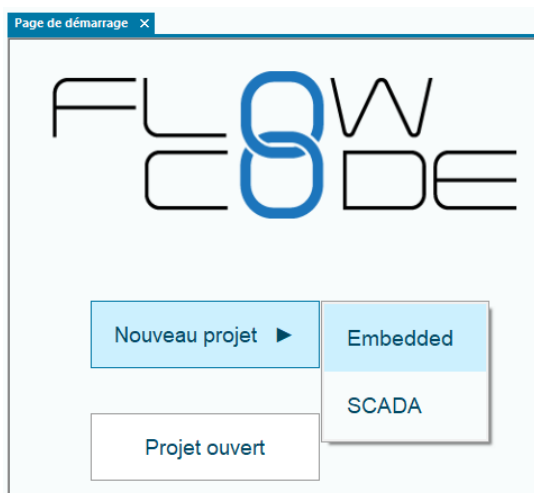


Table des matières

Création d'un projet.....	1
Créer votre espace de travail.....	2
Ajouter votre équipement.....	2
Inputs.....	2
Outputs.....	2
Affectez votre matériel au port du PIC.....	3
Puce.....	3
Réglages Entrées - Sorties.....	3
Pour un Switch :	3
Pour la LED.....	3
Présentation de l'interface :	4
Explorateur de projet :	4
Création de l'algorithme.....	5
Insérer un bloc " Entrée ".....	5
Insérer un bloc " Décision ".....	6
Insérer un bloc "Sortie ".....	6
Simulation :	7
Menu simulation.....	7
Lancer la simulation.....	7
Charger le programme dans le Micro-contrôleur.....	7

Création d'un projet

Cliquer sur nouveau projet. Embedded.



Choisir et sélectionner le micro-contrôleur (puce) dans la liste.
Choisir **16F88** ou (Formula Flowcode pour programmer le robot).

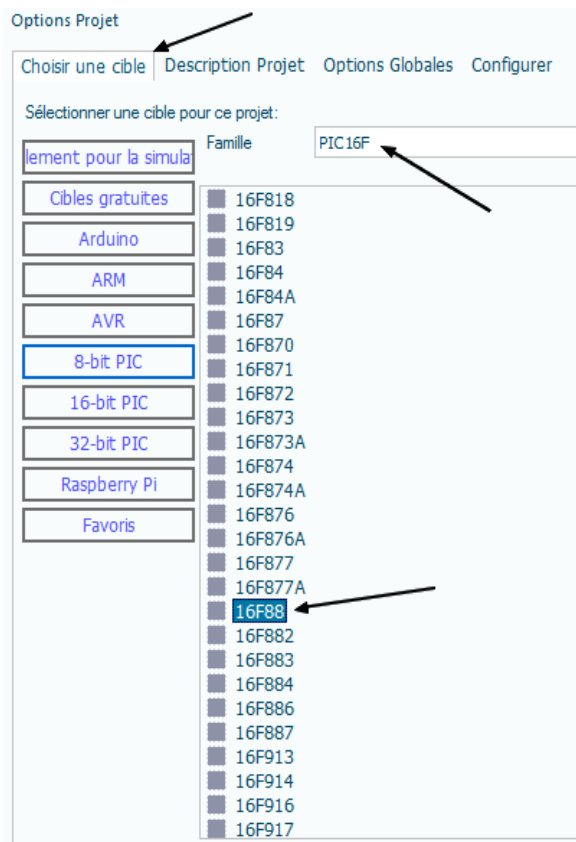
Attention :

Dans certains cas, il peut être difficile d'en changer ultérieurement !

Note : Si vous voulez changer vraiment de puce en cours de projet, vous pouvez essayer ceci : Menu « Générer » → « options projet » → « choisir puce cible »

Attention si le projet est avancé tout n'est pas possible (les broches du microcontrôleur peuvent être moins nombreuses).

Pour configurer Flowcode, voir le document : **Flowcode V8 – Configuration**



Créer votre espace de travail.

Affichage → 2D :Panneau de tableau de bord

C'est l'endroit où on va mettre notre équipement afin de simuler les entrées et sorties du micro-contrôleur :

*Nos **entrées** (Bouton, Potentiomètre...) et nos **sorties** (LED, moteur...).*

Ajouter votre équipement.

Inputs

Choisissez dans le menu déroulant les équipements d'**entrées** et placez-les dans le panneau **2D**.

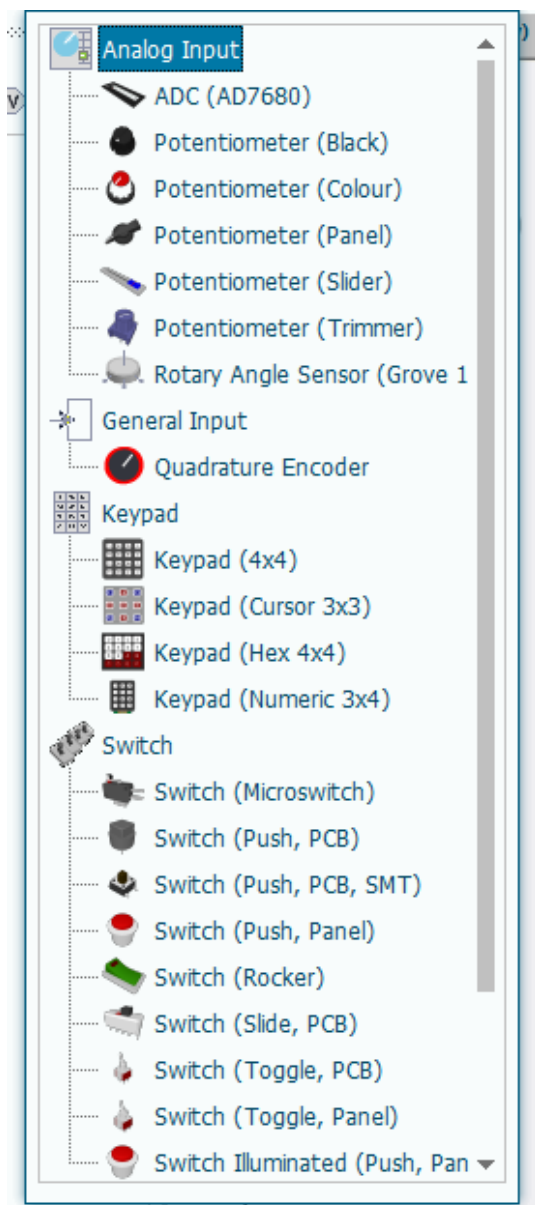
Potentiomètres, Interrupteurs et boutons (Switch)

Outputs

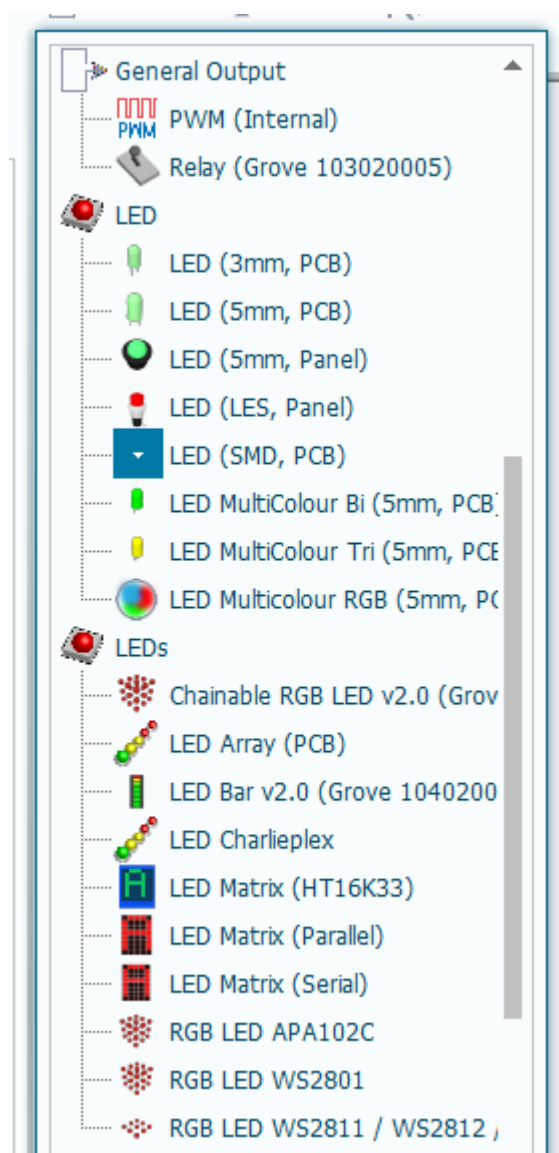
Choisissez dans le menu déroulant les équipements de **Sorties** et placez-les dans le panneau **2D**.

Voyants, LED, PWM (Pour la commande d'un moteur continu).

Les entrées (Inputs)



Les sorties (Outputs)



Affectez votre matériel au port du PIC

Pour chaque matériel Ajouter sur le panneau 2D, il faut lui indiquer son branchement sur le PIC :

Clic droit sur le matériel → **Propriétés**

Sur l'exemple de droite, on donne au Switch rouge la connexion : **\$PORTA.0**

Puce

Pour afficher la puce :

Affichage → Puces.

Réglages Entrées - Sorties

Il est possible de faire des réglages :

Pour un Switch :

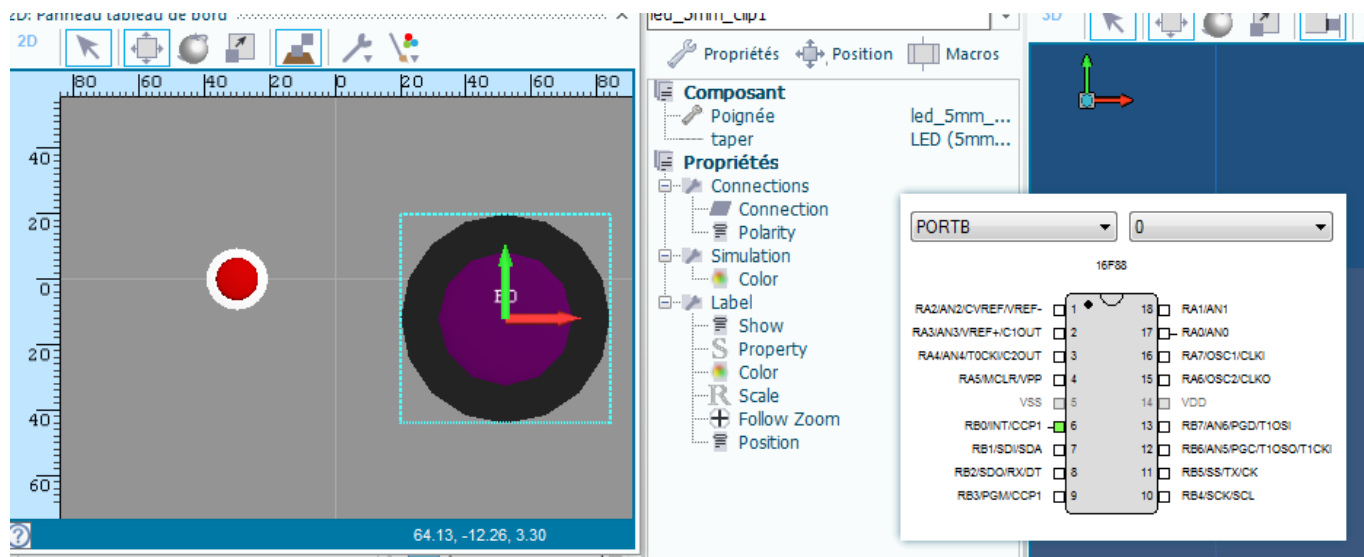
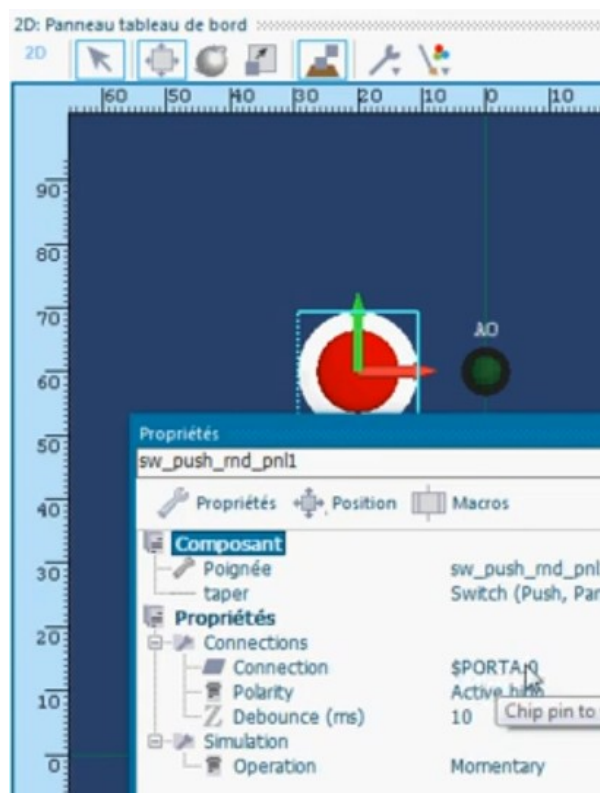
Momentary (Momentané, c'est à dire bouton poussoir)
ou fixed (Interrupteur)

Pour la LED

On peut changer la couleur, la taille...

Dans l'exemple ci-dessous :

Propriétés → **connexions** (Ici Port B entrée 0).



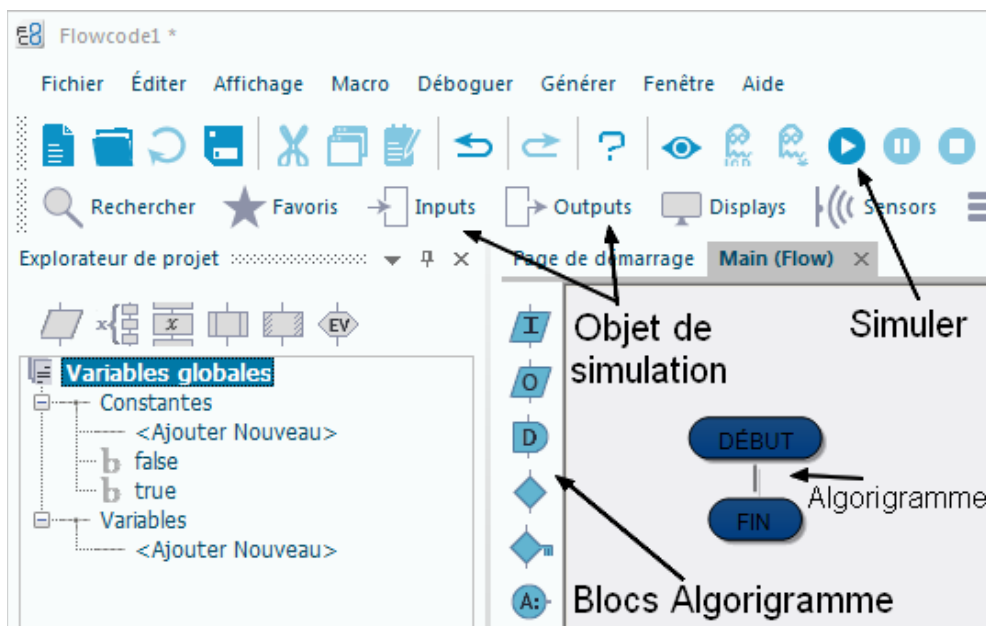
On va faire un programme tout simple : **Allumer une LED à partir d'un bouton.**

Sauvegarder votre travail !

Choisissez comme nom de fichier simple : Sans accent, aucun espace et limitez-vous à 8 caractères.

Présentation de l'interface :

	Lecture d'une entrée
	Sortie
	Délai (temporisation)
	Décision (test)
	Multi-décision
	Boucles
	Macro (sous-programme)
	Routine composant
	Calcul
	Traitement des chaînes de caractères
	Interruption
	Code-C



Explorateur de projet :

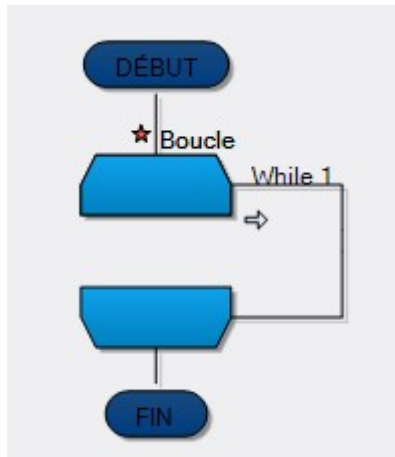
- Les ports
- Variables globales (Pour ajouter une variable et l'éditer)
- Main (Programme principal)
- Macros (Sous programmes)
- Composants (Présent sur le tableau de bord)

Création de l'algorithme

Afin de faire reboucler le programme sans fin il faut mettre en place une boucle :

Une boucle " Tant que " avec une condition égale à 1 (toujours vraie) est sans fin.

Il est possible d'éditer les propriétés de la boucle par un 'double clic'.



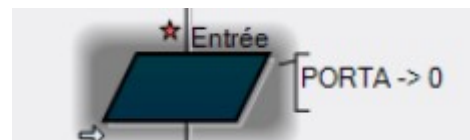
Propriétés: Boucle

Nom Affiché:
Boucle

☒ Tant que: Expression
☐ Jusqu'à: 1
☐ Compteur Boucle: Tester Boucle:
☒ Début
☐ Fin

Insérer un bloc " Entrée "

Affecter l'entrée du Bouton poussoir (A0) à une nouvelle Variable « Bouton » par exemple.



Propriétés: Entrée

Nom Affiché:
Entrée

Port :
PORTA

Variable :
0

Entrée depuis:
☐ Bit Unique: 0
☒ Port Complet:
☐ Masque: 7 6 5 4 3 2 1 0

Éditer Nom Variable

Saisir un nouveau nom de variable:
Bouton

Longueur de: Ajouter une dimension...

Valeur Initiale:

Description :

Type Variable:
☒ Bool (soit vraie, 1 soit fausse, 0)



Propriétés: Entrée

Nom Affiché:
Entrée

Port :
PORTA

Variable :
Bouton

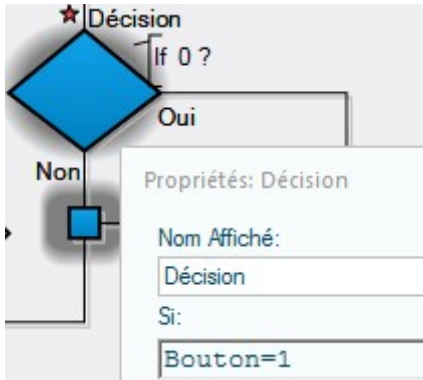
Entrée depuis:
☐ Bit Unique: 0
☒ Port Complet:
☐ Masque: 7 6 5 4 3 2 1 0

OK Annuler

Sauvegarder votre travail !

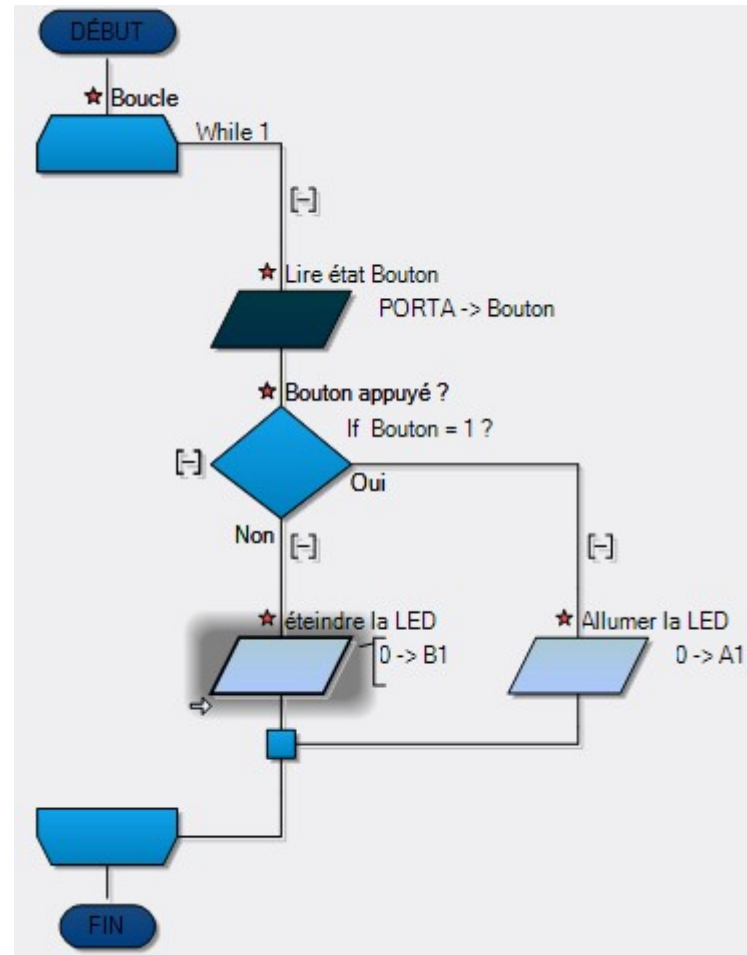
Insérer un bloc " Décision "

Utiliser Variable



Condition → Si Bouton=1 ?

Code complet du programme de la LED :



Insérer un bloc "Sortie "

- Sur la branche **OUI** : insérer un bloc " Sortie " et le nommer « **Allumer la LED** »

Double-clic sur la sortie

Mettre à 1 le **bit B1**

- Sur la branche **NON** : insérer un bloc " Sortie " et le nommer « **Éteindre la LED** »

Double-clic sur la sortie

Mettre à 0 le **bit B1**

Valeur de la
donnée ou
variable

Bit recevant la
donnée de la
variable

Valeur de la
donnée ou
variable










Bit recevant la
donnée de la
variable

Écriture en **binaire** : 0b101

Écriture en **Hexadécimal** : 0x8

Simulation :

Menu simulation

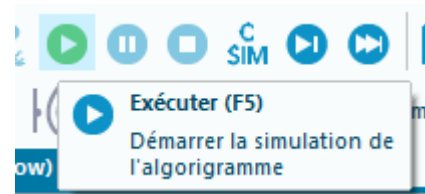
	GHOST
	Lancer la simulation en continu
	Effectuer une pause dans la simulation
	Arrêt de la simulation
	Pas à pas détaillé
	Pas à pas principal
	Compilation en code C (.ASM)
	Compilation en code exécutable (.HEX)
	Chargement de l'exécutable vers le micro-contrôleur

Lancer la simulation

Exécuter ou (F5)

Cliquer sur le bouton pour simuler un appui

Visualiser le résultat sur la LED



Charger le programme dans le Micro-contrôleur

Compiler et charger dans la puce :

(Nécessite une platine de programmation avec le PIC Choisi)

Par exemple : Matrix EB 006 ou HP 488



Sauvegarder votre travail !

Voir : <https://www.youtube.com/watch?v=Qp7ndYrzvdw>