

BE1

LE MICROCONTROLEUR

But : Programmer en assembleur un microcontrôleur AT89c51 (40 broches) cadencé par un quartz de fréquence 12MHz.

Sujet : une lampe est commandée par un bouton poussoir.
Lorsque la lampe change d'état, les informations :

---LAMPE---	ou	**--- LAMPE---**
-- ALLUMEE ---		**-- ETEINTE---**

doivent s'inscrire sur l'écran LCD de 2 lignes de 16 caractères.

La lampe est reliée à **P1.0**, le bouton poussoir à **P1.2**.

Actions sur le poussoir :

Bouton relâché 0V 0 logique

Bouton enfoncé 5V 1 logique

La lampe doit s'éteindre automatiquement au bout de 2s.

Le message envoyé utilisera le code **ASCII**

A la mise sous tension le message : doit être émis.

---SYSTEME---
---PRET---

Utiliser les sous programmes fournis de pilotage de l'écran LCD et d'envoi d'un caractère.

Organisation du TP :

Trois programmes de difficultés croissantes seront écrits :

- P1 Allumage simple d'une lampe par le bouton poussoir
- P2 Ecriture d'un message sur l'afficheur LCD
- P3 Commande de la lampe et envoi des messages

A chaque étape il faudra :

- Rédiger le programme, le compiler.
- Simuler
- Programmer le µC
- Vérifier le bon fonctionnement réel.

Programme 1

La lampe reliée à **P1.0** est allumée et éteinte par le bouton poussoir relié à **P1.2**.

1. Rédiger le programme

- 1.1. Définir un nouveau projet.
- 1.2. Définir les variables « Lampe » et « Bouton »
- 1.3. A quelle adresse doit-on commencer le programme ? Pourquoi ?
- 1.4. Comment programmer l'attente de l'action sur le poussoir ?
- 1.5. Sauvegarder le fichier : lampexxx.a51
- 1.6. Ajouter ce fichier en tant que nouveau « nœud » du projet.

2. Simuler

- 2.1. Placer des sondes en **P1.0** et **P1.2** et observer les traces correspondantes
- 2.2. Observer les changements qui interviennent dans les registres principaux (Main Registers) lors de l'action sur le poussoir.

3. Programmer le µC

- 3.1. La programmation est réalisée en utilisant le fichier : lampexxx.hex.
Charger ce fichier dans le programmeur et noter la valeur du « checksum »
- 3.2. Qu'est-ce que le « checksum » ?

- 3.3. Placer le composant sur son support en vérifiant son orientation (attention le composant a 40 broches, le support en a 48)
- 3.4. Effacer, programmer et vérifier
4. vérification du bon fonctionnement
 - 4.1. Placer le μ C programmé sur la maquette, relier lampe et bouton et valider le programme.

Programme 2

Ecrire, sur 2 lignes de l'afficheur LCD, le message d'accueil:

**---SYSTEME---
**---PRET---

Utiliser les sous programmes fournis.

Suivre les étapes 1 à 4 décrites dans le programme 1 en les adaptant au contexte.

1. Comment peut-on placer une chaîne de caractères en mémoire programme ?
2. Comment accéder à cette chaîne ?
3. Comment déterminer la fin d'une chaîne de caractères à envoyer ?

Programme 3

A la mise sous tension le message

est écrit sur l'afficheur LCD

**---SYSTEME---
**---PRET---

Le bouton poussoir enclenché allume la lampe et le message

**---LAMPE---
-- ALLUMEE ---

est affiché.

Après 2s la lampe s'éteint automatiquement et le message

**---LAMPE---
-- ETEINTE ---

est affiché pendant 2s.

Le système attend le prochain appui sur le poussoir en affichant le message d'accueil.

La temporisation « n fois t_0 » sera réalisée par :

Le Timer0 comptant 50ms, (appel au sous programme **tempo:**)

Le registre R7 de la banque 0 qui contient « n ».

1. Quelle valeur est chargée dans Timer0 ? Pourquoi ?
2. Quel est le mode de fonctionnement du Timer0 ?
3. Comment démarre-t-on le comptage ?
4. Comment sait-on que le comptage est terminé ?