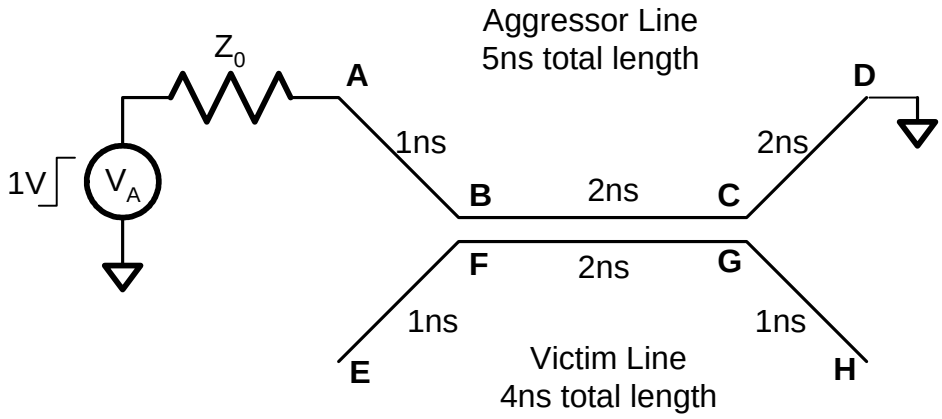
The minimum delay you can guarantee is $(t_{dCQ}-t_{cCQ}) + t_s + t_h$ *=(160ps-120ps) + 100ps + 50ps =* *190ps*

M. You need to move a signal from a CPU running on a 266MHz clock to a network running on a 200MHz clock with minimum delay. What type of synchronizer should you use? What is the minimum delay you can guarantee for this synchronizer? (Assume you are using flip-flops with the same parameters as in Question L. Also as in L include only the additional delay of the synchronizer).

a) You would use a clock prediction circuit (or periodic synchronizer).
b) The minimum delay you can guarantee is again $(t_{dCQ}-t_{cCQ}) + t_s + t_h$ *=(160ps-120ps) + 100ps + 50ps =* *190ps* *When, from the clock prediction circuit, it is determined that the 200MHz clock would sample in the keepout region (which includes the clock keepout region and the data keepout region) the the 200MHz clock samples from a delayed version of the data. The data is delayed by the keepout region.*

Problem 2: Transmission Lines [15 Points]

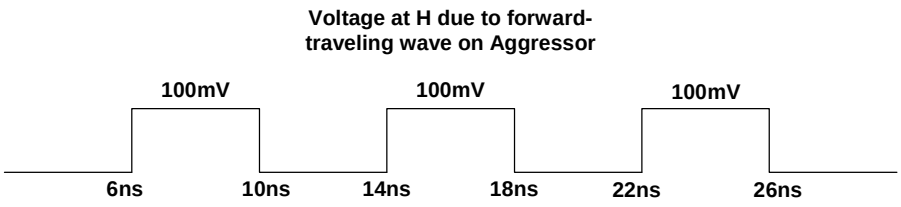
A pair of 50-Ohm transmission lines are terminated and coupled as illustrated in the drawing below. The aggressor line is 5ns long and the victim line is 4ns long. The two lines are coupled over 2ns of their length with a near-end crosstalk coefficient of $k_{rx} = 0.1$ and a far-end crosstalk coefficient of $k_{fx} = 0$. The aggressor is driven with a 1V step with a 500ps rise time through a matched impedance at time zero. Sketch the waveform at point H. (You need only show the first 20ns of waveform on H. Also, ignore any waves with a magnitude of less than 10mV).



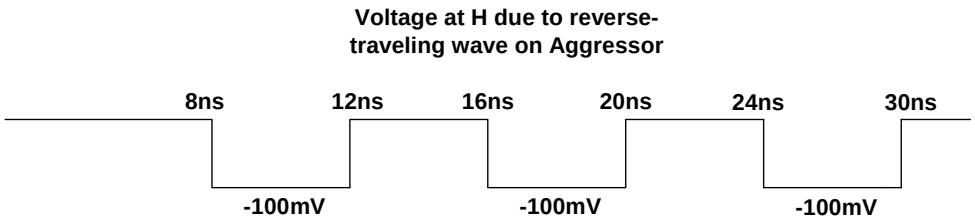
In this system, we will get crosstalk at H from two sources: (1) the near-end crosstalk induced at F reflects off of the open termination at E and travels to the far end, and (2) the aggressor wave reflects off of D and induces near-end crosstalk induced at the far end (at G).

Let's analyze these two sources separately and then superimpose them.

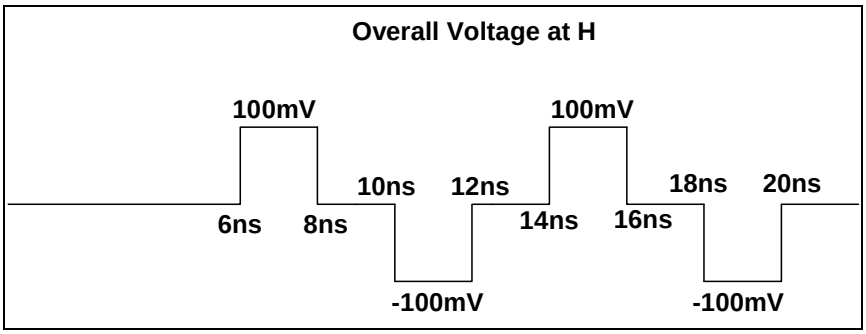
(1) *The aggressor wave at A is of magnitude 0.5 V at time t=0. At t=1ns, the 0.5 V wave arriving at B induces near-end cross talk of 0.1*0.5V=0.05V at F that travels toward E. At 1ns later (t=2ns), this 0.05 V wave completely reflects off of E (open termination) and begins to travel to H. 4 ns later (at t=6ns), this 0.05 V wave reaches H and reflects off of the open termination at H. Thus, the magnitude at H is 0.1 V since it sees the superposition of the forward and the reverse-traveling waves. This voltage (0.1V) lasts for 4ns (i.e. twice the coupled length of 2ns). Meanwhile, the reverse traveling wave reaches E (at t=10ns). Again it reflects off of E and returns to H 4 ns later (at t=14ns). This crosstalk will continue bouncing back-and-forth between points E and H forever (assuming they are ideal transmission lines and there is no loss. Below is a diagram of the voltage induced at H due to the forward traveling wave on the aggressor.*



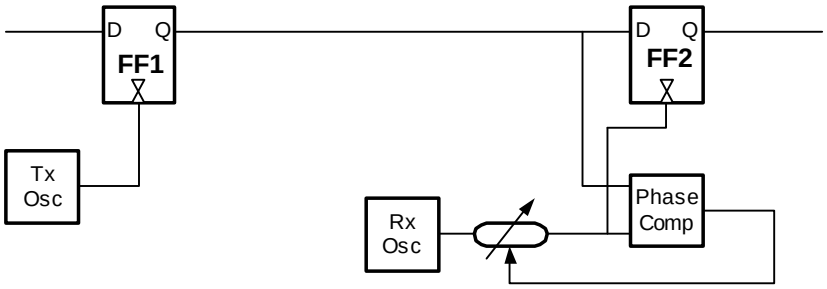
(2) *Now let's analyze the crosstalk caused by the aggressor wave reflecting off of D. At t=5ns, the 0.5V forward-traveling wave on the aggressor reflects off of D with a coefficient of k=-1 giving a reverse-traveling wave of –0.5V. At t=7ns, this –0.5V reverse-traveling wave induces crosstalk at G of (0.1 * -0.5V)=-0.05V. This crosstalk travels toward H and reaches H at t=8ns. At t=8ns, this wave completely reflects off of H and gives a magnitude of –0.1V at H (the sum of the forward traveling wave and the reverse-traveling wave). This crosstalk lasts for 4ns (twice the length of the coupled lines). As in the previous case, this cross talk reaches point E 4ns later (t=12ns) and completely reflects. 4ns later (t=16ns) this –0.05 V wave will reach H. Below is a diagram of the voltage induced at H due to the reverse-traveling wave on the aggressor (the wave reflected off of D). Note that this reverse traveling wave is absorbed into the source termination at time t=10ns and there is no more crosstalk induced by the aggressor line.*



Superimposing these two effects, we get the overall crosstalk waveform at H to be:



Problem 3: Timing [15 Points total]



Consider the transmission system shown above. A data signal is transmitted on each edge of a transmit clock. At the receiver, a phase comparator measures the time from the last clock edge to each data edge and adjusts the phase of a receive oscillator (with a variable delay line with a delay between 0 and 1 cycle) to make this time equal to a half a bit period. The specifications of the flip-flops, delay line, and oscillators are shown in the table below. Assume that the phase comparator is ideal. The rise/fall time of the signal is 25ps.

Component	Specification	Value	Units
Oscillator	Jitter	10	ps (p-p)
Flip-flop	Jitter	10	ps (p-p)
	Skew	±10	ps
	Aperture time	20	ps
	Aperture offset	±10	ps
Delay line	Jitter	10% of delay	(p-p)

(a) [8 points] What is the fastest data rate at which this system will operate reliably?

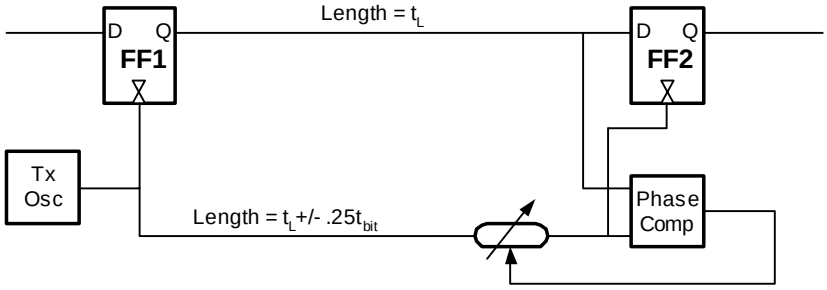
Only the skew from FF1 is cancelled. So our overall timing uncertainty is:
$tu = t_{jitter(FF1)} + t_{jitter(TxOsc)} + t_{jitter(RxOsc)} + t_{ao} + 0.1(1t_{bit})$
$= 10ps + 10ps + 10ps + 20ps + 0.1\ t_{bit}$
$= 50ps + 0.1\ t_{bit}$

Thus we have:
$t_{bit} \geq t_r + t_a + t_u = 25ps + 20ps + (50ps + 0.1t_{bit}) = 95ps + 0.1t_{bit}$

Solving for t_{bit} *we get:*
$t_{bit} \geq 105.6ps.$

Thus, the maximum bitrate we can have is:
$\boxed{bitrate \geq 9.47Gb/s}$

(b) [7 points] The system is redesigned as shown below so that the transmit clock is forwarded to the receiver on a line whose length is within ±1/4 bit time of the data line. With this arrangement, what is the fastest data rate at which the system will operate?



The maximum delay of the delay line is now ¾ of a bit period. (I.e. when the lower line is skewed by –0.25 t_{bit}, the delay line must be ¾ of a bit period.

But since this is a closed-loop system, the skew between the lines does not add into our timing margins – only the jitter of the delay line.

So, now our uncertainty is:

$tu = t_{jitter(FF1)} + t_{ao} + 0.1(0.75t_{bit})$
$= 10ps + 20ps + 0.075\ t_{bit}$
$= 30ps + 0.075\ t_{bit}$

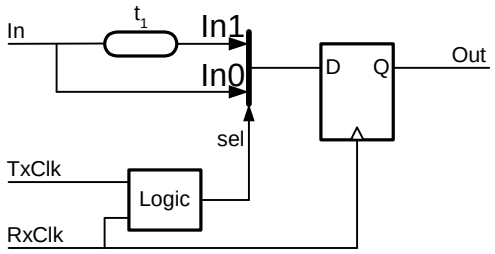
Thus we have:
$t_{bit} \geq t_r + t_a + t_u = 25ps + 20ps + (30ps + 0.075t_{bit}) = 75ps + 0.075t_{bit}$

Solving for t_{bit} we get:
$t_{bit} \geq 81.1ps.$

Thus, now the maximum bitrate we can have is:
$\boxed{bitrate \geq 12.3Gb/s}$

Note: The above analysis is assuming that jitter from the oscillator is mostly low frequency (which is usually the case). That is, the phase of the clock varies only a small amount from cycle to cycle. When the jitter is low-frequency and the data and clock are sent clocked through the same buffer, as above, then the jitter of the common buffer is cancelled. (Credit was given for various jitter numbers used depending on your stated assumptions.

Problem 4: Synchronization [18 points total]



The delay-line synchronizer shown above is used to move signal *In* from the domain of *TxClk* to the domain of *RxClk*. These two clocks are plesiochronous 1GHz clocks with a mismatch in frequency of 100ppm (10^{-4}). The flip flop has the following parameters: t_s = 20ps, t_h = 20ps, t_{cCQ} = 100ps, and t_{dCQ} = 120ps. The incoming data stream, which comes out of an identical flip-flop clocked by TxClk, includes a NULL symbol every 100 data symbols. You may ignore the delay of the multiplexer.

A. [4 points] What is the minimum delay t_1 for which this synchronizer will work properly.

The minimum delay of t_1 is the sum of the clock keepout region (of RxClk) and the data keepout region (of data In). The data keepout region is: $t_{dCQ} - t_{cCQ}$. The clock keepout region is $t_s + t_h$. So, the total keepout region is:

$$t_1 = t_{ko} = (t_{dCQ} - t_{cCQ}) + t_s + t_h = (120ps\text{-}100ps) + 20ps + 20ps = \boxed{60ps}$$

B. [5 points] Write down the rules for when the *sel* signal should be switched from 0 to 1 and from 1 to 0. (Note, *sel* = 1 selects the upper input of the multiplexer). For now ignore whether or not symbols are NULL. That is, your rules need only consider the delay from the rising edge of *TxClk* to the rising edge of *RxClk*.

For the 1→0 transition on select:

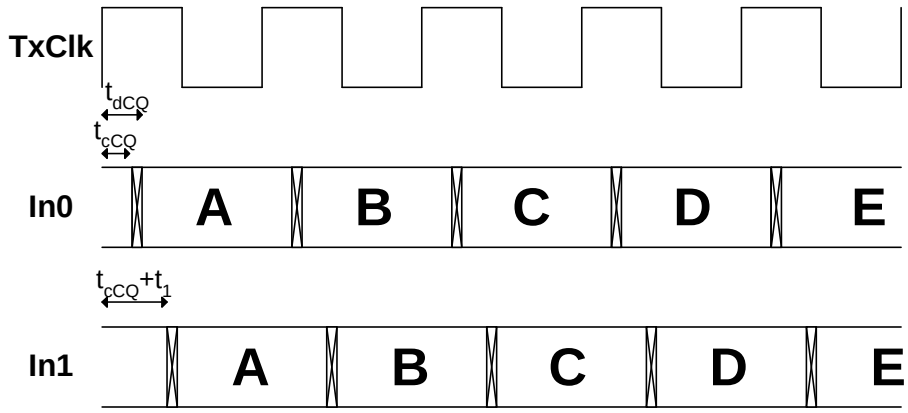
If RxClk is in (i.e. just at the boundaries of) the keepout region of In0, namely: $\boxed{\text{RxClk leads TxClk by} \geq 80ps}$ (i.e. $t_{cCQ} - t_h$) or $\boxed{\text{RxClk leads TxClk by} \leq 140ps}$ (i.e. $t_{dCQ} + t_s$).

For the 0→1 transition on select:

If RxClk is in (i.e. just at the boundaries of) the keepout region of In1, namely: $\boxed{\text{RxClk leads TxClk by} \geq 140ps}$ (i.e. $t_1 + t_{cCQ} - t_h$) or $\boxed{\text{RxClk leads TxClk by} \leq 200ps}$ (i.e. $t_1 + t_{dCQ} + t_s$)

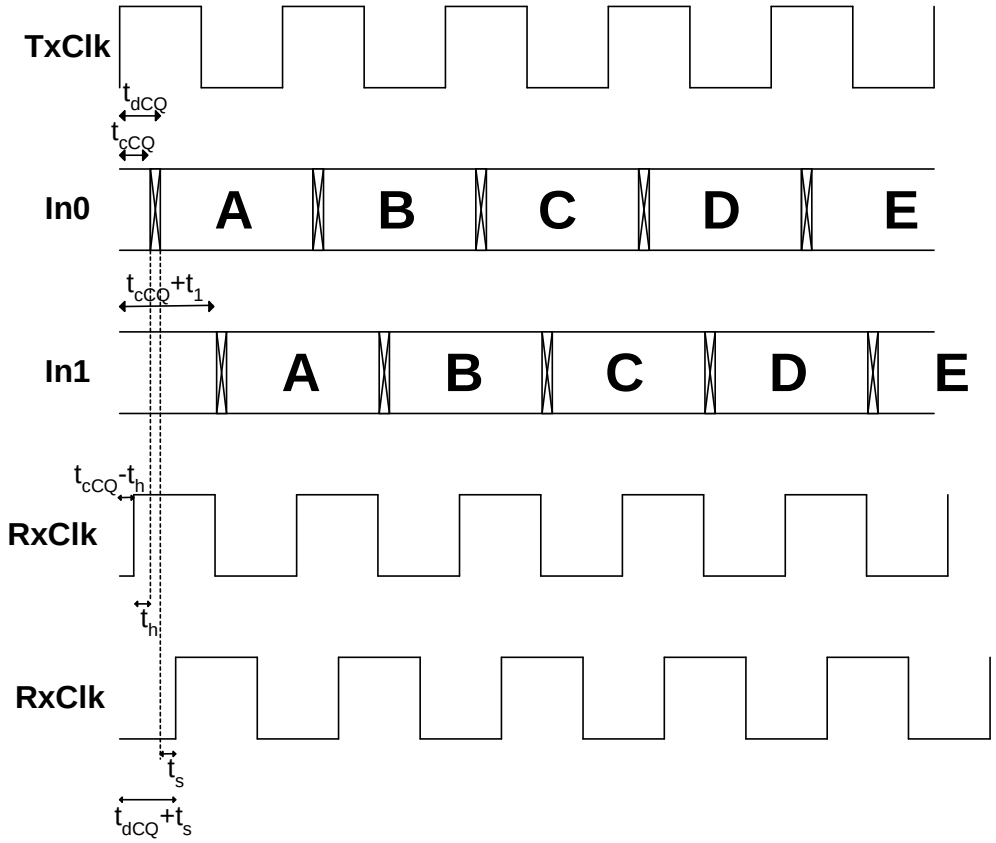
Actually the simplest solution is one of the following two options: when RxClk is in (i.e. just at the boundaries of) the keepout region of In1, select In0. Otherwise, select In1. Or, similarly, when RxClk is in the keepout region of In0, select In1, Otherwise, select In0.

Below is the timing diagram for the transmitted data, In0 and In1 relative to the transmitting clock, TxClk.

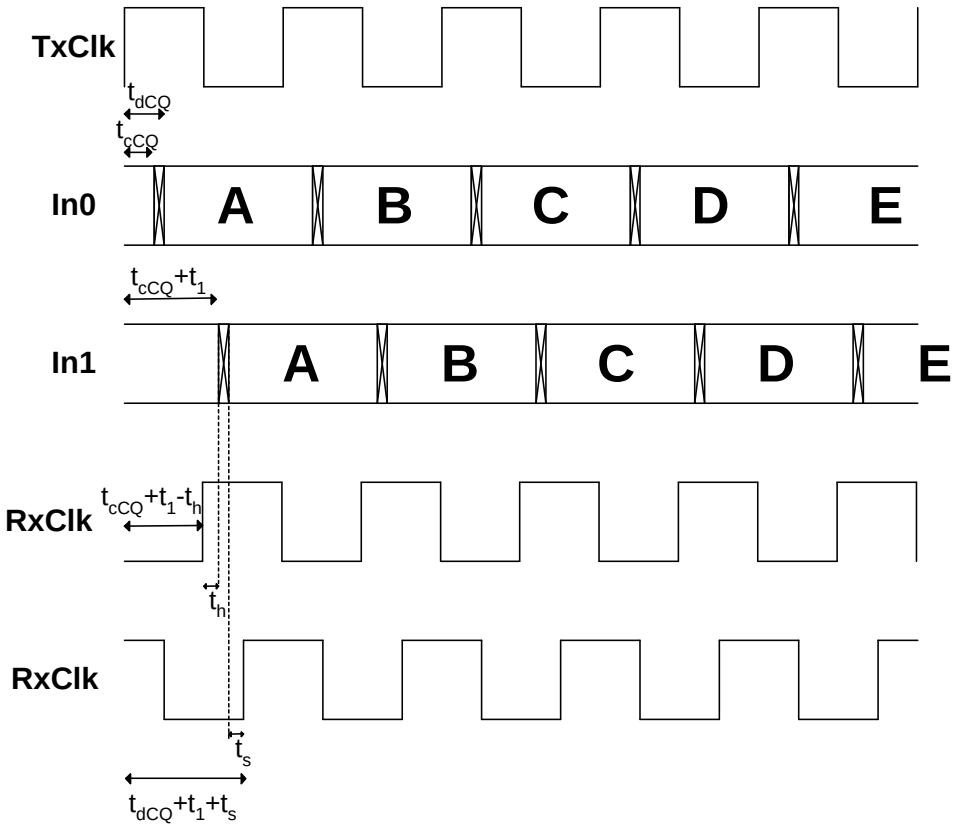


(Note that the cycle time is not to scale.)

Below is an illustration of the In0 keepout boundaries with respect to RxClk:



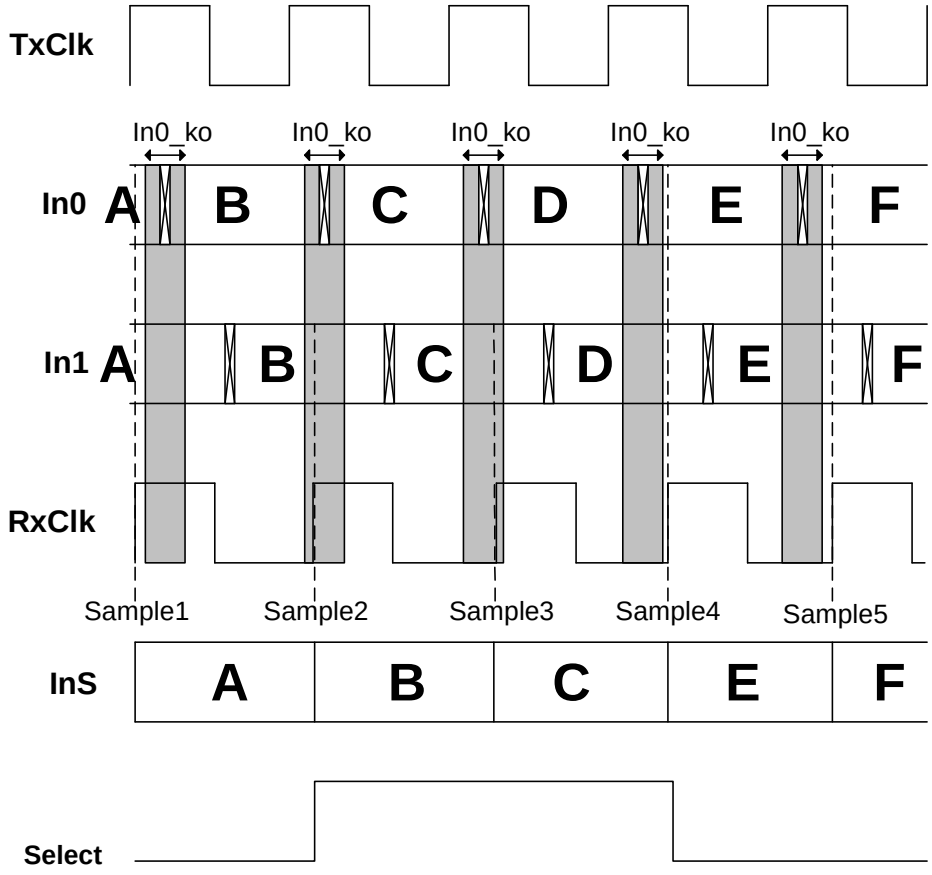
Note that the two versions of RxClk are possible versions of RxClk at two different times. Below is an illustration of the In1 keepout boundaries with respect to RxClk:



Again, note that the two versions of RxClk are possible versions of RxClk at two different times.

C. [4 points] Modify your rules from part B to take into account NULL symbols and to guarantee that no non-null symbols are duplicated or dropped.

Below is an example of what would happen if we didn't take the NULLs into account when switching. In the case below, RClk is slow with respect to TClk. We note that when Select transitions from 0→1 there is no data duplicated or dropped. However on a transition from 1→0 the data value D is skipped. So unless D is in fact a NULL (i.e. not actual data), we will have lost data. Similar diagrams can be drawn for the case when RClk is fast with respect to TClk and you will see that in this case data will be duplicated unless we take into account the NULL's.



Now for our analysis on how to ensure that data is neither skipped or duplicated:

Each cycle the phase of RxClk drifts (either leads or lags) TxClk by 100ppm – i.e. $100/10^6$ of a cyle = 10^{-4} of a cycle. So in 100 bits, the phase will drift by 10^{-2} of a cycle (i.e. $100*10^{-4}$). Since the cycle time is 1ns, this is 10ps of drift every 100 bits. Thus, to guarantee that a NULL will be either duplicated (when RxClk is fast with respect to TxClk) or dropped (when RxClk is slow with respect to TxClk), we add 10ps bands around each of the keepout regions above. So the new keepout regions are:

In0 keepout region: 70ps ≤ phase by which RxClk leads TxClk ≤ 150ps
In1 keepout region: 130ps ≤ phase by which RxClk leads TxClk ≤ 210ps

We notice now that the keepout regions overlap, so we increase the delay, t_1, by 20ps so that the new value of t_1 is 80ps. So, the new keepout regions are:

In0 keepout region: 70ps ≤ phase by which RxClk leads TxClk ≤ 150ps
In1 keepout region: 150ps ≤ phase by which RxClk leads TxClk ≤ 230ps

(Note, that if we use the solution of choosing In1 except for when RxClk is in In1's keepout region, we only have to increase t_1 to 70ps although this will increase our average data delay. The keepout regions would then as shown below.)

In0 keepout region: 70ps ≤ phase by which RxClk leads TxClk ≤ 150ps
In1 keepout region: 140ps ≤ phase by which RxClk leads TxClk ≤ 220p)

Now we decide when it is safe to toggle the select signal so that only NULLs are skipped or duplicated and not actual data. There are two cases: when RxClk is faster than TxClk, and when RxClk is slower than TxClk:

Case 1 - RxClk faster than TxClk
- ***1→0*** *transition on select: safe, no data will be duplicated.*
- ***0→1*** *transition on select: not safe, the previous bit will be duplicated. Must wait until the previous symbol was a NULL.*

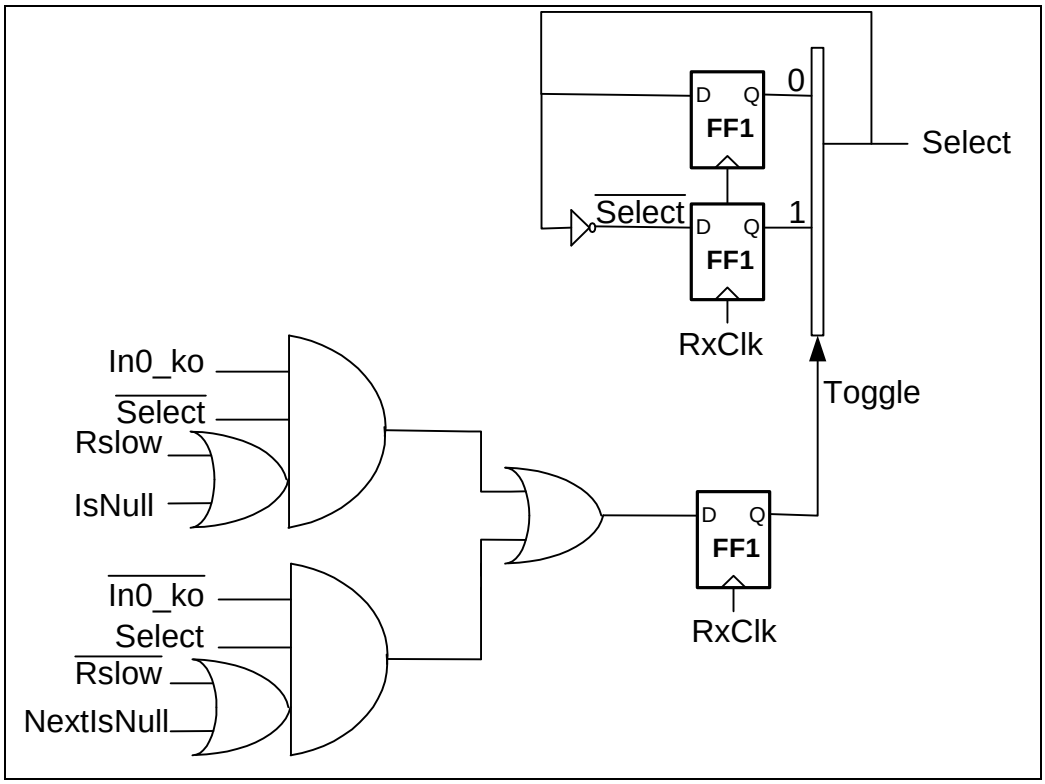
Case 2 - RxClk slower than TxClk
- ***1→0*** *transition on select: not safe, the current bit will be skipped. Must wait until the current bit is a NULL.*
- ***0→1*** *transition on select: safe, no data will be duplicated.*

D. [5 points] Sketch the contents of the Logic block that implement your rules from part C.

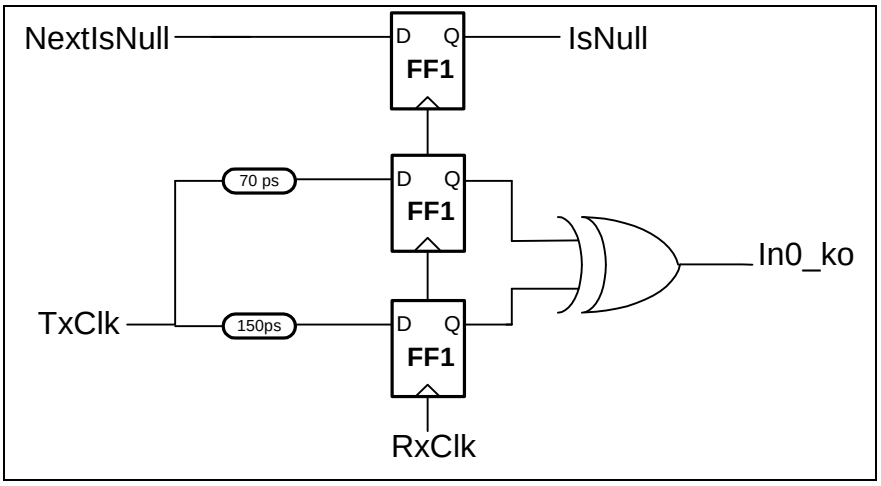
We choose the simple solution of choosing In0 except for when Rclk is in In0's keepout region. The schematic is shown on the next page and the equation for Toggle is:

*Toggle = In0_ko * Select' * (RSlow+IsNull) + In0_ko' * Select * (RSlow' + IsNull)*

Where:
In0_ko: is true when RxClk is in the In0 keepout region.
old: the previous value on Select.
RSlow: is true when RClk is slow with respect to TClk.



Note that the toggled value of Select won't take effect until the next rising edge of RClk after which it is generated. The circuit for generating In0_ko is shown below.



Note: to ensure stability of the signal, another stage of FF's (implementing a brute-force synchronizer) should be inserted after the phase comparators (the lower 2 FF's).