

Nom : \_\_\_\_\_

Matricule : \_\_\_\_\_

INF3500 : Conception et réalisation de systèmes numériques

Examen intra #2 – 22 mars 2021

1. Conditions de l'examen

- Cet examen est calibré pour une durée de 60 minutes. Une période additionnelle de 6 heures est allouée pour la remise de l'examen dans Moodle.
- L'examen est à livres ouverts, toutes les sources sont permises. Si vous utilisez du matériel trouvé en ligne ou ailleurs, citez bien vos sources dans votre réponse.
- L'examen doit être fait individuellement. Toute forme de communication avec d'autres personnes que le surveillant est interdite pendant l'examen. Une attestation à cet effet vous sera demandée dans la dernière question.
- La pondération de l'examen est de 10 %. La valeur de chaque question est indiquée.

2. Communication avec le surveillant pendant l'examen

- Le surveillant sera disponible de façon intermittente via le serveur Discord du cours pour aider à résoudre les problèmes techniques. Si le serveur Discord du cours ne fonctionne pas, envoyez un courriel à pierre.langlois@polymtl.ca.
- Ne posez pas de questions au surveillant concernant les questions de l'examen. Aucune réponse ne sera donnée. En cas de doute sur le sens d'une question, faites une supposition raisonnable, énoncez-la clairement dans votre réponse et poursuivez.

3. À propos de la remise

- Remettez vos réponses dans l'activité devoir de Moodle au plus tard à 23 h 59 (heure de Montréal).
- Les réponses en retard sont permises mais seront pénalisées de 1 point (10 %) par minute après minuit.
- Les réponses doivent être remises dans un seul fichier .pdf. Dans le document .pdf, en plus des réponses tapées et des tableaux remplis, il est acceptable d'inclure des captures d'écran, des diagrammes faits à la main, du texte écrit à la main etc.
- Sauvegardez souvent votre travail. Déposez régulièrement votre fichier le plus récent dans Moodle.
- Pour les images incluses, assurez-vous :
  - qu'elles soient orientées correctement, bien cadrées, bien rognées et bien alignées;
  - qu'elles soient nettes partout, que l'éclairage soit uniforme et que le contraste soit suffisamment élevé;
  - d'en limiter la taille – le document au complet doit faire moins de 5 Mo.
- Commencez la réponse à chaque question sur une nouvelle page.

À l'usage des correcteurs :

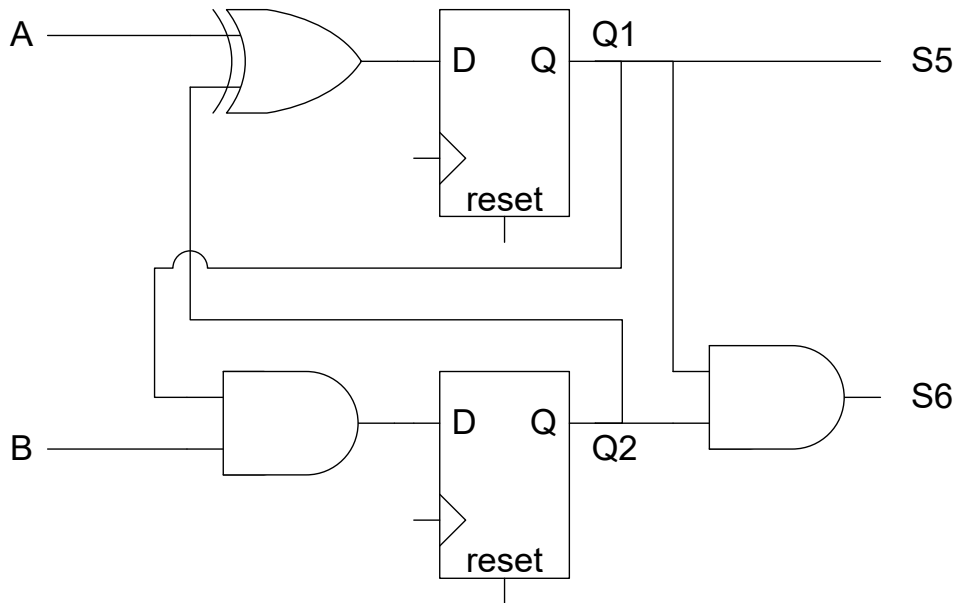
Q1	Q2	Q3	Q4	Q5 ok ?	Total sur 10

**Question 1. (2 points : 12 minutes)**

Considérez le circuit séquentiel suivant. Les ports A et B sont des entrées du système, et les ports S5 et S6 sont des sorties.

Donnez un modèle VHDL correspondant à ce circuit.

Remettez votre code dans votre fichier .pdf.



{Exemple de code avec un formatage agréable à lire :}

```
-- commentaires et déclarations

entity C2_202101_Q1 is
  port (
    -- une liste de ports, etc.
  );
end C2_202101_Q1;

-- d'autre code ici
--
-- La police Courier New pour du code n'est pas trop mal à lire dans un fichier pdf.
--
-- Pour les indentations, *** il faut éviter les tabulations **.
-- Il faut plutôt privilégier les espaces : 4 espaces par niveau de tabulation.
--
-- Vous pouvez utiliser votre simulateur de VHDL préféré pour confirmer la syntaxe,
-- mais ce n'est pas essentiel.
--
```

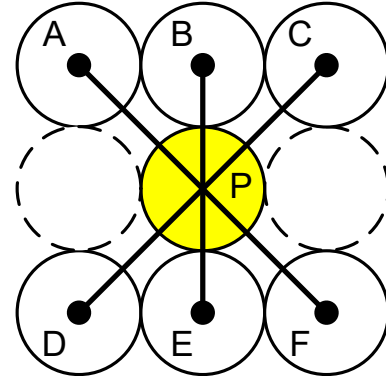
**Question 2. (3 points : 18 minutes)**

L'algorithme ELA (Edge-based Line Average) double la résolution verticale d'une image en créant une nouvelle ligne de pixels entre chaque paire de lignes existantes.

Pour chaque paire de lignes, l'algorithme prend trois pixels de la ligne du haut (A, B, C) et trois pixels de la ligne du bas (D, E, F), et calcule un nouveau pixel P de la nouvelle ligne intermédiaire.

Les pixels sont représentés en tons de gris sur 8 bits.

L'algorithme accepte en entrée les valeurs des six pixels A, B, C, D, E et F, et produit en sortie le pixel P. Il calcule trois différences absolues :  $|A - F|$ ,  $|B - E|$  et  $|C - D|$ . La plus petite de ces trois valeurs indique la direction d'interpolation. La valeur de P est alors donnée par la moyenne des deux pixels le long de la direction d'interpolation. Par exemple, si  $|A - F|$  a la plus petite valeur, alors  $P = (A + F) / 2$ . Si  $|B - E|$  a la plus petite valeur, alors  $P = (B + E) / 2$ . Si  $|C - D|$  a la plus petite valeur, alors  $P = (C + D) / 2$ .

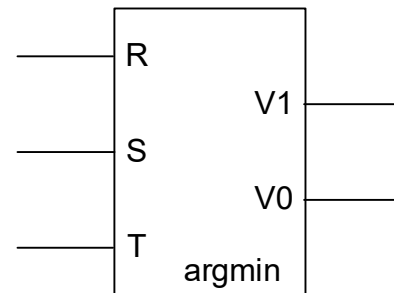


Donnez le diagramme d'un chemin des données pour implémenter l'algorithme ELA.

Supposez que les entrées A, B, C, D, E et F sont placées dans des registres. La sortie P doit aussi être placée dans un registre.

Vous pouvez utiliser, entre autres, les opérations arithmétiques et logiques, le décalage, des multiplexeurs, et la comparaison de grandeurs.

Vous pouvez aussi utiliser un bloc 'argmin'. La fonction 'argmin' indique laquelle de ses trois entrées R, S, et T est la plus petite. Si  $R \leq S$  et  $R \leq T$ , alors  $V = "00"$ . Si  $S \leq R$  et  $S \leq T$ , alors  $V = "01"$ . Si  $T \leq S$  et  $T \leq R$ , alors  $V = "10"$ . Le cas  $V = "11"$  n'est pas possible.



**Question 3. (3 points : 18 minutes)**

a. (2 points : 12 minutes) Combien de tranches de type L/M d'un FPGA Virtex-7 sont-elles nécessaires pour implémenter les opérations suivantes ? Pourquoi ?

Opération	# Tranches	Brève explication
Addition de deux nombres de 9 bits		
Multiplication d'un nombre de 16 bits par la constante 17		
Vérification de l'égalité entre un nombre de 12 bits et la constante 999 (base 10) = 3E7 (base 16)		
Modulo 4 d'un nombre de 16 bits		

b. (1 point : 6 minutes) Donnez la représentation binaire signée des quantités suivantes.

Quantité	Partie entière (4 bits)	Partie fractionnaire (8 bits)
$9 \times \sin(\pi / 4)$	,	
$-13 / 9$	,	

**Question 4. (2 points – 12 minutes)**

Un microcontrôleur contient 16 registres de 32 bits, un compteur de programme de 32 bits, et 16 registres de contrôle de 1 bit. Il a un port d'adresses bidirectionnel de 32 bits, un port de données bidirectionnel de 32 bits et un port de contrôle de 16 bits en entrée pour faire l'interface avec des capteurs.

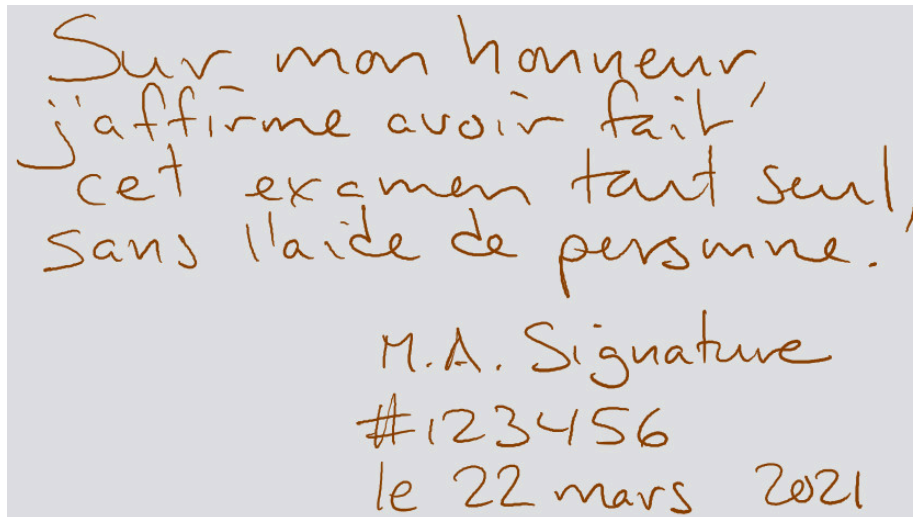
Combien de vecteurs de test seraient nécessaires, au minimum, pour effectuer un test exhaustif de ce microcontrôleur ? Énoncez clairement toutes vos suppositions et montrez tous vos calculs.

Ce nombre de vecteurs de test est-il raisonnable ? Pourrait-on penser faire un test exhaustif du microcontrôleur ? Justifiez votre réponse.

**Question 5. (0 points)**Déclaration sur l'honneur

Produisez une affirmation solennelle sur votre honneur, écrite à la main, signée et datée, à l'effet que vous avez complété cet examen seul/e et sans l'aide de personne.

Joignez une image de votre affirmation en guise de réponse. Voir l'exemple ci-dessous.



Sur mon honneur,  
j'affirme avoir fait  
cet examen tout seul,  
sans l'aide de personne.

M.A. Signature  
#123456  
le 22 mars 2021

Solutions

#1

```

library IEEE;
use IEEE.std_logic_1164.all;

entity cctsequentielex5 is
  port (
    clk, reset : in STD_LOGIC;
    A, B : in STD_LOGIC;
    S5, S6 : out STD_LOGIC
  );
end cctsequentielex5;

architecture arch of cctsequentielex5 is
  signal Q1, Q2 : std_logic;
begin

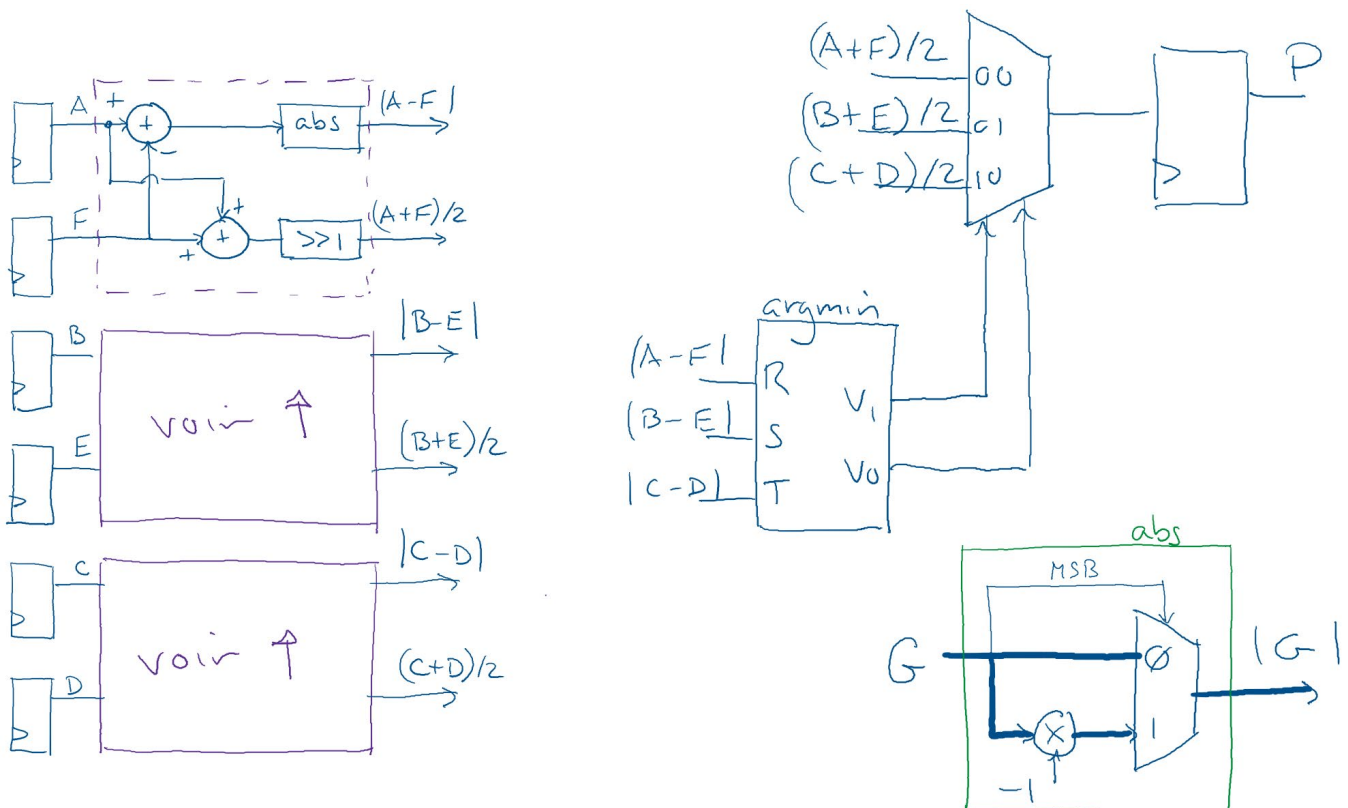
  process(CLK, reset) is
  begin
    if (reset = '0') then
      Q1 <= '0';
      Q2 <= '0';
    elsif (rising_edge(CLK)) then
      Q1 <= A xor Q2;
      Q2 <= Q1 and B;
    end if;
  end process;

  process(Q1, Q2) is
  begin
    S5 <= Q1;
    S6 <= Q1 and Q2;
  end process;

end arch;

```

#2 Solution



#3

a.

Opération	# Tranches	Brève explication
Addition de deux nombres de 9 bits	2.25 ou 3	Il faut une LUT par paire de bits à additionner, donc 9 LUT ici, donc 9 / 4 tranches.
Multiplication d'un nombre de 16 bits par la constante 17	5	On peut effectuer cette multiplication constante par $N \times 17 = N \times 16 + N = N \ll 4 + N$ . $N \ll 4$ s'exprime sur 20 bits. On fait donc une addition sur 20 bits, il faut 20 / 4 tranches.
Vérification de l'égalité entre un nombre de 12 bits et la constante 999 (base 10) = 3E7 (base 16)	3 / 4 ou 1	La solution la plus simple consiste à séparer le nombre de 12 bits en deux parties de 6 bits chacune. On prend chaque moitié du nombre et on la donne à une LUT de 6 bits. Dans chaque LUT on implémente une fonction booléenne pour vérifier l'égalité avec la partie correspondante de la constante. (La constante 3E7 peut être séparée en deux parties de 6 bits chacune. La partie la plus significative est 001111 et la partie la moins significative est 010111.) Il faut finalement vérifier si les deux LUT ont une sortie de 1, à l'aide d'une fonction ET dans une troisième LUT.
Modulo 4 d'un nombre de 16 bits	0	Le modulo par une puissance de deux s'effectue sans opération logique ni arithmétique, une sélection de bits suffit.

b.

Quantité	Partie entière (4 bits)	Partie fractionnaire (8 bits)
$9 \times \sin(\pi / 4)$ $=$ 6.3639610306789277196075992589436 $\approx 6 + 93 / 256$	<b>0110</b> ,	<b>0101 1101</b>
$-13 / 9 = -1.444444444444 \dots$ $1.44444 \approx 1 + 114 / 256 =$ 0001,01110010 On obtient 1110,10001101 + ,00000001	<b>1110</b> ,	<b>10001110</b>

#4

Par test exhaustif, on veut dire ici que chaque état est vérifié et que chaque transition possible entre les états est vérifiée aussi.

Chaque élément à mémoire du microcontrôleur est considéré entreposer un bit d'état. On a donc  $16 \times 32 + 32 + 16 = 560$  bits d'état. Le nombre d'états possibles est  $2^{560}$ .

Le nombre d'entrées est de  $32 + 32 + 16 = 80$  bits, pour  $2^{80}$  combinaisons possibles.

Le nombre minimal de vecteurs de test pour un test exhaustif serait donc de  $2^{560+80} = 2^{640} \approx 4.5 \times 10^{192}$ , qui est un nombre astronomiquement trop grand pour considérer faire un test exhaustif.

Il faudrait aussi considérer les vecteurs de test nécessaires pour se placer dans un état particulier afin de pouvoir exercer toutes les transitions possibles. Et on n'a pas considéré ici les registres de pipeline du processeur. Clairement cette approche ne peut pas être mise en pratique, et elle n'est pas nécessaire non plus.