

Licence Mécatronique Exercices (suite)

I) Chenillar sur LEDs

On vous donne un sous programme destiné à afficher le contenu d'un registre 8 bits : '*' quand il y a un 1 et ' ' autrement :

```
void affiche(char port){
    char i;
    system("clear");
    for (i=7;i>=0;i--){
        if ((port & (1<<i))==(1<<i))
            putchar('*');
        else
            putchar(' ');
    }
    printf("\n");
}
```

Ce sous-programme peut être utilisé pour simuler des leds allumées et éteintes.

1°) Essayer le programme principal :

```
#include <stdio.h>
#include <stdlib.h>
void affiche(char port);
main(){
    char port;
    char i;
    int j;
    while(1) {
        for (i=0;i<8;i++){
            port = 1 << i;
            affiche(port);
            // attente a adapter a votre processeur
            for (j=0;j<20000000;j++);
        }
    }
    return 0;
}
```

qui simule un chenillard simple. (Vous pouvez être amenés à changer la valeur 20000000 de la boucle d'attente suivant la vitesse de votre processeur).

2°) Modifier le programme pour qu'il fasse un chenillard double avec croisement.

3°) Modifier ce programme pour qu'il fasse un chenillard à entassement.

4°) Modifier le sous-programme void affiche(char port) pour améliorer l'affichage des leds simulées : par exemple avec des cases en utilisant les caractères "|" et "_". Cette question permettra de comprendre ce sous-programme.

II) Afficheur sept segments

Un afficheur sept segments est géré ("simulé") par le sous programme suivant :

```
void affiche7segs(char port){
    system("clear");
    if ((port & 1) == 1)
        printf(" **** \n");
    else
        printf("      \n");
    if ((port & 0x22)==0x22){
        printf("*      *\n");
    }
}
```

```

    printf("*      *\n");
    printf("*      *\n");
}
if ((port & 0x22)==0x02){
    printf("      *\n");
    printf("      *\n");
    printf("      *\n");
}
if ((port & 0x22)==0x20){
    printf("*      \n");
    printf("*      \n");
    printf("*      \n");
}
if ((port & 0x22)==0x00){
    printf("      \n");
    printf("      \n");
    printf("      \n");
}
if ((port & 0x40)==0x40)
    printf(" **** \n");
else
    printf("\n");
if ((port & 0x14)==0x14){
    printf("*      *\n");
    printf("*      *\n");
    printf("*      *\n");
}
if ((port & 0x14)==0x04){
    printf("      *\n");
    printf("      *\n");
    printf("      *\n");
}
if ((port & 0x14)==0x10){
    printf("*      \n");
    printf("*      \n");
    printf("*      \n");
}
if ((port & 0x14)==0x00){
    printf("      \n");
    printf("      \n");
    printf("      \n");
}
if ((port & 0x08)==0x08)
    printf(" **** \n");
else
    printf("\n");
}

```

Un essai est réalisé avec le programme principal :

```

#include <stdio.h>
#include <stdlib.h>
void affiche7segs(char port);
main(){
    char bonjour[7]={0x7C,0x3F,0x54,0xE,0x3F,0x1C,0x50};
    char i;
    int j;
    while(1) {
        for (i=0;i<7;i++){
            affiche7segs(bonjour[i]);
            // attente a adapter a votre processeur
            for (j=0;j<200000000;j++);
        }
    }
}

```

```
    return 0;  
}
```

1°) Modifier le programme pour qu'il affiche les chiffres de 0 à 9.

2°) Modifier le programme pour qu'il affiche les chiffres de 0 à F.

3°) Modifier le programme pour qu'il permette l'affichage sur deux afficheurs commandés par un multiplexeur.