

0101 Vue d'ensemble

1. Nommez les quatre niveaux d'abstraction des systèmes numériques. Identifiez, dans votre programme d'études de baccalauréat, les cours qui se rapportent à chacun des quatre niveaux.
2. Énumérer et décrire les quatre considérations principales pour l'implémentation d'un système numérique.
3. Expliquez l'effet de doubler la précision des calculs, par exemple en les faisant passer de 16 à 32 bits, sur les trois autres considérations d'implémentation d'un système numérique.

0102 Options d'implémentation

4. Vous travaillez pour une compagnie montréalaise dans le domaine du traitement d'images en temps réel. Votre compagnie a obtenu un contrat en sous-traitance d'un important intégrateur de systèmes avioniques pour la segmentation automatique d'images saisies par une caméra infra-rouge installée dans le nez d'un appareil Airbus A220. À ce jour, Airbus a des commandes fermes pour 203 appareils. Votre compagnie entrevoit la possibilité de vendre son produit à d'autres avionneurs, avec un potentiel de plus de 10K (dix mille) unités.

Votre patron vous demande de l'aider à déterminer s'il faut viser une implémentation en logique fixe ou sur FPGA. Pour la solution FPGA, vous estimez le temps de conception à 4 mois pour 3 ingénieurs, et à 6 mois pour 3 ingénieurs pour la logique fixe. Un ingénieur coûte \$100 k\$/an. Les licences d'outils de design pour toute l'équipe coûtent \$5 k\$ pour le FPGA et \$20 k\$ pour la logique fixe. Les frais fixes de la fonderie s'élèvent à \$250 k\$ puis les puces coûtent \$10 l'unité, et les premières puces seraient livrées 4 mois après la commande. Le FPGA qui accommoderait votre design coûte \$250 et peut être livré dans 4 semaines. Les frais de production et de distribution sont les mêmes pour les deux technologies. Que dites-vous à votre patron? Montrez tous vos calculs et énoncez toutes vos suppositions.

5. C'est l'été et vous travaillez chez FPGA Solutions Consultants Inc. Tous les employés, sauf vous et la présidente, sont partis en vacances. La présidente vous demande de l'aider à choisir parmi trois technologies pour l'implémentation d'un processeur pour un système de guidage automatisé de drone aérien. Les données de comparaison sont présentées au tableau suivant.

Technologie	Frais fixes supplémentaires	Coût unitaire	Fréquence d'horloge max.	Ressources de calcul (blocs logiques)
A. Logique fixe	\$95 000	\$1	1.3 GHz	Max 10000
B. FPGA haut de gamme	\$0	\$50	0.5 GHz	1500
C. FPGA bas de gamme	\$0	\$15	0.2 GHz	1000

Le système doit traiter des données à un débit de 650×10^6 résultats par seconde. Le processeur produit un résultat par cycle d'horloge. Il est possible d'en instancier plusieurs en parallèle et ainsi de multiplier le débit. Le processeur occupe un espace équivalent à 525 blocs logiques. La présidente prévoit que 2000 systèmes seront vendus. Quelle technologie privilégiez-vous? Énoncez clairement toutes les suppositions raisonnables que vous faites et montrez tous vos calculs.

6. Vous travaillez chez IoT-Montréal Inc. sur le sujet de l'internet des objets. Votre équipe travaille à la conception d'un senseur qui inclut une caméra embarquée. Une grande quantité de ces senseurs doit être déployée le long des artères principales de Montréal pour surveiller la circulation. Un algorithme adaptatif basé sur un réseau de neurones doit analyser les images captées afin de distinguer différentes marques de voitures. Vous estimez que l'étape déterminante du calcul a un débit de 50×10^9 multiplications par seconde. Vous considérez trois options d'implémentation : un processeur embarqué ARM Cortex-A7 dans un système Raspberry Pi 2, un FPGA et une puce conçue sur mesure en logique fixe.

Le processeur ARM a une fréquence d'horloge de 900 MHz et peut faire quatre multiplications par cycle. La puce est montée sur une carte avec les interfaces nécessaires, et le coût total du système est de 50 \$. L'effort de développement est purement logiciel et est limité à 2 ingénieurs pendant 1 mois. Les outils sont en libre accès. La livraison est très rapide : une seule semaine.

Le FPGA est le même que celui utilisé dans les laboratoires cette session. Il peut opérer à une fréquence de seulement 250 MHz, mais il contient 240 tranches DSP48E1. Vous estimez cependant qu'il faudrait 6 mois de travail à 3 ingénieurs pour faire la conception du système, et les frais de licence des outils sont de 5 k\$ par an par poste de travail (on les paye seulement pour la durée du projet). Il faudrait 2 ingénieurs de plus pour faire la conception d'une carte d'interfaces, ce qui prendrait un mois. On pourrait recevoir les FPGA, qui coûtent 49 \$ chacun, en 4 semaines après avoir passé une commande. Les frais pour la fabrication des cartes d'interface sont de 200 \$ par carte.

La logique fixe offre une fréquence d'horloge maximale de 1.8 GHz et on peut instancier jusqu'à 1000 multiplieurs en parallèle. Mais il faudrait compter 9 mois de travail pour 4 ingénieurs pour faire la conception, et les outils de conception coûtent 10 k\$ par an par poste. Il faut en plus compter 750 k\$ de frais fixes à la fonderie, puis les puces coûteraient 2 \$ l'unité. Il faut aussi 2 ingénieurs pendant un mois pour la carte d'interfaces, et ses frais sont les mêmes que pour le FPGA. La livraison serait 6 mois après la fin de la conception.

On estime que les ingénieurs coûtent 100 k\$ / an à l'entreprise, incluant salaires et avantages sociaux.

Quelle technologie privilégiez-vous? Pourquoi? Justifiez complètement votre argumentation. Énoncez clairement toutes vos suppositions et montrez tous vos calculs.

0103 Flot de conception

7. Réponses brèves :

- a. Quelles contraintes peuvent guider l'outil de synthèse dans son travail?
- b. Quelles métriques peuvent être obtenues concernant un design après l'étape d'implémentation?
- c. Quelle est la différence entre la simulation d'un module avant la synthèse, après la synthèse et après l'implémentation?

0104 Introduction à VHDL

8. Donnez 3 exemples de types scalaires et 2 exemples types composés du langage VHDL.

9. Donnez deux exemples d'utilisation du symbole '<=' en VHDL.

10. Décrire un module combinatoire en VHDL synthétisable qui accepte en entrée un nombre non signé de six bits, de type `unsigned`, et qui indique si le nombre est divisible par sept à l'aide d'une sortie de type `std_logic`.

11. Un système de contrôle de la température ambiante d'une pièce reçoit une entrée d'un thermomètre numérique et a deux sorties, une pour activer la climatisation et une pour activer le chauffage. Le thermomètre numérique exprime les températures selon un code Gray sur 4 bits, selon le tableau suivant. On vise à maintenir une température entre 20°C et 24°C. Le système de contrôle est très simple : le climatiseur doit être activé quand il fait trop chaud, et la fournaise doit être activée quand il fait trop froid.

Donnez une architecture en VHDL synthétisable d'un module combinatoire pour implémenter le contrôleur de température qui correspond à la définition d'entité suivante.

<i>T3</i>	<i>T2</i>	<i>T1</i>	<i>T0</i>	<i>Température</i>
0	0	0	0	$T \leq 0^\circ \text{C}$
0	0	0	1	$0^\circ \text{C} < T \leq 2^\circ \text{C}$
0	0	1	1	$2^\circ \text{C} < T \leq 4^\circ \text{C}$
0	0	1	0	$4^\circ \text{C} < T \leq 6^\circ \text{C}$
0	1	1	0	$6^\circ \text{C} < T \leq 8^\circ \text{C}$
0	1	1	1	$8^\circ \text{C} < T \leq 10^\circ \text{C}$
0	1	0	1	$10^\circ \text{C} < T \leq 12^\circ \text{C}$
0	1	0	0	$12^\circ \text{C} < T \leq 14^\circ \text{C}$
1	1	0	0	$14^\circ \text{C} < T \leq 16^\circ \text{C}$
1	1	0	1	$16^\circ \text{C} < T \leq 18^\circ \text{C}$
1	1	1	1	$18^\circ \text{C} < T \leq 20^\circ \text{C}$
1	1	1	0	$20^\circ \text{C} < T \leq 22^\circ \text{C}$
1	0	1	0	$22^\circ \text{C} < T \leq 24^\circ \text{C}$
1	0	1	1	$24^\circ \text{C} < T \leq 26^\circ \text{C}$
1	0	0	1	$26^\circ \text{C} < T \leq 28^\circ \text{C}$
1	0	0	0	$28^\circ \text{C} < T$

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity temperature is
port (
temperature : in unsigned(3 downto 0);
climatisation, chauffage: out std_logic
);
end temperature;
```

Solutions

1. Réponse : Voir les diapos et notes de cours. Circuit, portes logiques, module, système.

2. Réponse :

La puissance consommée, la précision des calculs, le taux de traitement (ou débit d'information) et la taille du circuit (ou nombre de ressources utilisées).

3. Réponse :

Si on double la précision des calculs, la taille du système et la puissance consommée vont augmenter, probablement d'un facteur de 2 pour les parties directement concernées par les calculs. Le taux de traitement va être réduit si on garde le reste du circuit pareil, c'est-à-dire sans utiliser de techniques de parallélisation ou de pipeline. Le facteur de ralentissement devrait être inférieur à 2 cependant.

4. Réponse

A. Analyse des coûts

Quoi	FPGA	Logique fixe
Main d'oeuvre	$3 \times 4/12 \times 100 \text{ k\$} / \text{an} = 100 \text{ k\$}$	$3 \times 6 / 12 \times 100 \text{ k\$} / \text{an} = 150 \text{ k\$}$
Licences	5 k\$	20 k\$
Autres frais	\$0	(fonderie) 250 k\$
Total des frais fixes	\$105 k\$	420 k\$
Par puce	250 \$	10 \$
Airbus : 203 systèmes	$105 \text{ k\$} + 203 \times 0.25 \text{ k\$}$ = 156 k\$	$420 \text{ k\$} + 203 \times 0.01 \text{ k\$}$ = 422 k\$
Autres contrats : 10 k systèmes	$105 \text{ k\$} + 10000 \times 0.25 \text{ k\$}$ = 2.61 M\$	$\$420 \text{ k\$} + 10000 \times \$0.01 \text{ k\$}$ = 520 k\$

Pour le contrat de Airbus, le FPGA est plus avantageux par un facteur de $422/156 = 2.7\times$.

Si les autres contrats se matérialisent, alors il vaudrait mieux investir dans la logique fixe (facteur de $2610/520 = 5\times$).

B. Le facteur temps peut être important aussi. La logique fixe a un délai de 4 mois. Les risques sont grands. Si on découvre un bogue, il faut attendre à nouveau 4 mois et repayer les frais fixes de fonderie pour faire une nouvelle puce. Avec le FPGA, les puces seraient disponibles dans un mois. Mais si un bogue était découvert, il ne serait pas nécessaire de faire une nouvelle commande, seulement reprogrammer la puce.

C. La performance n'est pas discutée dans la question, alors on suppose que le FPGA et la logique fixe rencontrent aussi bien les performances requises l'une que l'autre.

D. Conclusion. Le FPGA est nettement préférable à tous points de vue pour le contrat de 203 systèmes. Par contre, si des contrats pour 10000 systèmes se présentaient, il y aurait alors un net avantage à aller en logique fixe (2 M\$ de moins en dépenses). Il semblerait donc prudent de débiter avec la solution FPGA et puis d'attendre ensuite pour voir si les autres contrats se matérialisent. Les 156 k\$ de la solution FPGA sont assez faibles: c'est le salaire de 1.5 personnes pendant un an.

5. Réponses

1. Combien de processeurs ?

En logique fixe, la fréquence d'horloge est suffisamment élevée pour qu'un seul processeur suffise.

Pour le FPGA HG, il faut deux processeurs en parallèle pour dépasser 650 Mhz, pour un total de 1050 blocs logiques. Donc une seule puce suffit.

Pour le FPGA BG, il faut 4 processeurs en parallèle pour dépasser 650 MHz, donc 2100 blocs. Il faut donc trois puces.

2. Quels coûts ?

Coûts LF : $\$95 \text{ k} + \$2 \text{ k} = \$97 \text{ k}$

Coûts FPGA HG: $\$50 \times 2 \text{ k} = \100 k

Coûts FPGA BG : $\$15 \times 3 \times 2 \text{ k} = \90 k

L'option FPGA bas de gamme est donc moins chère.

Cependant, étant donné qu'il faudra utiliser trois puces, l'option FPGA haut de gamme semble être bien meilleure. Elle serait aussi meilleure par rapport à la logique fixe à cause de la flexibilité qu'elle offre.

6. Réponse

Le ARM ne convient pas, $900 \text{ MHz} \times 4$ multiplications par cycle $< 50\text{E}9$, il faudrait 14 systèmes en parallèle, c'est un non-sens. FPGA OK : $250 \text{ MHz} \times 240 = 60\text{E}9$ multiplications par seconde. Logique fixe ok.

	FPGA	Logique fixe
Ingénieurs	$3 \times 6 / 12 \times 100 \text{ k\$}$ $+ 2 \times 1 / 12 \times 100 \text{ k\$}$ $= 167 \text{ k\$}$	$4 \times 9 / 12 \times 100 \text{ k\$}$ $+ 2 \times 1 / 12 \times 100 \text{ k\$}$ $= 317 \text{ k\$}$
Licences	$3 \times 6 / 12 \times 5 \text{ k\$}$ $= 7.5 \text{ k\$}$	$4 \times 9 / 12 \times 10 \text{ k\$}$ $= 30 \text{ k\$}$
Fonderie	-	750 k\\$
Total frais fixes	$\sim 175 \text{ k\$}$	$\sim 1097 \text{ k\$}$
Total des frais unitaires	$49 \$ + 200 \$$ $= 249 \$$	$2 \$ + 200 \$$ $= 202 \$$

On pose $175000 + 249 n = 1097000 + 202 n$ et on trouve $n \sim 19.6 \text{ k}$ unités.

Pour moins de 20 k unités, la solution avec FPGA est la moins chère.

Si on construit plus de 20 k unités environ, il devient rentable d'utiliser la logique fixe, mais le coût total dépasserait alors 5 M\$. Étant donné la nature du projet et la valeur totale, il apparaît raisonnable de débiter avec une solution sur FPGA, à tout le moins un prototype. Puis, si le projet va de l'avant avec une commande de plus de 20 k unités, alors on penserait à une solution en logique fixe.

Il faudrait aussi tenir compte des délais de livraisons, dont les impacts financiers ne sont pas chiffrés dans la question.

7. Réponses :

a. Famille de puce, fréquence d'horloge cible, emphase sur la latence ou l'espace

b. Nombres de ressources, délai et fréquence d'horloge maximale, puissance consommée

c. Avant la synthèse : fonctionnelle seulement; après la synthèse : avec une estimation des délais; après l'implémentation : avec une estimation plus précise, qui reflète les choix de ressources sur la puce.

8. Réponse

scalaires : boolean, bit, character, integer, positive, real, std_logic, etc.

composés: std_logic_vector, unsigned, signed, bit_vector, string, etc.

9. Réponse

Assignation d'une valeur à un signal: B <= '0'; comparaison de taille : if (A <= 5) then ...

10. Réponse

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity divpar7 is
port (
    nombre : in unsigned(5 downto 0);
    divpar7: out std_logic
);
end divpar7;

architecture arch1 of divpar7 is
begin
    with to_integer(nombre) select
        divpar7 <=
            '1' when 0 | 7 | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63,
            '0' when others;
end arch1;

```

11. Réponse :

Deux solutions sont proposées ici.

```

architecture codegray1 of temperature is
begin

    with temperature select
    chauffage <=
        '0' when "1110" | "1010" | "1011" | "1001" | "1000",
        '1' when others;

    with temperature select
    climatisation <=
        '1' when "1011" | "1001" | "1000",
        '0' when others;

end codegray1;

architecture codegray2 of temperature is
begin

    with to_integer(temperature) select
    chauffage <=
        '0' when 14 | 10 | 11 | 9 | 8,
        '1' when others;

    with to_integer(temperature) select
    climatisation <=
        '1' when 11 | 9 | 8,
        '0' when others;

end codegray2;

```