|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **BTS S .N.** | Sainte Marie - Ecole Catholique LÃ©donienne | **« Systèmes Numériques »** | | 14/03/2019 | |
| **Projet porte de poulailler automatique** | | | |
| **Année**: 2018 / 2019 | **1ière Année** | | **Samuel LITZLER** |

**Objectif :**

Le but de ce **TP** est de réaliser la programmation d’un **afficheur 2x16 caractères** ainsi que d’un **clavier 16 touches**. Les deux seront reliés à un microcontrôleur **ATMEGA32**.

**Prérequis :**

Cours : « Le microcontrôleur ATMEGA32 »

Cours : « Programmation du microcontrôleur ATMEGA32 »

Cours : « Gestion des interruptions sur ATMEGA32 »

**Documents constituants le dossier :**

Page de garde (page 1)

TP de

**Remarque :**

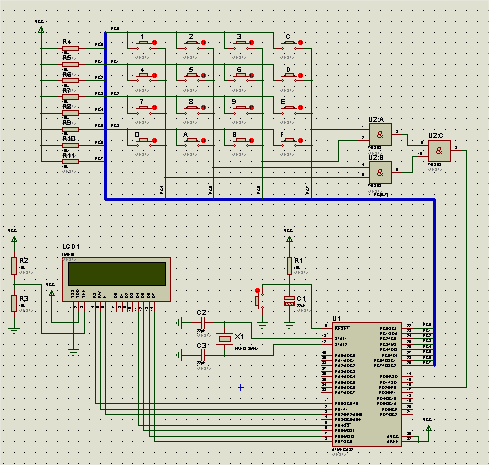
Nous utiliserons le logiciel **ISIS** pour faire les tests des programmes. Le programme sera fait avec le logiciel **CodeVisionAVR**.

**1 / Utilisation et mise en œuvre d’un afficheur 2 x 16 caractères**

Nous avons besoin d’une bibliothèque pour gérer des afficheurs LCD donc nous devons ajouter la bibliothèque « alcd.h ».



**2 / Utilisation et mise en œuvre d’un afficheur 2 x 16 caractères**

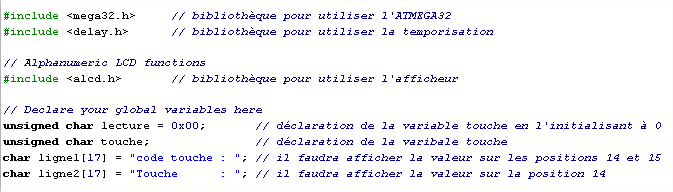


**3 / Détection d’une touche**

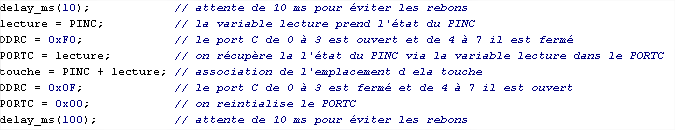
* La partie du port où se trouve les lignes est le port D.
* La partie du port où se trouve les colonnes est le port C.
* Le rôle des portes logiques est de signaler quand la touche est

enclenchée.

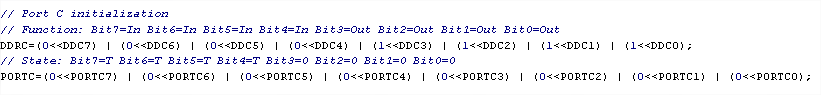
* Le **PORT C de 0 à 3** sont initialement instanciés à **1** et **le PORT C** de **4 à 7** sont eux initialement à **0.**  
  Lorsque qu’aucune pression n’est effectuée sur une des touche, **PC4 à PC7** passe à **1** et de **PC4 à PC7** à **0** c’est-à-dire que **PC4, PC5, PC6 et PC7** reste à **0** et l’information qu’une touche a été pressé est envoyé.
* Nous initialisons deux variables **lecture** et **touche** ainsi que deux tableau **ligne1** et **ligne2** pour l’afficheur **LCD**.



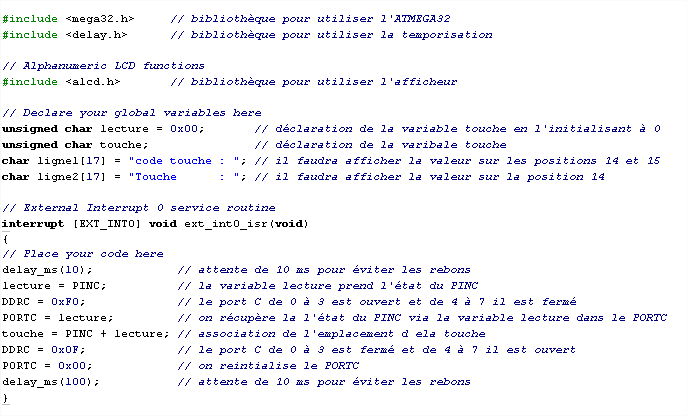
Le programme suivant permet de positionner le clavier dans une position d’attente d’appuis sur une touche.



Nous aurons besoin aussi donc d’initialiser les ports de **PC0 à PC3** sur **1** et les ports de **PC4 à PC7** sur **0.**



* Voici ensuite donc le programme d’interruption agissant sur INT0 qui permet de détecter une touche enfoncée, qui stocke son code dans une variable, et qui initialise le port afin de positionner à nouveau le clavier dans une position d’attente d’appuis sur une touche.



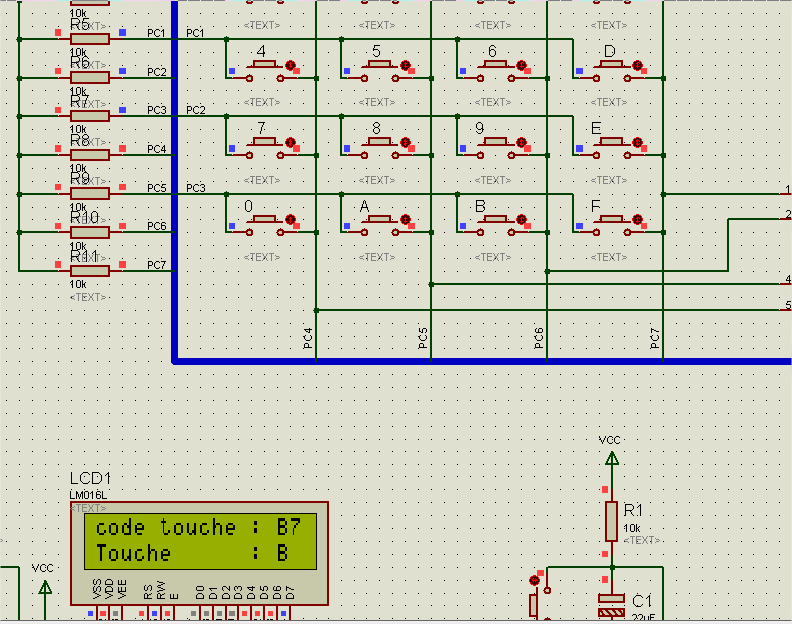
**4 / Utilisation du clavier et de l’afficheur**

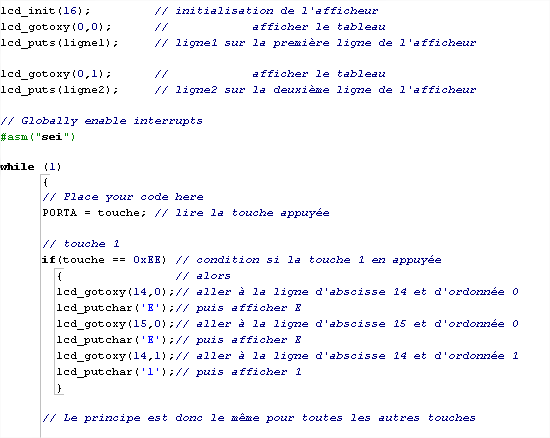
Nous devons réaliser un programme qui permet de réaliser les fonctions suivantes :

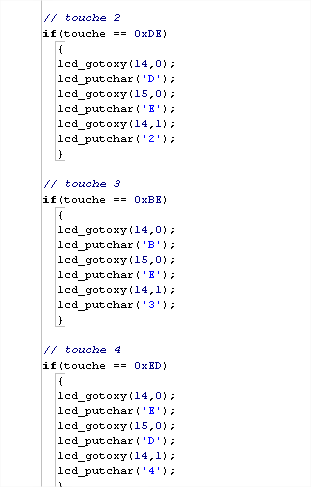
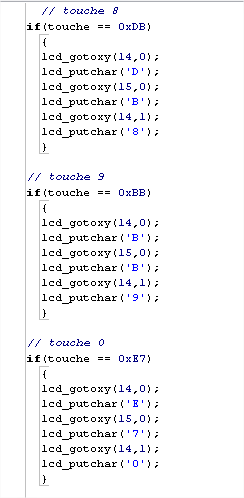
* + Initialisation du système (clavier + afficheur)
  + Détection de l’appui sur une touche du clavier
  + Affichage sur l’écran du code de la touche appuyée, ainsi que le numéro de la touche appuyée

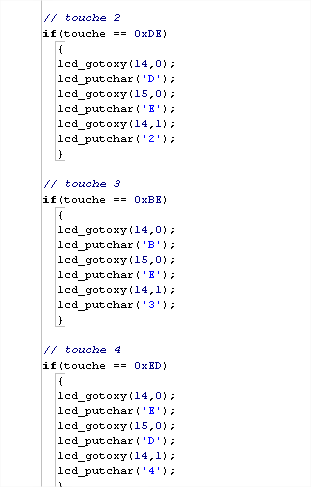
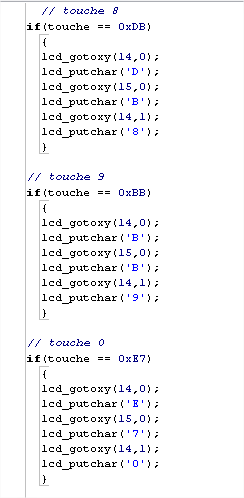
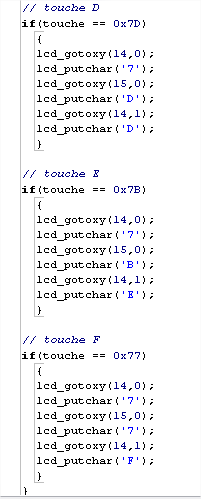
**Exemple :** Code touche : B7

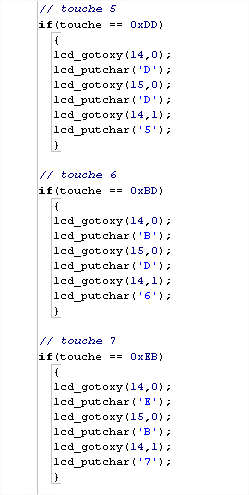
Touche : B

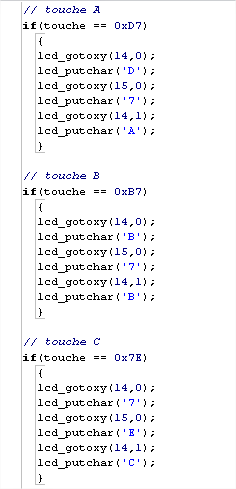












**5 / Compte Rendu**

Nous avons donc réalisé un programme qui utilise un **afficheur 2x16** **caractères** ainsi qu’un **clavier 16 touches**, les deux reliés à un **microcontrôleur ATMEGA32**.

L’afficheur nous donne les informations de l’interruption de la touche en fonction de ce que l’on a mis dans les conditions **des boucles if**.