TP : INITIATION A LA PROGRAMMATION DES MICROCONTROLEUR AVEC FLOWCODE

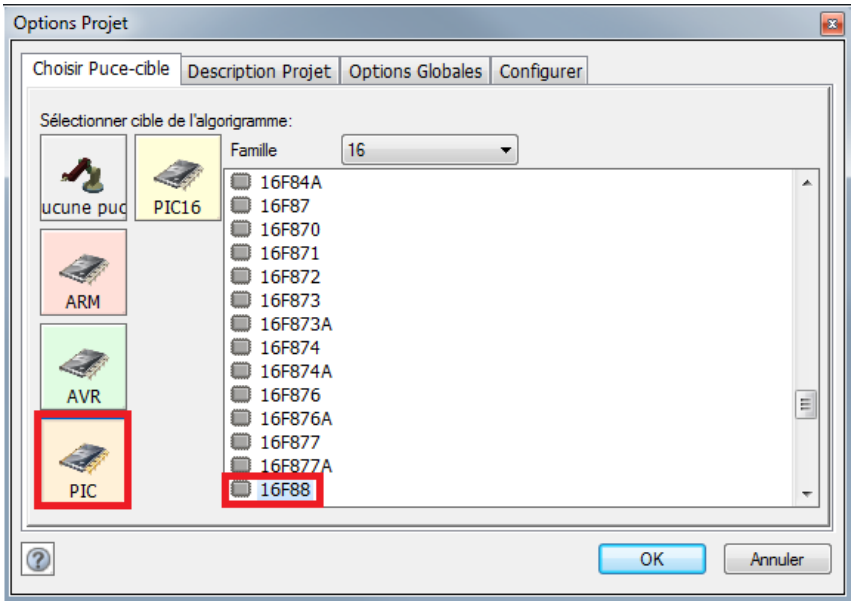
Nom des élèves du groupe :	Classe :	Date :

L'objectif de cette séance est de vous initier à la programmation des microcontrôleurs. L'utilisation du logiciel Flowcode va vous permettre de vous affranchir d'utiliser langage de programmation, trop complexe et/ou trop long à apprendre mais en conservant la structure réelle d'un programme.

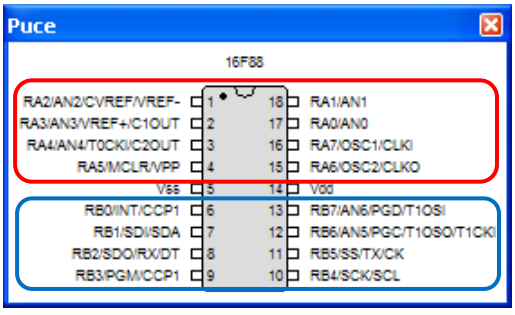
I. UTILISATION DU LOGICIEL FLOWCODE

Ouvrir le logiciel Flowcode V6 en cliquant sur l'icône

Choisir de créer un nouveau projet puis sélectionner le microcontrôleur cible PIC16F88 (par défaut) :

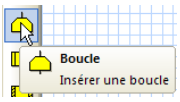


Le PIC16F88 est un microcontrôleur 18 broches qui possède 16 entrées/sorties réparties en 2 ports A et B. Les broches du **port A** sont nommées **RA0 à RA7** et les broches du **port B** sont nommées **RB0 à RB7**. *On notera que certaines de ces broches peuvent avoir plusieurs fonctions.*

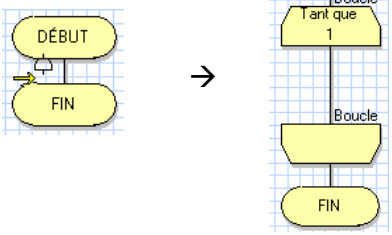


La création d'un programme sous Flowcode consiste à insérer des symboles algorithmiques et de les paramétrer en suivant les étapes suivantes :

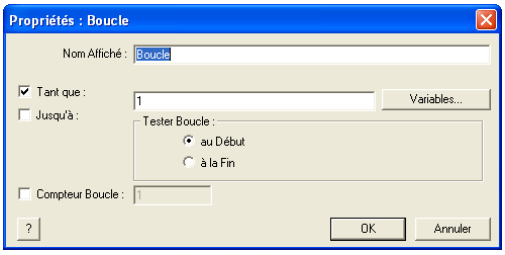
1. Sélectionnez le symbole à insérer.



2. Insérez le symbole dans l'algorigramme par cliquer/glisser.



3. Double cliquez sur le symbole pour le paramétrer.



En laissant les paramètres par défaut, on réalise une boucle infinie, c'est-à-dire que toutes les instructions qui se trouvent à l'intérieur se répéteront indéfiniment jusqu'à arrêt du programme.

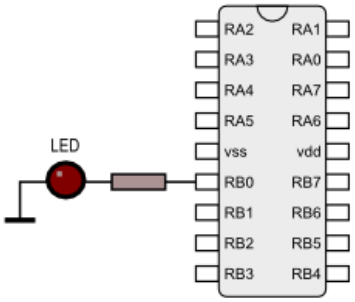
On rappelle qu'un microcontrôleur peut s'apparenter à un mini-ordinateur sur lequel on peut connecter des « périphériques » d'entrée qui vont acquérir des informations et des « périphériques » de sortie qui vont commander des actionneurs ou informer l'utilisateur.

	Entrée	**Sortie**
Ordinateur	Clavier Souris Webcam Joystick…	Ecran Imprimante Clé USB…
Microcontrôleur	Bouton Poussoir Mini clavier Potentiomètre Capteurs divers	Led Afficheur LCD Moteur…

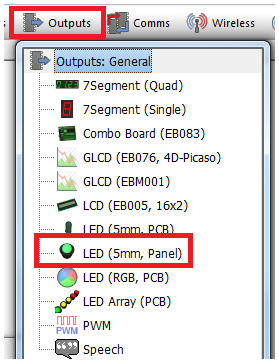
Si l'ordinateur utilise des connecteurs spécifiques (port USB, connecteur RJ45, VGA…), la connexion du microcontrôleur avec ses « périphérique » se fait par ses broches d'entrées/sorties.

1 - Création d'un premier programme simple : Allumer une Led

ENVIRONNEMENT : Une LED rouge est connectée sur la broche RB0 du microcontrôleur



Afin de pouvoir tester le programme, il faut au préalable insérer les composants à simuler (ici une LED rouge).



Dans la barre d'outils Objets, sélectionnez LED comme « périphérique » de sortie.

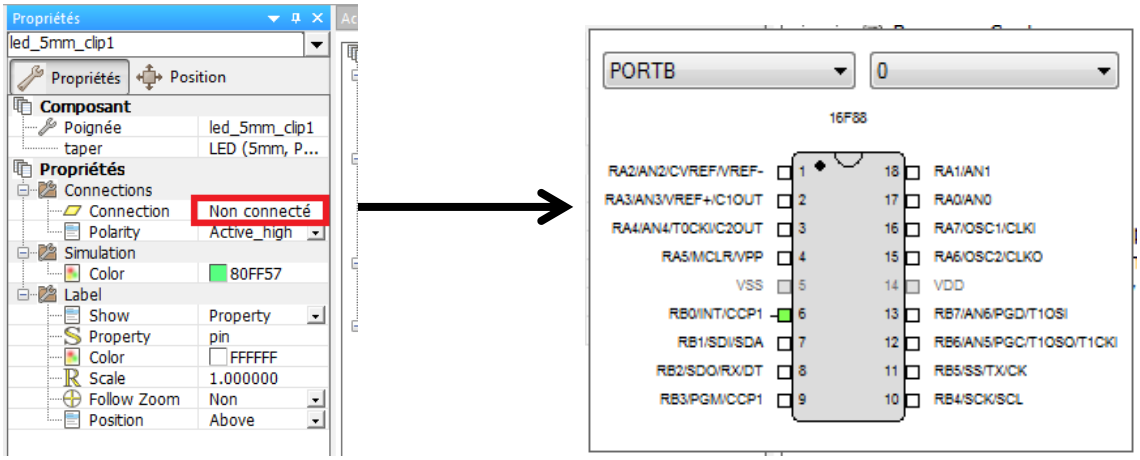
Une LED apparait sur le panneau !

Un clic droit sur la LED permet de la paramétrer :

- **Propriétés étendues** : Modification de l'apparence (forme, couleur, taille…)
- **Connexions** : Permet de connecter le composant au microcontrôleur.

Clic droit sur la LED puis Propriétés

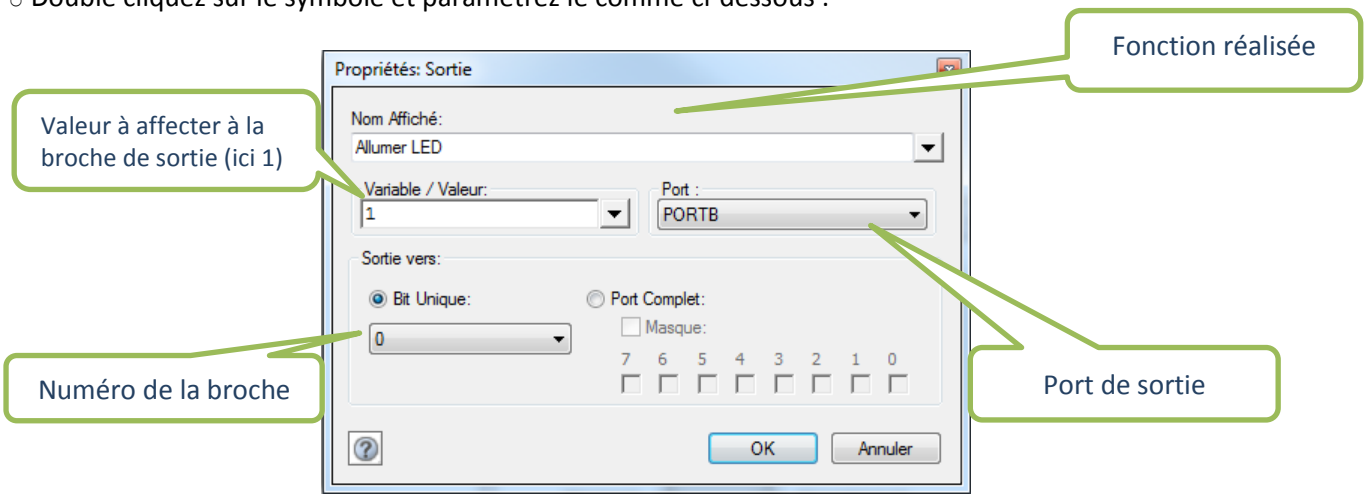
Connectez virtuellement la LED à la broche RB0 du microcontrôleur en paramétrant la fenêtre comme ci-dessous :



L'environnement de simulation est maintenant configuré, nous allons pouvoir passer à la programmation !

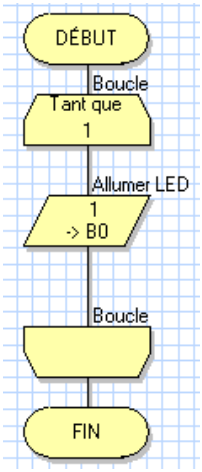
Une LED étant un « périphérique » de sortie, il va falloir insérer, dans la boucle créée précédemment, un symbole Sortie ou Output en anglais et le paramétrer correctement.

Insérez un symbole Output dans l'algorigramme.
Double cliquez sur le symbole et paramétrez le comme ci-dessous :



La configuration ci-dessus permet de « mettre à 1 » la broche RB0 (broche 0 du port B).

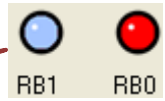
Programme final :



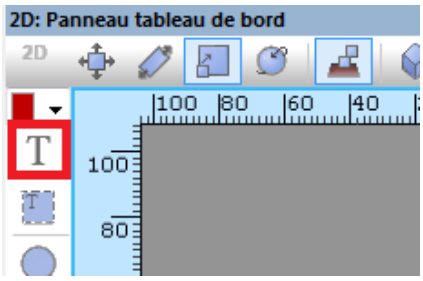
Il ne reste plus qu'à le simuler en cliquant sur l'icône ▶
La LED rouge doit s'allumer !

EXERCICE 1

Insérer une LED bleue puis connectez-la sur la broche RB1. Allumez-la !



L'insertion d'un texte sur le panneau s'effectue dans l'onglet TEXT du panneau



EXERCICE 2

Allumez alternativement la LED rouge et la LED bleues

Attention, le temps d'exécution de la boucle étant très rapide (plus de 10000 fois par seconde), vous ne verrez pas le clignotement. Il faut donc insérer un symbole Pause ou Delay *entre chaque symbole et le paramétrer (500ms par exemple).*

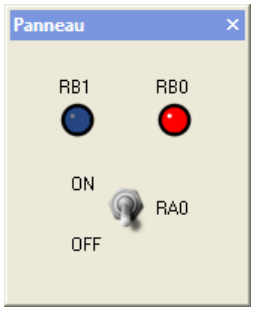
2 – Les « périphériques » d'entrée et les procédures de TEST

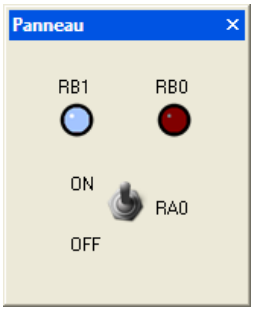
Jusqu'à présent, le pilotage des sorties s'est effectué par programmation mais une fois le programme lancé, l'utilisateur n'a plus son mot à dire. Nous allons voir dans ce chapitre comment faire interagir un utilisateur avec le programme.

De la même façon qu'un ordinateur acquiert des informations venant d'un clavier ou d'une souris, un microcontrôleur peut recevoir des informations venant de l'extérieur. La plus simple étant une information « tout ou rien », se traduisant par un état logique 0 ou 1 sur l'une de ses broches. Nous allons donc utiliser pour générer cette information logique un interrupteur.

ENVIRONNEMENT : 2 LEDs, une bleue et une rouge sont connectées sur les 2 broches du port B, RBO et RB1, et un interrupteur est connecté sur la broche RA0

On souhaite que la LED rouge soit allumée si l'interrupteur est en position OFF et que lorsque l'interrupteur est en position ON, ce soit la LED bleue qui soit allumée.



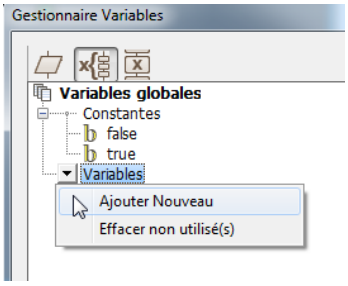


Le programme va donc se faire en 2 étapes :

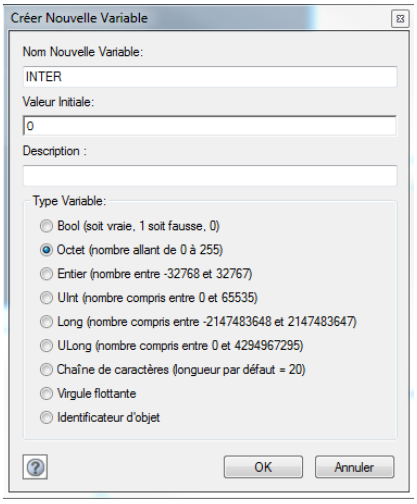
Etape 1 - Acquisition de la position de l'interrupteur par lecture de l'état logique présent sur l'entrée RA0.

Il va falloir créer un espace de stockage pour pouvoir « stocker » la valeur logique lue sur l'entrée RA0. Cet espace a un nom et s'appelle en langage informatique **une variable** qui est identifiée par un nom !

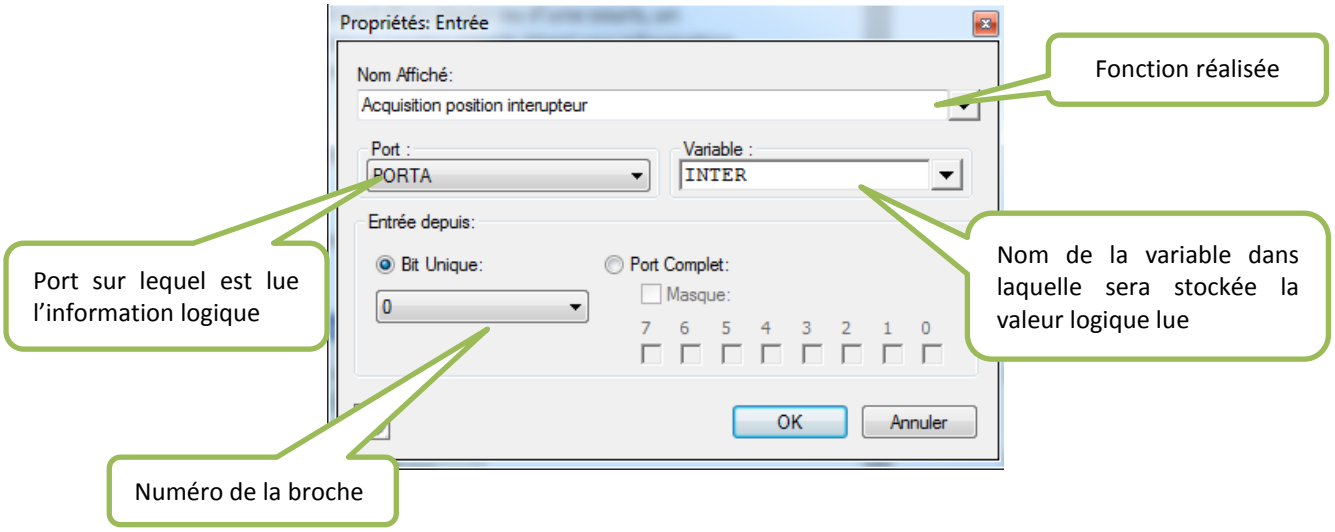
Pour la créer, aller dans Editer/Variables… puis cliquer sur Ajouter Variable…



Nommez-là INTER et laisser le type par défaut (Octet) et donnez-lui la valeur « 0 » (zéro) par défaut :



L'acquisition se fait en insérant un symbole Entrée (ou Input) et en le paramétrant comme ci-dessous :

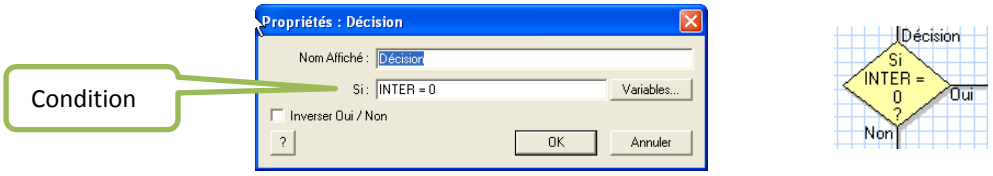


Etape 2 – Utilisation de cette variable pour déclencher une action en utilisant une procédure de TEST

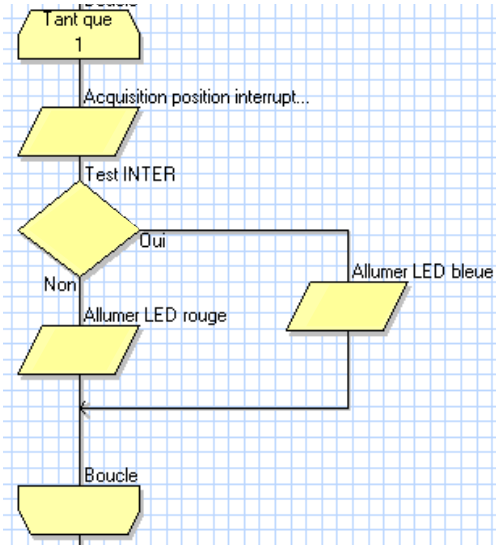
La procédure de test peut se traduire par l'algorithme suivant :

SI *condition vraie*		**SI** *INTER = 0*
ALORS *action1*	→	**ALORS** *allumer LED rouge*
SINON *action2*		**SINON** *allumer LED bleue*
FINSI		**FINSI**

Sous Flowcode, il faut utiliser le symbole Décision en le paramétrant comme ci-dessous :

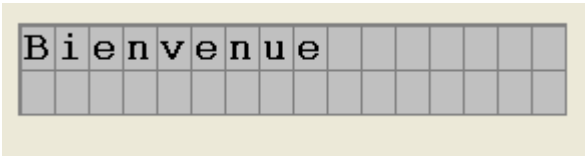


Vous pouvez maintenant créer le programme complet remplissant la fonction demandée. Il devra donc avoir la structure suivante :



3 – Un périphérique de sortie plus évolué : l'afficheur LCD

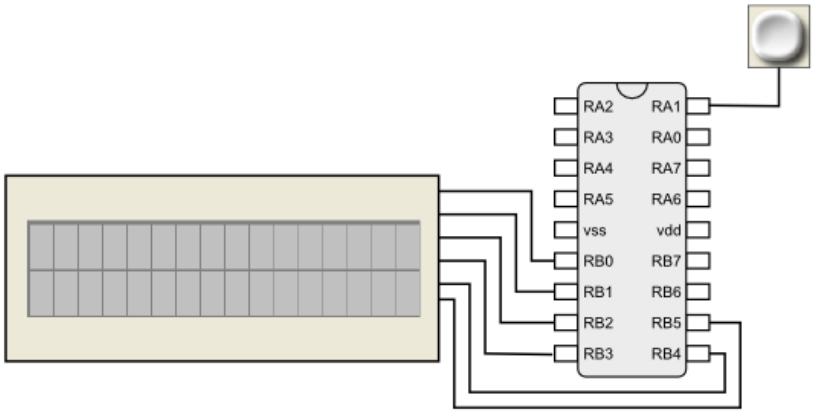
Cet afficheur LCD (Liquid Crystal Display ou afficheur à cristaux liquides) va permettre d'afficher du texte et des nombres ce qui va améliorer l'interaction entre l'utilisateur et le système microprogrammé. En effet, que serais un ordinateur sans son écran !



La programmation d'un afficheur LCD étant assez complexe dans la réalité, nous allons utiliser des Routines Composant spécifiques qui sont en fait des ensembles de petits morceaux de programme qui ont déjà été écrit et qui vont grandement nous faciliter le travail.

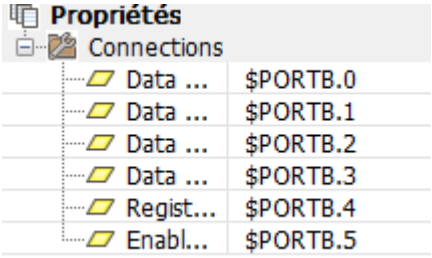
Nous allons créer un programme qui va compter le nombre de fois que l'utilisateur appuie sur un bouton poussoir

ENVIRONNEMENT : Un afficheur LCD connecté aux broches RB0 à RB5 et un bouton poussoir connecté à RA0

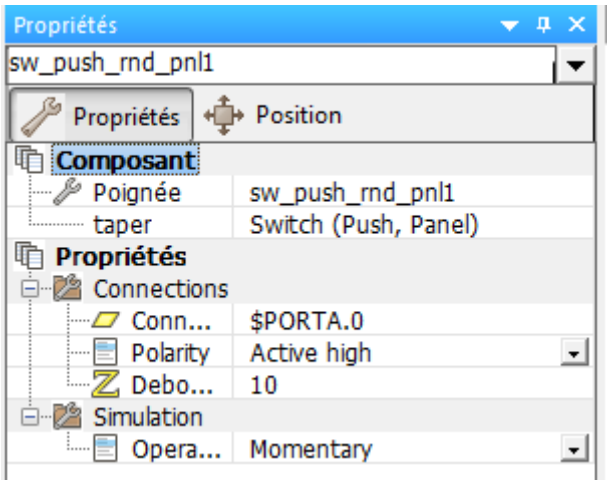


Créer un nouvel projet (microcontrôleur cible : PIC16F88)

Insérez un afficheur LCD LCD (EB005, 16x2) (voir dans les OUTPUTS) et connectez-le aux broches RB0 à RB5 (elles le sont normalement déjà par défaut).

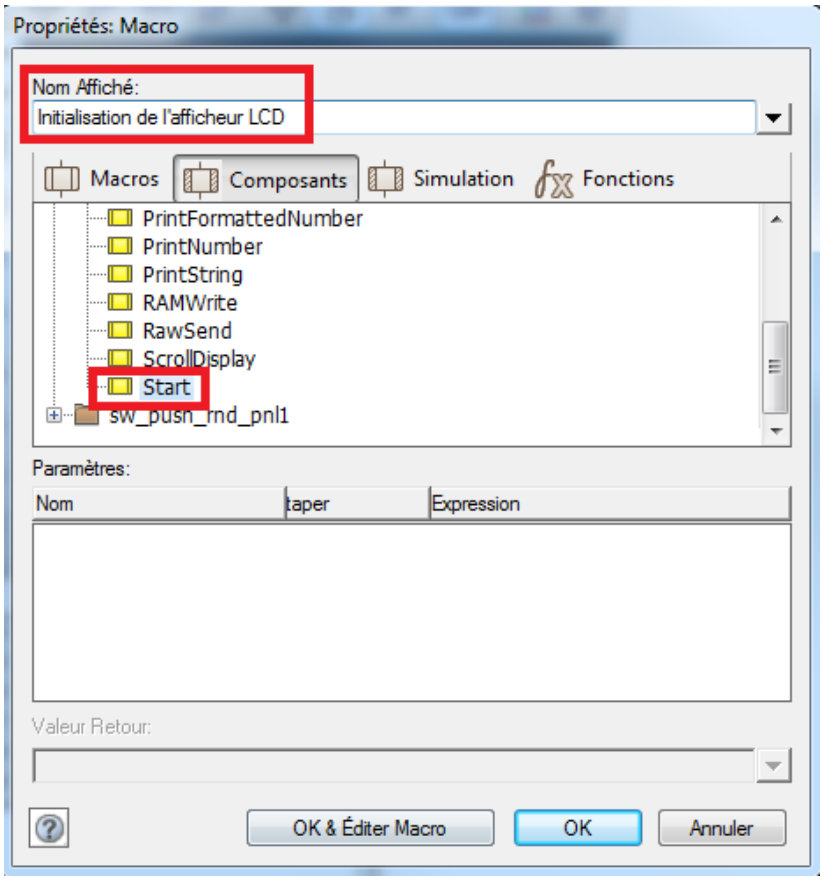


Insérez un Bouton poussoir (Input → Switch (push pannel) et configurer le comme ci-dessous puis connectez-le à RA0



L'utilisation de l'afficheur LCD impose qu'il soit initialisé au lancement du programme. Cette action ne devant être effectuée qu'une seule fois, elle ne sera pas insérée dans la boucle infinie « Tant que 1 ».

Insérez une Routine composant et paramétrez la comme ci-dessous :

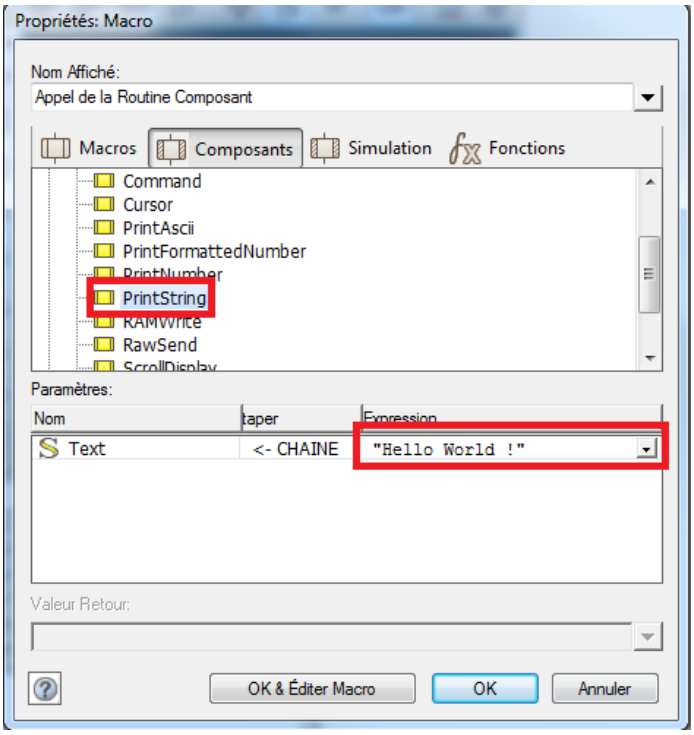


La Routine LCDDisplay comprend 10 Macro de commande de l'afficheur !

Une macro est en fait un petit morceau de programme appelé aussi sous-programme

3-1 Affichage d'un texte « Hello World »

Insérez à la suite de la macro « Start » une macro « PrintString » (Ecrire chaine de caractères) et paramétrez-là en écrivant le texte à afficher entre guillemets :



Testez le bon fonctionnement du programme !

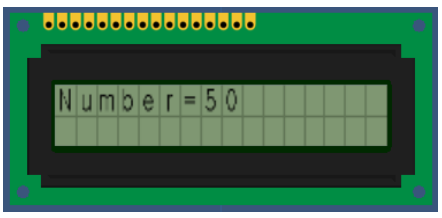
3-2 Affichage d'un nombre lié à une variable

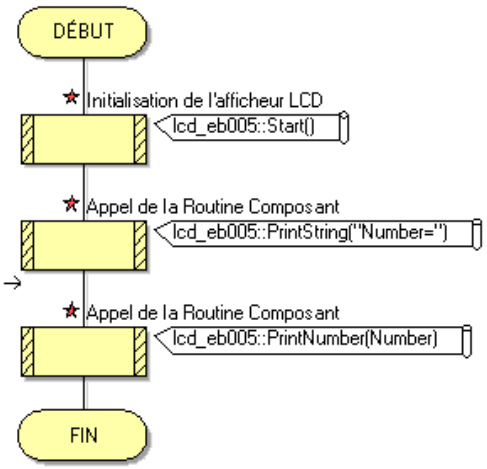
Créez une variable « **Number** » de type Octet.

Affectez-lui une valeur initiale (50 par exemple)

Insérez une macro « PrintNumber » (Ecrire Nombre) à la place de « PrintString » (*le nombre à afficher est la variable Number*)

On peut améliorer l'affichage en rajoutant un texte devant le nombre tel que « Number = » :

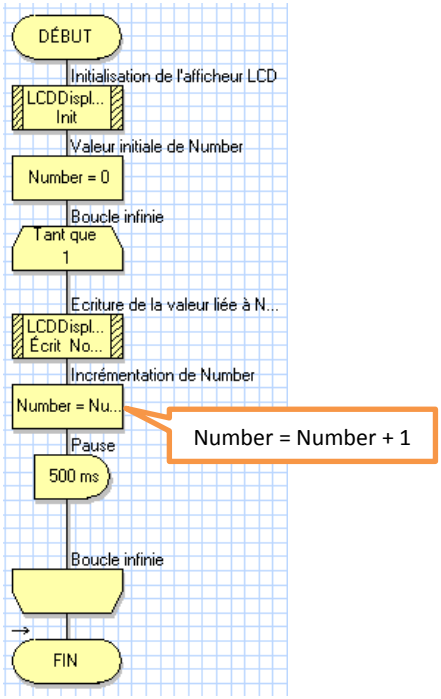




Testez le bon fonctionnement du programme !

5-3 Affichage d'un nombre variant au cours du programme

Nous allons maintenant programmer l'incrémentation de la variable Number et l'afficher en temps réel. Cela nécessite une mise à jour permanente de l'afficheur.



- **Reproduire l'algorigramme ci-contre et observez le résultat de la simulation !**

Explication : lors de l'écriture d'un caractère sur l'afficheur, le curseur se décale automatiquement vers la droite et après un certain temps passe à la 2e ligne.

Il faut donc à chaque nouvel affichage remettre le curseur en position initiale. Cela se fait notamment par la procédure d'effacement de l'afficheur.

- **Insérez une macro « Clear » (effacer) avant la macro PrintNumber**

La macro doit se placer avant afin que la durée d'affichage soit très supérieure à la durée d'effacement (persistance rétinienne)

5-4 Comptage du nombre d'appui sur le bouton poussoir

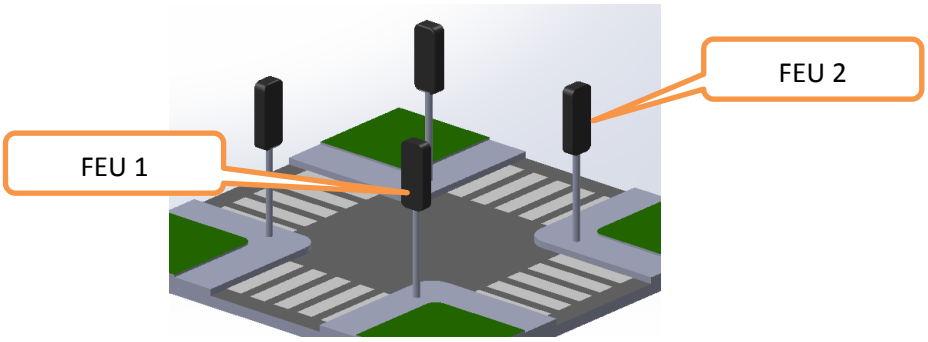
Vous avez maintenant toutes les cartes en mains pour réaliser ce petit programme. Voici néanmoins l'algorithme à respecter

Initialisation LCD
Initialisation de Number (valeur initiale = 0)
Répéter
  SI *appui sur bouton poussoir*
    ALORS *incrémenter Number*
    SINON *rien*
  FINSI
  EFFACER LCD
  AFFICHER Number
  PAUSE 500ms
Recommencer

EXERCICE 9
Modifiez le programme en rajoutant un Bouton poussoir permettant la remise à zéro de la valeur affichée.

Exercice d'application

Gestion de feux tricolores à un carrefour



Vous aller au cours de cet exercice effectuer la programmation de la gestion de feux tricolores à un carrefour selon les indications suivantes :

- Chaque feu reste au vert pendant 5 secondes.
- La durée du feu orange est de 1 seconde.
- Lorsqu'un feu passe au rouge, un délai de 1 seconde doit être respecté avant que l'autre ne passe au vert.

Les feux sont connectés comme suit :

FEUX 2			FEUX 1		
A5	A4	A3	A2	A1	A0

- Copier le fichier fichier Excel « carrefour.xls » qui se trouve dans le répertoire ressource du TP dans votre répertoire personnel et remplir le tableau en respectant les indications données ci-dessus.
- *Le feu 1 est au vert au début de la séquence et le feu 2 au rouge.*

Un nombre décimal apparait dans la colonne Port A ! Pour comprendre à quoi elle correspond, lire la page suivante.

Ouvrir le fichier Flowcode « Carrefour » et établir l'algorigramme de gestion des feux tricolore.