

Circuits d'interfaces

KIT MDA 8086

Préparé par : TLILI KAIS

Année Universitaire : 2012/2013

1- Configuration MDA-8086

Introduction :

8255 est un circuit d'interface programmable (PPI : Programmable Peripheral Interface). Il est compatible avec les microprocesseurs Intel ou autres types et possède 3 ports d'entrées/sorties parallèles (I/O) port A, B, C. Il a 24 pins d'entrées/sorties programmable par groupe de 12 et fonctionne en 3 modes.

Objectif : se familiariser avec le kit MDA8086 et connaître les fonctions de ces composants

Equipement : le kit MDA-8086 et un PC avec WinnComm

Les fonctions des composants du kit MDA-8086 :

- 1- **CPU (Central processing unit)** : utilise un Intel 8086 à une fréquence 4.9152 MHz.
- 2- **ROM (Read Only Memory)**: contient un programme pour Contrôler les touches d'entrées de l'utilisateur. Programme utilisateur pour LCD, 64KByte
L'intervalle de l'adressage du ROM est F0000H ~ FFFFFH.
- 3- **SRAM (Static Random Access Memory)** : Entren les données et le programme utilisateur.
Son intervalle d'adressage mémoire est de 00000H ~ 0FFFFH, de taille 64Kbyte.
- 4- **AFFICHAGE**: c'est un LCD, 16(Caracteres) ×2(Lines).
- 5- **CLAVIER**: le clavier est utilisé pour entrer le langage machine, il est constitué de 16 touches hexadécimale et 10 touches de fonction
- 6- **SPEAKER** : il est utilisé pour tester un son et les synthétiseurs
- 7- **RS-232C** : Port de communication avec les PC
- 8- **MATRICE A DIODES LEDS** : Pour tester et comprendre le principe de fonctionnement des matrices à diodes commandées par un 8255A(PPI).
- 9- **CONVERTISSEUR A/D** : Convertir un signal analogique en un signal numérique avec un ADC0804
- 10- **CONVERTISSEUR D/A** : Convertir un signal numérique en un signal analogique avec un DAC0800 qui est branché sur une barre graphe pour voir les niveaux de tension
- 11- **INTERFACE MOTEUR PAS A PAS** : Utiliser pour contrôler les driver pour les moteurs pas à pas.
- 12- **ALIMENTATION**: AC 110 ~230V, DC +5V 3A, +12V 1A, -12V 0.5A.

2- Adressage MDA-8086

2-1- Adressage mémoire

ADRESSE	MEMOIRE	DESCRIPTION
00000H ~ 0FFFFH	RAM	Mémoire programme et données
F0000H ~ FFFFFH	ROM	ROM moniteur
10000H ~ EFFFFH		Intervalle utilisateur

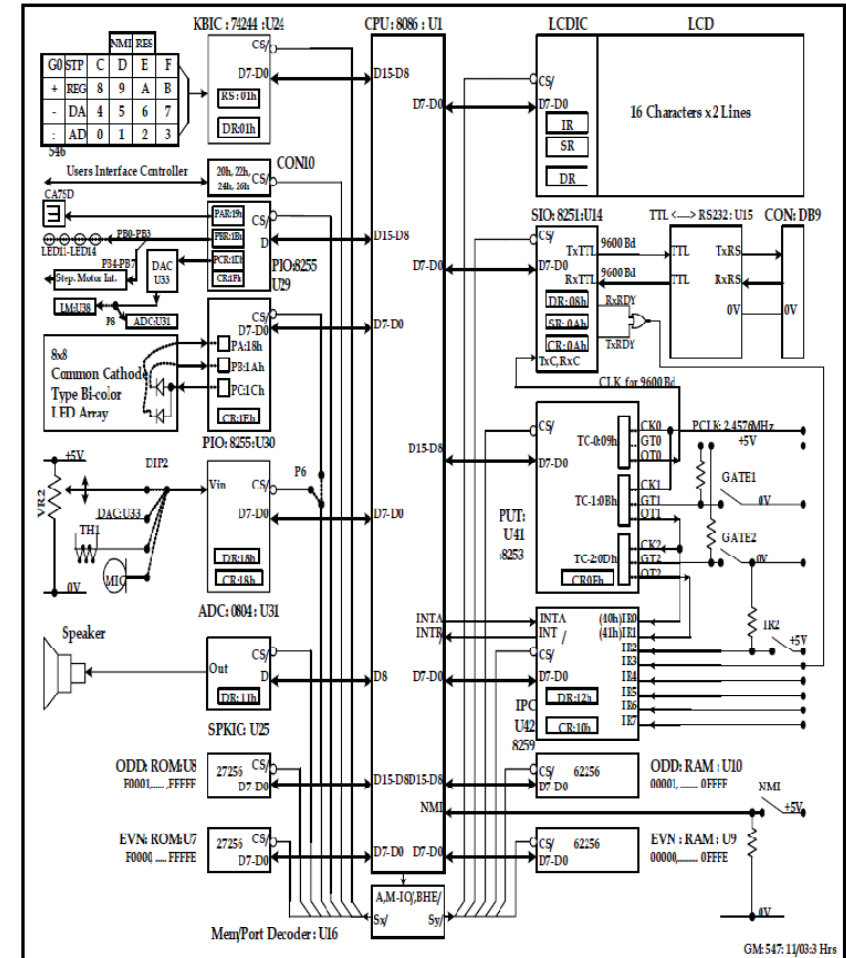
2-2- Adressage I/O (entrées sorties)

ADRESSE	PORT I/O	DESCRIPTION
00H ~ 07H	LCD et CLAVIER	LCD 00H : Registre d'instruction 02H : Registre d'état 04H : Registre données CLAVIER 01H : Registre clavier (lecture seule) 02H : drapeau(FLAG) clavier (écriture seule)
08H ~ 0FH	8251/8253	8251(pour la communication) 08H : Registre donnée 0AH : Registre d'état / d'instruction 8253(TIMER/COMPTEUR) 09H : Registre TIMERO 0BH : Registre TIMER1 0DH : Registre TIMER2 0FH : Registre DE CONTROLE
10H ~ 17H	8259/speaker	8259(Contrôleur d'interruption) 10H : Registre de commande 12H : Registre de données SPEAKER 11H : Speaker
18H ~ 1FH	8255A-CS1/ 8255A-CS2/	8255A-CS1 (interface ADC et MATRICE) 18H : Registre donnée PORT A 1AH : Registre donnée PORT B 1CH : Registre donnée PORT C 1EH : Registre DE CONTROLE 8255A-CS1 (interface ADC et MATRICE) 19H : Registre donnée PORT A 1BH : Registre donnée PORT B 1DH : Registre donnée PORT C 1FH : Registre DE CONTROLE

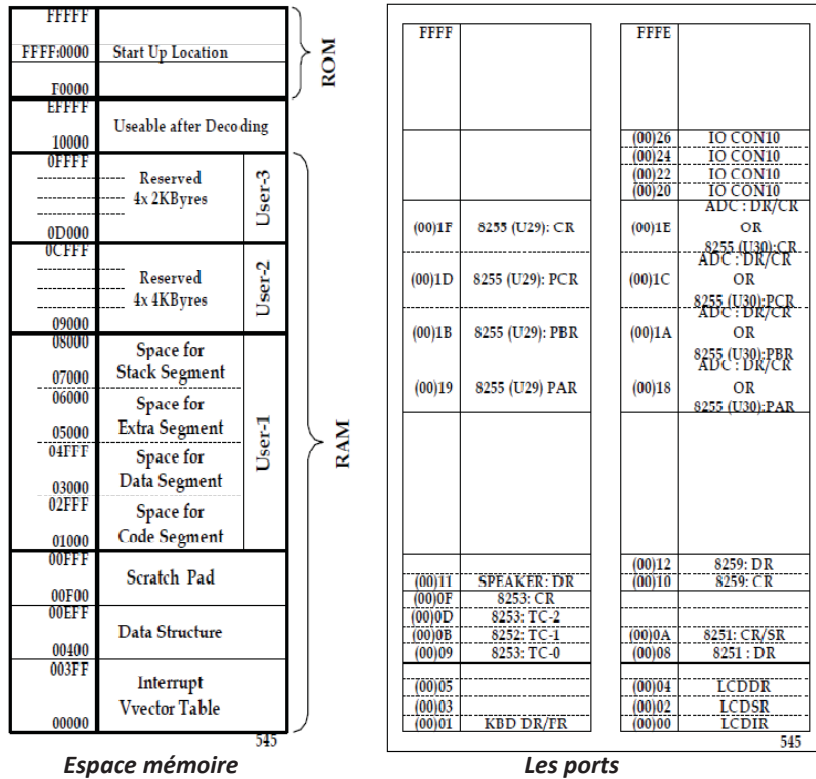
20H ~ 2FH	Connecteur d'extension I/O
30H ~ FFH	Intervalle utilisateur

2-3- Schéma bloc des circuits d'interface du kit MDA-8086

La figure ci-dessous représente le schéma bloc des circuits des interfaces relatifs au kit MDA-8086



2-3- Organisation de l'espace mémoire et des ports



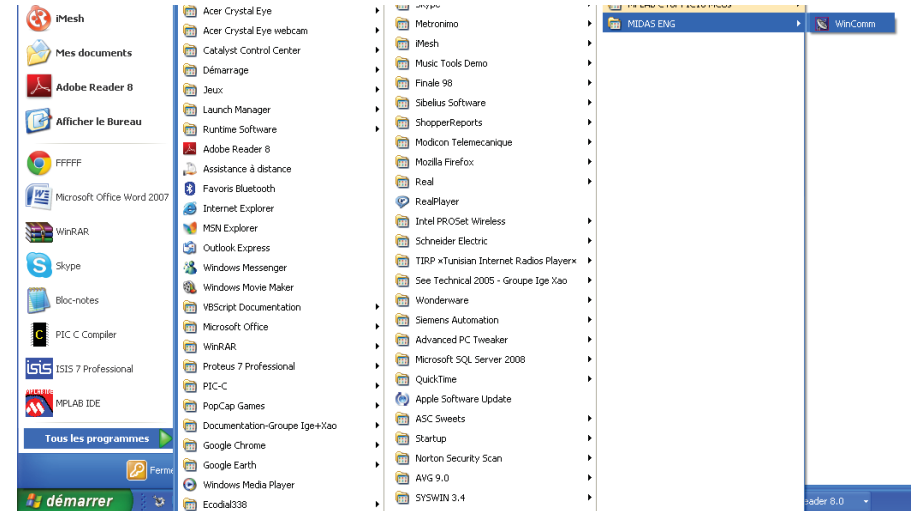
Espace mémoire

Les ports

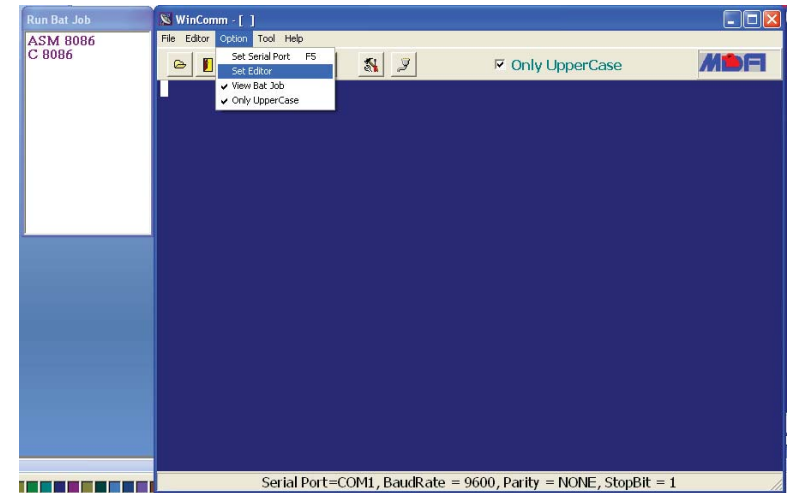
3- MDA-8086 pour Windows

Pour la communication et l'exécution avec le kit MDA-8086 on doit suivre les étapes suivantes :

- 1- Installer le programme : double click sur « **SETUP.EXE** » à partir du CD
- 2- Choisir « **Wincomm** » à partir de menu Démarrer



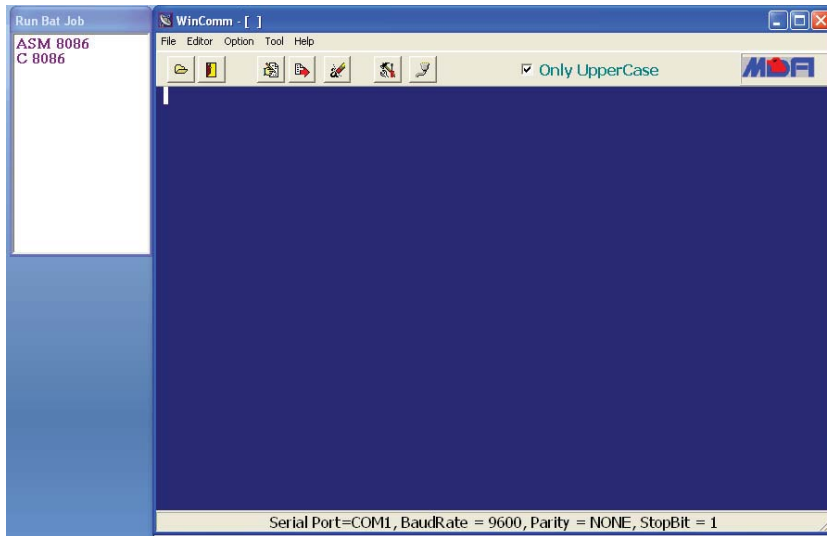
3- Choisir « Set Editor » dans le menu « Option »



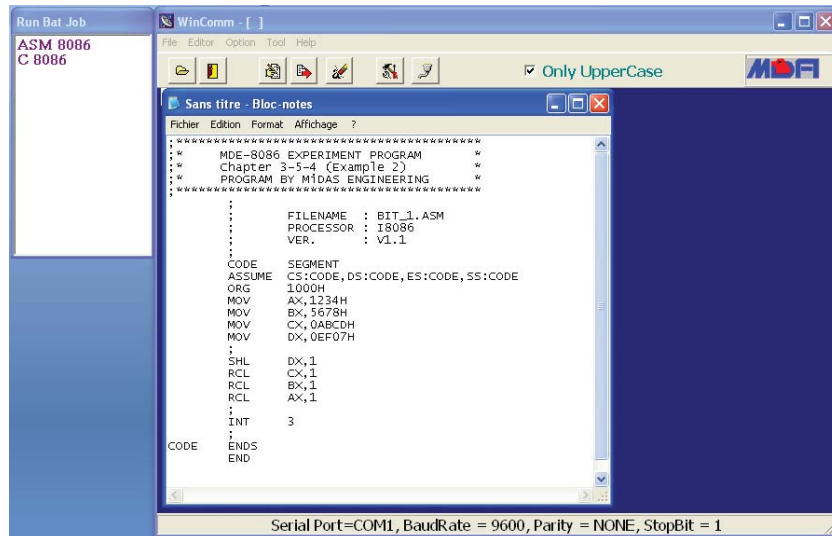
4- Choisir un éditeur de programme pour écrire un programme source

(exemple : « C:\windows\system32\notepad.exe »)

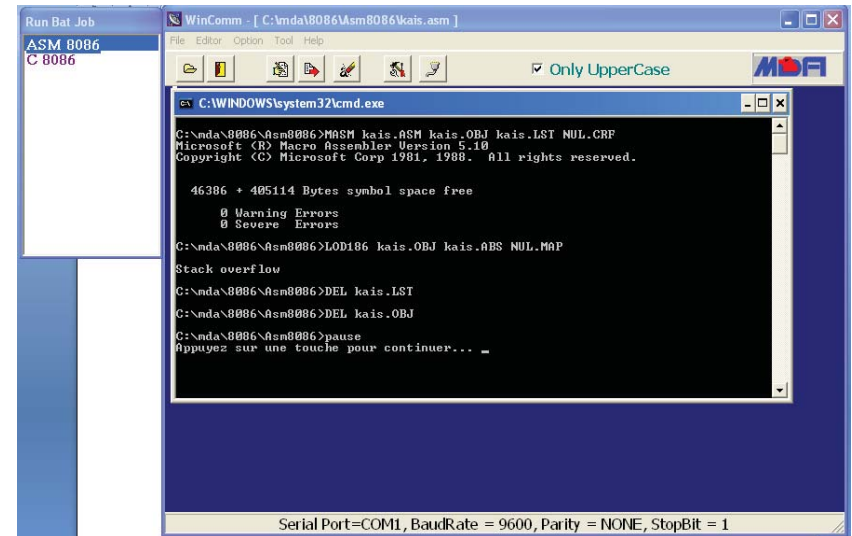
5- Cliquer « Editor » dans le menu pour exécuter l'éditeur du programme choisi



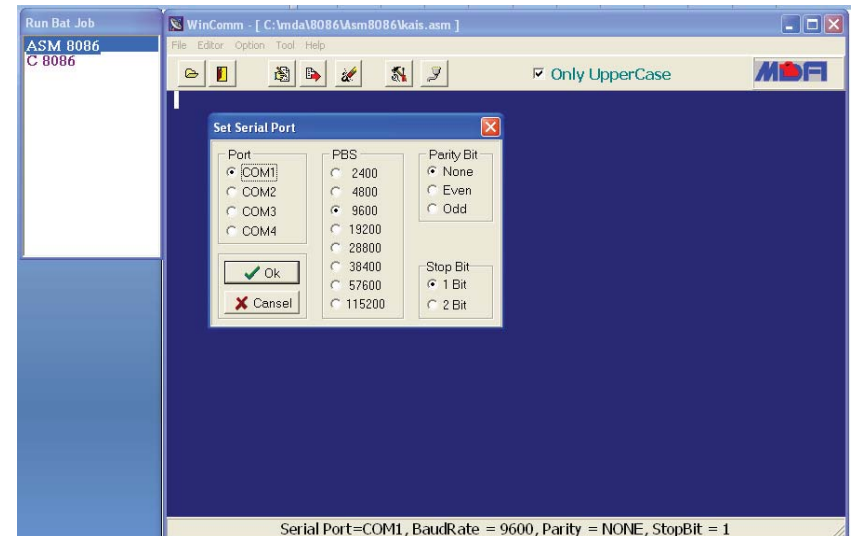
6- Ecrire le programme et l'enregistrer



7- Compiler le programme en utilisant « C 8086 » dans la fenêtre « Run Bat Job »

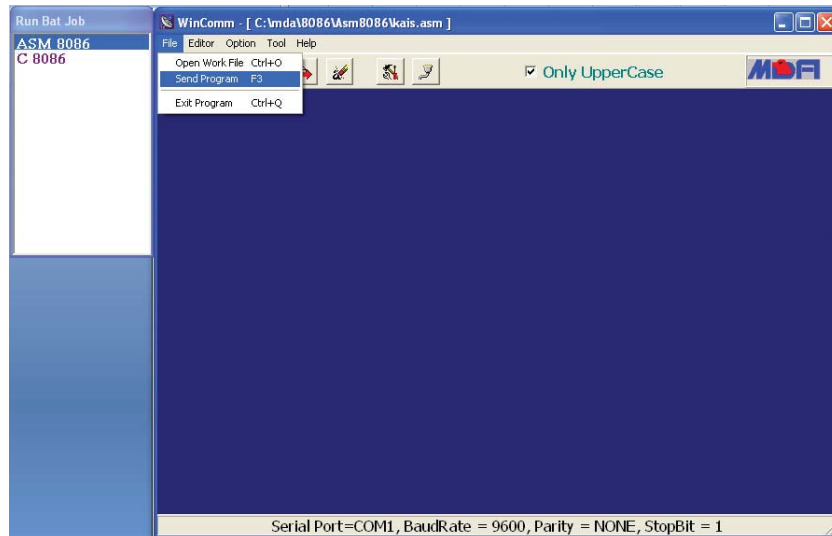


8- Configurer les paramètres de la communication série entre le PC et le KIT MDA-8086

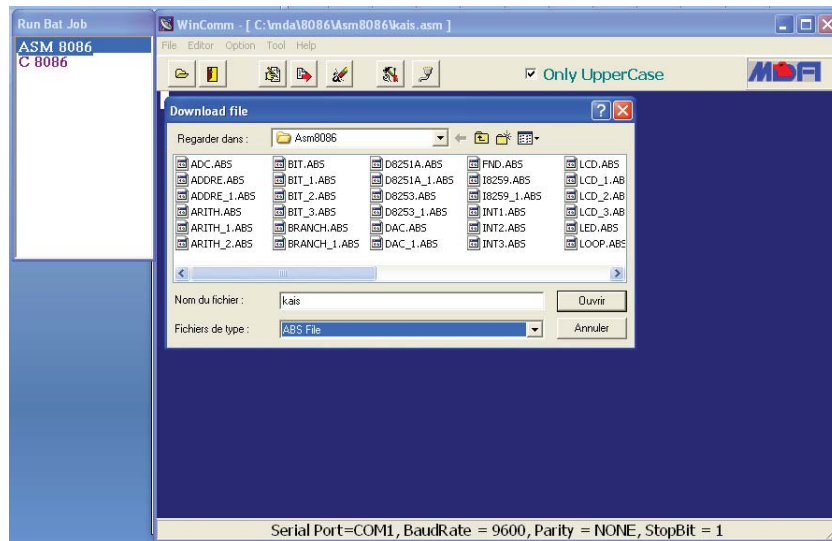


9- Enfoncer le bouton « RESET KEY » du kit MDA-8086.

10- Entrer la lettre « L » puis cliquer « Send programm » du menu « File »



11- Choisir le fichier compilé au format ABS



12- Charger le fichier

13- Taper «G 0000 :1000 » pour exécuter le programme

4- Manipulation

1- En appliquant les étapes vues précédemment, écrire et tester les deux programmes suivants

```

CODE    SEGMENT
        ASSUME    CS:CODE,DS:CODE,ES:CODE,SS:CODE
        ;
        PPIC_C    EQU    1EH ; control register
        PPIC      EQU    1CH
        PPIB      EQU    1AH
        PPIA      EQU    18H
        ;
        ORG    1000H
        MOV    AL,1000000B
        OUT    PPIC,CAL
        ;
        MOV    AL,11111111B
        OUT    PPIC,AL
        ;
        MOV    AL,11111111B
        OUT    PPIB,AL
        ;
L1:     MOV    AL,11111110B
L2:     OUT    PPIA,AL
        CALL    TIMER
        STC
        ROL    AL,1
        JC    L2
        JMP    L1
        ;
        INT    3
        ;
TIMER:  MOV    CX,0FFFFH
TIMER1: NOP
        NOP
        NOP
        NOP
        LOOP    TIMER1
        RET
        ;
CODE    ENDS
        END

```

```

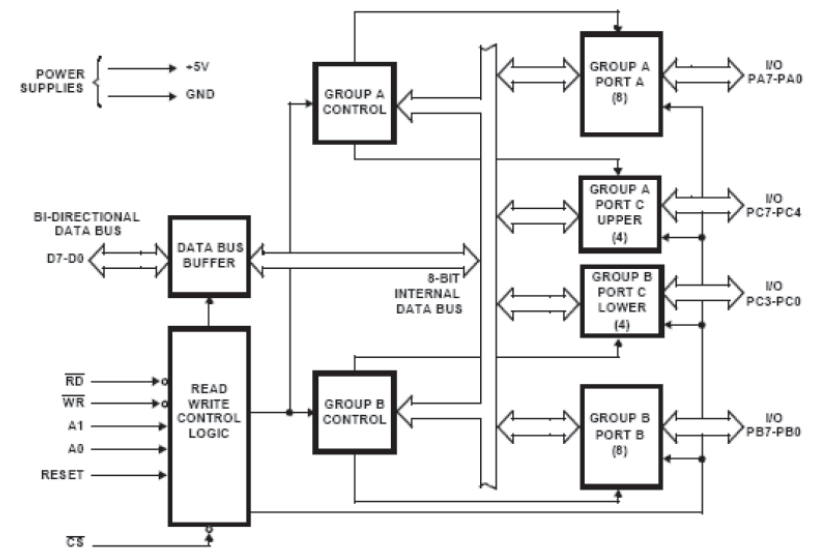
CODE SEGMENT
ASSUME CS:CODE,DS:CODE,ES:CODE,SS:CODE
SPK EQU 17H
PER EQU 0CH ;,0EH
ORG 1000H
L1: MOV SI,OFFSET FREQ_TAB
L2: MOV DH,BYTE PTR CS:[SI]
INC DH
JZ L1
DEC DH
MOV DL,PER
MOV AL,0FFH
L3: OUT SPK,AL
MOV CL,DH
AND CX,00FFH
LOOP $
XOR AL,01H
DEC DL
JNZ L3
INC SI
JMP L2
FREQ_TAB:
DB 25H ;,85H
DB 27H ;,7EH
DB 29H ;,77H
DB 2CH ;,70H
DB 2EH ;,6AH
DB 31H ;,64H
DB 34H ;,5EH
DB 37H ;,59H
DB 3BH ;,54H
DB 3EH ;,4FH
DB 42H ;,4AH
DB 46H ;,46H
DB 4AH ;,42H
DB 4FH ;,3EH
DB 54H ;,3BH
DB 59H ;,37H
DB 5EH ;,34H
DB 64H ;,31H
DB 6AH ;,2EH
DB 70H ;,2CH
DB 77H ;,29H
DB 7EH ;,27H
DB 85H ;,25H
DB 0FFH
CODE ENDS
END

```

5- Interface parallèle PPI 8255

Introduction :

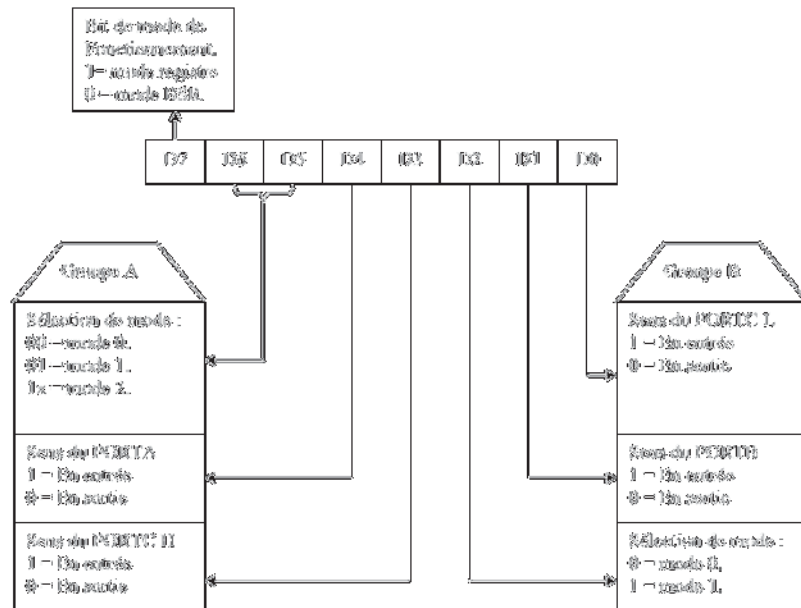
8255 est un circuit d'interface programmable (PPI : Programmable Peripheral Interface). Il est compatible avec les microprocesseurs Intel ou autres types et possède 3 ports d'entrées/sorties parallèles (I/O) port A, B, C. Il a 24 pins d'entrées/sorties programmable par groupe de 12 et fonctionne en 3 modes.



Accès registres :

AI	A0	RD\	WR\	CS\	Opération
0	0	0	1	0	PORTA → DATA BUS
0	1	0	1	0	PORTB → DATA BUS
1	0	0	1	0	PORTC → DATA BUS
1	1	0	1	0	CR → DATA BUS
0	0	1	0	0	DATA BUS → PORTA
0	1	1	0	0	DATA BUS → PORTB
1	0	1	0	0	DATA BUS → PORTC
1	1	1	0	0	DATA BUS → CR
X	X	X	X	1	DATA BUS Haute impédance
X	X	1	1	0	DATA BUS Haute impédance

Structure de registre CR :



Fonctionnement du PPI 8255 :

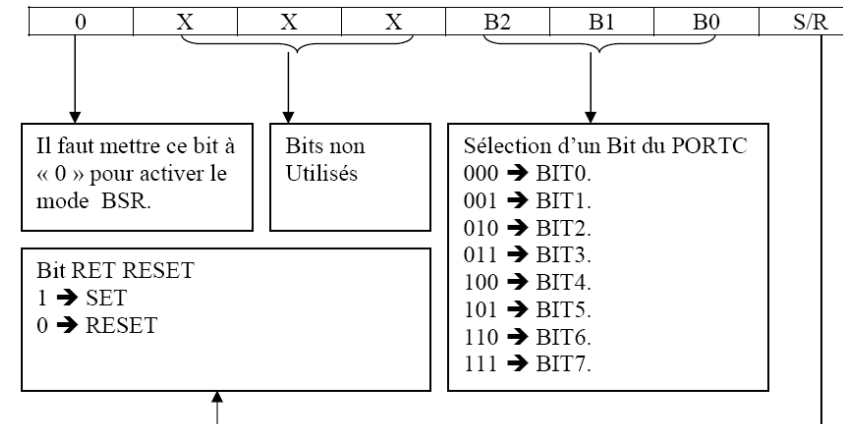
Le PPI 8255 fonctionne en 4 modes (mode0, mode1, mode2, mode BSR). Ces modes sont activés par la programmation de registre CR.

Mode BSR (BIT SET RESET):

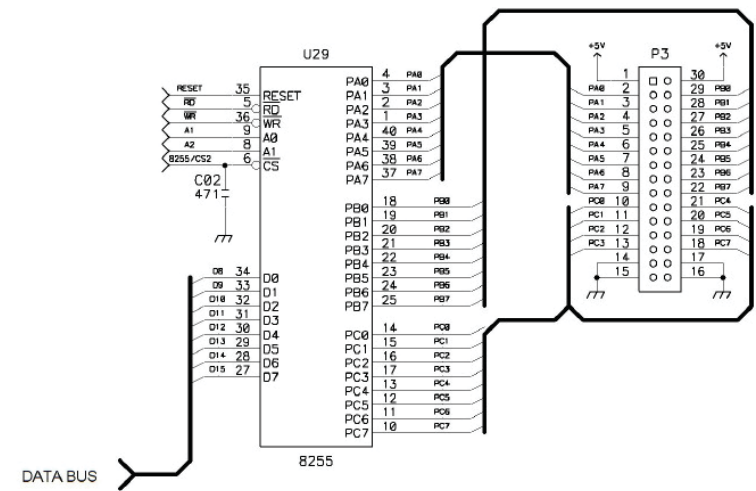
Ce mode utilise par le PORTC uniquement qui peuvent être mise à « 1 » ou « 0 » d'une ligne du PORTC individuellement. Dans ce mode la structure de registre CR est modifié mais cette modification ne dérange pas le mode de fonctionnement des ports (A et B).

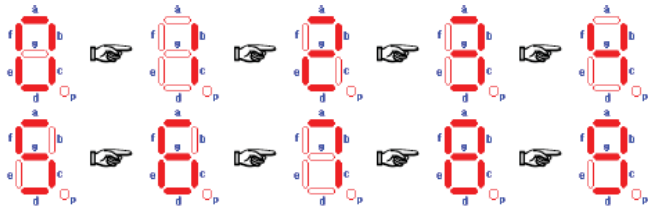
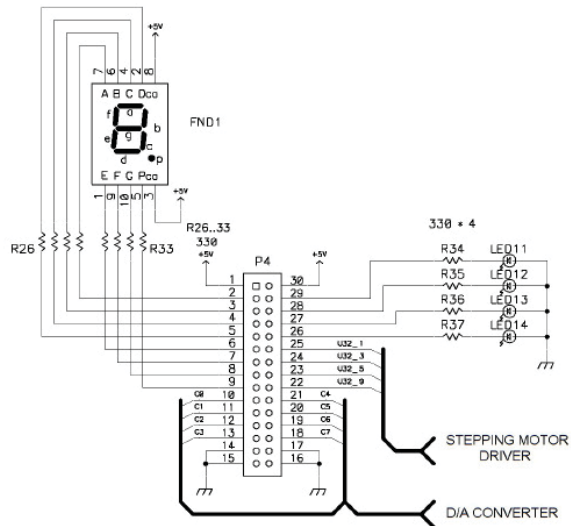
Remarque : n'oubliez pas de configurer le PORTC en sortie dans le registre CR.

Structure de registre CR en mode BSR :



6- Manipulation



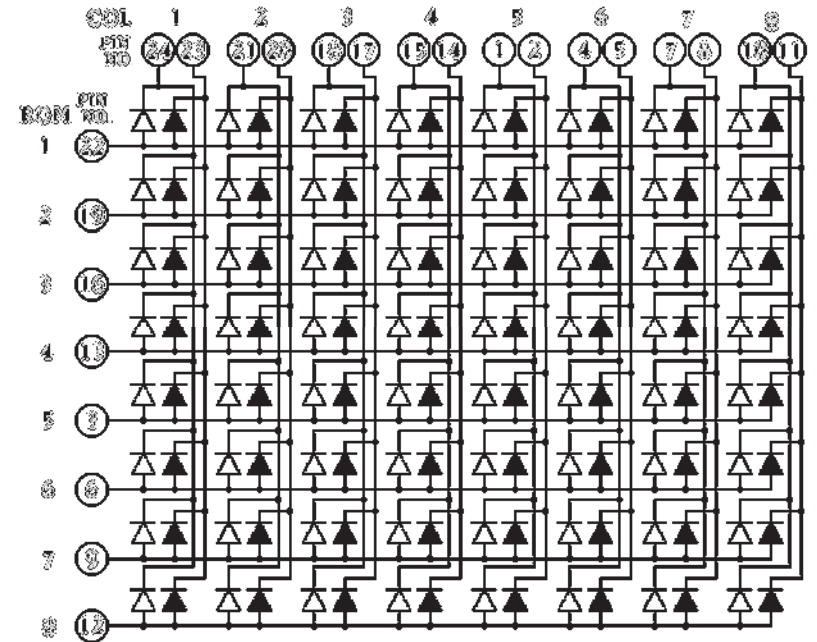


La figure (3) représente le circuit d'interface parallèle 8255 qui commande 4 diodes LED (LED11, LED12, LED13, LED14) à travers le port B et un afficheur 7 segment FND1 à travers le port A

- 1- Ecrire un programme en assembleur qui permet d'allumer les LED (LED11, LED12, LED13, LED14) en diagonale.
- 2- Ecrire un programme en assembleur qui permet d'afficher les nombres décimaux.
- 3- Modifier le programme précédent pour afficher les symboles hexadécimaux.

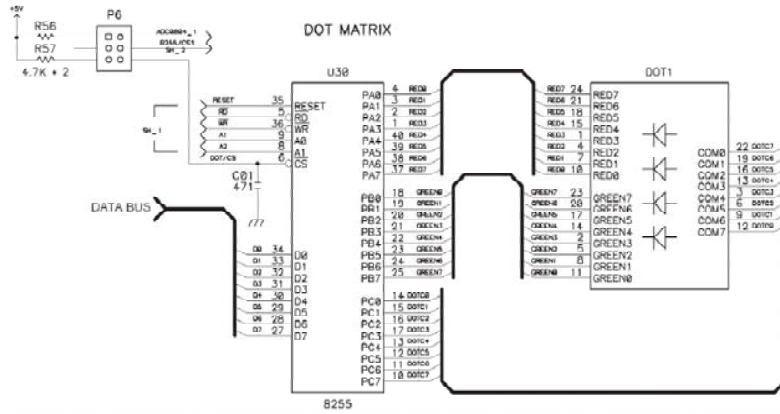
Matrice à diodes :

Le KMD D1288C est une matrice à diodes bi-couleurs rouge et vert constitués de 8 lignes et 8 colonnes.



La figure (5) représente le schéma commande de l'afficheur à matrice de diodes par un circuit 8255.

- 1- Ecrire un programme en assembleur qui permet d'allumer les diodes de couleurs vertes.
- 2- Ecrire un programme en assembleur qui permet d'allumer les diodes de couleurs rouges.
- 3- Ecrire un programme en assembleur qui permet d'écrire votre nom



7- Convertisseurs numériques analogiques

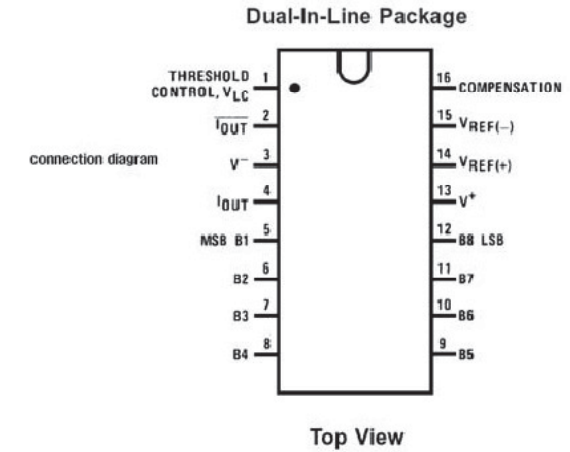
Objectif : l'objectif de cette manipulation est de se familiariser avec les convertisseurs numériques analogiques

Equipement :

Le matériel utilisé : PC, kit MDA 8086

Description générale :

Le circuit DAC0800 est un convertisseur numériques analogiques 8 bits monolithiques, de grande vitesse avec une sortie courant



La figure ci-dessous représente l'utilisation du convertisseur numériques analogiques DAC0800 dans le kit MDA 8086

8- Convertisseurs analogiques numériques

Objectif : l'objectif de cette manipulation est de se familiariser avec les convertisseurs analogiques numériques

Equipement :

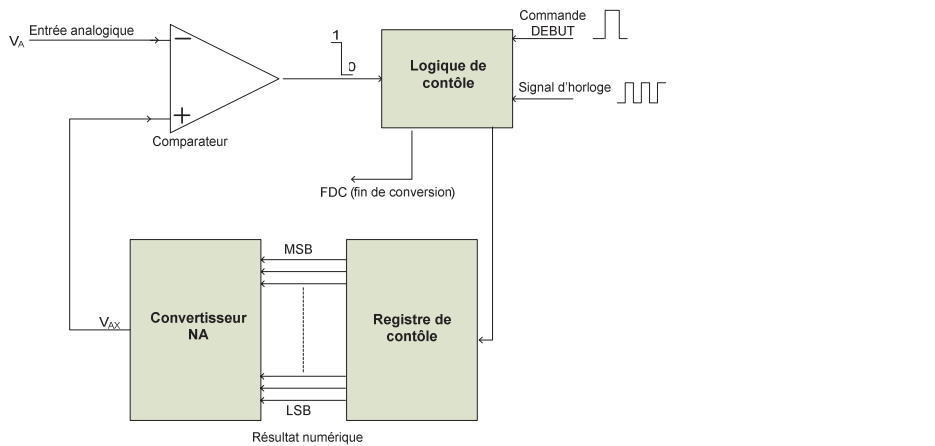
Le matériel utilisé : PC, kit MDA 8086

Description générale :

Le circuit ADC0800 est un convertisseur analogique numériques 8 bits monolithiques utilisant une technologie MOS à canal P. Il contient un comparateur à grande impédance d'entrée, 256 séries de résistances, un commutateur analogiques et un registre.

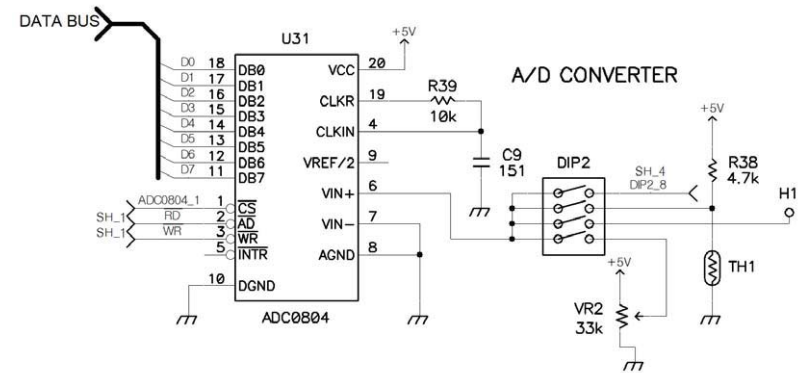
La technique utilisée pour ce convertisseur est par approximation successive. La logique de contrôle modifie le continu du registre bit par bit jusqu'à ce que la donnée qui s'y trouve soit l'équivalent numérique du signal analogique V_A .

Le schéma de principe est donné par la figure



La figure ci-dessous représente le montage de conversion analogique numérique utilisé dans le kit MDA 8086

ADC Interface:



Régler le **DIP2** comme montrée dans la figure



DIP2

Régler les **JUMPER P6** comme montrée



P6

Ecrire un programme qui permet d'afficher la valeur e la tension sur LCD si on ajuste V_A

