1• GÉNÉRALITÉS

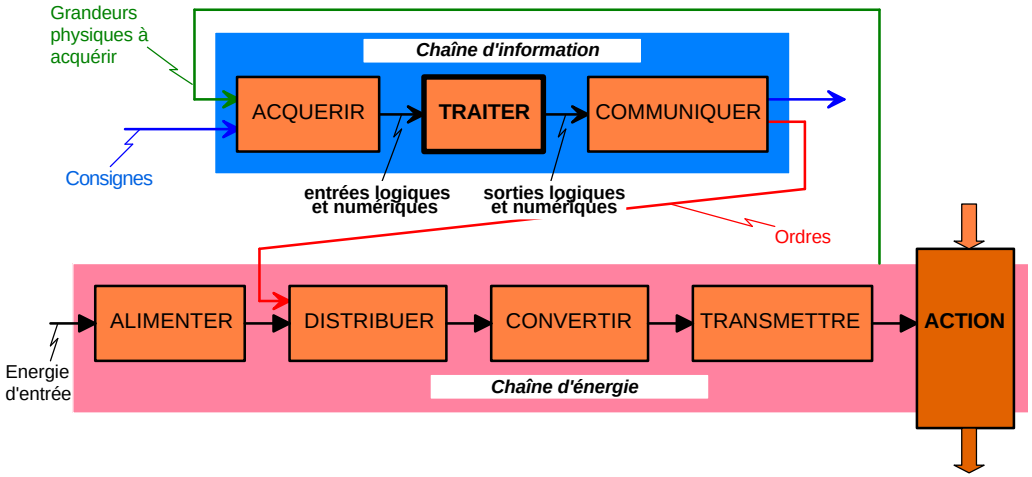
Les microcontrôleurs sont des composants programmables. Ils intègrent dans un seul boîtier l'environnement minimal d'un système à microprocesseur (l'UC, la RAM, l'EPROM et les interfaces). Ils sont présents dans la plupart des systèmes électroniques embarqués ou dédiés à une application unique. Il en existe de nombreux modèles différents avec parmi les plus courants : le 8051 de Intel, le 68HC11 de Motorola... et les PIC de Microchip

1.1• LES PIC 16F...

Les PIC de Microchip contiennent un processeur à jeu d'instruction réduit (RISC : Reduced Instructions Set Computer, constitué de 35 instructions seulement). La série 16F contient de la mémoire "Flash", reprogrammable des centaines de fois (idéale pour développer).
Il existe un grand nombre de PIC disponibles disposant d'un nombre d'entrée / sorties ou de ports différents, de mémoires de tailles variables, ou encore de capacités fonctionnelles différentes (pour la communication, la commande de moteurs à courant continu, etc...)

2• IDENTIFICATION DE LA FONCTION TECHNIQUE RÉALISÉE

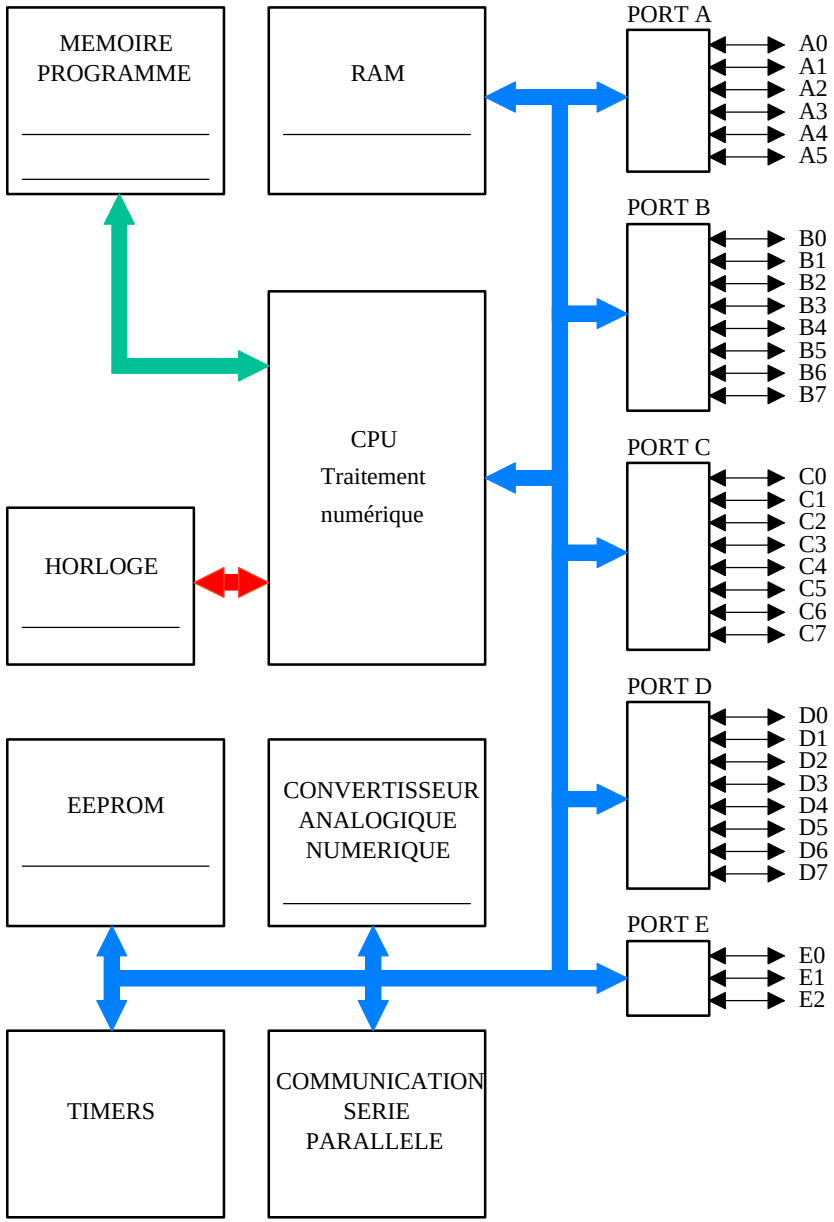
Les microcontrôleurs réalisent la fonction TRAITER de la chaîne d'information :



3• PRÉSENTATION DU PIC 16F877A

3.1• SYNOPTIQUE SIMPLIFIÉ DU 16F877A

C'est le modèle qui sera utilisé en TP. Le synoptique simplifié est le suivant :



3.2• L'UNITÉ DE TRAITEMENT (CPU : Central Processing Unit)

L'unité de traitement numérique exécute les instructions du programme (codées sur 14 bits). Il traite et produit des opérations sur des informations numériques uniquement.

3.3• LES MÉMOIRES

Elles se divisent en trois blocs distincts :

- la mémoire programmes Flash (8 k mots de 14 bits),
- la RAM (368 octets) est utilisée pour le stockage temporaire des données et résultats,
- l'EEPROM (256 octets) qui peuvent être lus et écrits depuis le programme. Ces octets sont conservés après une coupure de courant et sont très utiles pour conserver des paramètres semi-permanents.

3.4• L'HORLOGE

Associé à un quartz externe de 8 MHz, elle génère le signal qui cadence l'exécution des instructions. Chaque instruction du programme est traitée en un cycle machine (une période de l'horloge avec une division interne par 4 soit 2 MHz effectifs). La durée d'exécution d'une instruction est donc de 500ns.

3.5• LES PORTS

Pour communiquer avec l'extérieur le PIC dispose de 5 ports (PORT A, PORT B, PORT C, PORT D et PORT E). Les ports sont bi-directionnels, ce qui signifie qu'ils peuvent être configurés et utilisés comme des entrées ou des sorties.

Le microcontrôleur reçoit les informations sur un port d'entrée :

- informations logiques issues de capteurs sur un ou plusieurs bits d'un port d'entrée ;
- informations numériques codées sur 8 bits sur un port entier (le code d'une touche d'un clavier par exemple) ;
- informations analogiques variables dans le temps (une tension représentative d'une température par exemple) si le PIC est doté d'un convertisseur analogique / numérique.

Le microcontrôleur traite ces données et les utilisent pour commander des circuits qui sont connectés à un port de sortie.

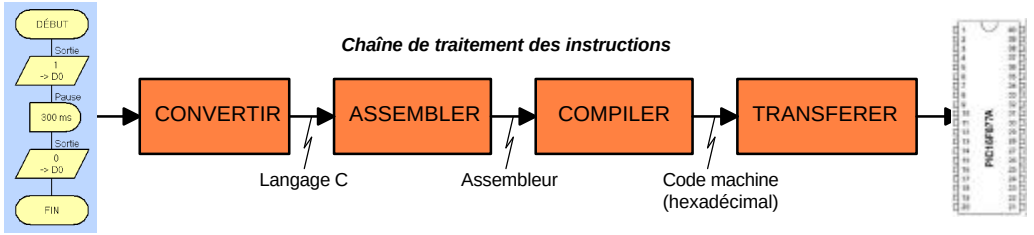
3.6• LE CONVERTISSEUR ANALOGIQUE NUMÉRIQUE (CAN)

Il peut convertir 8 tensions analogiques (variables entre 0 et 5V) appliquées sur ses 8 entrées (PORT A et PORT E) en nombres binaires codés sur 10 bits. Les résultats des conversions sont stockés dans des registres internes de l'UC).

4• PROGRAMMATION DES PIC

Les microcontrôleurs sont des composants programmables. Ils exécutent les instructions du programme et rien d'autre. Les instructions sont codées en binaire pour pouvoir être comprises par le microcontrôleur (code machine).

Pour programmer les PIC nous utiliserons un logiciel de programmation graphique, FLOWCODE. Ce logiciel permet, à partir de la saisie d'ordinogrammes, de créer des programmes pour les microcontrôleurs de la famille des PICmicro® de Microchip.
Une fois l'algorigramme élaboré, Flowcode permet de simuler et visualiser le comportement du programme en découlant, avant de le traduire en langage C, de l'assembler, de le compiler en hexadécimal et de le transférer dans le microcontrôleur cible.



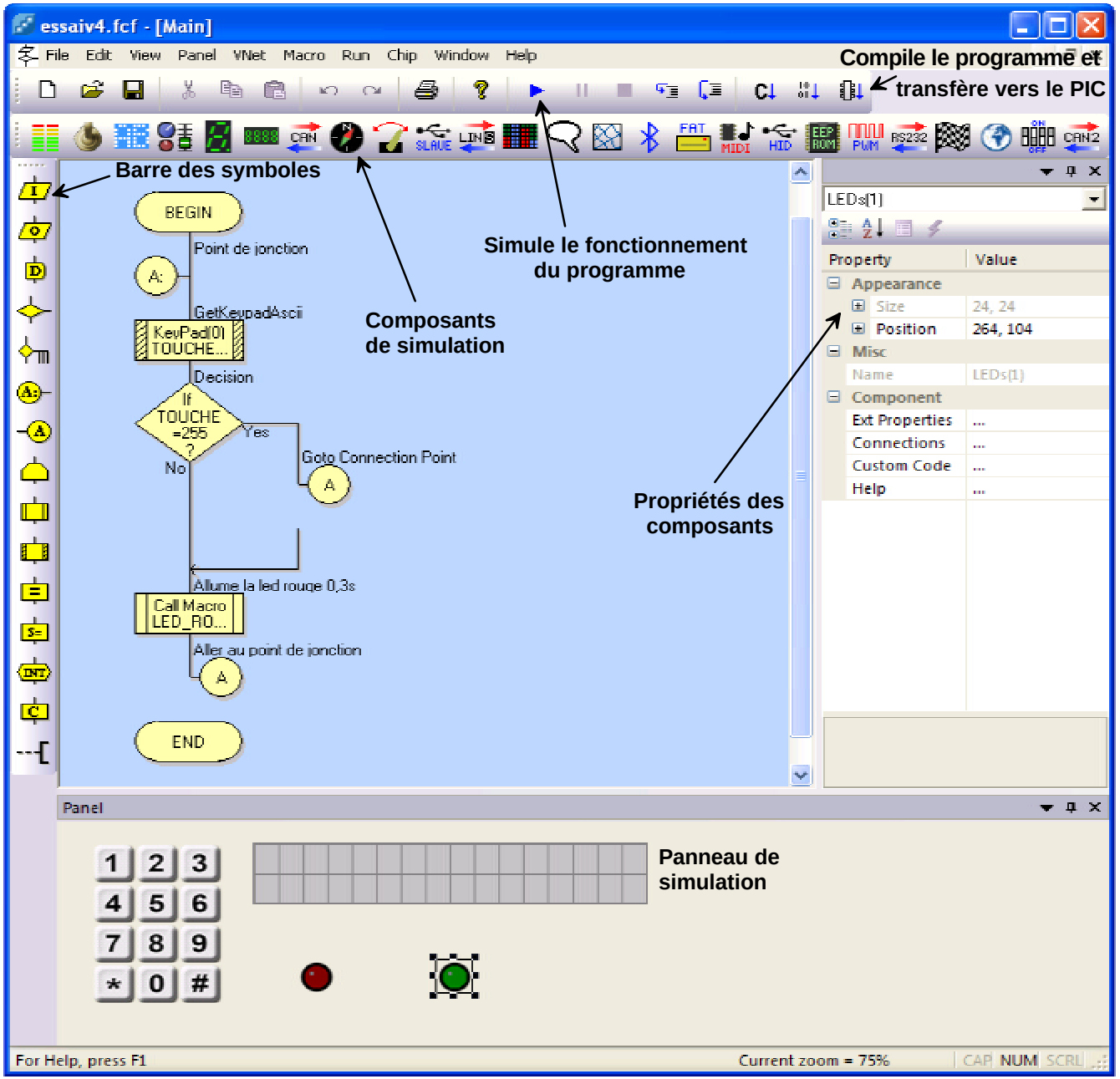
Le langage C est un langage de programmation de bas niveau conçu pour la programmation système. Inventé au début des années 1970 il est devenu un des langages les plus utilisés. De nombreux langages plus modernes comme C++, Java et PHP reprennent des aspects du langage C.

L' assembleur est un langage de bas niveau qui représente le langage machine sous une forme lisible. Les combinaisons de bits du langage machine sont représentées par des symboles appelés «mnémoniques», c'est-à-dire faciles à retenir.

Le langage machine, ou code machine, est la suite de bits qui est interprétée par l'unité de traitement. C'est le seul langage que le microprocesseur puisse traiter. Il est composé d'instructions et de données à traiter codées en binaire.

5- UTILISATION DU LOGICIEL FLOWCODE V4

FLOWCODE est un logiciel de programmation graphique. Il permet de représenter le programme à implanter dans le PIC sous forme d'ordinogramme. La figure ci-contre présente la page de travail du logiciel et ses principales fonctions.

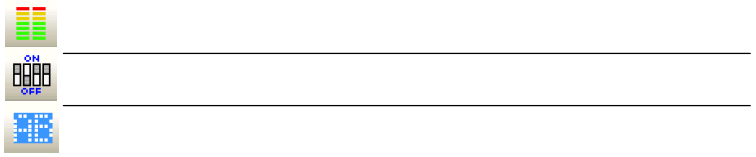


5.1- LA BARRE DES SYMBOLES

Pour dessiner l'ordinigramme, FLOWCODE utilise les symboles normalisés. On les insère dans le programme par un glisser-déposer à partir de la barre de symboles :

5.2- LES COMPOSANTS DE SIMULATION

Avant de programmer le PIC, on peut vérifier par simulation le fonctionnement du programme. Les composants peuvent être connectés au PIC pour réaliser cette vérification



5.3• LES VARIABLES

Une variable désigne une information qui peut être modifiée au cours du programme. Elle permet de :
- de mémoriser une information provenant de l'extérieur.
- de stocker le résultat d'une opération.

Dans FLOWCODE on distingue deux types de variables :
- les variables de type OCTET, codées sur 8 bits et non signées, sont des nombres qui seront compris entre 0 et 255 ;
- les variables de type ENTIER, codées sur 16 bits et signées, sont des nombres qui seront compris entre -32768 et 32767 ;
- Les variables de type VIRGULE FLOTTANTE, nombres signés à 32 bits, sont des nombres qui seront compris entre ($3,4 \bullet 10^{-38}$ à $3,4 \bullet 10^{38}$).

5.4• LES OPÉRATIONS

FLOWCODE permet d'effectuer une ou plusieurs opérations dans une boite de calcul. Toutes les opérations doivent contenir le nom d'une variable existante, le signe = et une expression faite de nombres, variables et des opérateurs suivants :

( )	- parenthèses,
= <>	- égal , différent,
+ - * / MOD	- addition, soustraction, multiplication, division et modulo (reste),
< <= > >=	- inférieur, inférieur ou egal, supérieur, supérieur ou égal,
>> <<	- décalage à droite, décalage à gauche,
NOT AND OR XOR	- complément, ET, OU, OU Exclusif.

5.5• LES NOMBRES

Dans les boites de calcul , les nombres peuvent être écrits en base 10, en base 2 ou en base 16. Pour identifier la base on utilise les symboles distinctifs *0b* pour la base 2 et *0x* pour la base 16. Exemples :
- 0b10101111 représente le nombre binaire $(10101111)_2$
- 0xFF représente le nombre héxadécimal $(FF)_{16}$

5.6• LES SOUS PROGRAMMES (MACROS)

Un sous-programme est un morceau de programme que l'on utilise souvent. Généralement, il réalise une fonction. Plutôt que d'écrire ce morceau de programme chaque fois qu'il est nécessaire, il est plus judicieux de l'écrire une seule fois et de l'intégrer dans une boîte appelée MACRO. On peut ensuite y faire appel aussi souvent que l'on veut. De plus, les sous-programmes facilitent la lisibilité et la mise au point du programme.
Le logiciel intègre aussi des macros prédéfines pour certains composants : ce sont les routines composant.

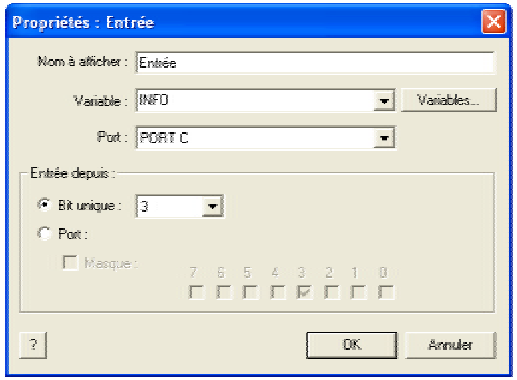
5.7• LES MASQUES

Les masques permettent :
- de tester l'état d'un seul bit parmi les 8 bits d'un port, sans se préoccuper de l'état des 7 autres (en les masquant).
- d'écrire une valeur sur un seul bit parmi les 8 d'un port de sortie sans modifier les autres.

Exemple : l'ordinogramme précédent réalise la lecture du bit 3 du PORTC et stocke cette information dans la variable INFO

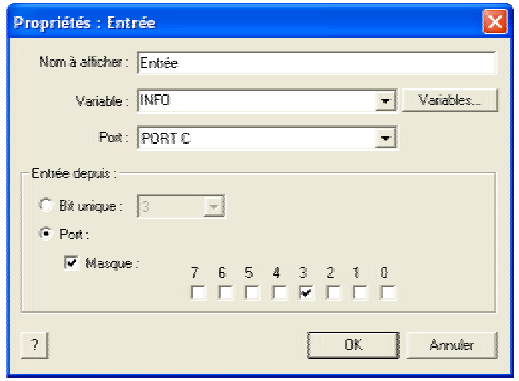
Pour obtenir ce fonctionnement, il faut double-cliquer sur le symbole INPUT. La boite de dialogue des propriétés s'ouvre. Elle doit alors être complétée :
- Cocher ***Bit unique*** puis sélectionner le bit 3.

Dans ce cas, la variable INFO est une variable logique (0 ou 1).



On peut obtenir un fonctionnement similaire avec la fonction Masque :
- Cocher ***Port*** et ***Masque***,
- Cocher ensuite le bit 3 (le bit à tester).

Dans ce cas, la variable INFO est une variable numérique. Elle pourra prendre la valeur 0 ou 8.



Cependant, on utilisera la fonction Masque plutôt quand on doit tester plusieurs bits ou lorsque certains bits d'un port sont des entrées et d'autres des sorties.