ECE 327 Solution to Final

2019t1 (Winter)

		Total Marks	Approx. Time	Page
Q1	Hardware Design	20	25	1
Q2	DFD	20	25	4
Q3	Performance	16	25	7
Q4	Elmore Delay	16	25	10
Q5	Work	14	20	14
Q6	Clock Gating	14	20	17
Totals		100	140	

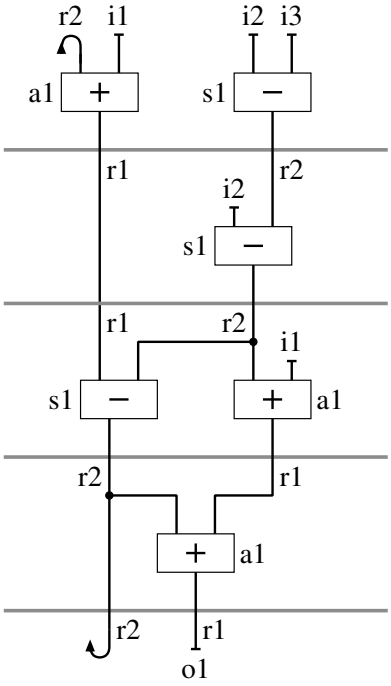
Q1 (20 Marks) Hardware Design

(*estimated time: 25 minutes*)

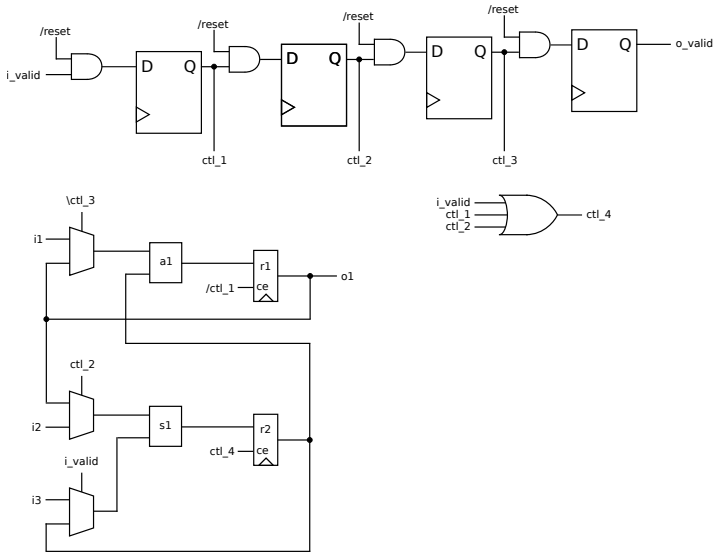
Design the hardware (draw a diagram of the hardware) to implement the dataflow diagram below.

NOTES:

1. Your system shall support an ASAP parcel schedule
2. Your drawing shall include both datapath and control circuitry, including the gates for the state machine.
3. Your drawing shall use the standard building blocks of RTL hardware: adders, subtracters, multiplexers, flip-flops, AND gates, OR gates, *etc.*
4. Marks will be earned for functional correctnes, simplicity and elegance of the design, and neatness of the drawing.



Answer:


/reset
/reset
/reset
/reset
i_valid
D
Q
D
Q
D
Q
D
Q
o_valid
ctl_1
ctl_2
ctl_3
\ctl_3
i1
a1
r1
/ctl_1
ce
o1
ctl_2
i2
s1
r2
ctl_4
ce
i_valid
i3
i_valid
ctl_1
ctl_2
ctl_4

Marking:

1 mark *2 registers for datapath*

1 mark *1 adder*

1 mark *1 subtracter*

1 mark *3 inputs, 1 output*

2 marks *4 flops for state*

2 marks *correct reset circuitry for state*

2 marks *r1 chip-enable=0 in cycle 1*

2 marks *r2 chip-enable=0 in cycle 3*

2 marks *if use valid-bit encoding: r2 chip-enable=0 when system contains a bubble (e.g., just after reset)*

2 marks *a1.src2 mux between r1 (cycle 3) and i1 (all other cycles)*

2 marks *s1.src1 mux between r1 (cycle 3) and i2 (all other cycles)*

2 marks *s1.src2 mux between i3 (cycle 0) and r2 (all other cycles)*

-2 marks *incorrect hardware for inter-parcel variable*

-1 mark *each mistake not listed above*

Q2 (20 Marks) DFD

(*estimated time: 25 minutes*)

You have graduated and are starting your dream job of designing hardware. Your first task is to design and analyze a dataflow diagram for the equation:

$$z = F^3(a) + 2F^2(a) \bullet F(a) \bullet F(a)$$

NOTES:

1. All of the signals are 16 bit `unsigned`.
2. F is a top-secret function (rumoured to be a key component in Waterluvian filters). The only information you have been given about F is:
   - F is purely combinational.
   - The input and output are both 16 bit `unsigned`.
   - $F^2(a) = F(F(a))$ (A chain of 2 Fs).
   - $F^3(a) = F(F(F(a)))$ (A chain of 3 Fs).
3. The area estimates of the components are:

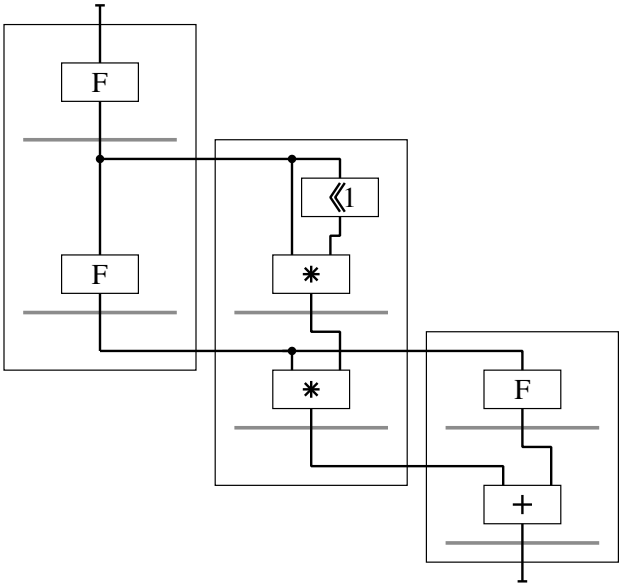
Adder	A
F	A
Multiplier	3A

4. The maximum area of the datapath is 9A.
5. Optimization goals, in order of decreasing importance
   - Minimize clock period
   - Maximize throughput
   - Minimize area of datapath components (sum of the area of adders, Fs, and multipliers)
   - Minimize number of registers
   - Minimize latency
6. Algebraic optimizations *are* allowed, as long as the final output (z) is correct.
7. You may ignore the area consumed by multiplexers.
8. You do *not* need to do any allocation.

Q2a (17 Marks) Design

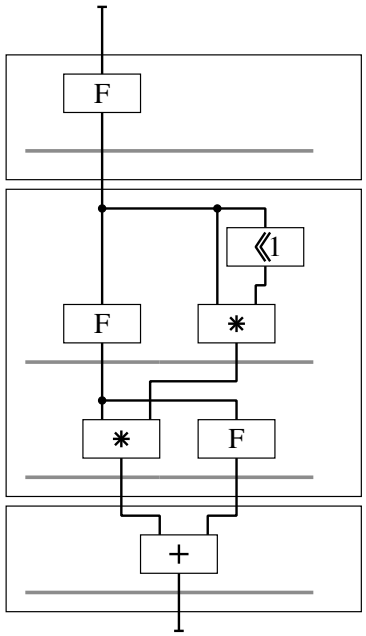
Draw your dataflow diagram.

Answer:



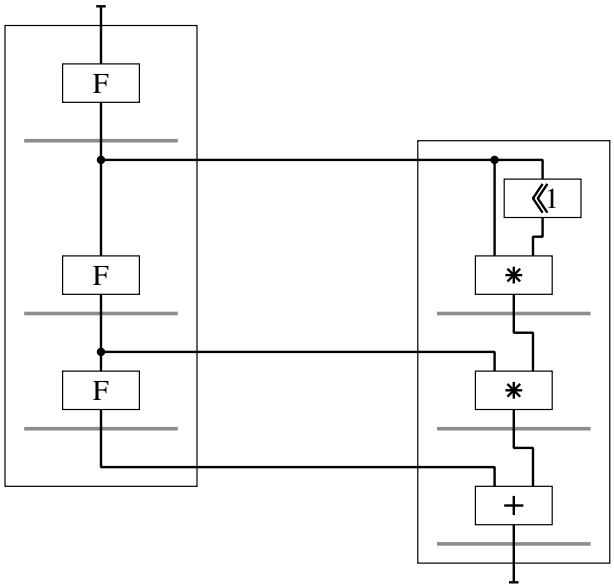
Clock period	*max(Mul, F) + Flop*
Throughput	*1/2*
Adders	*1*
Fs	*2*
Mults	*1*
Datapath area	*6*
Registers	*3*
Latency	*4*

17 marks



Clock period	*max(Mul, F) + Flop*
Throughput	*1/2*
Adders	*1*
Fs	*2*
Mults	*1*
Datapath area	*6*
Registers	*4*
Latency	*4*

15 marks



Clock period	*max(Mul, F) + Flop*
Throughput	*1/3*
Adders	*1*
Fs	*1*
Mults	*1*
Datapath area	*5*
Registers	*2*
Latency	*4*

13 marks

Marking:

- **-4 marks** *incorrect functionality*
- **-2 marks** *signal skips over clock cycle boundary*
- **-2 marks** *extra datapath area or register*
- **-2 marks** *extra latency*
- **-2 marks** *suboptimal clock period*
- **-2 marks** *suboptimal throughput*

Q2b (3 Marks) Analysis

See the example answers for part a

Q3 (16 Marks) Performance

(*estimated time: 25 minutes*)

You have heard rumours that Waterluvian filters can be helpful in medical imaging to diagnose mysterious abdominal pains. You have decided to jump on this latest fad and create a startup company "Waterluvian advanced systems for tomagraphy engineering" (Waste).

Your task is to choose which FPGA chip to use for your Waterluvian filter. Using an FPGA chip with hardware multipliers increases performance, but also increases cost.

NOTES:

1. Performance of Waterluvian filters is measured in megapixels per second (MPPS).
2. Performance increases by 4 MPPS for each multiplier on the FPGA.
3. The average price for a Waterluvian filter increases by 1% for each increase of 1 MPPS in performance.
4. The cost of an FPGA without any multipliers is \$10.
5. The cost of the FPGA increases by \$0.20 for each multiplier.
6. Profit is measured as the price that you charge for the Waterluvian filter minus the cost of the FPGA chip. (You shall ignore non-recurring engineering (NRE) costs.)
7. **For full marks, you must justify your answer.**

Which option will give you a higher profit for your Waterluvian filter: using an FPGA without any multipliers or using an FPGA with 20 multipliers?

Answer:

1. Equations

$$\begin{aligned} Cost_0 &= 10 \\ Perf_{20} &= 4 \bullet 20 + Perf_0 \\ Cost_{20} &= 0.20 \bullet 20 + Cost_0 \\ Price_{20} &= Price_0 \bullet 1.01^{Perf_{20} - Perf_0} \\ Profit_{20} &= Price_{20} - Cost_{20} \end{aligned}$$

2. *Solve the equations, working toward $Profit_0$ and $Profit_{20}$.*

$$\begin{aligned} Profit_0 &= Price_0 - Cost_0 \\ &= Price_0 - 10.00 \end{aligned}$$

$$\begin{aligned} Perf_{20} &= 4 \bullet 20 + Perf_0 \\ &= 80 + Perf_0 \\ Price_{20} &= Price_0 \bullet 1.01^{Perf_{20} - Perf_0} \\ &= Price_0 \bullet 1.01^{80 + Perf_0 - Perf_0} \\ &= Price_0 \bullet 1.01^{80} \\ &= Price_0 \bullet 2.2167 \\ Cost_{20} &= 0.20 \bullet 20 + Cost_0 \\ &= 0.20 \bullet 20 + 10 \\ &= \$14.00 \\ Profit_{20} &= Price_{20} - Cost_{20} \\ &= Price_0 \bullet 2.2167 - 14.00 \end{aligned}$$

3. *We need to compare $Profit_0$ and $Profit_{20}$. We have two options:*
   - *Calculate the* ratio *between the two profits:* $\frac{Profit_{20}}{Profit_0} > 1$
   - *Calculate the* difference *between the two profits:* $Profit_{20} - Profit_0 > 0$

   Because the equations for $Profit_0$ and $Profit_{20}$ are of the form $Price_0 - x$, calculating the difference will probably give a simpler expression. In comparison, if the equations had been of the form $x \bullet Price_0$, then we would have chosen to use the ratio, because $Price_0$ would cancel out.
4. *Solve for $Profit_{20} - Profit_0$. If the difference is greater than 0, then the system with 20 multipliers has a higher profit.*

$$\begin{aligned} \mathit{Profit}_{20} - \mathit{Profit}_0 &\geq 0 \\ (2.2167 \bullet \mathit{Price}_0 - 14.00) - (\mathit{Price}_0 - 10.00) &\geq 0 \\ 1.2167 \bullet \mathit{Price}_0 - 4 &\geq 0 \\ \mathit{Price}_0 &\geq \frac{4}{1.2167} \\ \mathit{Price}_0 &\geq \$3.29 \end{aligned}$$

If the price of the FPGA chip with 0 multipliers is more than $3.29, then the more profitable option is to use the chip with 20 multipliers.

5. *The cost of the chip with 0 multipliers is $10, so it is reasonable to assume that the price that is charged for the Waterluvian filter is more than $3.29, therefore the best choice is the chip with 20 multipliers.*

Marking:

3 marks *price increases exponentially with performance*

3 marks *cost and performance increase linearily with number of multipliers*

3 marks *solve for condition under which* $\mathit{Profit}_{20} \geq \mathit{Profit}_0$ *(or vice-versa)*

3 marks *the answer depends on the price of the chip with 0 multipliers*

4 marks *correct answer and quality of justification*

Q4 (16 Marks) Elmore Delay

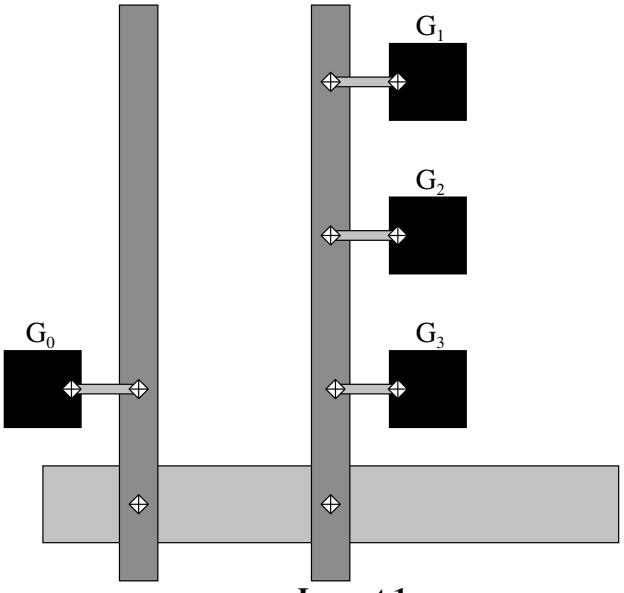
(*estimated time: 25 minutes*)

To maximize the clock speed of your Waterluvian filter, you are doing place-and-route of the FPGA cells by hand. Which layout below will have the higher clock speed?

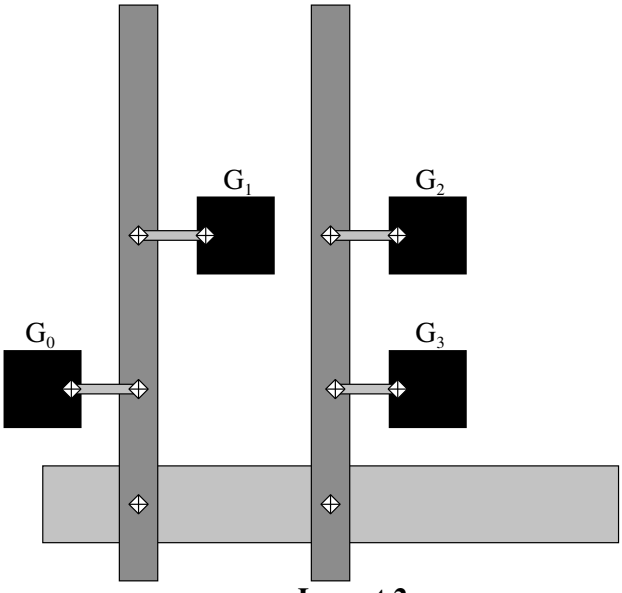
NOTES:

1. The source gate is G_0, the other gates ($G_1 - G_3$) are destinations.
2. If you do not have sufficient information to answer the question, find an equation for the relationship between C_X, C_Y, and C_L such that the two layouts have the same delay (An example equation is: $C_X = C_Y + C_L$).
3. The capacitance of a wire is independent of distance and location on the wire.
4. **For full marks, you must justify your answer.**

Symbol	Description	Capacitance	Resistance
	Interconnect level 3	C_X	0
	Interconnect level 2	C_Y	0
	Interconnect level 1	0	0
	Gate	C_L	0
	Switchbox	0	R



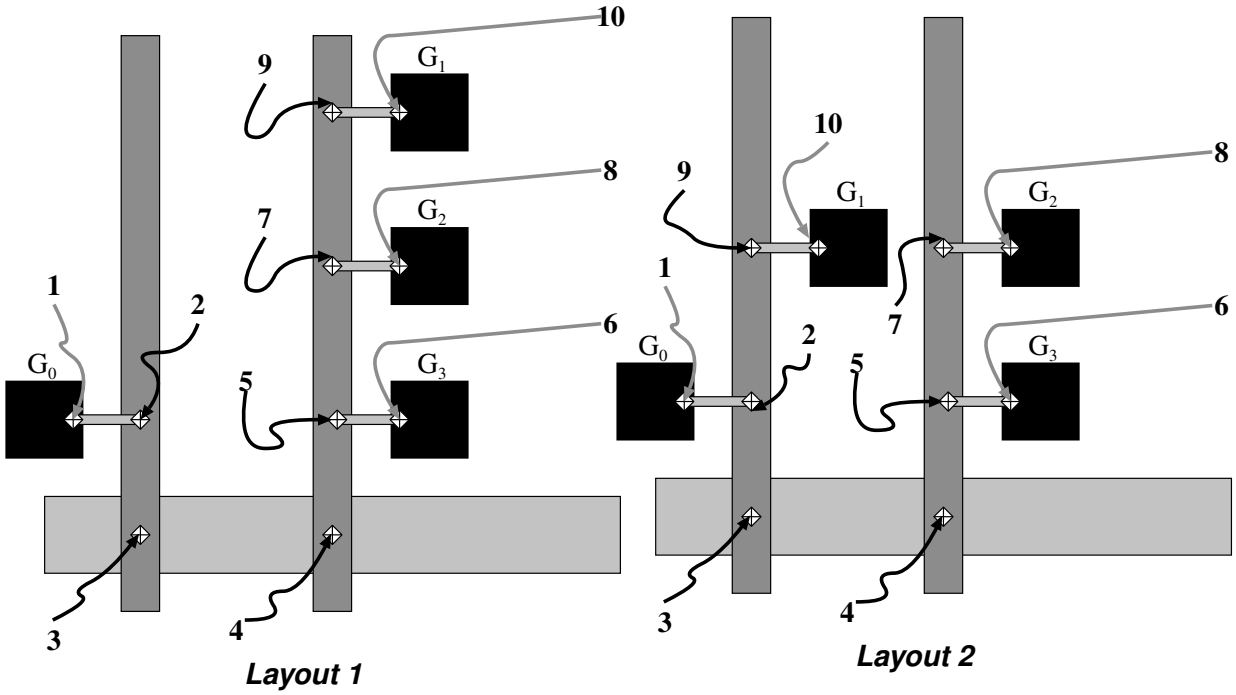
Layout 1



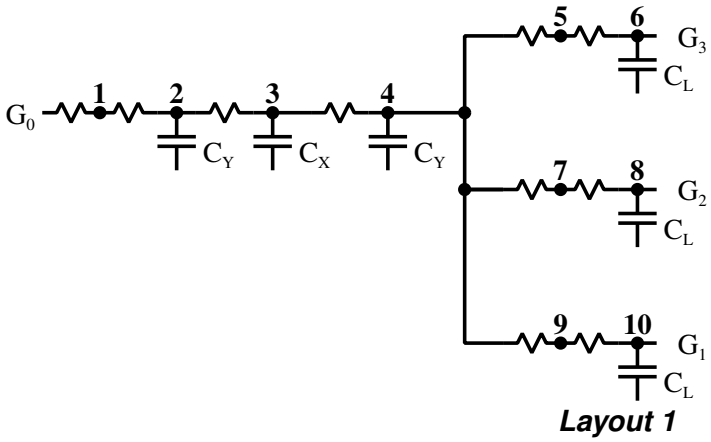
Layout 2

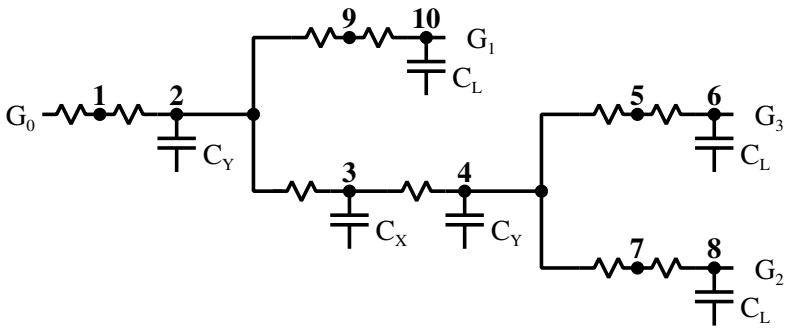
Answer:

1. Label the nodes



2. Draw RC networks





Layout 2

3. *Find slowest node in each network*

   Layout 1 G_1, G_2, *and* G_3 *all have the same delay*

   Layout 2 G_2, *and* G_3 *have the same delay and are slower than* G_1.

4. *Compare a slowest node in Layout 1 with a slowest node in Layout 2:* G_3 *in Layout 1 with* G_3 *in Layout 2.*

5. *The resistors on the path to* G_3 *are the same in both layouts:* R_1, R_2, R_3, R_4, R_5, *and* R_6

   Each of the resistors R_1, R_2, R_5, *and* R_6 *has the same set of downstream capacitors in both layouts. Therefore, we can ignore them and examine just* R_3 *and* R_4.

   We can see from the layouts that the difference in the layouts for R_3 *and* R_4 *are the location of the* C_L*s. At this point, we can determine that* Layout 2 will be faster*, because* R_3 *and* R_4 *have only* $2C_L$*, in comparison to Layout 2, where these resistors have* $3C_L$ *downstream.*

   Alternatively, we can take a more detailed approach and write the expressions for the delay of R_3 *and* R_4.

   Layout 1

$$\begin{array}{llllll} R(& C_X+ & C_Y+ & 3C_L) & (\textit{from } R_3) \\ R(& & C_Y+ & 3C_L) & (\textit{from } R_4) \end{array}$$

   Layout 2

$$\begin{array}{llllll} R(& C_X+ & C_Y+ & 2C_L) & (\textit{from } R_3) \\ R(& & C_Y+ & 2C_L) & (\textit{from } R_4) \end{array}$$

   For complete completeness, the complete expressions for the delay are:

   Layout 1

$$\begin{array}{llllll} 2R(& C_Y+ & C_X+ & C_Y+ & 3C_L) & (\textit{from } R_1+R_2) \\ R(& & C_X+ & C_Y+ & 3C_L) & (\textit{from } R_3) \\ R(& & & C_Y+ & 3C_L) & (\textit{from } R_4) \\ 2R(& & & & C_L) & (\textit{from } R_5+R_6) \end{array}$$

   Layout 2

$$\begin{array}{llllll} 2R(& C_Y+ & C_X+ & C_Y+ & 3C_L) & (\textit{from } R_1+R_2) \\ R(& & C_X+ & C_Y+ & 2C_L) & (\textit{from } R_3) \\ R(& & & C_Y+ & 2C_L) & (\textit{from } R_4) \\ 2R(& & & & C_L) & (\textit{from } R_5+R_6) \end{array}$$

6. *Layout 2 is faster.*

Marking:

4 marks *Draw RC networks*

3 marks *Goal is to compare Elmore delay of slowest node in each layout*

6 marks *Correct delay expressions*

3 marks *Correct analysis and conclusion*

Q5 (14 Marks) Work

(*estimated time: 20 minutes*)

What is the minimum set of parameters needed to calculate the impact on the amount of work your battery powered Waterluvian filter can accomplish on 1 battery charge if you decrease VDD by 20%?

NOTES:

1. The impact on work is measured as a percentage, *e.g.*, "with a 20% decrease in VDD, the filter can process at most 10% fewer pixels".
2. All of the parameters below refer to the original system, before VDD is reduced.
3. If there are multiple sets that have the same number of parameters, choose the set with the minimum sum of the indices (*e.g.*, 1: Clock frequency + 2: Cycles per instruction = 3 is better than 1: Clock frequency + 3: Power = 4).
4. **For full marks, you must justify your answer.**

		Needed?	
Index	**Parameter**	**Yes**	**No**
1	Clock frequency	☐	☐
2	Cycles per instruction	☐	☐
3	Power	☐	☐
4	Energy in battery	☐	☐
5	VDD	☐	☐
6	Threshold voltage	☐	☐
7	Number of instructions to process one pixel	☐	☐
8	Millions of instructions per second	☐	☐

Answer:

1. *When we lower VDD, we increase the delay through a circuit and we reduce power consumption.*
2. *Increasing the delay will force us to reduce the clock frequency.*
3. *Reducing power consumption will allow the Waterluvian filter to run for a longer time on one battery charge.*
4. *Equations:*

$$\begin{aligned} Delay &\propto \frac{VDD}{(VDD - Vt)^2} \\ ClkFreq &\propto \frac{1}{Delay} \\ \frac{Pixels}{Battery} &= \frac{Pixels}{Instr} \bullet \frac{Instrs}{Cycle} \bullet \frac{Cycles}{Sec} \bullet \frac{Secs}{Joule} \bullet \frac{Joules}{Battery} \end{aligned}$$

We are solving for a percentage change, so cancel out all parameters that don't change

$$\begin{aligned} \frac{Pixels}{Battery} &\propto \frac{Cycles}{Sec} \bullet \frac{Secs}{Joule} \\ \frac{Cycles}{Sec} &\propto \frac{(VDD - Vt)^2}{VDD} \end{aligned}$$

Assume that the changes in short-circuiting and leakage power are negligible in comparison to the change in switching power.

$$\begin{aligned} \frac{Secs}{Joule} &\propto \frac{1}{VDD^2} \\ \frac{Pixels}{Battery} &\propto \frac{(VDD - Vt)^2}{VDD} \bullet \frac{1}{VDD^2} \end{aligned}$$

Marking:

1 mark *Delay and/or clock frequency is dependent on VDD and Vt*
1 mark *Power is dependent on VDD*
1 mark *Assume short-circuiting and leakage power are negligible*
1 mark *Solving for ratio, so cancel out values that do not change*
1 mark *Clock frequency is irrelevant*
1 mark *CPI is irrelevant*
1 mark *Power is irrelevant*
1 mark *Energy in battery is irrelevant*
1 mark *VDD is needed*
1 mark *Vt is needed*

1 mark *Number of instructions to process one pixel is irrelevant*

1 mark *MIPS is irrelevant*

3 marks *Clear and correct logical argument*

Q6 (14 Marks) Clock Gating

(*estimated time: 20 minutes*)

Your manager has told you that she wants to use clock gating to reduce power consumption. In your Waterluvian filter, there are 6 candidate modules to which you could apply clock gating. But, your manager has said that there is time to design and implement clock gating for only one module. Which one of the modules below should you choose?

NOTES:

1. The time to design and implement clock gating is the same for each module.
2. Each row has at most 2 unique values (*e.g.*, 1.2 and 1.4 for VDD). To make it easier to see the differences between the modules, gray boxes are used to distinguish the two values.
3. **For full marks, you must justify your answer.**

	Module					
	1	**2**	**3**	**4**	**5**	**6**
VDD (V)	1.2	1.2	1.4	1.4	1.4	1.4
Clock frequency (MHz)	500	500	500	600	600	600
Area (FPGA cells)	200	200	200	300	300	200
Latency (cycles)	6	6	6	6	6	6
Activity factor	0.25	0.25	0.25	0.30	0.25	0.25
Average number of contiguous parcels	10	10	10	10	10	10
Average number of contiguous bubbles	70	14	70	14	14	14

Answer:

Our goal is to determine which system will have the maximum power reduction with clock gating. We will look at the current power of the system and the percentage of power that can be reduced by clock gating.

1. *Assumptions*
   - *Static power is negligible*
   - *Short circuiting power is negligible*
   - *Clock gating scheme is 100% effective*
   - *Power of clock enable state machine will be the same for all options*
2. *Calculate PctBusy for each system*

$$PctBusy = \frac{NumPcls + Latency}{NumPcls + NumBubbles}$$

	Module					
	1	***2***	***3***	***4***	***5***	***6***
VDD (V)	*1.2*	*1.2*	*1.4*	*1.4*	*1.4*	*1.4*
Clock frequency (MHz)	*500*	*500*	*500*	*600*	*600*	*600*
Area (FPGA cells)	*200*	*200*	*200*	*300*	*300*	*200*
Activity factor	*0.25*	*0.25*	*0.25*	*0.30*	*0.25*	*0.25*
PctBusy	*0.20*	*0.67*	*0.20*	*0.67*	*0.67*	*0.67*

3. *Compare options that differ in only one parameter, eliminate the option that either has lower original power or less potential for power reduction from clock gating.*
   - *1 is better than 2, because 1 has a lower PctBusy, which means it has more potential for power reduction with clock gating.*
   - *3 is better than 1, because 3 has a higher VDD, which gives it a higher original power.*
   - *5 is better than 6, because 5 has more area, which gives it a higher original power.*
   - *4 is better than 5, because 4 has a higher activity factor, which gives it a higher original power*
4. *We are now left with two options: 3 and 4. Option 3 has a lower PctBusy, which means that clock gating can save a higher* percentage *of power than with Option 4. However, Option 4 has a higher original power, which means that the absolute amount of power saved might be higher than Option 3.*

   For each option, we calculate the relative power and the percentage of time that the clock can be turned off.

   The equation for power saved does not include VDD, because both options have the same VDD.

$$\begin{aligned} \textit{PwrSave} &= (1-\textit{PctBusy}) \bullet \textit{PwrSwitch} \\ &\propto (1-\textit{PctBusy}) \bullet \textit{ActFact} \bullet \textit{ClkFreq} \bullet \textit{Area} \\ \textit{PwrSave}_3 &\propto (1-0.20) \bullet 0.25 \bullet 500 \bullet 200 \\ &\propto 20\,000 \\ \textit{PwrSave}_4 &\propto (1-0.67) \bullet 0.30 \bullet 600 \bullet 300 \\ &\propto 18\,000 \end{aligned}$$

5. *Option 3 saves more power.*

Marking:

2 marks *Assumptions*

2 marks *Find module for which clock gating will save the maximum amount of power*

2 marks *Power saved is* $(1-\textit{PctBusy}) \bullet \textit{PwrOrig}$

2 marks *Calculate PctBusy*

2 marks $\textit{PwrOrig} \propto \textit{ActFact} \bullet \textit{ClkFreq} \bullet \textit{Area} \bullet \textit{Vdd}^2$

4 marks *Choice and justification for best module*