

QUESTIONNEMENT

Ce sujet comporte 5 parties indépendantes organisées autour de 3 centres d'intérêt :

Autour du pilotage d'axes...

Partie 1 : Dimensionnement du calculateur page 2/24

Partie 2 : Etude de la chaîne d'acquisition de la position des axes page 5/24

Partie 3 : Conception de la commande des variateurs de vitesse page 13/24

Autour de la communication...

Partie 4 : Organisation de centres d'usinages en réseau page 14/24

Autour de l'interface Homme / Machine...

Partie 5 : Interface graphique avec l'utilisateur page 16/24

Vous trouverez également 13 documents réponse à joindre avec votre copie.

Partie 1 : Dimensionnement du calculateur

Nous vous proposons dans cette partie de valider le dimensionnement du DSP en fonction des opérations à effectuer pour réaliser le pilotage des axes puis d'en déduire le diagramme de déploiement du système.

1.1 Lecture de la charge de travail sur le diagramme de séquence UML :

Le diagramme de séquence UML définit une vision dynamique des objets participant à la génération des consignes de position à partir d'un fichier ISO.

La charge maximale de travail du DSP est atteinte **uniquement après le chargement** du fichier d'usinage lors du pilotage des axes et du calcul d'enveloppe.

- Q1.1. En vous référant au diagramme de séquence (Dossier de Présentation page 11), indiquer pour chacun des objets les fonctions actives **dans ces conditions**. (préciser pour chaque objet les fonctions s'exécutant séquentiellement ou en parallèle)

1.2 Etude de l'objet : Interpréteur ISO

On donne un extrait de programme dans l'exemple suivant :

<i>Programme</i>	<i>Commentaires</i>
%1	En entête de tout programme
T1	Sélection de l'outil 1
S30000 M3	Démarrage de la broche à 30000 tr/mn
G1 X1000.000 Y500.000 F3000	(bloc de trajectoire 1)
G1 X2000.000Y700.000 F1000	(bloc de trajectoire 2)
G1 X1000.000Y2000.000 F3000	(bloc de trajectoire 3)
M5	Arrêt de la broche
M2	Fin de programme

La position initiale est : X=0, Y=0 et Z=0

1.2.1 Représentation de la trajectoire :

- Q1.2. En vous référant au document Doc 2 (annexes et documents constructeurs) « Commandes ISO », représenter la trajectoire réalisée par ce programme sur le document réponse DR1
(Vous préciserez pour chacune des trajectoires la vitesse d'usinage)

1.2.2 Réalisation de la fonction « Calcul bloc() » :

- Q1.3. En vous référant à la description de l'interpréteur ISO (Dossier de Présentation page 7), compléter dans le tableau donné sur le document réponse DR1 la construction des trois blocs du programme donné en exemple

Nota1 : L'accélération γ est constante et égale à 3m/s^2

Nota2 : Le calcul de V_{finb} , Pos_dec et V_{max} sera abordé lors des questions suivantes

1.3 Etude de l'objet : Pilotage Axes

1.3.1 Réalisation de la fonction « Calcule enveloppe » :

Analyse du cas donné en exemple :

- Q1.4. Représenter sur le chronogramme « Vitesse d'enveloppe » document réponse DR1, les vitesses d'usinage pour les 3 blocs du programme donné en exemple.
- Q1.5. Compléter l'enveloppe élaborée par la fonction « Calcule enveloppe() » de l'objet **Pilotage Axes**.

Analyse du cas général :

Le squelette du pseudo-code du programme de calcul d'enveloppe pour un déplacement linéaire en XY est donné sous la forme d'un tableau dans le document réponse DR2.

L'illustration des différents cas pouvant se produire est donnée colonne de droite.

- Q1.6. En vous référant au diagramme de séquence et à la documentation correspondante, compléter sur ce document pour les cas 2,3 et 4:
- l'enveloppe de vitesse générée (sur le croquis)
 - l'emplacement de $V_{max}[i]$ et de $Pos_dec[i]$, (sur le croquis)
 - Le pseudocode associé

- Q1.7. Montrer que $Pos_dec = LTC - (V_{max}^2 - V_{fin}^2)/2\gamma$

- Q1.8. Montrer que : $V_{xd} = \sqrt{2 * \gamma * LCT[i] + V_{finbloc[i-1]}^2}$

1.3.2 Réalisation de la fonction « Calcul de la vitesse d'usinage » :

La fonction « Calcul de la vitesse d'usinage » détermine du début de la pile vers la fin de la pile la vitesse à appliquer à la broche en respectant l'enveloppe et en ajoutant des rampes d'accélération.

- Q1.9. Superposer sur le chronogramme « Vitesse d'enveloppe », document réponse DR1, les vitesses d'usinage ainsi déterminées pour les 3 blocs.

1.3.3 Réalisation de la fonction « Calcul de la vitesse d'usinage » :

Nous ne nous intéresserons pas ici, au passage d'un bloc à l'autre.

- Q1.10. En utilisant les grandeurs calculées par la fonction détection d'enveloppe, compléter document réponse DR3 le pseudocode permettant de calculer, sur un bloc, les grandeurs :
- Vit_cur : la vitesse curviligne de la trajectoire,
 - X_cur : la position curviligne de la trajectoire.
- Cet élément prépare le départ à chaque interruption temps réel HTR (2ms)
- Q1.11. Déterminer DR3 les consignes de position X_{cons_pos} et Y_{cons_pos} en fonction de X_{cur} , ΔX , ΔY et LTC.

1.4 Calcul du taux d'occupation du DSP :

Q1.12. En vous référant à la documentation du DSP, Déterminer le nombre d'instructions par seconde exécutées par celui-ci (le quartz est porté à 16MHz)

Q1.13. Compléter dans le tableau DR4 :

- Le nombre d'instructions exécutables par HTR
- Le taux d'occupation du DSP pour réaliser l'asservissement, le calcul de vitesse et de position
- Les ressources utilisées et le taux d'occupation pour le calcul d'enveloppe

Nota : L'essentiel du temps de calcul lors de la détermination de l'enveloppe concerne les opérations sur les flottants :

- addition, multiplication : 40 instructions
- division : 240 instructions
- racine carrée : 250 instructions

On estime que les signaux issus des codeurs auront au maximum une fréquence de 100kHz et que le temps de traitement associé serait de 30 instructions

Q1.14. Montrer que le DSP n'est pas capable d'assurer la gestion du décodage des impulsions. Justifier le choix d'un FPGA pour réaliser cette tâche.

1.5 Diagramme de déploiement du système :

Q1.15. Compléter DR4 le diagramme de déploiement du système en précisant :

- les Applications / Objets réalisées par les structures,
- la dénomination de la liaison entre le bloc microprocesseur et le bloc DSP.

En vous référant à la documentation du DSP, expliquer très succinctement les caractéristiques de cette dernière liaison.

Partie 2 : Etude de la chaîne d'acquisition de la position

Nous vous proposons dans cette partie de valider le choix des composants de la chaîne d'acquisition de la position sur les axes, puis de proposer une séquence pédagogique sur le traitement des signaux issus des capteurs.

2.1. Choix du codeur de position

Le constructeur prévoit des surfaces de travail pouvant atteindre en standard 7m50.

Des surfaces plus importantes doivent toutefois pouvoir être réalisées pour répondre à des demandes spécifiques.

La chaîne cinématique des axes est standard et est toujours réalisée de la manière suivante :

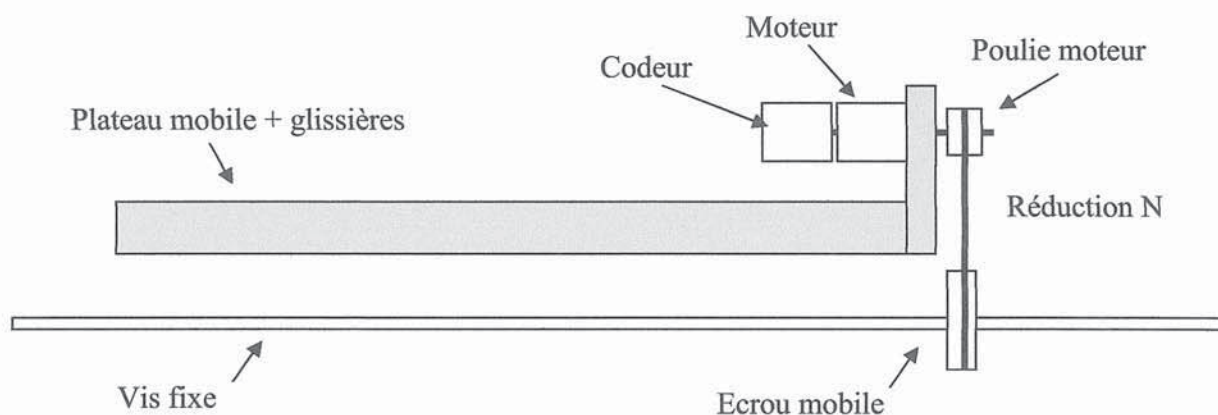


Figure1 : Le moteur monté sur la table d'usinage entraîne un écrou mobile ($\Omega_{\text{écrou}} = \Omega_{\text{moteur}} / N$) via une courroie crantée. L'écrou qui tourne autour d'une vis fixe provoque le déplacement de la table.

On donne les éléments cinématiques mis en jeu :

	Axe X	Axe Y	Axe Z
Masse en déplacement :	300 kg	250kg	70kg
Vitesse maximale de déplacement :	417mm/s	417mm/s	417mm/s
Pas de la vis à bille :	20mm	25mm	10mm
Diamètre de la vis à bille :	32mm	25mm	25mm
Rapport de réduction N :	2	2,5	1

- Q.2.1. Comparer les avantages et inconvénients d'un codeur incrémental et d'un codeur absolu. Justifier le choix du constructeur pour un capteur incrémental dans ce cas.
- Q.2.2. Calculer pour chaque axe le nombre de points nécessaires sur le codeur associé sachant que l'on souhaite pour chacun un pas de mesure de $0,84\mu\text{m}$.

On monte le codeur avec une fixation par bride synchro de 6mm

Q.2.3. Compte tenu de cette contrainte, et en vous référant au schéma structurel d'interfaçage du codeur DSP_IO.SCH, définir la référence complète de chaque codeur dans la documentation de la série DGS60 donnée documentation constructeur DOC3 .

Q.2.4. Calculer la fréquence maximale du signal issu du codeur d'axe X.

2.2. Choix du câble de liaison

La note d'application du récepteur de ligne précise pour le choix du câble de liaison :

Losses in a transmission line are a complex combination of DC conductor loss, AC losses (skin effect), leakage, and AC losses in the dielectric.

On se propose ici de valider le choix du constructeur sur ces différents points

Le schéma ci-dessous caractérise la modélisation d'une ligne :

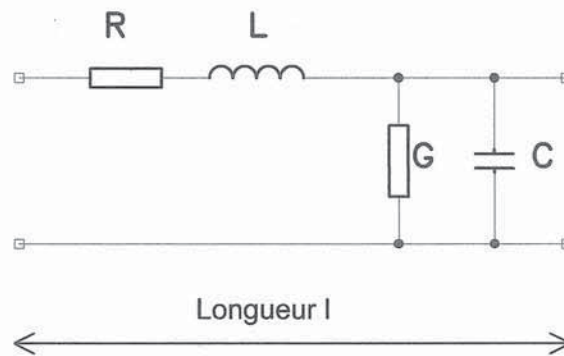
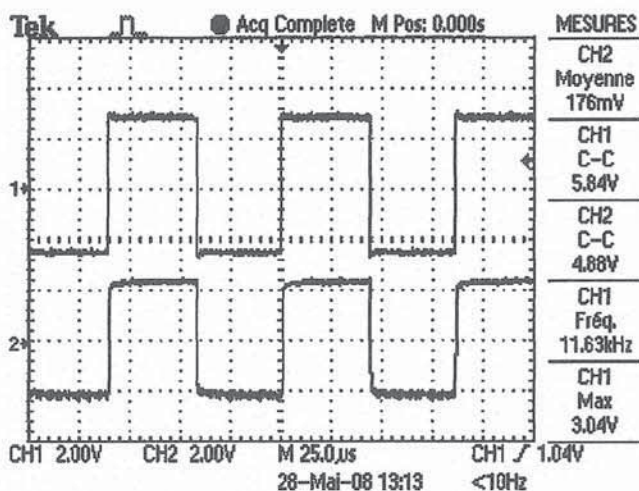


Figure 2 : Schéma équivalent de la ligne

Q.2.5. Indiquer ce que représentent R, G, L et C par rapport à cette ligne.

2.2.1. DC et AC Conductor loss

Relevés effectués :



Sur le canal CH1 de l'oscilloscope a été relevé un signal généré par le codeur au cours d'un usinage.

Sur le canal CH2 de l'oscilloscope a été relevé le signal en sortie du câble utilisé.

Figure 3 : Relevé effectué dans des conditions normales de fonctionnement en entrée et en sortie d'une ligne de 100m.

Q.2.6. S'agit-il d'une ligne sans perte? Expliquer et chiffrer.

La documentation constructeur du câble testé (ACOLAN) est donnée en annexe3.

Q.2.7. Compte tenu de la fréquence maximale de fonctionnement précisée dans la documentation du codeur DGS60, un effet de peau est-il à craindre? Expliquer ce phénomène.

Q.2.8. Rappeler l'expression de R en fonction de la résistivité du conducteur, de sa longueur et de sa section.

On donne : ρ : résistivité en $\Omega.m$ $\rho = 1,72.10^{-8} \Omega.m$ pour le cuivre

On caractérise l'effet de peau (en sinusoïdal) par l'épaisseur suivante :

$$\delta = \sqrt{\frac{\rho}{\pi \times \mu \times f}}$$

Avec :

- ρ : résistivité du matériau en $\Omega.m$
- μ : perméabilité magnétique du conducteur en H/m (on rappelle : $\mu = \mu_r \cdot \mu_0$ avec : $\mu_0 =$ perméabilité du vide $= 4.\pi.10^{-7} \text{H/m}$ et $\mu_r = 1$ pour le cuivre)
- f : fréquence en Hz du signal sinusoïdal.

Un calcul nous donne la courbe DR5 donnant la densité de courant à la périphérie par rapport à la profondeur en mm pour une fréquence de 300kHz. Cette courbe correspond à un câble de section infinie.

Q.2.9. Exploiter cette courbe de façon à faire apparaître les caractéristiques physiques du câble choisi.

Q.2.10. Calculer la résistance totale R d'une ligne de longueur 100m avec et sans effet de peau. Comparer au relevé de la figure 3 et conclure.

2.2.2. Leakage loss

On donne l'expression des paramètres linéiques dans le cas d'une paire torsadée :

- Inductance linéique ($\mu\text{H/m}$) :

$$L_1 = 0,92. \text{Log} \left(\frac{D}{d} + \sqrt{\left(\frac{D^2}{d^2} - 1\right)} \right) \text{ avec :}$$

D : distance entre axe des 2 conducteurs (en m)

d : diamètre du conducteur (en m)

- Capacité linéique (pF/m) : $C_1 = 12. \frac{\epsilon_r}{\text{Log} \left(\frac{D}{d} + \sqrt{\left(\frac{D^2}{d^2} - 1\right)} \right)}$

- Conductance en Siemens/m :

$$G_1 = 2\pi \cdot f \cdot \frac{C_1}{Q} \text{ avec :}$$

f : fréquence en Hz

C_1 : Capacité linéique en F/m

Q : Facteur de qualité (inverse de l'angle de perte) = 5000 pour le cuivre

- La valeur de la permittivité relative pour différentes natures de diélectriques :

Type	Nature	Permittivité
PE	Polyéthylène	2.2 à 2.3
PEA	Polyéthylène spiralé	1.2 à 1.6
PEC	Polyéthylène cellulaire	1.3 à 1.6
PEF	Polyéthylène mousse	1.4 à 1.6

Q.2.11. Calculer l'inductance et la capacité linéique de notre ligne toujours réalisée avec le câble ACOLAN dont la documentation est fournie en annexe.

Q.2.12. Calculer la conductance de cette ligne et conclure.

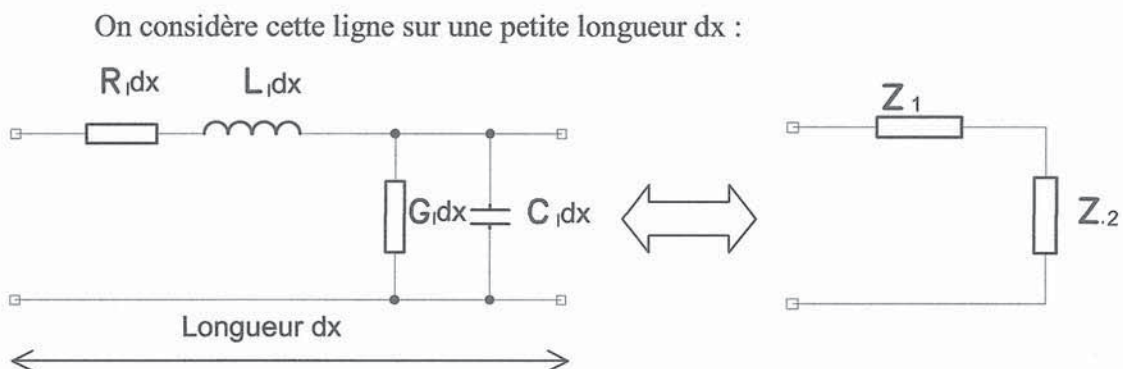


Figure 4 : Portion de ligne dx et son schéma équivalent

Cette portion est reproduite à l'infini sur toute la longueur de la ligne.

Le dernier élément Z_1/Z_2 de la ligne est chargé par une impédance Z_0 .

- Q.2.13. Expliquer comment doit être l'impédance d'entrée de cet élément pour un fonctionnement optimal.
- Q.2.14. Exprimer Z_0 en fonction des éléments de la portion de ligne pour une infinité de tronçons.
- Q.2.15. Calculer cette impédance pour la fréquence maximale du codeur choisi puis à la fréquence maximale de fonctionnement calculée. Conclure.

Une dilatation de l'échelle du chronogramme de la figure 3 nous donne le résultat ci-dessous :

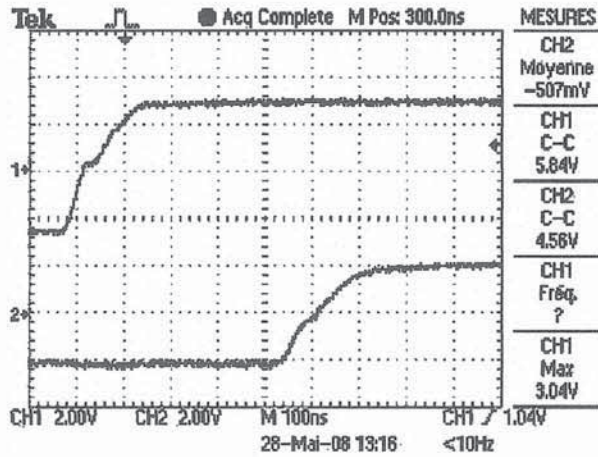


Figure 5 : Dilatation de l'échelle pour le relevé de la figure 2. Entrée et sortie de la même ligne.

Dans le cas d'une ligne à faibles pertes, la vitesse de propagation pourrait être approchée par l'expression suivante :

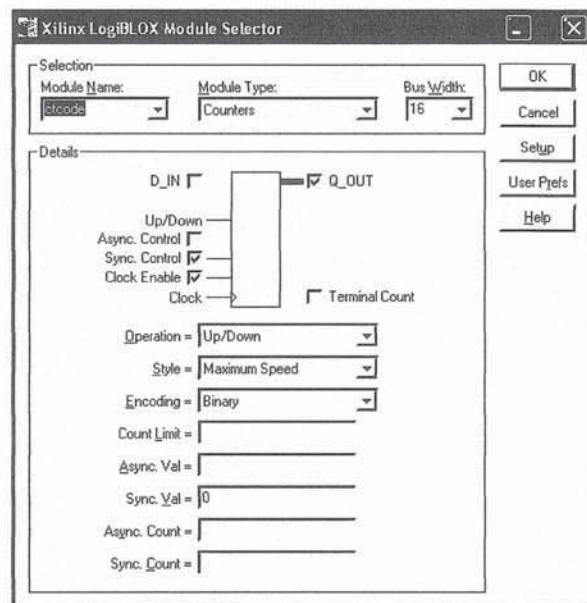
$$v = \frac{1}{\sqrt{L_1 \cdot C_1}}$$

- Q.2.16. Vérifier cette propriété sur le chronogramme de la figure 5 et expliquer la forme arrondie du front.
- Q.2.17. La vitesse de propagation nominale est également spécifiée dans la documentation constructeur. Retrouver cette valeur et comparer de nouveau aux relevés de la figure 5.

2.3. Décodage des informations issues du codeur

- Q.2.18. Quel est le rôle des composants LTC488 ? Justifier leur présence.
- Q.2.19. Quel est le type de composant utilisé pour le décodage des informations codeurs, l'interfaçage avec le DSP et le décodage d'adresse. Justifier ce choix.
- Q.2.20. En vous référant aux documents relatifs à U6 (FPGA2), Identifier la fonction chargée de lire les codeurs. Combien de fois est elle implantée ?
- Q.2.21. Quelle est l'étendue de mesure de la position des axes obtenue avec cette solution ?
Cette étendue de mesure est elle compatible avec le cahier des charges de la machine ? Justifiez cette différence !

On donne la logibox associée à
CTCode :



Activité Pédagogique sur la mesure de la position :

Cette activité pédagogique s'adresse à des étudiants de BTS SE, IRIS ou d'IUT GEII. Il s'agit d'illustrer le principe de la mesure de position par capteur incrémental à partir du support proposé.

- Q.2.22. Présenter les différents capteurs de position usuellement utilisés. Préciser lequel semble le mieux adapté compte tenu de l'application.
- Q.2.23. Proposer une séquence pédagogique (cours/TP/synthèse) permettant, au travers du système étudié, d'illustrer le principe de traitement des informations issues d'un codeur incrémental.

2.4. Interfaçage du FPGA avec le DSP

On se propose ici d'analyser la structure implantée dans le FPGA afin d'en déduire les opérations à réaliser par le DSP pour lire et réinitialiser les compteurs d'axe.

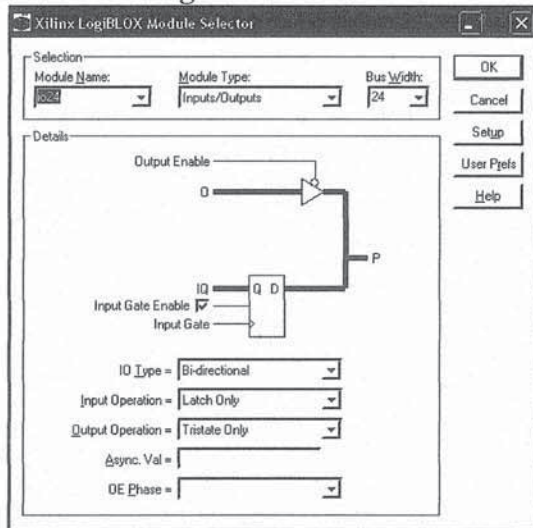
Q.2.24. En vous référant au schéma de la carte électronique DSP.SCH 2/9 et à la documentation du DSP, décrire DR5 les caractéristiques des signaux échangés entre le DSP et le FPGA.

Q.2.25. En vous référant :

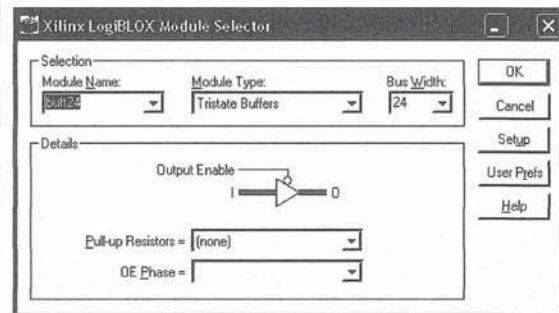
- au schéma structurel 2/9,
- aux schémas internes du FPGA2 « Core » et « CPU »,
- aux logibox,

compléter DR5 l'organisation matérielle représentant l'interfaçage des bus de données du DSP, de la RAM et du FPGA :

On donne la logibox de IO24 :



Et de buff24 :



Q.2.26. En vous référant au schéma FPGA2/CPU, préciser la condition de passage à '1' de EXTPER .

Q.2.27. En déduire la condition de sélection de la RAM externe du DSP.

Q.2.28. En vous référant à la structure implantée dans le FGPA2 représentée ci-dessous:

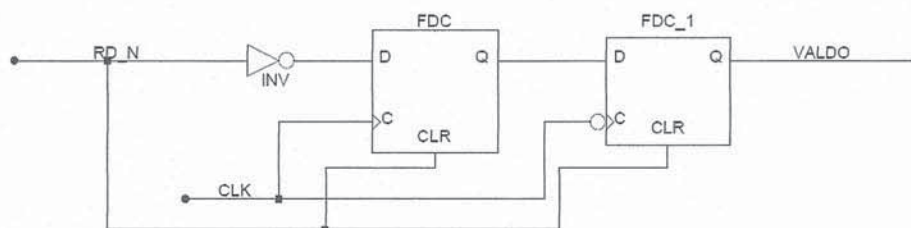


Figure 6 : Elément de structure implanté dans FPGA2

et aux chronogrammes d'adressage externe du DSP ci-dessous :

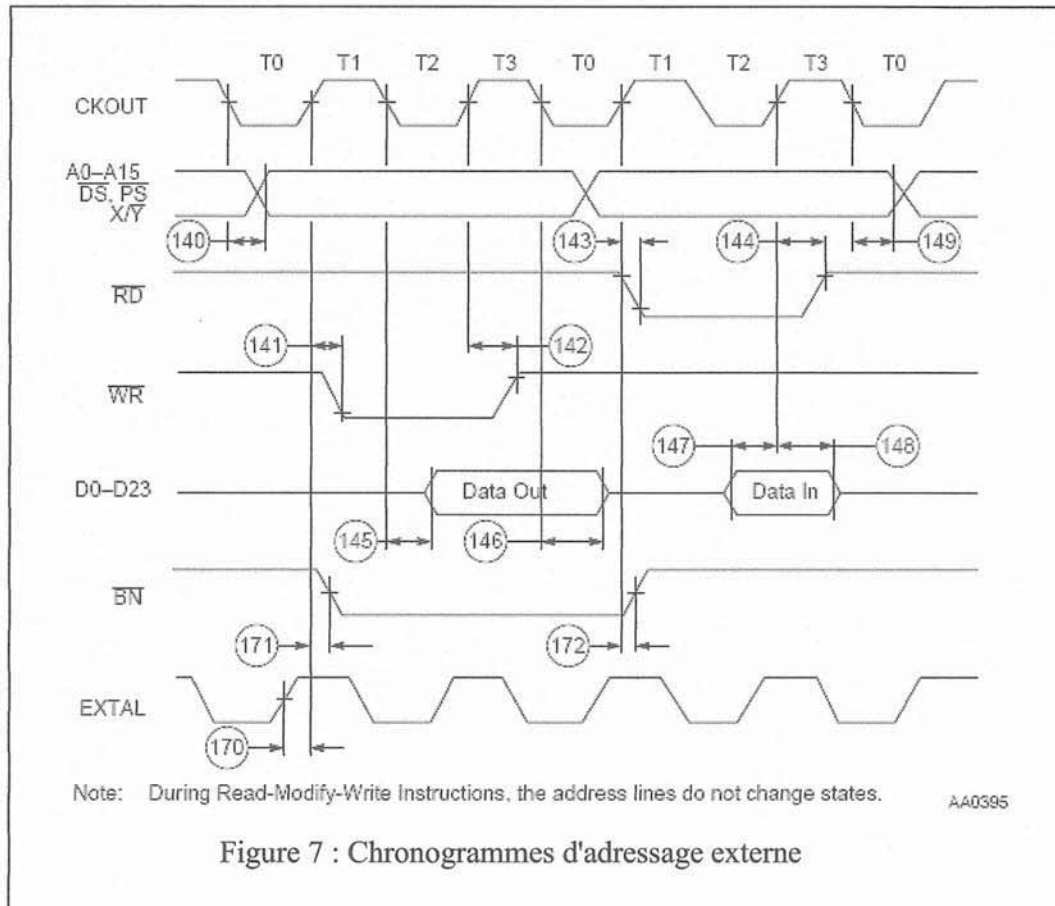


Figure 7 : Chronogrammes d'adressage externe

Justifier le rôle du signal Valdo .

- Q.2.29. En déduire la condition de lecture de la donnée présente sur le bus de données du FPGA.
- Q.2.30. Quelle opération permet la mémorisation d'un mot en sortie de la bascule D de la logibox IO24? (répondre DR5)
- Q.2.31. Justifier le rôle du Buffer IO24.
- Q.2.32. Résumer dans le tableau document DR6 les opérations à effectuer par le DSP pour interfacer son bus de données avec celui de la RAM et du FPGA.
- Q.2.33. Décrire le fonctionnement des signaux RD et WR.
- Q.2.34. Déduire des résultats précédents l'opération à réaliser par le DSP pour lire la valeur du codeur1.
- Q.2.35. Déduire des résultats précédents l'opération à réaliser par le DSP pour réinitialiser simultanément la valeur de tous les compteurs et de valider leur fonctionnement (on laissera les bits de commande de test et de SAMPLE à 0).

Partie 3 : Conception de la commande des variateurs de vitesse

Nous vous proposons dans cette partie de produire la solution constructive, tant sur le plan matériel que logiciel, du pilotage des convertisseurs numérique/analogique du système.

La commande des variateurs est réalisée en analogique selon le cahier des charges suivant :

- Consigne Vitesse = +2047 \Rightarrow VANx = +10V \Rightarrow Vmoteur = +3000 tr/min
- Consigne Vitesse = -2048 \Rightarrow VANx = -10V \Rightarrow Vmoteur = -3000 tr/min

L'interfaçage DSP / variateurs de vitesse est partiellement représentée DR7

3.1 Conception de la solution constructive matérielle du pilotage des DACs

Q.3.1 Déterminer la valeur des tensions VA et VB.

Q.3.2 Réaliser l'interconnexion entre les composants du schéma afin de permettre :

- la sélection de U1 par SC0,
- la sélection de U2 par SC1,
- la validation du chargement de U1 et U2 par SC2,
- la remise à 0 des valeurs par le signal SC3,
- le dialogue entre le DSP et les DACS,
- plus toutes les connexions que vous jugerez utiles...

Q.3.3 Quel est le courant d'entrée sur les broches Vref_H et Vref_L ? Cette valeur est elle compatible avec votre schéma ?

Q.3.4 Représenter DR6 le chronogramme des signaux de commande lors de l'écriture de la valeur 0x89A8 sur U2.

Q.3.5 Quelle sortie est alors pilotée ? Quelle est sa tension de sortie ?

3.2 Configuration des registres du module SSI du DSP

Q.3.6 En vous référant à la documentation du DSP, identifier DR8 la valeur à configurer dans les registres HSAR, HCKR, et HCSR.

Vous indiquerez pour l'état des bits soit :

- leur valeur (0 ou 1)
- la valeur X si l'état est indifférent

3.3 Procédure d'écriture dans les DACs

Cette procédure devra admettre trois arguments :

- ✓ *numero_DAC : 1 ou 2 permet la sélection de U1 ou 2*
- ✓ *numero_sortie : 0,1,2,3 permet la sélection de la sortie A, B, C ou D*
- ✓ *consigne_vitesse : valeur comprise entre -2048 et + 2047 correspondant à une consigne de vitesse de -3000 à +3000tr/min*

Q.3.7 Compléter le pseudo-code DR6 permettant de piloter les moteurs.

Partie4 : Organisation en réseau de centres d'usinages :

Nous vous proposons dans cette partie de produire un plan d'adressage permettant l'intégration de plusieurs centres d'usinage dans le réseau d'une entreprise.

Rappel modèle OSI : (Répondre sur DR9)

- Q4.1. Donner le nom de la couche 2 du modèle OSI ?
Indiquer le nom de l'entité traitée par la couche 2 ?
- Q4.2. Donner le nom de la couche 3 du modèle OSI ?
Indiquer le nom de l'entité traitée par la couche 3 ?

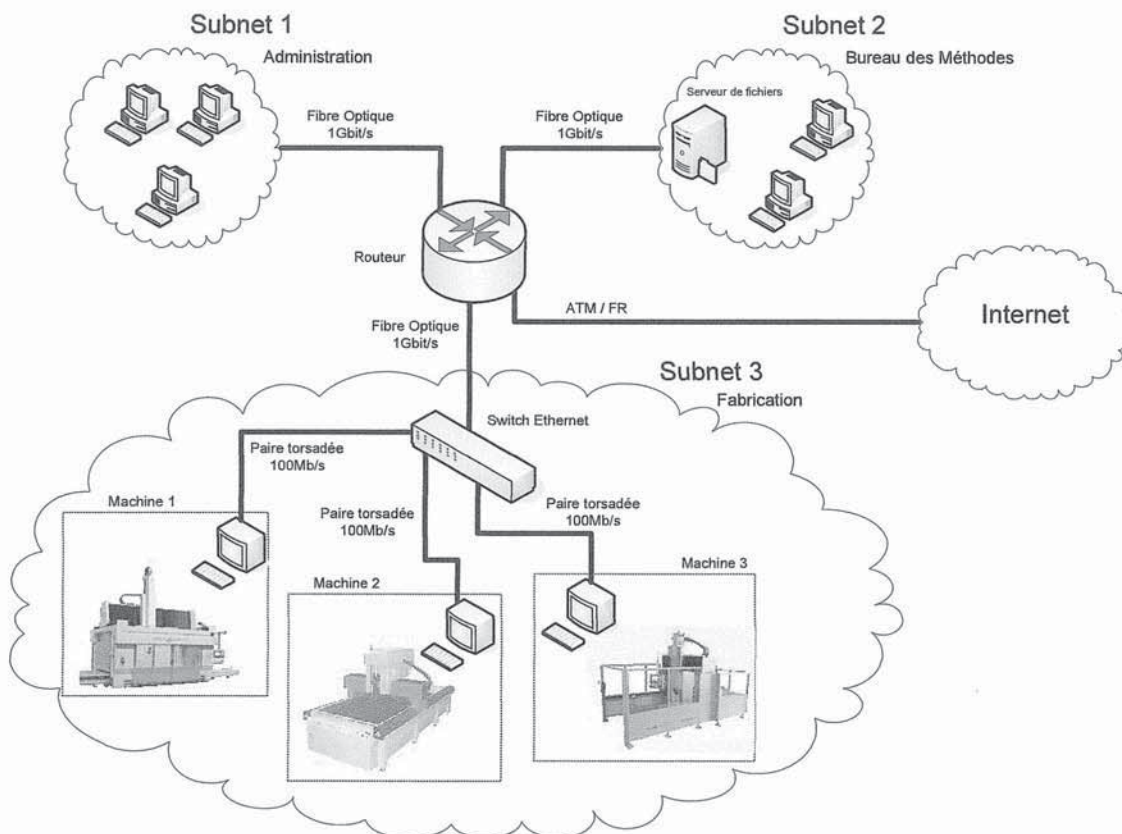
Achat d'une adresse publique : (Répondre sur DR9)

L'entreprise s'est adressée à un FAI pour obtenir une adresse publique. Elle a obtenu une adresse publique en 200.20.2.XXX.

- Q4.3. Quelle est la classe de cette adresse ?
- Q4.4. Préciser dans le tableau DR9 son masque par défaut.
- Q4.5. Préciser de même sa NetID, son HostID.
- Q4.6. Indiquer le nombre maximal de machines pouvant y être connectées ?

Organisation du réseau interne : (Répondre sur DR9)

L'entreprise souhaite réaliser une organisation de son réseau informatique interne selon la figure ci-dessous :

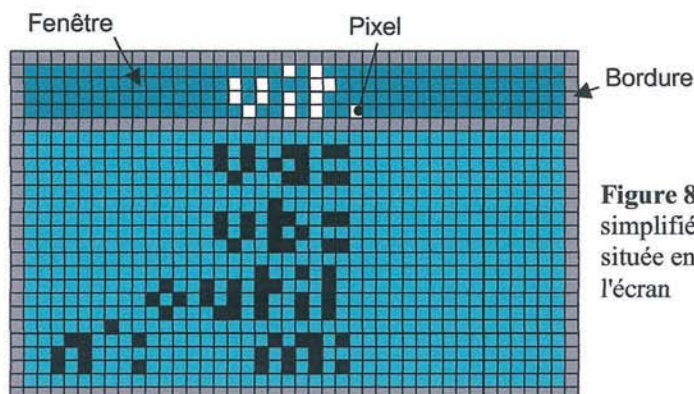


- Q4.7. Sachant que l'on n'utilisera pas les adresses n° réseau et broadcast, proposer l'adressage des sous réseaux (on utilisera les MSB pour définir les sous réseaux).
- Q4.8. Y préciser le nouveau masque.
- Q4.9. Indiquer l'adresse complète de chaque sous réseau.
- Q4.10. Déterminer le nombre maximal de machines sur chacun des sous réseaux.
- Q4.11. Attribuer pour chaque machine une adresse IP et un masque en partant de l'adresse basse de la plage.

Partie 5 : Interface Graphique avec l'utilisateur

Nous nous proposons, dans cette partie, d'étudier dans un premier temps la solution retenue par le constructeur pour générer les signaux vidéos à partir d'une matrice graphique puis d'aborder la construction de cette matrice par des applications orientées objet. Nous nous interrogerons ensuite sur la pédagogie à mettre en œuvre pour traiter ce sujet avec des étudiants.

L'écran utilisé est de type TFT-LCD référence LQ104V1DG51 de SHARP.
Il s'intègre dans un pupitre qui permet à l'utilisateur de dialoguer avec le système



5.1. Génération de la matrice graphique :

Le microcontrôleur calcule et transfère, en mémoire, la matrice graphique en fonction de fenêtres et de textes prédéfinis permettant l'interfaçage avec l'utilisateur.

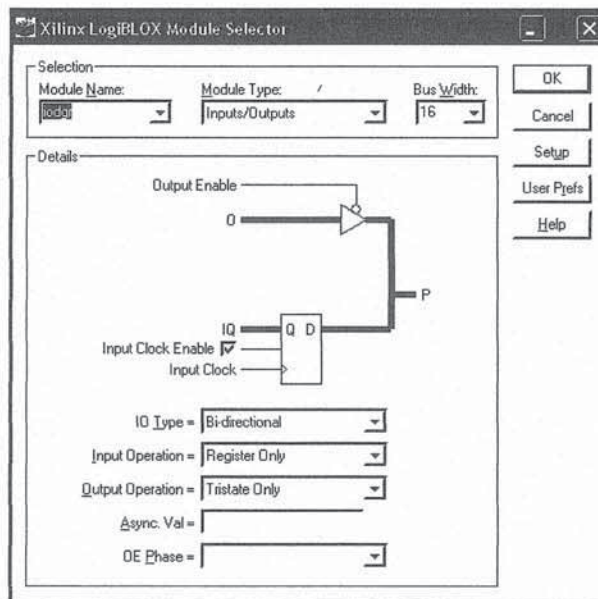
Le FPGA1 génère, à partir de cette matrice graphique, les codes de couleur nécessaires à l'interprétation par l'écran. Il produit également les signaux de commande utiles.

Q5. 1. A partir du schéma principal de la carte CN6000-A.sch et de la documentation constructeur, indiquer DR10 le rôle des broches permettant le pilotage de l'écran.

Dans la suite du sujet, il va s'agir d'étudier les structures intégrées dans le FPGA1 pour permettre la génération de ces signaux. Ces structures sont réparties sur plusieurs schémas qu'il est conseillé de lister pour répondre efficacement aux questions.

Lorsque la représentation symbolique ne sera pas suffisante à la compréhension du fonctionnement, une image de la "logibox" sera fournie en entrée de question.

La description de la composante IODGR du schéma GRDATA est donnée ci-dessous :



Q5. 2. En vous référant à l'ensemble des schémas implantés dans le FPGA1, compléter **DR10**, le schéma fonctionnel de la gestion des bus et de l'affichage.

L'élaboration des signaux de pilotage nécessite une horloge d'échantillonnage H25.

Cette horloge est obtenue à partir du bloc GRPLL du FPGA1 et d'un composant PLL AV9173-01CS08 dont la documentation est jointe.

Q5. 3. Extraire un schéma synoptique décrivant le principe de l'élaboration de H25.

Q5. 4. En déduire la fréquence de ce signal afin de valider la structure choisie.

Le schéma GRCMD montre comment sont générés les divers signaux de commande en fonction de H25 et des caractéristiques de l'écran.

Q5. 5. En vous référant au schéma structurel de GRCMD proposer **DR10** un chronogramme permettant de caractériser les signaux PHsync, CH[9..0] et TCH en fonction de H25.

En déduire la valeur des grandeurs TH et THp puis valider ces valeurs par rapport à la documentation constructeur :

Q5. 6. Proposer **DR10** un chronogramme permettant de caractériser les signaux PVsync en fonction de CV[9..0] et de TCH.

En déduire la valeur des grandeurs TV et TVp puis valider ces valeurs par rapport à la documentation constructeur et en déduire la configuration de l'écran.

Q5. 7. Représenter **DR11** la forme du signal PENAV en fonction de CV[9..0] et de TCH.

En déduire la valeur des grandeurs TVs et TVd puis valider ces valeurs.

Q5. 8. Représenter **DR11** la forme des signaux PENAHI en fonction de CH[9..0]

En déduire la valeur des grandeurs THe et Tep puis valider ces valeurs.

Q5. 9. Compléter **DR11** la zone de validité des pixels en fonction des valeurs de CH[9..0] et de CV[9..0].

Une étude rapide des schémas de commande montre que deux signaux sont prépondérants dans l'ordonnement de l'apparition des pixels. Il s'agit des signaux CYCEND et GRAPH.

Q5. 10. Compléter les chronogrammes CYCEND à LOADDIN du document DR12.

Q5. 11. En vous référant au schéma GRDATA, déduire la taille des mots traités par le module graphique du FPGA et le nombre de pixel codé dans chaque mot. Préciser le nom des signaux permettant de décomposer ces informations.

Le composant LUT de ce schéma est une PROM dont le contenu est donné ci-dessous :

<code>;memfile prom.mem for LogiBLOX symbol prom</code>	<code>3:0003F</code>
<code>; Header Section</code>	<code>4:3F03F</code>
<code>DEPTH 16</code>	<code>5:3FFC0</code>
<code>WIDTH 18</code>	<code>6:00FFF</code>
<code>DEFAULT 0</code>	<code>7:2A000</code>
<code>;</code>	<code>8:00580</code>
<code>; Data Section</code>	<code>9:00028</code>
<code>; Specifies data to be stored in different addresses</code>	<code>A:3F7C0</code>
<code>; e.g., DATA 0:A, 1:0</code>	<code>B:28028</code>
<code>RADIX 16</code>	<code>C:007DF</code>
<code>DATA</code>	<code>D:1F7DF</code>
<code>0:00000</code>	<code>E:2FBF0</code>
<code>1:3F000</code>	<code>F:3FFFF</code>
<code>2:00FC0</code>	<code>; end of LogiBLOX memfile</code>

Figure 9 : Contenu de la ROM du FPGA1

Q5. 12. D'après la documentation constructeur, indiquer le nombre de couleurs pouvant être affichées par l'écran et valider le choix du concepteur en précisant le nombre de couleurs utilisées par l'application.

Q5. 13. Préciser les avantages de ce choix au niveau de la conception.

Le schéma GRADR montre le multiplexage du bus d'adresse.

Q5. 14. Sur le chronogramme **DR12**, indiquer les périodes pendant lesquelles le microcontrôleur peut accéder au bus d'adresse de la RAM graphique puis celles pendant lesquelles le contrôle s'effectue par le compteur de balayage des pixels.

5.2. Génération de la matrice graphique :

C'est la partie logicielle, intégrée dans le microcontrôleur qui calcule et sauvegarde en mémoire graphique la matrice d'affichage.

5.2.1 Contraintes logicielles :

Pour ses activités, l'utilisateur a accès à 6 écrans d'affichage différents comportant une barre de menu (Bas d'écran) accessible par appui sur la touche Maj.

Chaque écran comporte plusieurs fenêtres de fonte et taille différente, porteuses d'informations et réunies par type dans une fenêtre globale.

Les fenêtres associées à chaque famille d'indications sont listées ci-dessous :

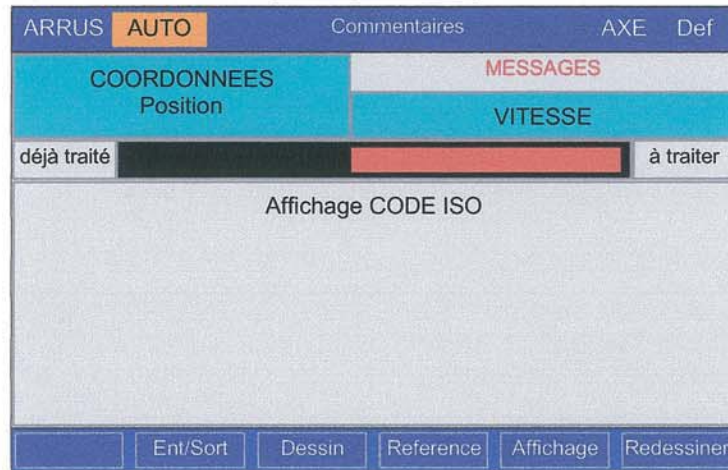
- 7 fenêtres en fonte 8x12 composent la barre de menu (Barre de fond, Menu Entrée/sortie, Menu Dessin, Menu Référence, Menu Affichage, Menu Redessiner et Menu gauche),
- 6 fenêtres en fonte 12x16 indiquent, en haut de l'écran, le mode de fonctionnement (ARRUS, MANU/AUTO, axe, commentaires et défaut),
- 5 fenêtres en fonte 12x16 permettent d'accéder aux coordonnées (Titre, cadre coordonnées, valeur de x, valeur de y et valeur de z),
- 2 fenêtres en fonte 8x12 sont réservées aux messages d'information,
- 5 fenêtres en fonte 12x16 constituent la zone des vitesses (valeur va, valeur vb, numéro outil, mesure outil),
- 4 fenêtres en fonte 8x12 réalisent le panneau de titre et d'état des entrées et sorties,
- 5 fenêtres en fonte 8x12 gèrent l'état des buffers,

De la même façon, il y a les fenêtres relatives à l'affichage du code iso, au dessin (Ex : représentation du circuit de déplacement de l'outil), aux positions de référence, aux positions de référence courante et aux outils.

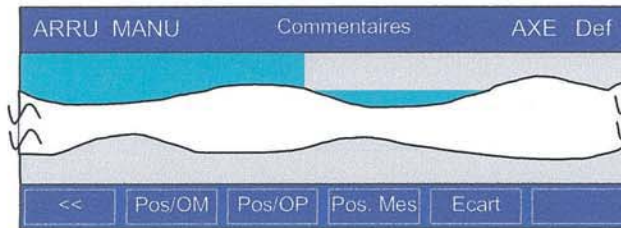
Quelques exemples d'écran de différents menus sont représentés page suivante.

Figure 10 : Exemple de menus :

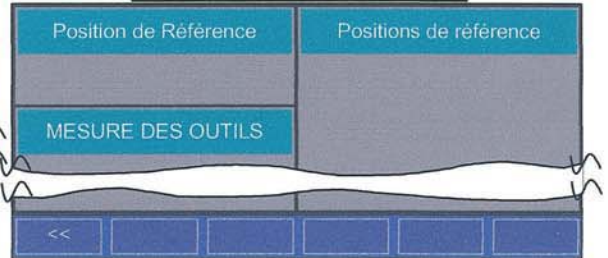
Écran principal :



Écran principal + Menu Affichage :



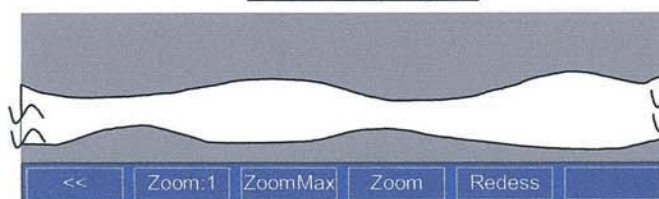
Écran : références et outils



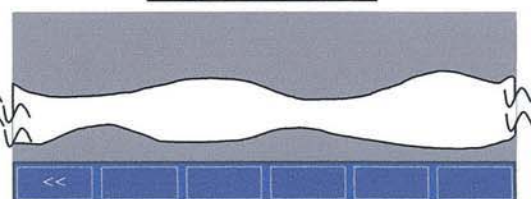
Écran des Entrées/Sorties :



Écran Graphique :



Écran Paramètre :



5.2.2 Diagramme de Classes :

Il est aisé de déduire de cette étude les classes et les interactions entre classes nécessaires à la mise en oeuvre de l'affichage :

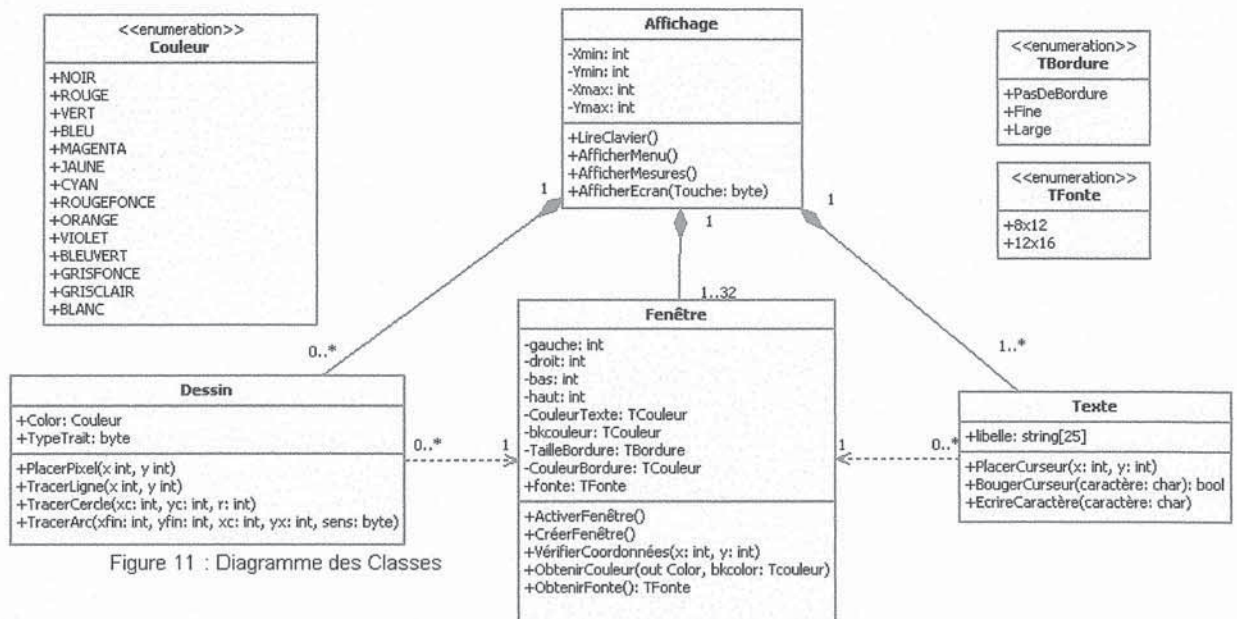


Figure 11 : Diagramme des Classes

Q5. 15. Afin d'illustrer le modèle précédent, fournir un diagramme d'objet relatif à l'écran des entrées/sorties hors la barre de menu (Touche Maj non enfoncée).

Le diagramme étant relativement complexe, on propose de se pencher plus précisément sur la fenêtre titre des entrées et de son texte.

Q5. 16. Compléter le diagramme de séquence **DR12** du cas particulier de l'écriture du titre "Etat des entrées" dans la fenêtre correspondante.
Préciser les modifications à apporter au diagramme des classes pour le rendre conforme au diagramme de séquence ainsi complété.

5.2.3 Implémentation :

Q5. 17. D'après la documentation constructeur de l'écran LCD, compléter le tableau **DR13** précisant l'ordre de remplissage des pixels du panneau.

Les pixels seront numérotés de 0 à N ou N sera à indiquer.

Q5. 18. Donner la taille maximale de la RAM graphique utilisée puis valider ce choix.

On donne, ci-dessous, les chronogrammes décrivant le fonctionnement de la génération des adresses graphiques pour la ligne 0 de l'écran.

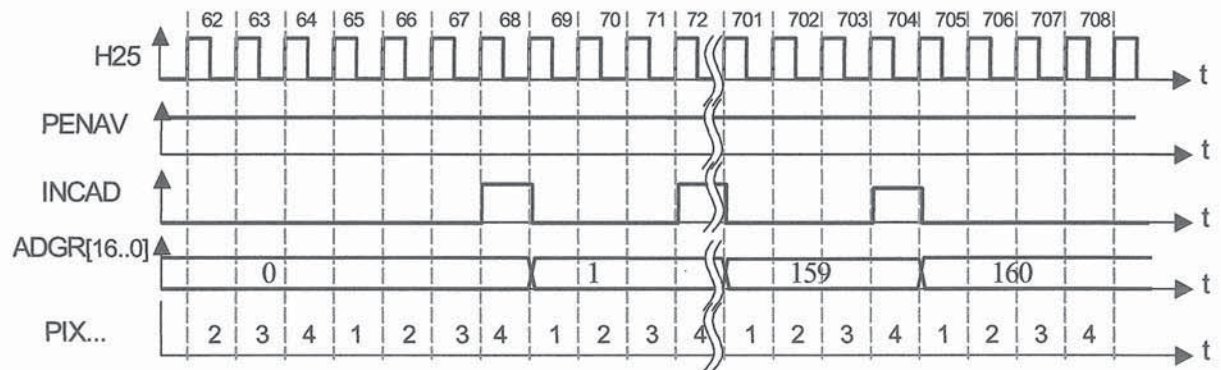


Figure 12 : Génération des adresses graphiques

Q5. 19. En déduire le plan de la RAM graphique en terme de pixels dans le tableau donné DR13.

On notera, dans ce tableau, le mot "neutre" pour les pixels dits "inactifs".

Pour la suite de l'étude, On donne, ci-dessous, l'entête const.h contenant la liste des constantes relatives à l'interface graphique.

```
#define RAMGBASE 0x00080000
#define NOIR 0x00
#define ROUGE 0x01
#define VERT 0x02
#define BLEU 0x03
#define MAGENTA 0x04
#define JAUNE 0x05
#define CYAN 0x06
#define ROUGE FONCE 0x07
#define VERT FONCE 0x08
#define BLEU FONCE 0x09
#define ORANGE 0x0A
#define VIOLET 0x0B
#define BLEUVERT 0x0C
#define GRISFONCE 0x0D
#define GRISCLAIR 0x0E
#define BLANC 0x0F
```

Figure 13 : Fichier "const.h"

L'affichage d'un texte est consécutif à la création et l'activation d'une fenêtre. Cette classe comporte les méthodes suivantes :

- CréerFenêtre qui :
 - active les coordonnées de la fenêtre,
 - colore le fond de la fenêtre
 - ajoute une bordure
 - positionne le point courant en bas à gauche
 - positionne le curseur en haut à gauche (Adr_curseur)
- ActiverFenêtre qui permet d'activer une fenêtre déjà créée mais ayant perdu le contrôle au profit d'une autre appartenant au même écran d'affichage. Pour ce faire elle :
 - active les paramètres et les coordonnées de la fenêtre
 - positionne le point courant en bas à gauche
 - positionne le curseur en haut à gauche (Adr_curseur)

Les méthodes de la classe "Texte" servent à :

- positionner le curseur à l'endroit où l'on souhaite écrire un caractère (PlacerCurseur(x,y) =>Adr_curseur),
- procéder à un déplacement (BougerCurseur(caractère)) en avant, en arrière, 1 ligne au dessus, 1 ligne au dessous, en début de ligne ou en début de plan en fonction du caractère transmis en paramètre (Ex : "CR" = Retour à la ligne)

Note : Les caractères de contrôle peuvent être inclus dans la chaîne de caractère du libellé.

1. Classe : Fenêtre

La méthode "CréerFenêtre()" permet, entre autre, de colorier la fenêtre dans la couleur spécifiée en propriété.

Q5. 20. En déduire le pseudo-code de la fonction Colorer() qui permettra de remplir le fond de la fenêtre définie dans le diagramme de classes.

Il est possible d'utiliser la fonction suivante :

ColorerQuartet(n1 : entier, n2 : entier, @Quartet : entier)

qui permet de colorer les pixels n1 à n2 du quartet transmis.

2. Classe : Texte - Affichage d'un caractère :

Le système peut afficher 123 caractères distincts (Majuscules, minuscules, lettres accentuées et éléments graphiques) dans 2 fontes différentes :

- la fonte 8x12
- la fonte 12x16.

Pour ce faire, deux tables de caractères sont stockées en EPROM. La couleur étant un attribut de la fenêtre d'affichage, une table binaire est suffisante pour coder un caractère.

Un exemple de déclaration de "Bit map" de la fonte 8x12 est donné ci-dessous :

```

FONTE_X EQU 8
FONTE_Y EQU 12
...
font equ *
...
Car6   dc.b %00000000 ;*F
        dc.b %01111000
        dc.b %00110000
        dc.b %00110000
        dc.b %00110010
        dc.b %00111110
        dc.b %00110010
        dc.b %00110000
        dc.b %00110001
        dc.b %01111110
        dc.b %00000000
        dc.b %00000000

```

Figure 14 : Bitmap F

Q5. 21. En déduire le pseudo-code décrivant l'écriture d'un caractère de la fonte 8x12 dans la fenêtre Titre des entrées du diagramme d'objet précédent. Cette fenêtre aura été précédemment activée.

3. **Exploitations pédagogiques :**

Il va s'agir, ici, d'établir deux séquences pédagogiques.

- L'une d'elle s'adressera à des élèves de BTS SE et concernera le principe de fonctionnement de l'écran LCD.
- la seconde concernera des élèves de BTS IRIS et l'aspect programmation d'une interface homme machine.

Q5. 22. Proposer une séquence pédagogique complète permettant de mettre en évidence le principe de fonctionnement d'un écran LCD. Une suggestion de présentation des documents, du matériel utilisé, des conditions de mise en oeuvre est attendue. Les résultats à obtenir et/ou à observer seront, également, précisés.

Q5. 23. Elaborer les documents et la démarche qui vous permettront d'exploiter ce système afin d'illustrer le concept de programmation objet.

Attentes du jury

Pour les documents réclamés, on attend une production maximale de 4 pages format A4.

Critères d'évaluation du candidat :

- *L'exactitude technique des documents présentés et le respect du système proposé,*
- *La pertinence pédagogique de la mise en situation choisie et des caractéristiques principales retenues,*
- *La cohérence et la pertinence pédagogique des représentations utilisées au niveau choisi,*
- *La pertinence des choix des travaux proposés et des résultats attendus des élèves,*
- *La qualité des descriptions et des explications*

NE RIEN ÉCRIRE DANS CE CADRE

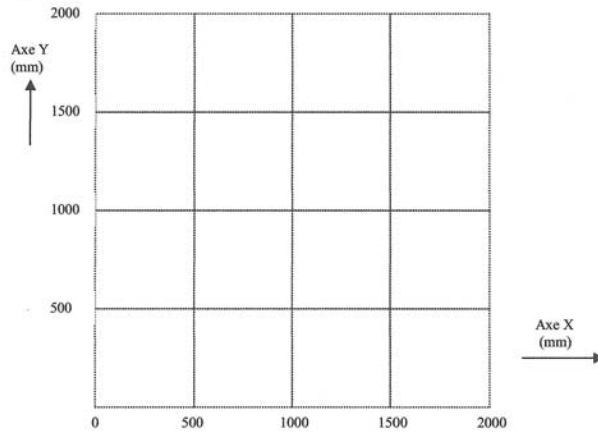
MINISTÈRE DE L'ÉDUCATION NATIONALE

Académie : _____ Session : _____
 Concours : _____
 Spécialité/option : _____ Repère de l'épreuve : _____
 Intitulé de l'épreuve : _____
 NOM : _____
(en majuscules, suivi s'il y a lieu, du nom d'épouse)
 Prénoms : _____ N° du candidat

(le numéro est celui qui figure sur la convocation ou la liste d'appel)

EAI GEA 1

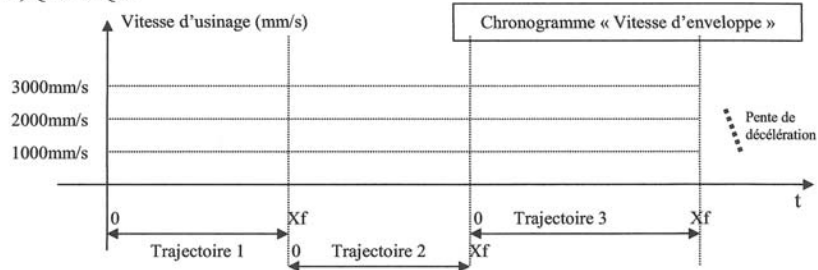
Q1.2.



Q1.3.

Mnémonique	Trajectoire 1 G1 X1000.000 Y500.000 F3000	Trajectoire 2 G1 X 2000.000 Y700.000 F1000	Trajectoire 3 G1 X1000.000 Y2000.000F3000
Xf			
Yf			
Vusinage			
ΔX			
INCX			
ΔY			
INCY			
Axe P			
LTC			
Vfinb			
Pos_dec			
Vmax			

Q1.4, Q1.5 et Q1.9



NE PAS DÉGRAFER CETTE LIASSE

NE RIEN ÉCRIRE

DANS CE CADRE

Q1.6.

<p>i = longueur pile $V_{finbloc}[i] = \dots\dots\dots$ (à compléter) Tant que $i > 0$</p>	
<p>Si $V_{usage}[i] < V_{finbloc}[i-1]$ Alors $V_{finbloc}[i-1] = V_{usage}[i]$</p>	
<p>Si $V_{usage}[i] < V_{finbloc}[i]$ (cas1) Alors $V_{max}[i] = V_{usage}[i]$ $Pos_dec[i] = LTC[i]$</p>	
<p>Sinon Calcul de V_{xd} : $V_{xd} = \sqrt{2 * \gamma * LTC[i] + V_{finbloc[i-1]}^2}$ V_{xd} est la vitesse théorique correspondant à une vitesse en début de bloc permettant d'obtenir $V_{finbloc}[i]$ par décélération constante (voir dessins ci-dessous)</p>	
<p>Si $V_{xd} \leq V_{finbloc}[i-1]$ (cas2) Alors $V_{max}[i] = \dots\dots\dots$ $V_{finbloc}[i-1] = \dots\dots\dots$ $Pos_dec = \dots\dots\dots$ (à compléter)</p>	
<p>Sinon (cas3) Si ($V_{xd} > V_{usage}[i]$ et $V_{finbloc}[i-1] = V_{usage}[i]$) Alors $V_{max}[i] = \dots\dots\dots$ (à compléter)</p>	
<p>Sinon (cas 4) Si ($V_{xd} > V_{usage}[i]$ et $V_{finbloc}[i-1] < V_{usage}[i]$) Alors $V_{max}[i] = \dots\dots\dots$ (à compléter)</p>	<p>Deux cas possibles :</p>
<p>Pos_dec (cas 3 et 4) = $LTC - (V_{max}^2 - V_{finp}^2) / 2\gamma$</p>	
<p>$i = i-1$ Fin Tant Que</p>	

NE RIEN ÉCRIRE

DANS CE CADRE

Q1.10 Pseudo code à compléter :

Initialisation Programme d'interruption

```
Vit_cur = Vfinbloc[i-1]
X_cur=0
```

Programme d'interruption sur une HTR

```
SI (X_cur ≤ Pos_dec) alors // cas position < Pos_dec
  SI (Vit_cur < Vmax) alors
    Vit_cur = Vit_cur + INC
  SI (Vit_cur > Vmax) alors
    Vit_cur = ..... ???
  FIN SI
FIN SI
SINON alors // cas position > Pos_dec
  Vit_cur = Vit_cur - ..... ??????????
FIN SI

X_cur = X_cur + ..... // Mise à jour position curviligne
FIN SI

SI(X_cur > LTC) alors // On ne s'intéressera pas au changement de bloc
```

Q1. 11

X_Cons_Pos =

Y_Cons_Pos =

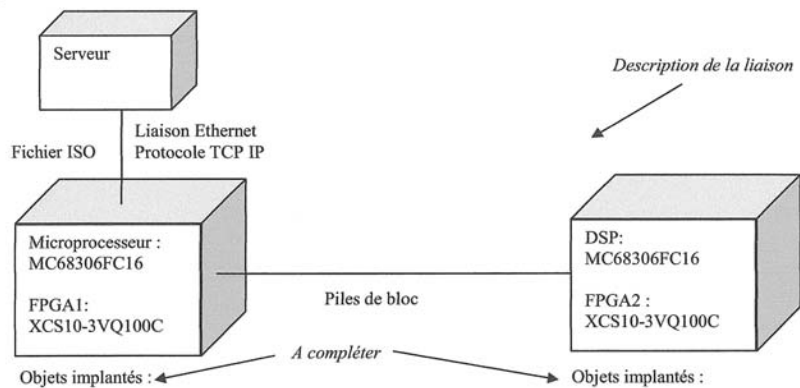
NE RIEN ÉCRIRE

DANS CE CADRE

Q1.13

Nombre d'instructions par HTR				
Ressources utilisées par l'asservissement :				
Nombre instructions / axe		Nombre d'axes	Nbr instructions / HTR	
100		5	500	
Taux d'utilisation du DSP				
Ressources utilisées par le calcul d'enveloppe :				
Opérations flottants	Addition	Multiplication	Division	Racine carrée
Instructions/Opération	40 instructions	40 instructions	240 instructions	250 instructions
Nombre par bloc				
Instructions/Bloc				
Nombre instructions / bloc		Nombre maxi de blocs	Nbr instructions / HTR	
		10		
Taux d'occupation du DSP				
Ressources utilisées par le calcul de la vitesse curviligne :				
Opérations flottants	Addition	Multiplication	Division	Racine carrée
Instructions/Opération	40 instructions	40 instructions	240 instructions	250 instructions
Nombre par HTR				
Instructions/Bloc				
Nombre instructions				
Taux d'occupation du DSP				
Ressources utilisées par le calcul des consignes de position :				
Opérations flottants	Addition	Multiplication	Division	Racine carrée
Instructions/Opération	40 instructions	40 instructions	240 instructions	250 instructions
Nombre par HTR				
Instructions/Bloc				
Nombre instructions / axe		Nombre d'axes	Nbr instructions / HTR	
		4 (X1,X2,Y1,Y2)		
Taux d'occupation du DSP				
Ressources utilisées par l'objet pilotage axe sur le DSP :				
Taux d'occupation total du DSP				

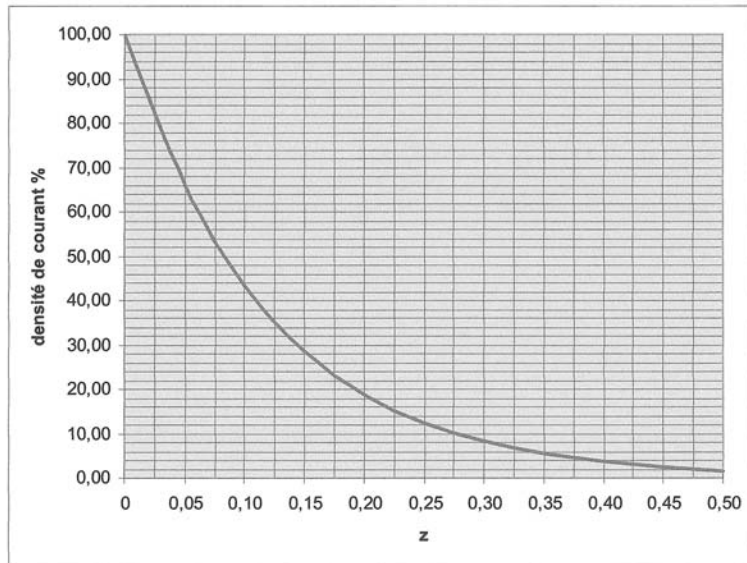
Q1.15



NE RIEN ÉCRIRE

DANS CE CADRE

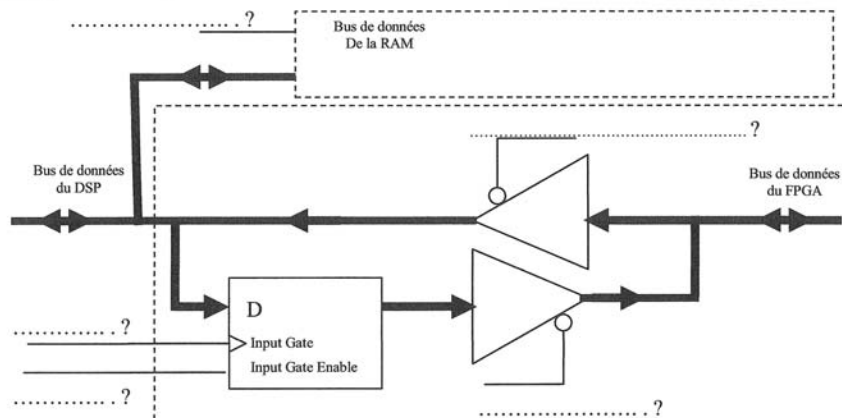
Q2.9 Densité de courant par rapport à la profondeur dans le matériau

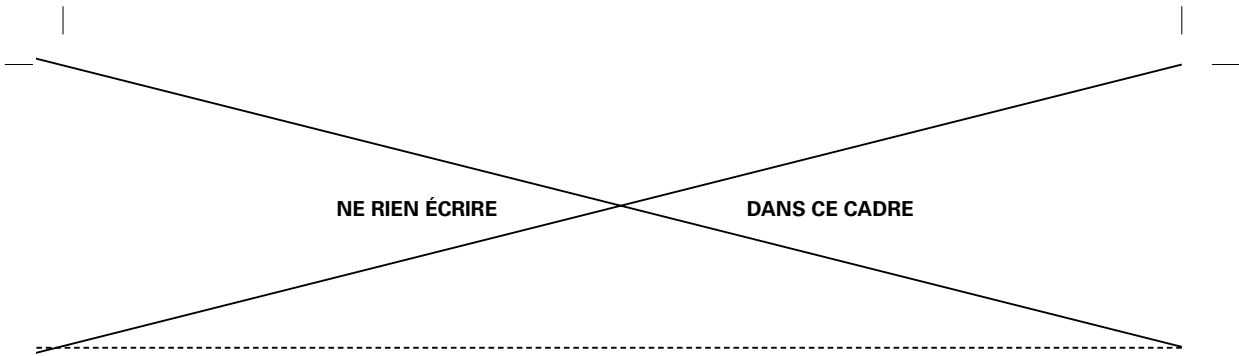


Q2.24

SD23..0 :
SA15..0 :
/DS :
X/Y :
/WR :
/RD :
CLK : Horloge 80MHz délivrée par FPGA2

Q2.25

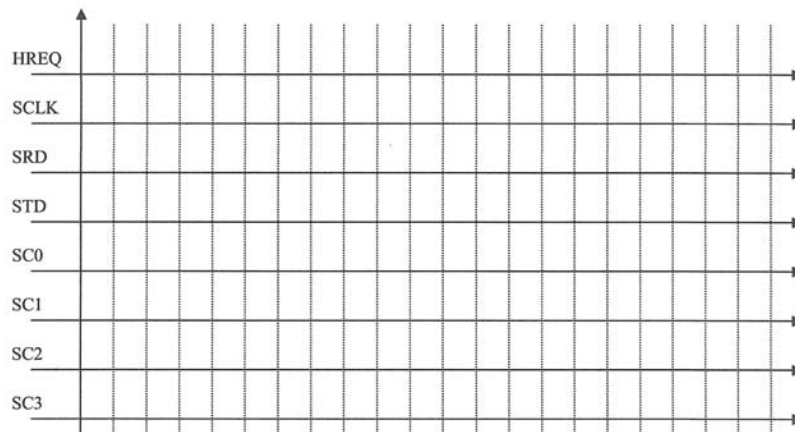




Q2.32

	Transfert DSP -> RAM	Transfert RAM -> DSP	Transfert DSP -> FPGA	Transfert FPGA -> DSP
Opération				

Q3.4



Q3.7

```

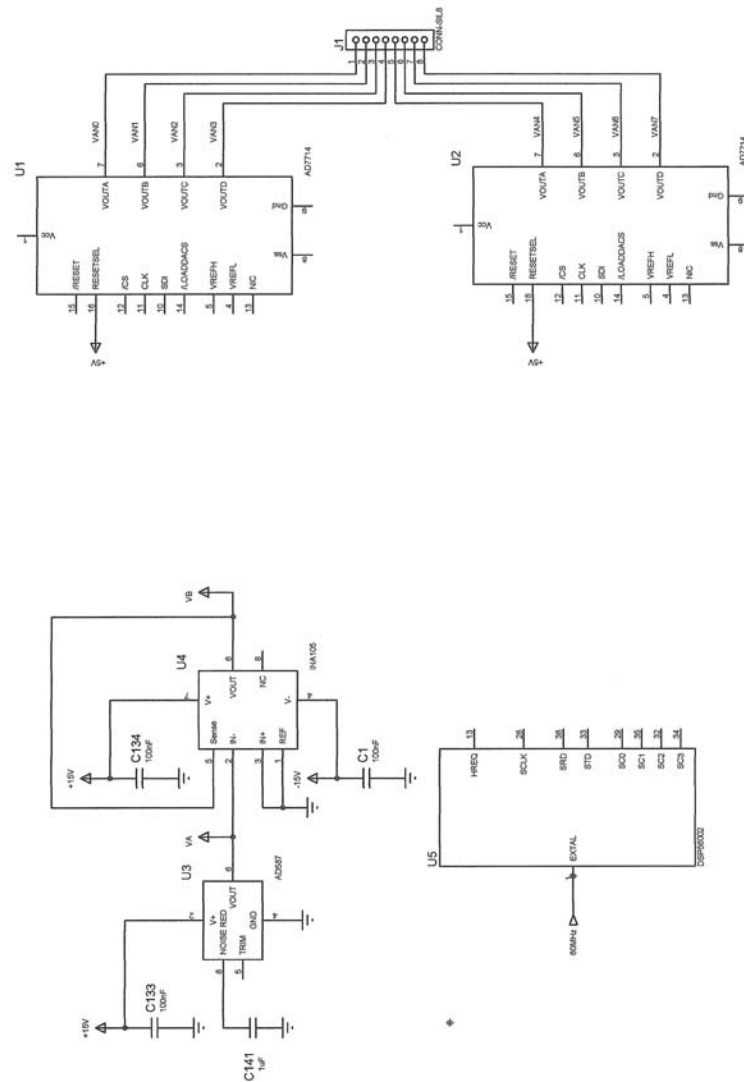
procédure ecrire_DAC(entier numero_DAC, entier numero_sortie, entier consigne_vitesse) est
déclarer entier valeur ;                               // valeur est un entier

fin procédure ;
  
```

NE RIEN ÉCRIRE

DANS CE CADRE

Q3.2 Schéma partiel de commande des DACs à compléter :



NE RIEN ÉCRIRE

DANS CE CADRE

Q3.6

Configuration de HSAR					
Bits	HA6	HA5	HA4	HA3	HA1
Etat					

Justification :

Configuration de HCKR						
Bits	HFM1	HFM0	HDM5...HDM0	HRS	CPOL	CPHA
Etat						

Justification :

Configuration de HCSR (LSB)									
Bits	HTIE	HBIE	HIDLE	HRQE1..0	HMST	HFIFO	HM1..0	HI2C	HEN
Etat									

Justification :

Configuration de HCSR (MSB)								
Bits	HBUSY	HBER	HROE	HRFF	HRNE	HTDE	HTUE	HRIE1..0
Etat								

Justification :

NE RIEN ÉCRIRE

DANS CE CADRE

Q4.1. Couche 2 du modèle OSI :
Entité traitée sur cette couche :

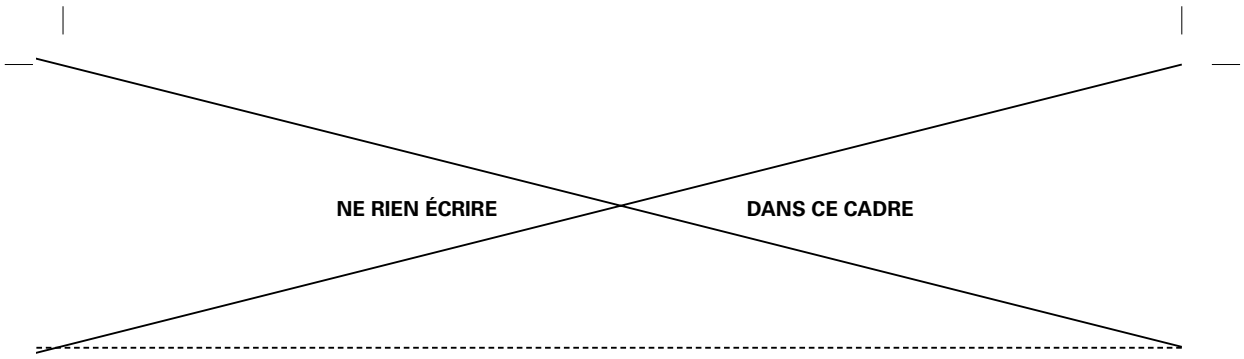
Q4.2. Couche 3 du modèle OSI :
Entité traitée sur cette couche :

Q4.3 à Q4.6

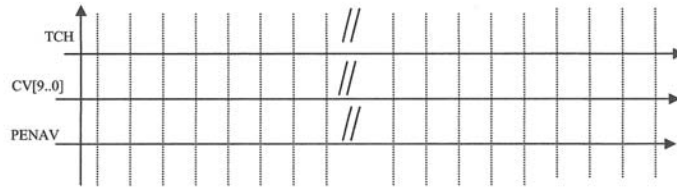
<i>Achat d'une adresse publique :</i>			
Réseau Public	Adresse (en BCD) :		Masque (en BCD) :
	200.20.2.XXX		
	Net ID (en BCD) :	Host ID (en BCD) :	
	Nombre maximal de machines connectées :		

Q4.7 à Q4.11

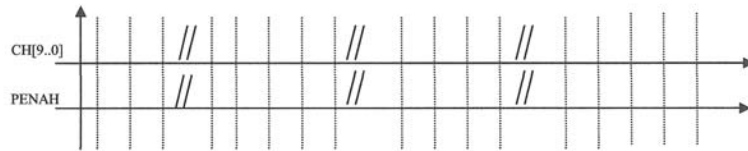
<i>Organisation du réseau interne :</i>					
Subnet 1	Net ID : (en BCD)	SubNet ID : (en binaire)	Host ID : (en binaire)	Adresse IP : (en BCD)	Masque : (en BCD)
	Nombre maximal de machines connectées :				
Subnet 2	Net ID : (en BCD)	SubNet ID : (en binaire)	Host ID : (en binaire)	Adresse IP : (en BCD)	Masque : (en BCD)
	Nombre maximal de machines connectées :				
Subnet 3	Net ID : (en BCD)	SubNet ID : (en binaire)	Host ID : (en binaire)	Adresse IP : (en BCD)	Masque : (en BCD)
	Nombre maximal de machines connectées :				
Machine 1	Adresse IP : (en BCD)		Masque : (en BCD)		
Machine 2	Adresse IP : (en BCD)		Masque : (en BCD)		
Machine 3	Adresse IP : (en BCD)		Masque : (en BCD)		



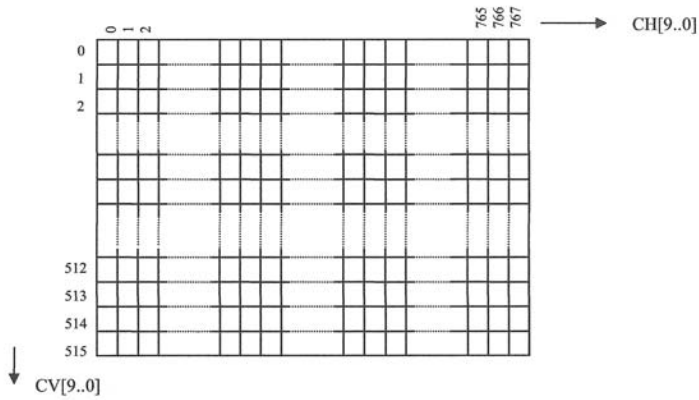
Q5.7



Q5.8



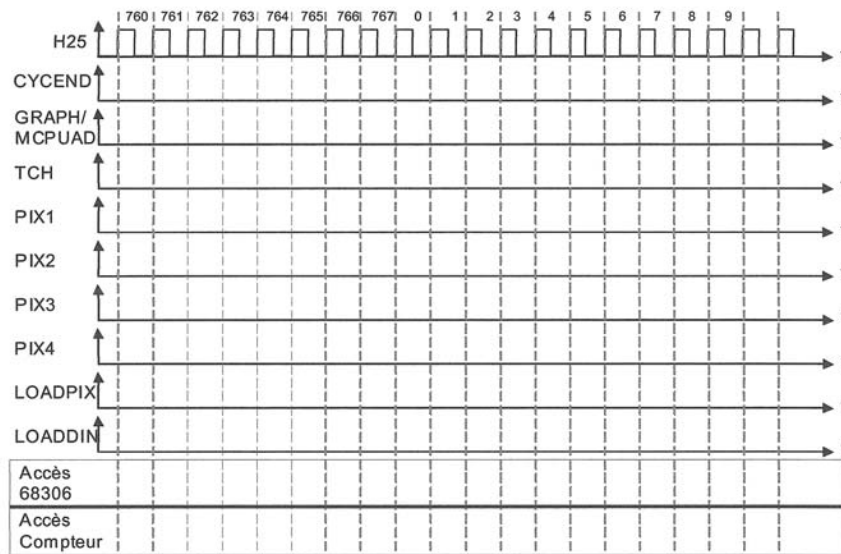
Q5.9



NE RIEN ÉCRIRE

DANS CE CADRE

Q5.10 et Q5.14



Q5.16

