NTU 6342 / EE 241 Homework #3
SOLUTIONS

Problem #1:

Delay times:

$t_{p\_FO4\_20} := 0.424\text{ns}$

$t_{p\_FO4\_100} := 2.12\text{ns}$

Simulation values:

$T_{min} := 2 \cdot t_{p\_FO4\_20}$ $\qquad T_{max} := 2 \cdot t_{p\_FO4\_100}$

$T_{min} = 0.848\,\text{ns}$ $\qquad T_{max} = 4.24\,\text{ns}$

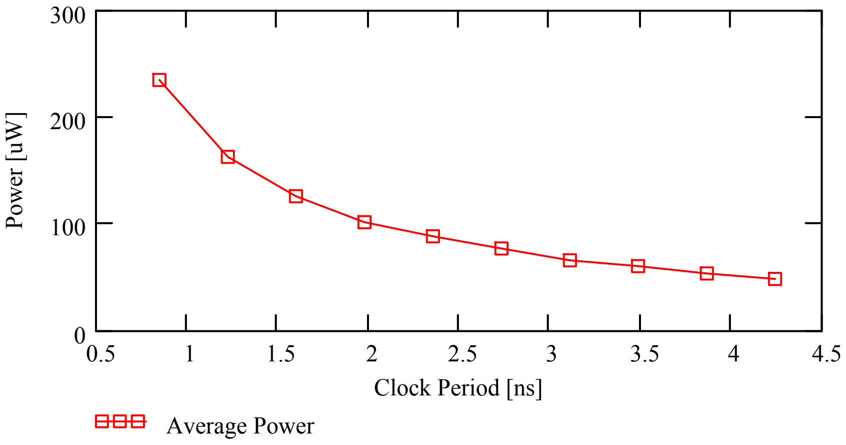
$f_{max} := \dfrac{1}{T_{min}}$ $\qquad f_{min} := \dfrac{1}{T_{max}}$ $\qquad f_{max} = 1.179\,\text{GHz}$ $\qquad f_{min} = 235.849\,\text{MHz}$

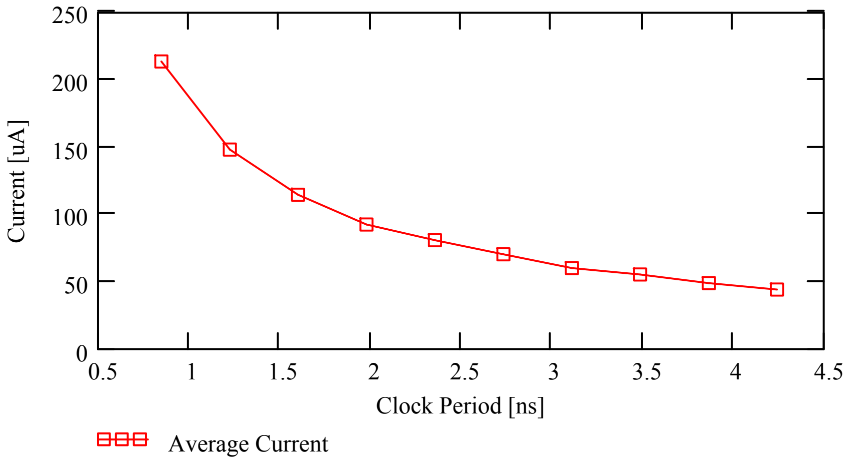
Part a:

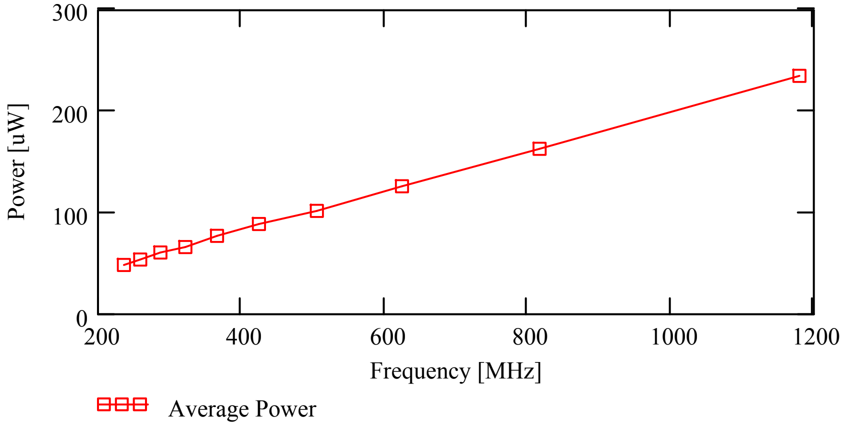
$\text{PartAData} := \text{READPRN}(\text{"inverter\_delay\_fo4\_20.txt"})$

$\text{PulseWidth} := \text{PartAData}^{\langle 0 \rangle}\,\text{s}$ $\qquad V_{dd} := 1.1\text{V}$ $\qquad I_{ave} := \text{PartAData}^{\langle 2 \rangle}\,\text{A}$

$\text{Frequency} := \overrightarrow{\dfrac{1}{2 \cdot \text{PulseWidth}}}$ $\qquad P_{ave} := \overrightarrow{\left(\left|I_{ave}\right| \cdot V_{dd}\right)}$



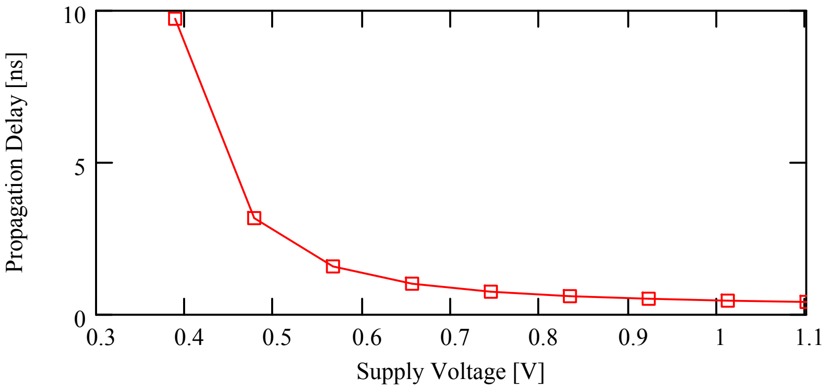

250
200
150
100
50
0
Current [uA]
0.5
1
1.5
2
2.5
3
3.5
4
4.5
Clock Period [ns]
Average Current


300
200
100
0
Power [uW]
200
400
600
800
1000
1200
Frequency [MHz]
Average Power

Part b:

$$\mathrm{PartBData} := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd.txt"})$$

$$\mathrm{SupplyVoltage} := \mathrm{PartBData}^{\langle 0\rangle}\,\mathrm{V} \qquad t_p := \mathrm{PartBData}^{\langle 4\rangle}\,\mathrm{s}$$



Interpolating the delay-Vdd curve to get the needed Vdd for a given delay:

```
$V_{dd\_opt} :=$ | for  $n \in 0..9$
               |     $x_n \leftarrow \mathrm{linterp}\left(\frac{-t_p}{s}, \frac{\mathrm{SupplyVoltage}}{V}, \frac{-\mathrm{PulseWidth}_n}{s}\right)$
               | $x \cdot V$
```

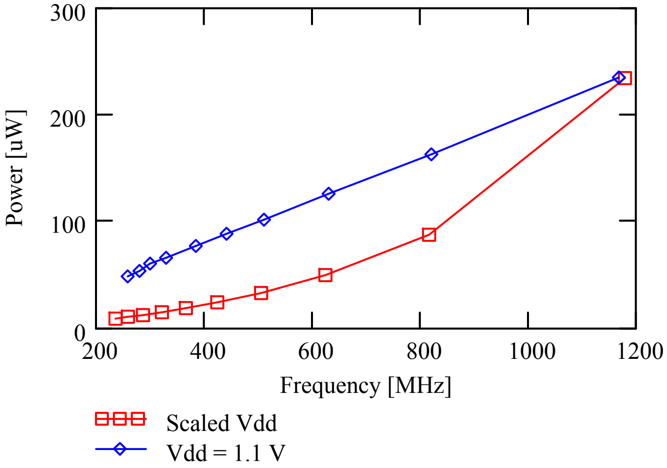
$$V_{dd\_opt} = \begin{pmatrix} 1.099 \\ 0.841 \\ 0.731 \\ 0.667 \\ 0.631 \\ 0.602 \\ 0.573 \\ 0.559 \\ 0.548 \\ 0.537 \end{pmatrix} V \qquad \mathrm{PulseWidth} = \begin{pmatrix} 0.424 \\ 0.612 \\ 0.801 \\ 0.989 \\ 1.178 \\ 1.366 \\ 1.555 \\ 1.743 \\ 1.932 \\ 2.12 \end{pmatrix} \mathrm{ns}$$

$VddOptData := READPRN(\text{"inverter\_delay\_fo4\_20\_vdd\_opt.txt"})$

$I_{ave\_opt} := \overrightarrow{\left|VddOptData^{\langle 4\rangle}\right|}\ A \qquad t_{p\_opt} := VddOptData^{\langle 7\rangle}\ s$

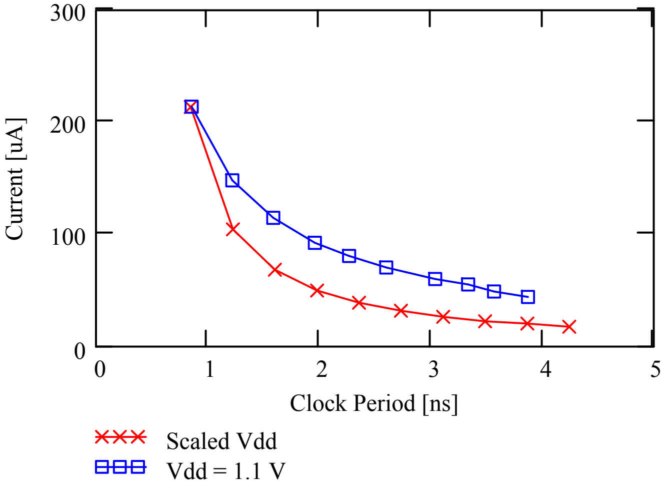
$freq_{opt} := \overrightarrow{\frac{1}{2 \cdot t_{p\_opt}}}$

```
P_ave_opt := | for  n ∈ 0 .. 9
             |   $x_n \leftarrow I_{ave\_opt_n} \cdot V_{dd\_opt_n}$
             | x
```



We can see from the graph that reducing the supply voltage reduces the speed to the amount required.

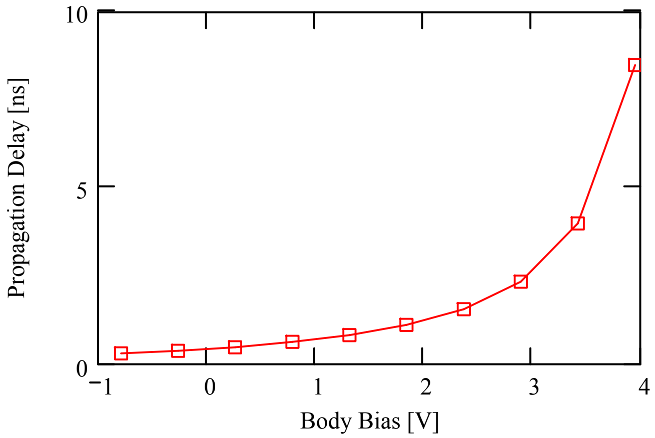
This results in a quadratic reduction in power as opposed to a linear decrease with only frequency scaling.



Part c:

$\text{BodyBiasData} := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vbb.txt"})$

$\text{BodyBias} := \text{BodyBiasData}^{\langle 0\rangle}\ \text{V}$ $\qquad\qquad$ $\text{BodyDelay} := \text{BodyBiasData}^{\langle 3\rangle}\ \text{s}$



We can see that the inverter propagation delay decreases as the threshold voltage is increased (reverse body bias) and increases when the threshold voltage is decreased (forward body bias).

From this graph, we can determine the required body bias needed to achive the desired propagation delay.

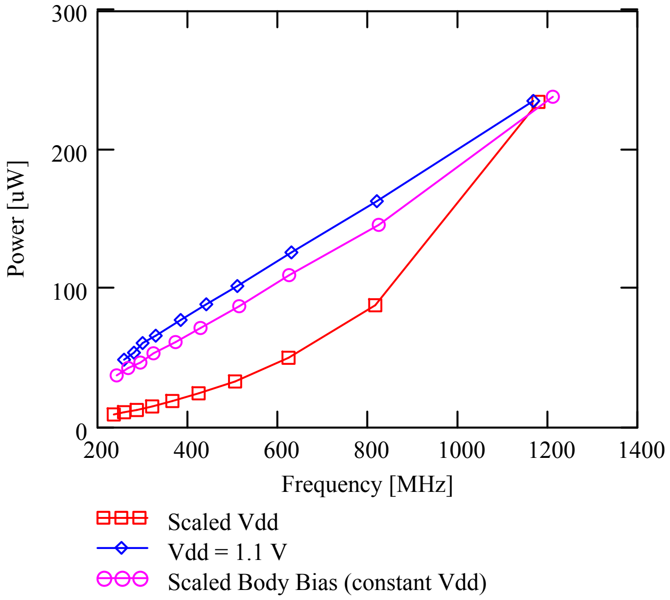
$$V_{body\_opt} := \left|\begin{array}{l} \text{for}\ \ n \in 0..9 \\ \quad x_n \leftarrow \text{linterp}\left(\dfrac{\text{BodyDelay}}{\text{s}}, \dfrac{\text{BodyBias}}{\text{V}}, \dfrac{\text{PulseWidth}_n}{\text{s}}\right) \\ x \cdot \text{V} \end{array}\right.$$

$$V_{body\_opt} = \begin{array}{|r|}\hline -0.062 \\ \hline 0.699 \\ \hline 1.223 \\ \hline 1.596 \\ \hline 1.907 \\ \hline 2.132 \\ \hline 2.358 \\ \hline 2.488 \\ \hline 2.615 \\ \hline 2.741 \\ \hline \end{array}\ \text{V} \qquad \text{PulseWidth} = \begin{array}{|r|}\hline 0.424 \\ \hline 0.612 \\ \hline 0.801 \\ \hline 0.989 \\ \hline 1.178 \\ \hline 1.366 \\ \hline 1.555 \\ \hline 1.743 \\ \hline 1.932 \\ \hline 2.12 \\ \hline \end{array}\ \text{ns}$$

$\mathrm{VbbOptData} := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vbb\_opt.txt"})$

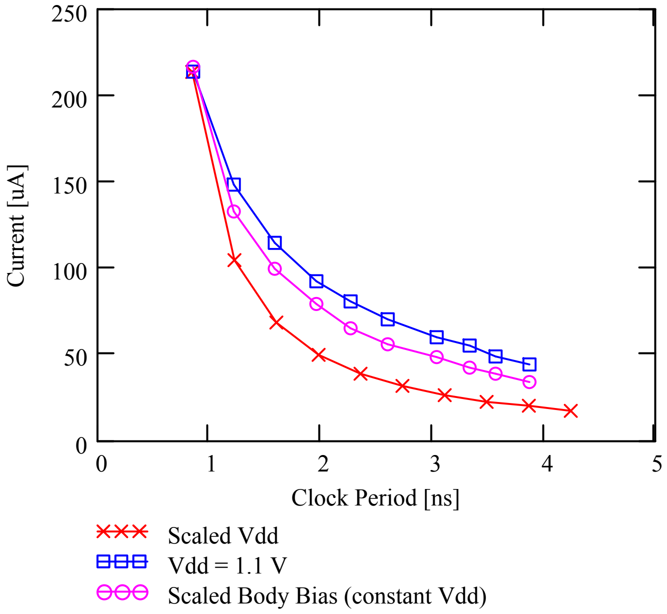
$I_{ave\_vbb\_opt} := \overrightarrow{\left|\mathrm{VbbOptData}^{\langle 4\rangle}\right|}\ \mathrm{A}$ $\qquad t_{p\_vbb\_opt} := \mathrm{VbbOptData}^{\langle 7\rangle}\ \mathrm{s}$ $\qquad V_{dd} = 1.1\ \mathrm{V}$

$freq_{opt\_vbb} := \overrightarrow{\dfrac{1}{2\cdot t_{p\_vbb\_opt}}}$ $\qquad P_{ave\_opt\_vbb} := \left|\begin{array}{l}\text{for } n \in 0..9\\ \quad x_n \leftarrow I_{ave\_vbb\_opt_n}\cdot V_{dd}\\ x\end{array}\right.$



We can see from this curve that we can increase the threshold voltage by varying the substrate bias, slowing down the transistor.

The increased threshold voltage also results in decreased leakage current, thus reducing the overall power.



Part d:

Forward biasing the transistor bulk:

$\mathrm{VddVbbNegData}_0 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg1.txt"})$

$\mathrm{VddVbbNegData}_1 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg2.txt"})$

$\mathrm{VddVbbNegData}_2 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg3.txt"})$

$\mathrm{VddVbbNegData}_3 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg4.txt"})$

$\mathrm{VddVbbNegData}_4 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg5.txt"})$

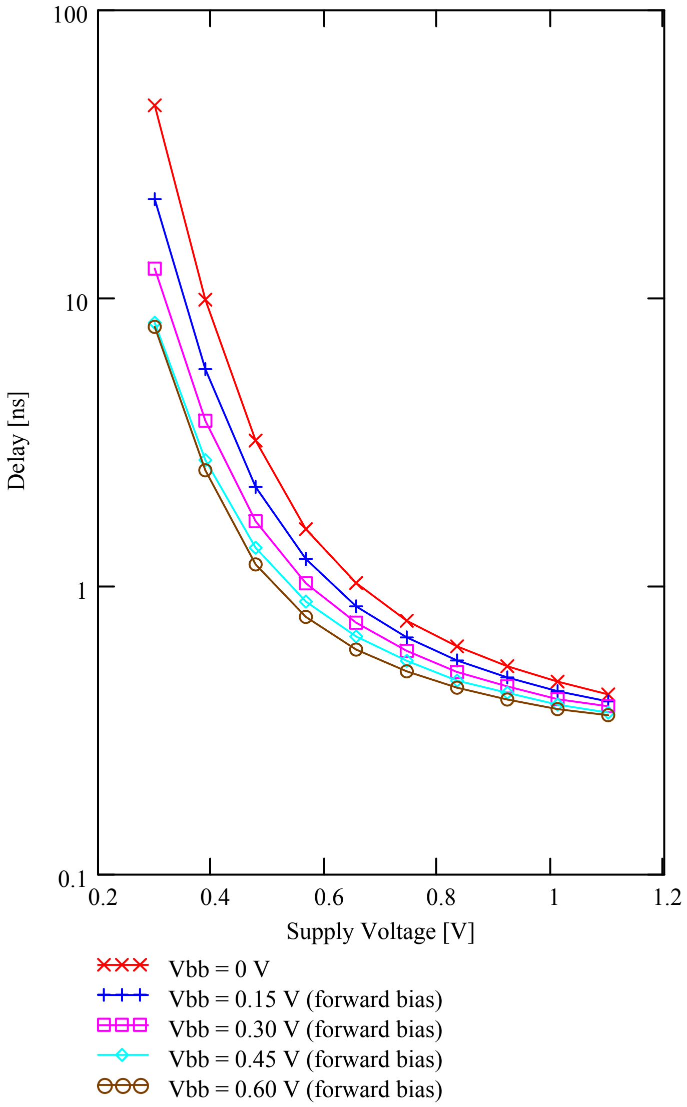
$\mathrm{VddVbbNegData}_5 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg6.txt"})$

$\mathrm{VddVbbNegData}_6 := \mathrm{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg7.txt"})$

$$\mathrm{Delay}_{\mathrm{db\_neg}} := \left|\begin{array}{l} \text{for } n \in 0..4 \\ \quad x_n \leftarrow \left(\mathrm{VddVbbNegData}_n\right)^{\langle 5\rangle}\,\mathrm{s} \\ x \end{array}\right.$$

$$V_{\mathrm{dd\_db\_neg}} := \left(\mathrm{VddVbbNegData}_0\right)^{\langle 0\rangle}\,\mathrm{V}$$

$$V_{\mathrm{bb\_db\_neg}} := \begin{pmatrix} 0 \\ -0.15 \\ -0.3 \\ -0.45 \\ -0.6 \\ -0.75 \\ -0.9 \end{pmatrix}\mathrm{V}$$

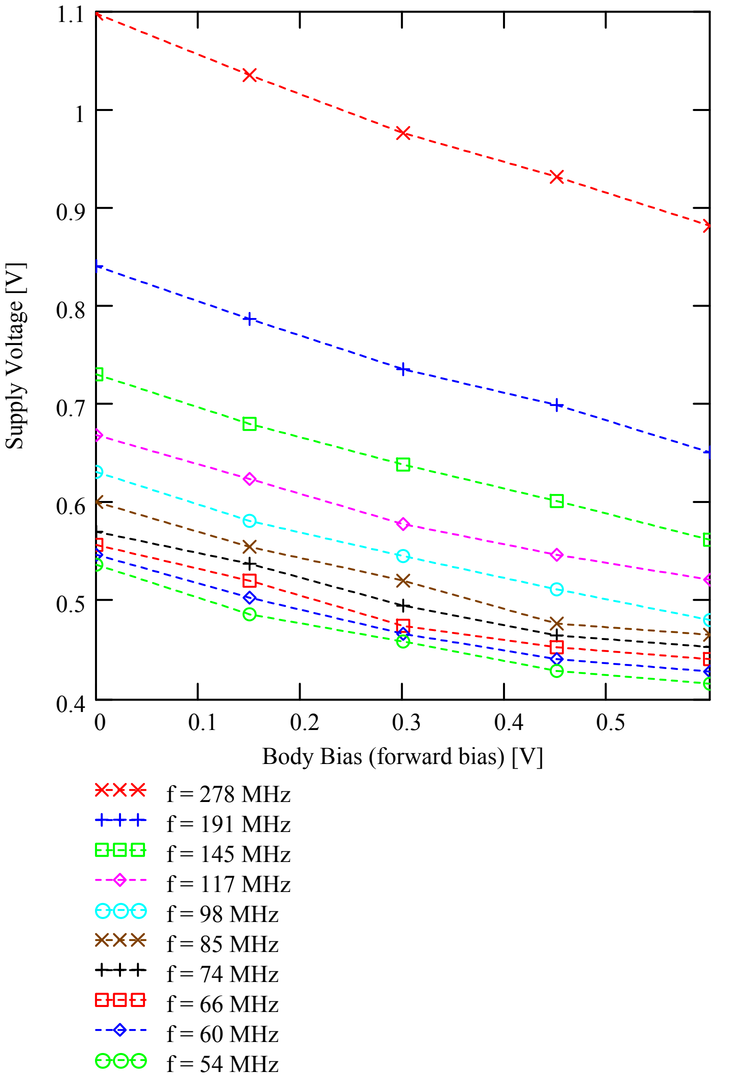


Note that by forward biasing the bulk of the transistors, the threshold voltage is reduced, increasing drive current and therefore speed.

We can use this additional slack to further reduce the supply voltage, and consequently the power.

Finding the corresponding Vdd and Vbb pairs for a given operating frequency:

$$V_{dd\_body\_opt}(\text{delay}) := \left| \begin{array}{l} \text{for } \; n \in 0..4 \\ \quad x_n \leftarrow \text{linterp}\left(\dfrac{-\text{Delay}_{db\_neg_n}}{s}, \dfrac{V_{dd\_db\_neg}}{V}, \dfrac{-\text{delay}}{s}\right) \\ x \cdot V \end{array} \right.$$



Using this data, we can then simulate the inverter chain and find the optimal Vdd-Vbb pair for each frequency point that will result in the lowest power.

$\text{VddVbbNegOptData}_0 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt1.txt"})$

$\text{VddVbbNegOptData}_1 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt2.txt"})$

$\text{VddVbbNegOptData}_2 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt3.txt"})$

$\text{VddVbbNegOptData}_3 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt4.txt"})$

$\text{VddVbbNegOptData}_4 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt5.txt"})$

$\text{VddVbbNegOptData}_5 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt6.txt"})$

$\text{VddVbbNegOptData}_6 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt7.txt"})$

$\text{VddVbbNegOptData}_7 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt8.txt"})$

$\text{VddVbbNegOptData}_8 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt9.txt"})$

$\text{VddVbbNegOptData}_9 := \text{READPRN}(\text{"inverter\_delay\_fo4\_20\_vdd\_vbb\_neg\_opt10.txt"})$

$$V_{\text{supply}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \left(\text{VddVbbNegOptData}_n\right)^{\langle 1\rangle} \\ x \cdot V \end{array}\right. \qquad V_{\text{substrate}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \left(\text{VddVbbNegOptData}_n\right)^{\langle 2\rangle} \\ x \cdot V \end{array}\right.$$

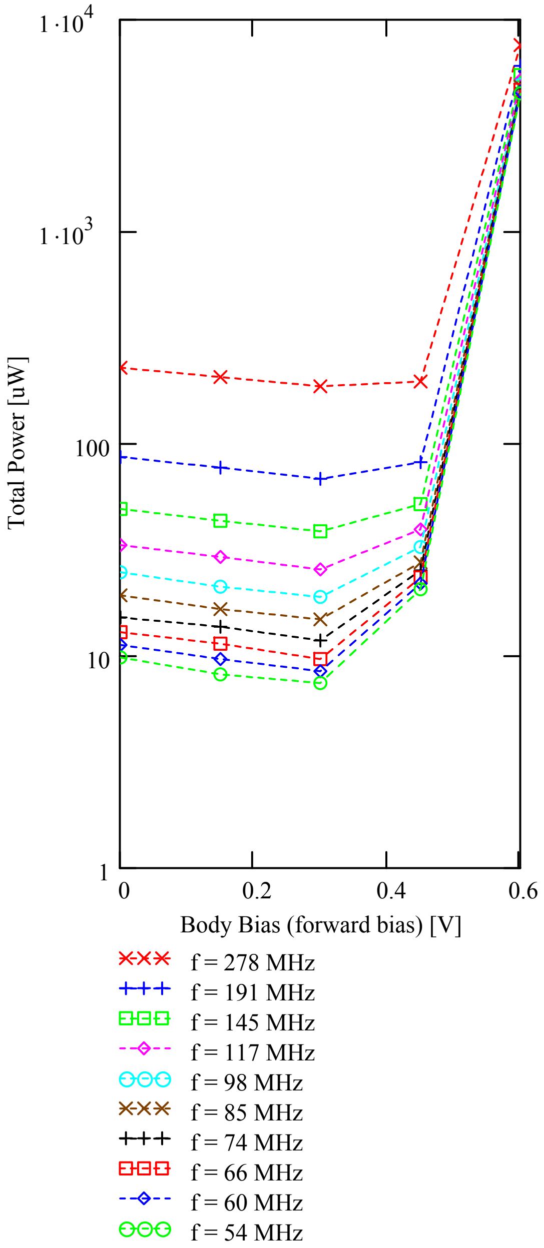
$$I_{\text{ave\_inv20}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left|\left(\text{VddVbbNegOptData}_n\right)^{\langle 4\rangle}\right|} \\ x \cdot A \end{array}\right. \qquad t_{\text{delay}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left|\left(\text{VddVbbNegOptData}_n\right)^{\langle 9\rangle}\right|} \\ x \cdot s \end{array}\right.$$

$$I_{\text{ave\_nbulk}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left|\left(\text{VddVbbNegOptData}_n\right)^{\langle 5\rangle}\right|} \\ x \cdot A \end{array}\right. \qquad I_{\text{ave\_pbulk}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left|\left(\text{VddVbbNegOptData}_n\right)^{\langle 6\rangle}\right|} \\ x \cdot A \end{array}\right.$$

$$P_{\text{supply}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left(V_{\text{supply}_n} \cdot I_{\text{ave\_inv20}_n}\right)} \\ x \end{array}\right. \qquad P_{\text{nbulk}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left|V_{\text{substrate}_n} \cdot I_{\text{ave\_nbulk}_n}\right|} \\ x \end{array}\right.$$

$$P_{\text{pbulk}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left|\left(\left(V_{\text{supply}_n} - V_{\text{substrate}_n}\right)\right) \cdot I_{\text{ave\_pbulk}_n}\right|} \\ x \end{array}\right.$$

$$P_{\text{total}} := \left| \begin{array}{l} \text{for } n \in 0..9 \\ \quad x_n \leftarrow \overrightarrow{\left(P_{\text{supply}_n} + P_{\text{nbulk}_n} + P_{\text{pbulk}_n}\right)} \\ x \end{array}\right.$$



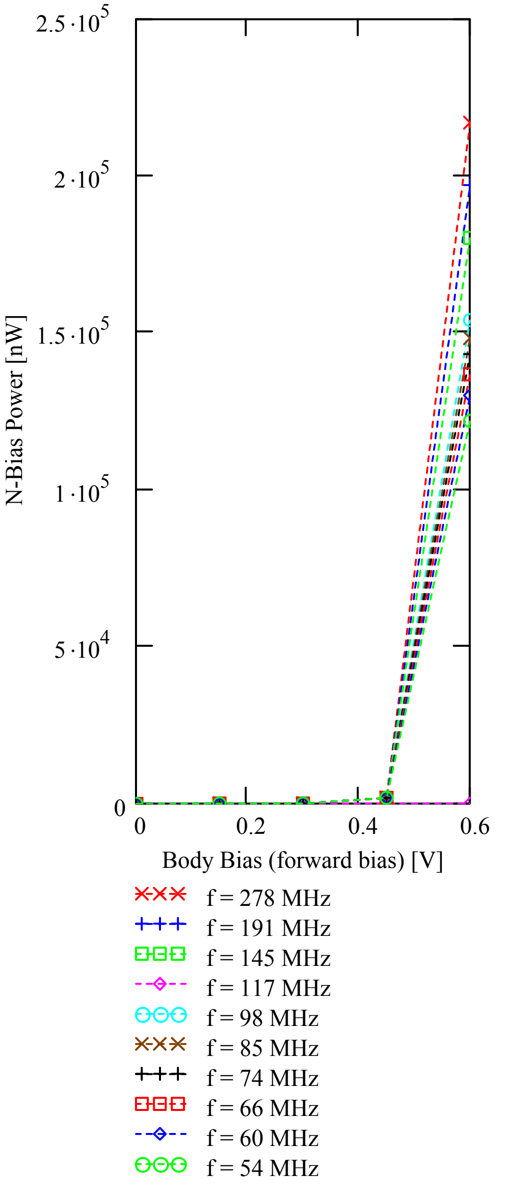
Note that as the substrate is forward biased, we can reduce the supply voltage. We expect the substrate current to increase with forward bias.

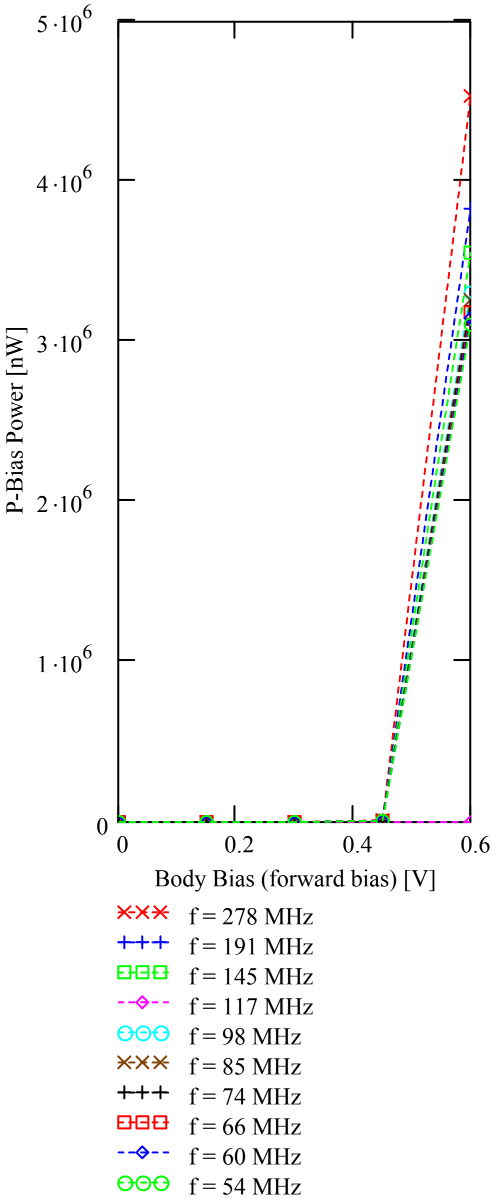
However, as seen from the plot, the power starts to increase as the S/D junction is biased in the forward active region.

This can be attributed to the fact that the substrate current is starts to dominate the overall power.

Thus we see that the minimum power occurs when the bulk is forward-biased at around 0.3 volts. This buys enough timing slack to reduce the supply voltage but still drawing a small enough current through the S/D junctions.

From these graphs, we can see the turn-on voltage of the S/D junctions:





Problem #2:

Part a:

The two circuit styles compared in the paper are the standard CMOS inverter circuit and the Charge Recovery (or Adiabatic) logic. The delay and energy models for each can be expressed as:

$$D_{CMOS} = \frac{C \cdot V_{dd}}{I} = \frac{C \cdot V_{dd}}{k \cdot \left(V_{dd} - V_{th}\right)^2} = R_{CMOS} \cdot C$$

$$E_{CMOS} = \frac{1}{2} \cdot C \cdot V_{dd}^{\;2}$$

$$D_{CR} = \frac{\pi}{\omega_d} + R_{CR} \cdot C$$

$$E_{CR} = \frac{1}{2} \cdot C \cdot \Delta V^2 \cdot \left[1 - e^{\left(\frac{-\pi \cdot \alpha}{\omega_d}\right)}\right]$$

$$\alpha = \frac{R_{CR}}{2 \cdot L}$$

SInce both energy expressions are dependent on the square of the supply voltage (or in the case of CR, a fraction of the supply voltage), we can use voltage scaling to quadratically reduce the energy.

Part b:

Without considering switching power, CMOS logic dissipates less energy until the voltage is scaled down to 1V. The point where CR starts to consume less energy has a 10X reduction in frequency and a 150X reduction in power.

With switching power considered, CMOS logic now dissipates less energy over a wider range. The voltage crossover point is reduced from 1V to voltages close to the transistor threshold voltage. Thus, with switching power considered, CR starts to consume less energy when the power is reduced 2500X at a frequency of 100X the peak.

Part c:

For low power designs with increased computation requirements such as portable phones, computers, PDAs, and the like, CMOS logic will be the better choice.

However, for ultra-low power, very low performance requirements, such as watches, simple sensors without much computational requirements, medical and biological implants, and the like, CR logic might be the logic syle of choice.