

## TD

### Architecture des Ordinateurs et Microprocesseurs

#### Exercice 1 :

Donnez la taille (nb d'octets) de la représentation-mémoire de ces instructions

*MOVE.W \$80004,D3*

*MOVEQ.B \$04,D0*

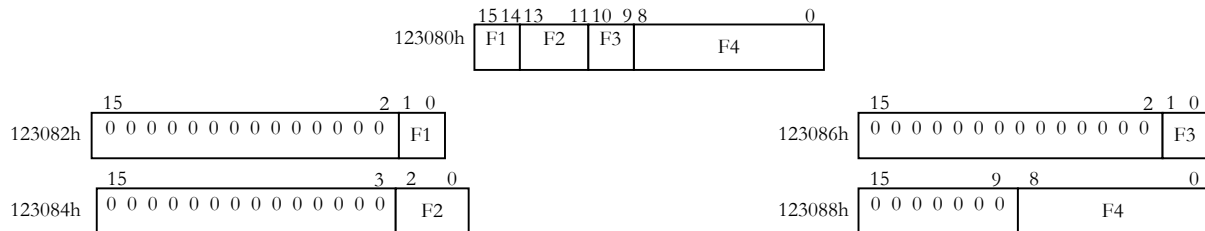
*MOVE.L (A0)+,D0*

#### Exercice 2 :

Proposez un sous programme qui découpe le mot mémoire d'adresse 123080h comme suit :

- les bits 14-15 sont rangés dans le mot d'adresse 123082h,
- les bits 11-13 sont rangés dans le mot d'adresse 123084h,
- les bits 9-10 sont rangés dans le mot d'adresse 123086h,
- les bits 0-8 sont rangés dans le mot d'adresse 123088h,
- les bits restants des quatre mots de destination sont à 0.

Ce sous programme réside dans la mémoire à partir de l'adresse 120100h.



#### Exercice 3 :

- 1- Le tableau suivant donne le contenu d'un bloc de la mémoire RAM. Il s'agit de la représentation-mémoire d'un petit programme assembleur 68000. Déduisez les instructions de ce programme (Désassemblage).

.....
.....
.....
.....
.....
.....
.....

\$ 400400	\$ 34	\$ 00
\$ 400402	\$ 70	\$ 70
\$ 400404	\$ 00	\$ 70
\$ 400406	\$ 34	\$ 00
\$ 400408	\$ E9	\$ 18
\$ 40040A	\$ 55	\$ 42

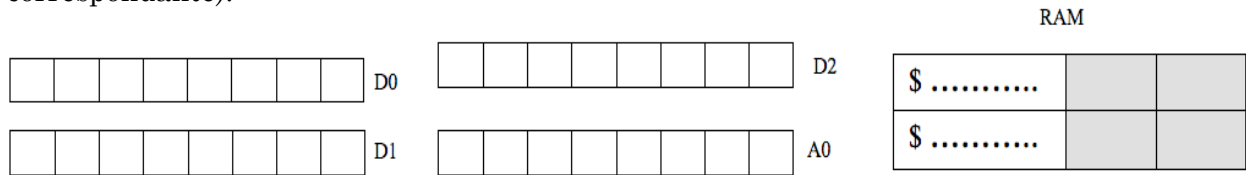
2- Soit le programme assembleur 68000 suivant

```

MOVEA.W    #$AF30, A0
SUBQ.W     #2,D2
MOVE       D0, (A0)+
MOVEQ      -(A0), D0
MOVE.B     #$70, D1

```

Sachant qu'initialement, les registres D0, D1 et D2 contiennent la valeur \$90909090, quels sont les résultats obtenus suite à l'exécution de ce programme ?  
(Donnez le contenu de chaque registre ainsi que l'adresse et le contenu de la case mémoire correspondante).



#### Exercice 4 :

Un tableau de \$200 octets réside dans la mémoire, à partir de l'adresse 500600h, d'un système informatique de contrôle d'une machine de production est à base du processeur 68000.  
Donner un sous programme 'separe' permettant de séparer les valeurs (octets) paires de celles impaires et de ranger les octets pairs à partir de l'adresse 500700h et les octets impairs à partir de l'adresse 500800h.

#### Exercice 5 :

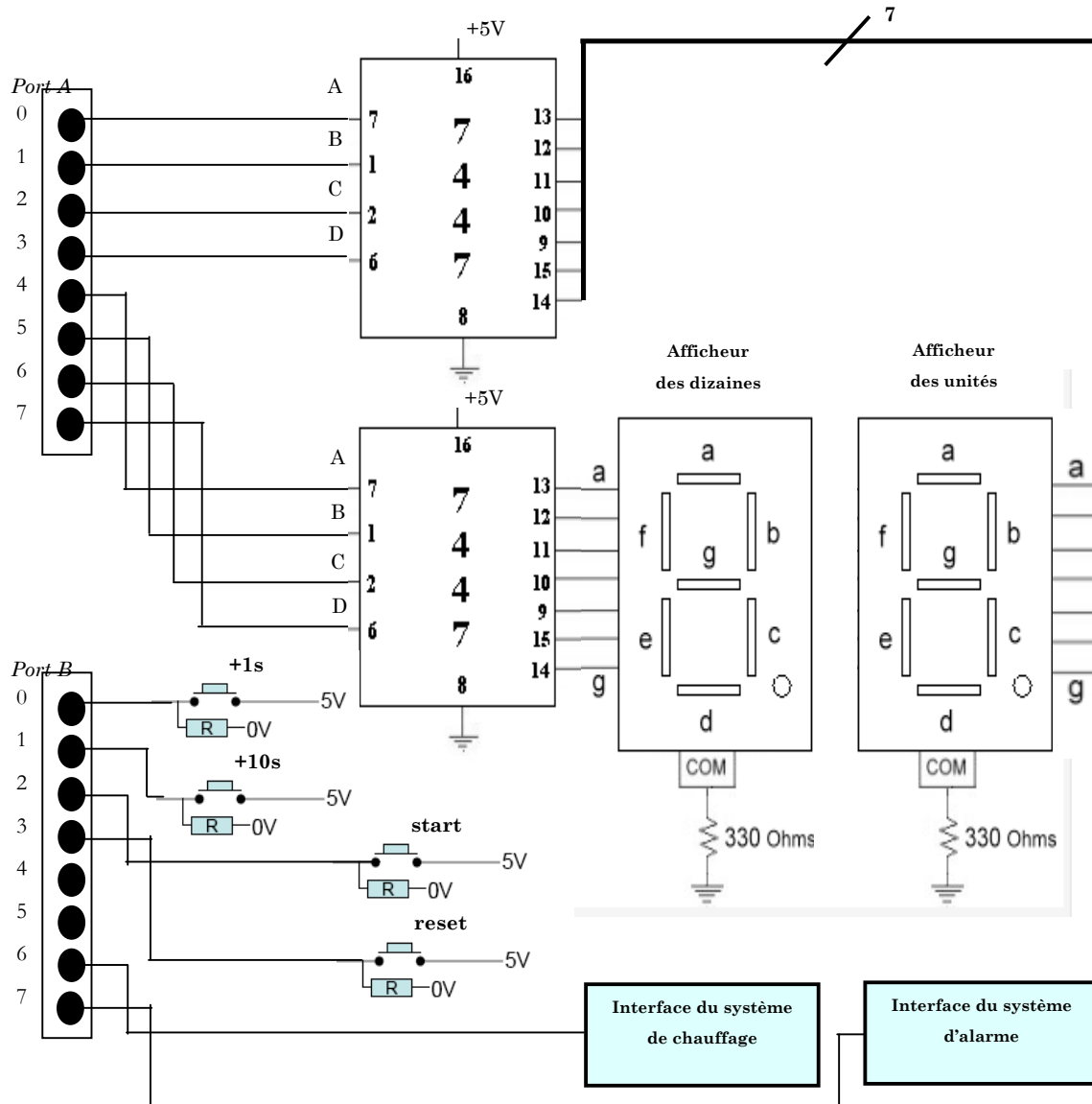
Il s'agit de réaliser un système destiné à la préparation de quelques produits biochimiques et chimiques au laboratoire. On se limitera dans cet exercice à l'étude de dispositif de chauffage. Ce dispositif est commandé par un système à base du microprocesseur 68000 à travers le circuit d'interface parallèle du type PI/T 68230.

Avec ce dispositif, l'utilisateur sélectionne un temps de chauffage exprimé en secondes, à l'aide de 2 boutons '+1s' et '+10s'. Puis, après l'appui d'un bouton '**start**', le chauffage démarre tandis que le temps restant à s'écouler est affiché sur une minuterie constituée de 2 afficheurs 7 segments. Si le temps choisi par l'utilisateur est écoulé, un petit système d'alarme est déclenché pendant 2 secondes. Un bouton '**reset**' permet d'arrêter et de remettre la minuterie à zéro.

Les caractéristiques de cette application sont les suivantes :

- Les broches 0 à 3 du port A (de PI/T 68230) sont connectées au premier afficheur 7 segments (afficheur des unités) par l'intermédiaire d'un décodeur BCD/7 Segments, le IC7447.
- Les broches 4 à 7 du port A sont connectées au deuxième afficheur 7 segments (afficheur des dizaines) par l'intermédiaire d'un autre décodeur de même type.
- Les broches 0 à 3 du port B sont connectées à 4 boutons permettant à l'utilisateur de programmer le temps de cuisson, de démarrer la cuisson ou d'interrompre celle-ci. Ils sont à l'état logique 0 au repos et à 1 lorsqu'ils sont appuyés.
- Les broches 6 et 7 du port B commandent respectivement le système de chauffage et celui d'alarme (0 : arrêter, 1 : Activer).

Ci-après le schéma de synthèse du circuit, utilisé par ce dispositif.



- 1) Soit A, B, C, D les entrées du décodeur IC7447 et a, b, c, d, e, f, et g les sorties qui commandent les segments de chaque afficheur. Veuillez remplir le tableau suivant

Code BCD				Sorties							Etat afficheur
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	1	1	1	1	1	1	0	0
											1
											2
											3
											4
											5
											6
											7
											8
1	0	0	1	1	1	1	1	0	1	1	9

NB : les combinaisons de 10 jusqu'à 15 sont ignorés.

- 2) Le programme principal appelle un sous-programme d'initialisation des registres de direction et de contrôle de deux ports A et B de PI/T. Donnez ce sous-programme qui débute à l'adresse '*Init*' en se referant au tableau suivant.

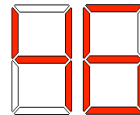
.....
.....
.....
.....
.....
.....
.....
.....

Registre	Abréviation	Adresse	Offs et
Registre de contrôle général du port	PGCR	\$800001	\$00
Registre de direction du port A	PADDR	\$800005	\$04
Registre de direction du port B	PBDDR	\$800007	\$06
Registre de contrôle du port A	PACR	\$80000D	\$0C
Registre de contrôle du port B	PBCR	\$80000F	\$0E
Registre de données du port A	PADR	\$800011	\$10
Registre de données du port B	PBDR	\$800013	\$12
Registre alterné du port A	PAAR	\$800015	\$14
Registre alterné du port B	PBAR	\$800017	\$16

- 3) Dans tous ce qui suit, on suppose que les registres de données D0 et D1 reflètent respectivement les valeurs affichées ou qui seront affichées par l'afficheur des unités et celui des dizaines.

**c-à-d :**

Si D0 contient \$06 et D1 contient \$04, la minuterie affiche la valeur suivante



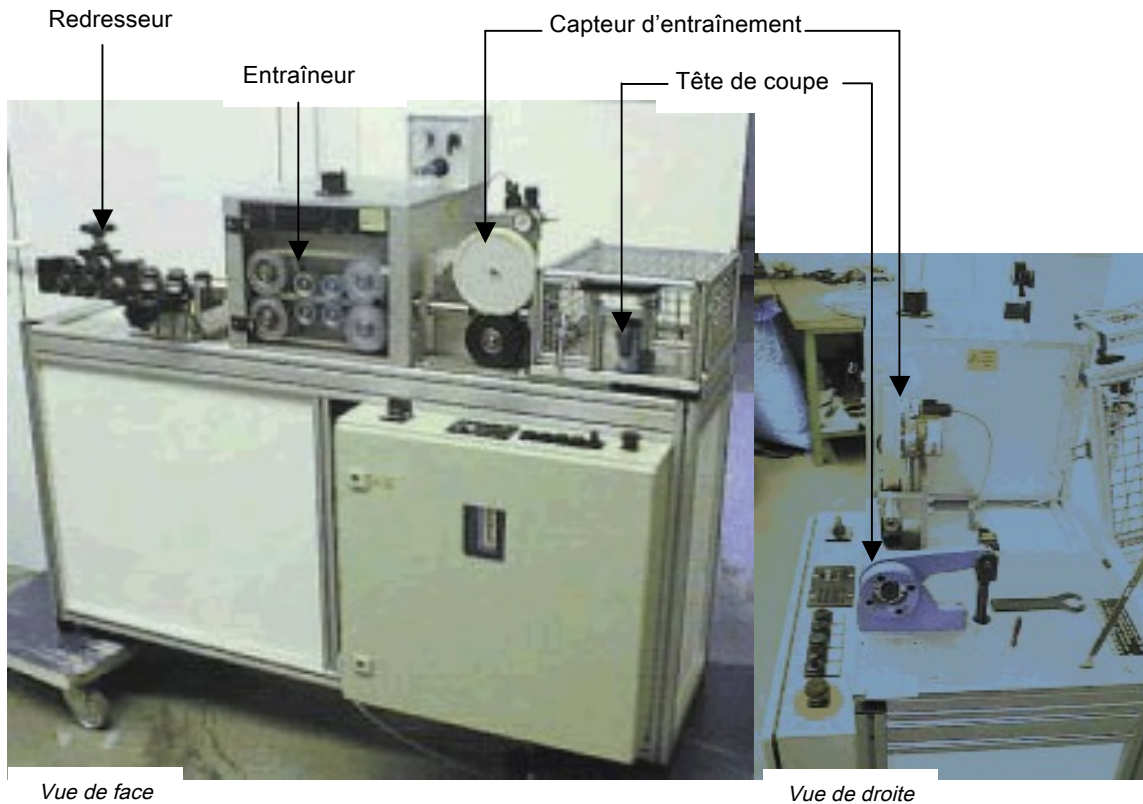
Proposez un sous-programme qui débute à l'adresse '*Display*' permettant d'afficher sur la minuterie les valeurs des unités et des dizaines lus respectivement à partir de D0 et D1. Ce sous-programme fait appel à un autre sous-programme de temporisation d'une seconde qui débute à l'adresse '*Delay*' pour pouvoir visualiser chaque changement. Donnez aussi ce sous-programme de temporisation.

<i>Display</i> .....
.....
.....
.....
.....
.....
.....
.....

<i>Delay</i> .....
.....
.....
.....
.....
.....
.....
.....

- 4) Complétez le programme principal, suivant, qui débute à l'adresse \$400400, en tenant compte des indications suivantes :
- \* Vous pouvez éventuellement changer la partie du programme donnée si elle ne correspond pas à votre propre choix des registres.
  - \* Si l'utilisateur appuie sur le bouton '**+1s**' et que l'afficheur des unités affiche la valeur 9, celui ci est mise à zéro tandis que l'afficheur des dizaines est incrémenté de 1 tant que ce dernier est inférieure à 9. Si la minuterie contient la valeur 99, tout appuis sur le bouton '**+1s**' sera ignoré par le microprocesseur.
  - \* Si l'utilisateur appuie sur le bouton '**+10s**' et que l'afficheur des dizaines affiche la valeur 9, cet appuie sera ignoré par le microprocesseur.





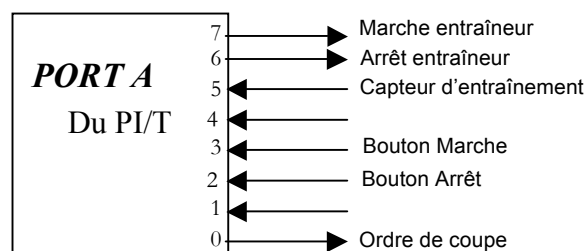
**Figure 3.** Machine de découpage des gaines en plastique

La gaine, généralement enroulée sur des bobines (figure 4), est tirée par un système d'entraînement. Elle est assez rigide, pour cette raison, à son entrée du côté gauche de la machine, elle est redressée afin de la rendre assez rectiligne facilitant ainsi l'opération de coupe.



**Figure 4.** Bobine de gaine en plastique

Le microprocesseur 68000 commande ce système par l'intermédiaire du port A d'un circuit d'interface PI/T 68230. La figure 5 schématise l'affectation des lignes du port A aux différentes lignes de commande et d'état du système et le tableau 2 donne les adresses des différents registres du PI/T.



**Figure 5.** Affectation des lignes du port A

**Tableau 2. Adresses des registres du PIT**

Registre	Abréviation	Adresse	Offset
Registre de contrôle général du port	PGCR	\$A10001	\$00
Registre de demande de service du port	PSRR	\$A10003	\$02
Registre de direction du port A	PADDR	\$A10005	\$04
Registre de direction du port B	PBDDR	\$A10007	\$06
Registre de direction du port C	PCDDR	\$A10009	\$08
Registre vecteur d'interruption du port	PIVR	\$A1000B	\$0A
Registre de contrôle du port A	PACR	\$A1000D	\$0C
Registre de contrôle du port B	PBCR	\$A1000F	\$0E
Registre de données du port A	PADR	\$A10011	\$10
Registre de données du port B	PBDR	\$A10013	\$12
Registre alterné du port A	PAAR	\$A10015	\$14
Registre alterné du port B	PBAR	\$A10017	\$16
Registre de données du port C	PCDR	\$A10019	\$18
Registre d'état du port	PSR	\$A1001B	\$1A
Registre de contrôle du Timer	TCR	\$A10021	\$20
Registre vecteur d'interruption du Timer	TIVR	\$A10023	\$22
Registre de préchargement du compteur Haut	CPRH	\$A10027	\$26
Registre de préchargement du compteur Moyen	CPRM	\$A10029	\$28
Registre de préchargement du compteur Bas	CPRL	\$A1002B	\$2A
Registre de comptage Haut	CNTRH	\$A1002F	\$2E
Registre de comptage Moyen	CNTRM	\$A10031	\$30
Registre de comptage Bas	CNTRL	\$A10033	\$32
Registre d'état du Timer	TSR	\$A10035	\$34

1- Le programme de gestion de ce système commence par appeler un sous programme d'initialisation des registres de direction. Donnez ce sous programme qui commence à l'adresse \$400560.

2- Le capteur d'entraînement fournit un signal à impulsions (Une impulsion après chaque centimètre de gaine) comme celui représenté sur la figure 6.



**Figure 6.** Signal fourni par le capteur d'entraînement

2-1- Donnez le sous programme de détection d'un front montant qui débute à l'adresse \$400590.

2-2- Donnez le sous programme de coupe qui commence à l'adresse \$4005C0. Ce sous programme contrôle la longueur demandée puis donne l'ordre de coupe en envoyant un niveau haut sur la sortie 0 du port A. Il finit en appelant un sous programme de temporisation à l'adresse \$400400.

1ère version : La longueur demandée des morceaux de gaine est de 1m20cm.

2ème version : La longueur demandée en cm est contenue dans le mot mémoire \$400600.

3- Donnez le programme principal qui commence par appeler le sous programme d'initialisation. Puis, il passe à une phase d'attente d'appuie sur le bouton de marche qui provoque un niveau haut sur l'entrée 'Marche'. Par la suite, le programme passe à une boucle infinie de coupe de morceaux de gaines. Dans cette boucle, on appelle le sous programme de contrôle de la longueur et de coupe. L'adresse de début du programme principal est \$400500. On demande deux versions de ce programme, la première utilise, pour appeler un sous programme, uniquement l'instruction de saut et la deuxième version utilise uniquement l'instruction de branchement à un sous programme.

4- Améliorez le programme principal de manière à arrêter l'entraînement du câble dès qu'on appuie sur le bouton d'arrêt (Niveau actif : Haut). On se contente ici d'un test de l'état du bouton d'arrêt à la fin de chaque boucle de coupe (Niveau actif : Haut).