

Nom :

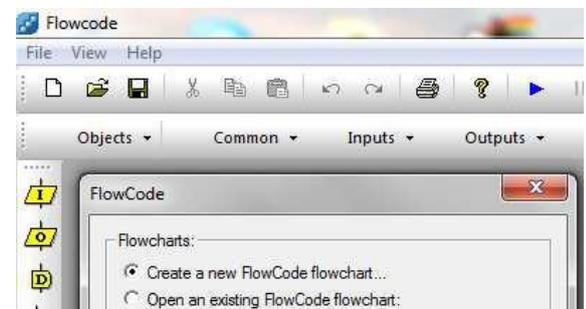
L'élaboration d'un programme à l'aide de l'outil R.A.D. Flowcode passe par plusieurs étapes qui sont pour l'essentiel :

- La création du projet qui consiste à choisir et configurer le μ contrôleur utilisé et les composants qui lui sont associés.
- L'édition du programme sous la forme d'un organigramme.
- La validation du programme par son exécution simulée, avec éventuellement une mise au point si le résultat n'est pas satisfaisant.
- La production du code machine destiné à être « gravé » dans la mémoire morte du circuit réel.

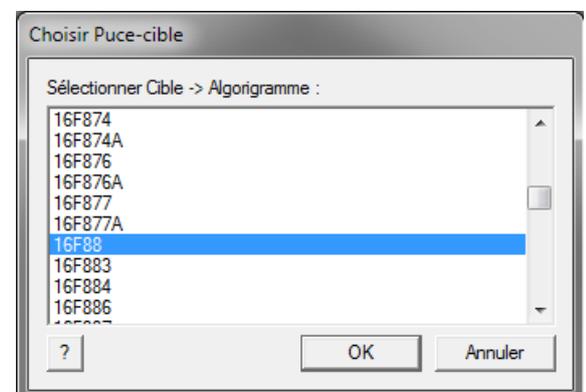
Des vidéos présentant la mise en œuvre des fonctionnalités de Flowcode sont disponibles [ici](#).

A - Création d'un projet Flowcode

1) Ouvrir Flowcode, une fenêtre apparaît dans la fenêtre principale, permettant d'ouvrir un fichier existant ou d'en créer un nouveau.

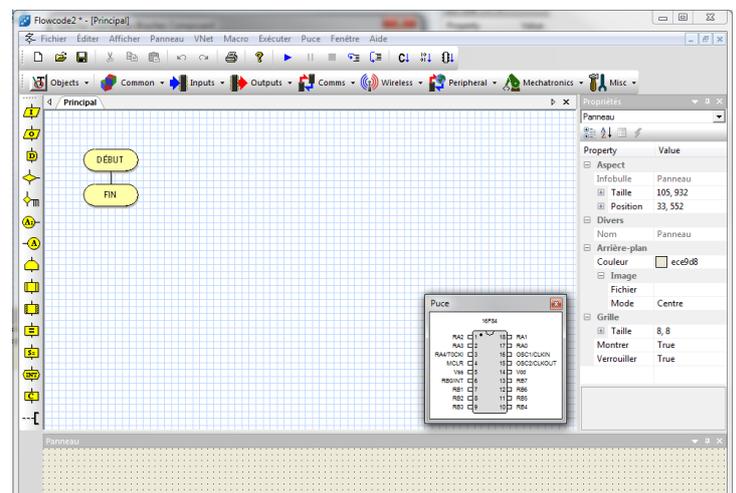


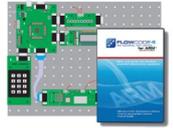
2) Choisir le microcontrôleur envisagé. On choisit le 16F88 qui est utilisé dans la pédale lumineuse



Un nouveau programme avec DEBUT et FIN, le brochage du 16F88 apparaît.

- A gauche, les symboles du logigramme
- A droite, la fenêtre des propriétés.
- En bas, le panneau destiné aux composants pilotés par le microcontrôleur.

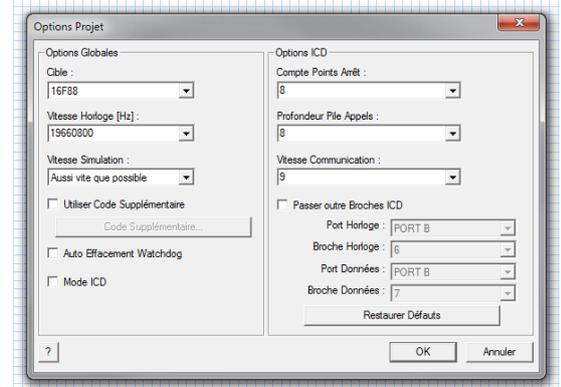




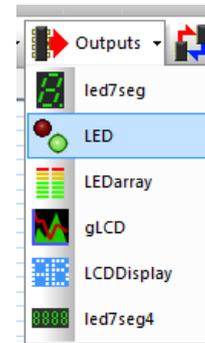
Nom :

2) Configurer les propriétés du microcontrôleur utilisé dans *afficher* → options projet :

- La fréquence de l'horloge du circuit (4MHz).
- la vitesse de simulation à 1s par étape qui laisse le temps d'observer le déroulement. Pour une simulation plus réaliste, on choisit *aussi vite que possible*.



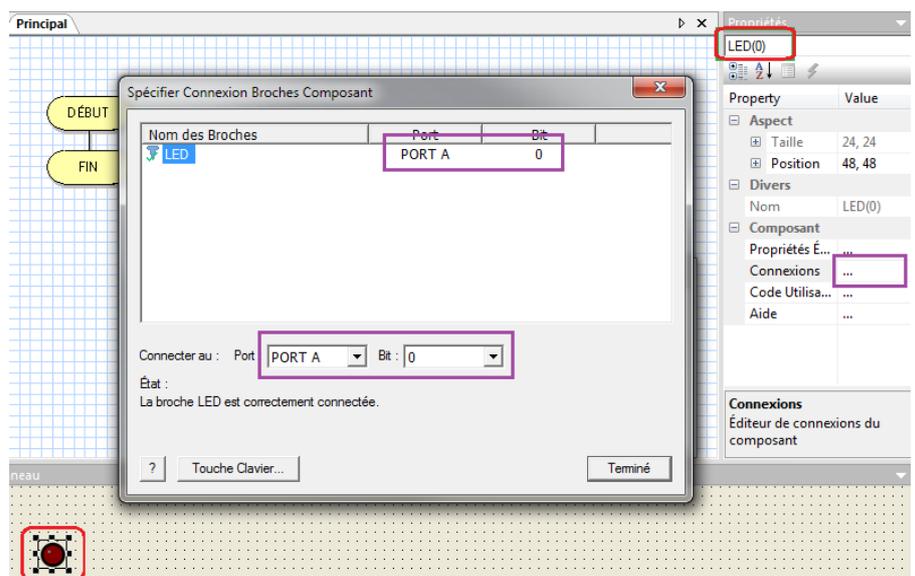
3) Mettre en place les composants périphériques connectés au microcontrôleur. Flowcode est capable d'en gérer une grande variété : E/S TOR, CAN, CNA, communication. Pour ce TP, on choisit de prendre une LED.

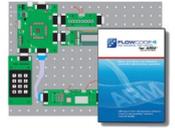


Définir les propriétés du composant répété *LED(0)* en cliquant sur :

- ⇒ *Propriétés étendues* pour la couleur, la forme, la polarité, etc.
- ⇒ *Connexions* pour relier la led à une broche du 16F88.

4) Dans notre cas, on choisit une led rouge, ronde, actif à l'état haut et reliée au port B2



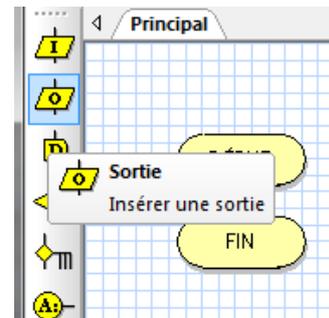


Nom :

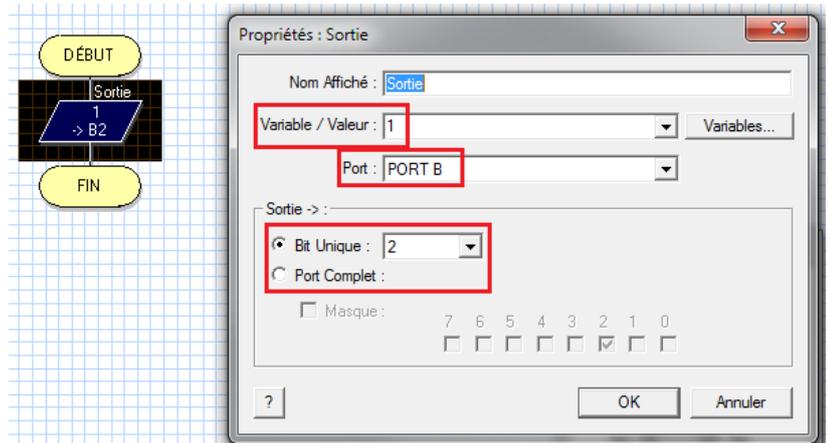
B - Edition de l'organigramme

Pour la prise en main de l'outil Flowcode, on va éditer un organigramme qui consiste simplement à allumer une led durant une seconde puis à l'éteindre.

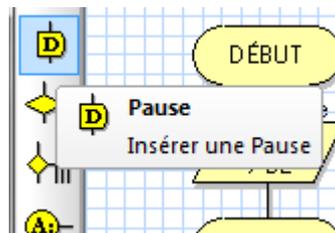
5) Insérer une sortie entre DEBUT et FIN (action O) par un cliquer – glisser.



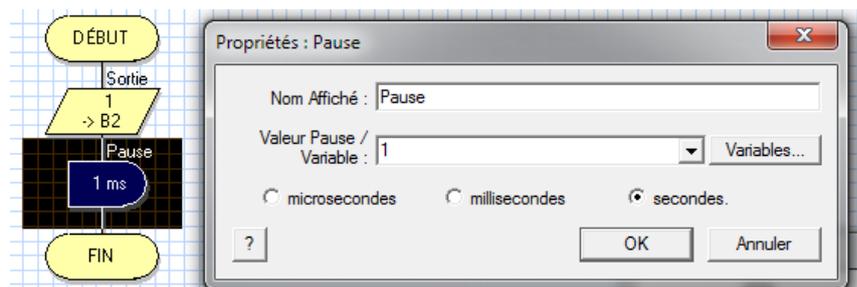
6) Cliquer sur l'action pour définir le port concerné (PB2). La valeur du port est fixé à 1 pour allumer la led (*niveau actif haut*)

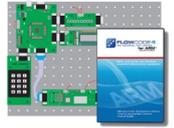


7) Fixer une pause en insérant une action D



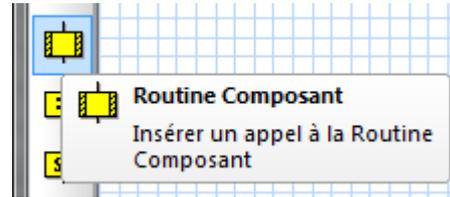
8) puis cliquer dessus pour fixer sa valeur à 1 seconde.





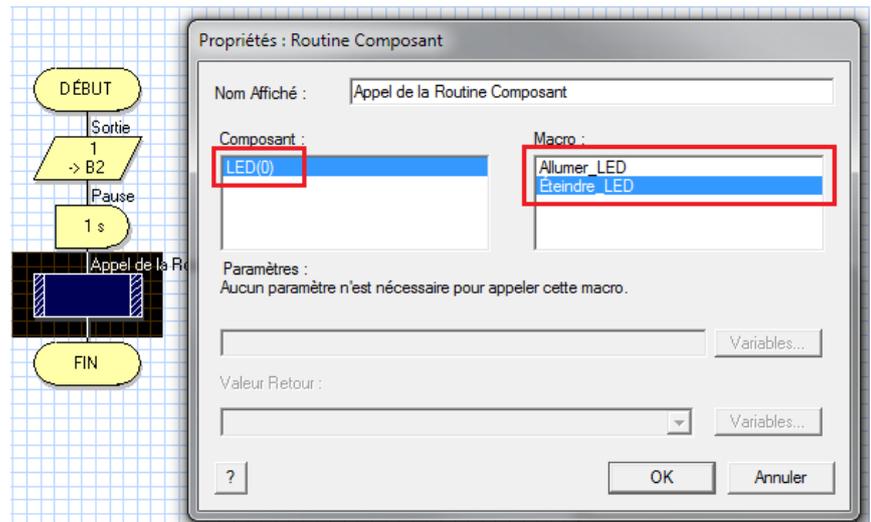
Nom :

Flowcode propose des fonctions qui facilitent la gestion des composants connectés au microcontrôleur.



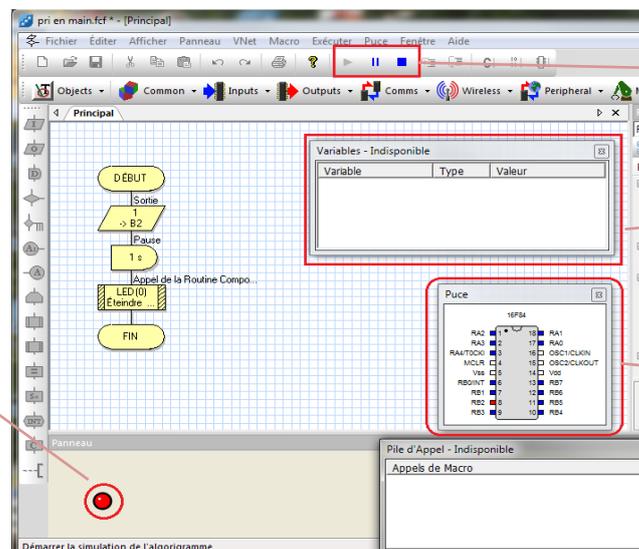
9) Insérer une « routine composant » puis cliquer dessus.

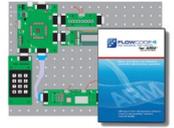
Choisir ensuite le composant concerné et la fonction désirée. Ici, on souhaite éteindre la led.



C - Exécution du programme

Une fois le programme édité, un clic souris sur le symbole ▶ démarre l'exécution simulée du programme.





Nom :

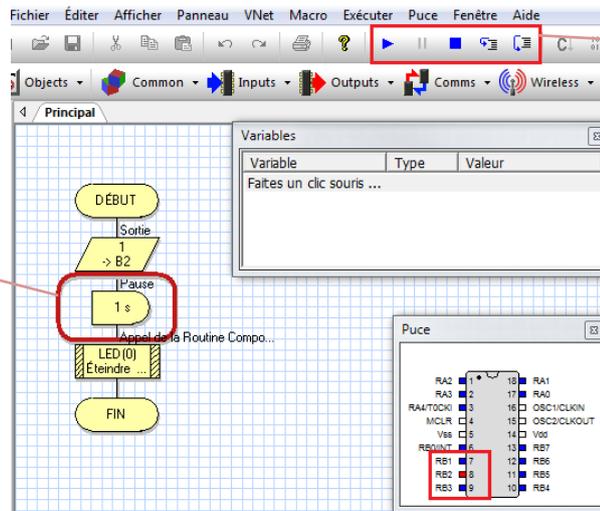
D - Mise au point du programme

Lorsque l'exécution ne donne le résultat attendu, on observe le déroulement du programme à la loupe pour détecter la source du problème. Les outils de développement logiciel comme Flowcode permettent :

- D'exécuter étape par étape le programme ;
- De suspendre l'exécution sur une étape particulière (point d'arrêt)
- D'observer le contenu des variables ;

11) Mettre en œuvre l'outil de mise au point en exécutant le programme étape par étape

Le programme a été exécuté jusqu'à cette étape. Le rectangle rouge indique la prochaine à être exécutée.



Exécution du programme :

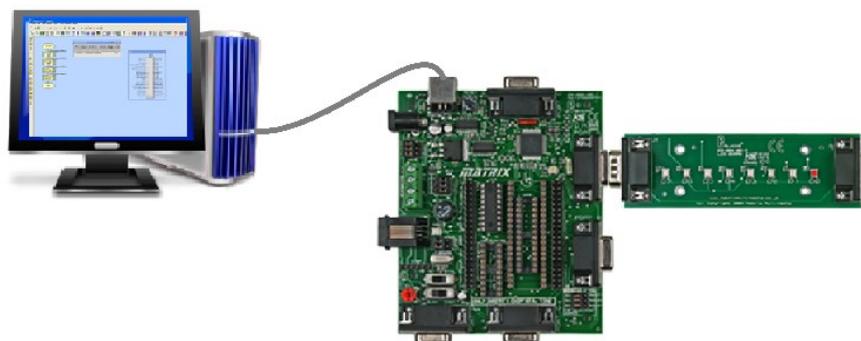
- Normal
- Pause
- Arrêt
- Pas à pas détaillé
- Pas à pas principal

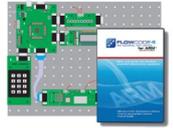
E - Programmation du microcontrôleur

Une fois le programme validé, il est temps de vérifier que son déroulement dans la puce même est conforme au cahier des charges. Flowcode permet de le traduire (compiler) :

- En langage C, permettant ainsi de l'optimiser avant de le traduire en langage machine,
- En langage machine (seul langage compréhensible par le μ contrôleur) puis stocké dans un fichier.
- En langage machine puis le stocke dans la mémoire programme du μ contrôleur (sous réserve d'être inséré dans une carte de programmation relié au PC).

12) Réaliser le montage ci-dessous (carte LED connectée sur port A) puis relier la carte EBLOCKS B006 à l'ordinateur à l'aide du câble USB fourni

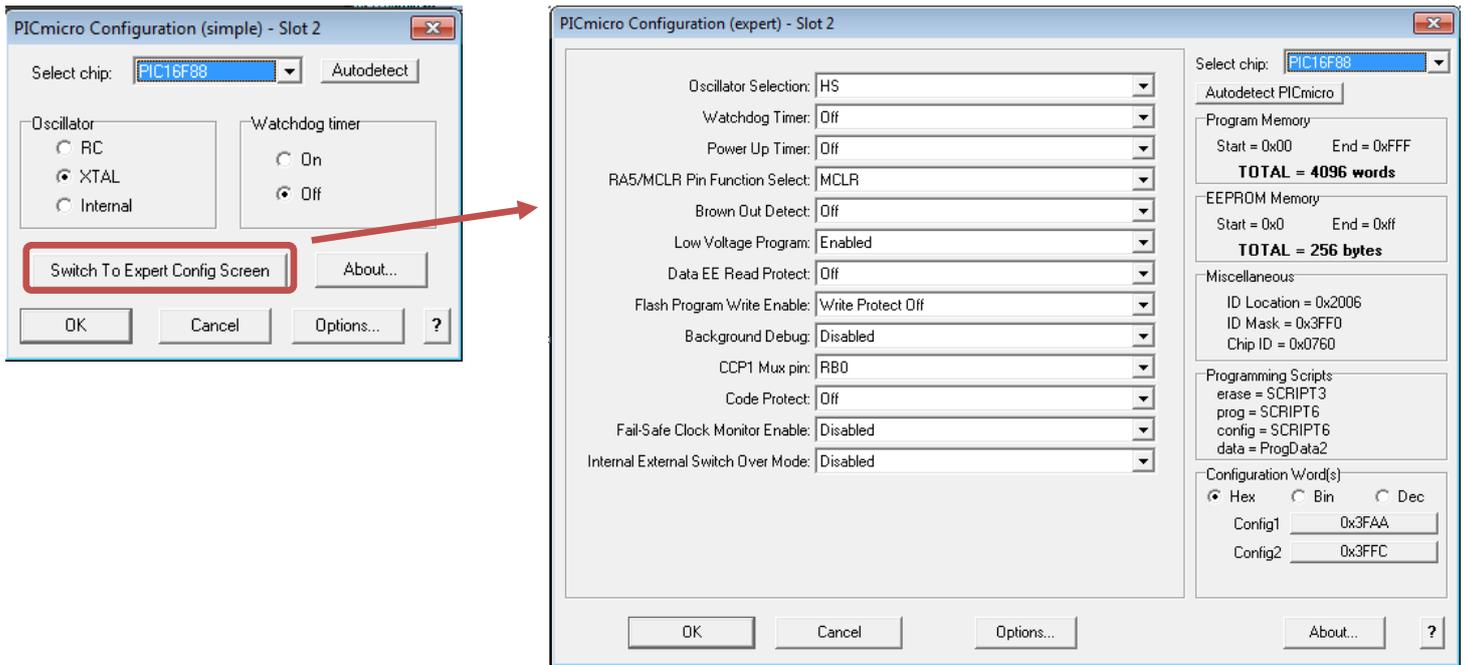




Nom :

13) Dans le menu « Puce », cliquer sur « Configurer ... ». Dans la boîte de dialogue « PICmicro configuration » cliquer sur « Autodetect » et le référence du PIC s'affiche.

14) Cliquer ensuite sur « Switch to Expert ... » puis configurez les éléments comme indiqué ci-dessous :



15) Compiler alors le projet afin d'obtenir un fichier .hex puis l'afficher dans le bloc notes. Décrire brièvement son contenu. Charger ensuite le programme dans le 16F88 en cliquant sur « Compiler -> Puce ... », puis valider son fonctionnement.

