

## TP n° 9 : UTILISATION DE L'USART (UNIVERSAL SYNCHRONOUS/ASYNCHRONOUS RECEIVER/TRANSMITTER)

L'USART n° 0 est relié au PC et sert au dialogue d'Insight avec la carte.

L'USART n° 1 est également relié au PC mais sur le port ttyS1.

Pour échanger avec ce port ttyS1, il faut ouvrir un terminal.

Pour se faire, dans un shell, taper

```
gtkterm --port=/dev/ttyS1 --speed=57600 &
```

Dans la fenêtre de gtkterm, sélectionner dans le menu *Configuration* l'item *Port* et vérifier que les paramètres sont bien les suivants (valeurs correspondant à l'initialisation faite dans les routines ci-dessous) :

- Port : /dev/ttyS1
- Speed : 57 600
- Parity : none
- Bits : 8
- Stopbits: 1
- Flow Control : none
- End of line delay : 10

Tester la configuration avec l'exemple 1 vu en cours.

### Exercice 1

Écrire un sous-programme qui affiche une chaîne de caractères. Ce sous-programme recevra en entrée (par la pile) l'adresse de la chaîne à afficher.

Utiliser ce sous-programme pour afficher une chaîne de caractères quelconque, puis pour effacer l'écran (console gtkterm) en suivant les codes de contrôle donnés en annexe.

### Exercice 2: alphabet (Evaluation TP 2013-2014)

En utilisant l'USART écrire un programme qui affiche des lettres sur le terminal selon le mode opératoire suivant. Au départ la lettre courante est la lettre 'A'

- si l'utilisateur appuie sur le bouton SW1 la lettre courante est envoyée sur le terminal.
- si l'utilisateur appuie sur le bouton SW2 la lettre courante est incrémentée (si la lettre courante est 'A' la lettre incrémentée sera 'B') puis envoyée sur le terminal. Lorsque l'on atteint la lettre 'Z' on reprend au début ('A')
- si l'utilisateur appuie sur le bouton SW3 la lettre courante est décrémentée (si la lettre courante est 'C' la lettre décrémentée sera 'B') puis envoyée sur le terminal. Lorsque l'on atteint la lettre 'A' on reprend à la fin ('Z')
- si l'utilisateur appuie sur le bouton SW4 on change de casse (si on était en majuscules, on passe en minuscules et vice versa) et la lettre courante est envoyée sur le terminal. Si la lettre courante était 'B', la lettre courante devient 'b' et est envoyée sur le terminal.

## Exercice 3

3. Écrire un sous-programme qui à partir d'un nombre entier <100 renvoie les 2 codes ASCII des 2 chiffres, celui des dizaines et celui des unités.
4. Écrire un sous-programme qui positionne le curseur à un endroit précis (les coordonnées, numéro de ligne et un numéro de colonne, seront passés en paramètre). La chaîne de contrôle permettant de positionner le curseur à un endroit précis est :

ESC[<ligne>;<colonne>H

où <ligne> et <colonne> sont les numéros de ligne et de colonne indiqués en ASCII, chacun sur 2 octets (un octet pour le code ASCII du chiffre des dizaines, et un octet pour le code ASCII du chiffre des unités). La chaîne de contrôle comporte donc 8 octets.

3. Utilisez le sous-programme précédent pour écrire un programme qui fait défiler la lettre 'A' sur la première ligne de l'écran. Quand on arrive au bout de la ligne, on revient au début.
4. Modifier le programme de façon à ce que lorsqu'on appuie sur un bouton, on change de direction
5. Modifier le programme de façon à ce que le changement de direction s'effectue avec les flèches du clavier.

## Exercice 4: le jeu du plus ou moins

On souhaite implémenter le jeu suivant :

**Première étape** : un nombre mystère entre 0 et 9 est choisi par le programme (ce nombre est écrit en dur dans le programme) L'utilisateur doit deviner le nombre mystère en faisant le moins d'essais possible.

Après chaque essai le programme affiche « plus » si le nombre mystère est supérieur au nombre essayé par l'utilisateur, le programme affiche « moins » si le nombre mystère est inférieur au nombre essayé par l'utilisateur et « gagné » si l'utilisateur devine le bon nombre.

**Deuxième étape** : le nombre mystère est compris entre 0 et 99

**Troisième étape** : le nombre mystère est une valeur entrée au clavier par l'utilisateur.

**Quatrième étape** : le nombre d'essais est affiché sur les LEDs en binaire au fur et à mesure du jeu.

**Cinquième étape** : Le temps est limité à 30s. Si au bout de ce temps, le nombre mystère n'est pas trouvé, le message « perdu » est affiché.

## Annexes : codes

Passage en mode graphique	<shift out> (0xe)
Effacement de l'écran	ESC[2J
Sortie du mode graphique	ESCn
Positionnement en haut à gauche (1,1)	ESC[H
Positionnement en ligne (ligne,colonne)	ESC[<ligne>;<colonne>H
Passage en mode vidéo inverse	ESC[7m
Effacement du curseur	ESC[?25l
Ré-affichage du curseur	ESC[?25h
Choix des couleurs	ESC[x1;x2m
Déplacement du curseur vers le haut	ESC[A
Déplacement du curseur vers le bas	ESC[B
Déplacement du curseur vers la gauche	ESC[D
Déplacement du curseur vers la droite	ESC[C

Code couleur x1	Code Couleur x2	Caract. graphiques	
30 à noir	41 à rouge	0x6c	┐
31 à rouge	42 à vert	0x6b	┑
32 à vert	43 à orange	0x6d	└
33 à orange	44 à bleu	0x6a	┘
34 à bleu	45 à magenta	0x74	┌
35 à magenta	46 à cyan	0x75	└
36 à cyan	47 à blanc	0x77	┐
38 à surbrillant	48 à surbrillant	0x76	┘
39 à blanc	49 à noir	0x6e	┌
		0x78	┐
		0x71	—