

SmileNG (Sébastien Gerlach , 1998)

SmileNG est un éditeur-assembleur facile à utiliser. L'éditeur fait en plus une mise en évidence typographique qui facilite la structuration et relecture des programmes.

L'environnement de SmileNG se charge depuis une disquette, un CD ou depuis le site www.didel.com/pic/SmileNG.zip . Il faut alors décompresser le fichier .zip dans un dossier SmileNG et établir un raccourci de SmileNG.exe dans le bureau.

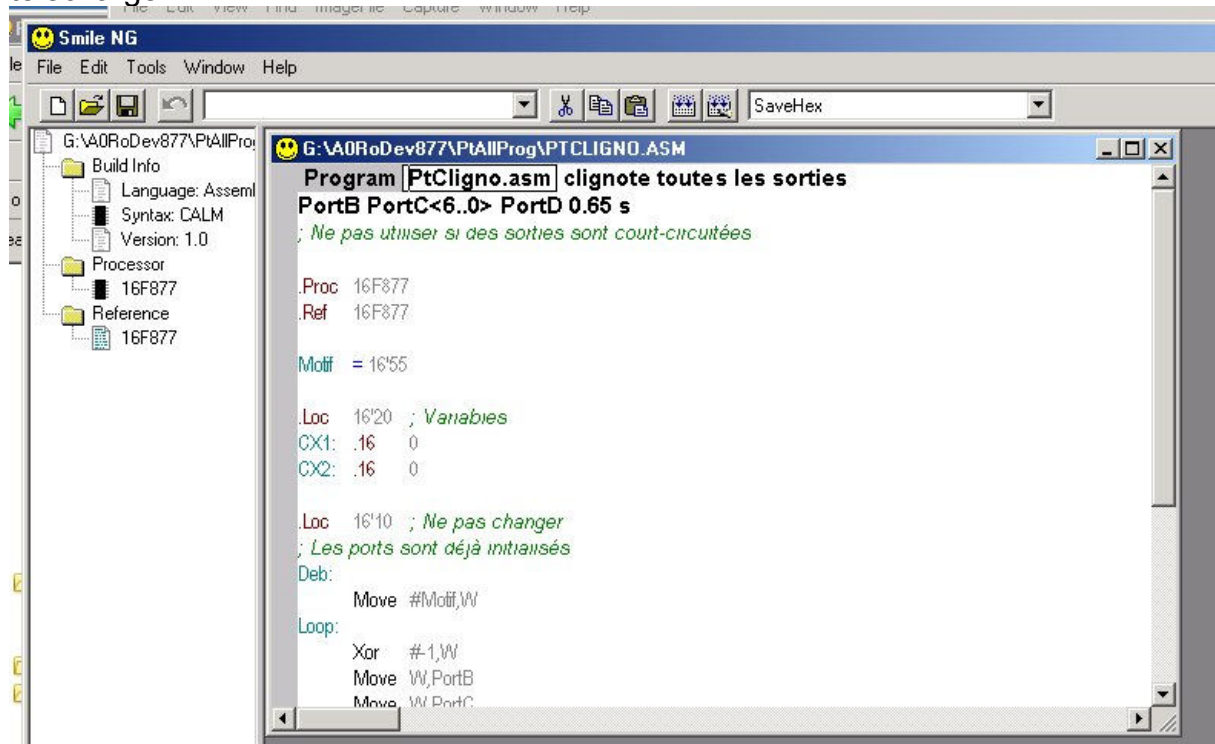
Le répertoire SmileNG contient les sous-dossiers et fichiers suivants:

SmileNG.exe	le programme
Modules	des DLL et fichiers de configuration
Scriptini	l'assembleur et d'autres fichiers utiles
Proc	assembleurs pour différents processeurs
Ref	déclaration des bits de modes et d'entrée/sortie

L'exécution de SmileNG donne une en-tête assez traditionnelle :



Charger un fichier, par exemple PtCligno.asm dans le dossier Pt des fichiers à télécharger :



On voit que la structure du programme a été extraite et que des mises en évidences typographiques ont été choisies lors de l'écriture du programme.

Ordres LILA

Les ordres LILA (Listing on Laser) ont été utilisés au LAMI-EPFL dès que les imprimantes laser ont permis des effets de taille de caractère et graphisme. Ces ordres sont compatibles avec tous les éditeurs, mais ne sont interprétés que par l'éditeur SmileNG, si on le demande. La touche F8 commute d'un mode à l'autre.



On voit que les ordres Lila commencent par un \ (boa ou barre oblique arrière) en début de ligne. Ce signe est considéré par l'assembleur comme un début de commentaire. Les ordres se terminent par un ; Un séparateur | (barre verticale, attention il y en a deux différentes sur les claviers de PC) est utilisable dans certains ordres. Les mots clés décodés sont les suivants:

```

\titre;xxxxxx
\prog;xxxx|zzzzz
\module;xxxx|zzzzz
\rout;xxxx|zzzz
\const;xxxx|zzzz
\var;xxxx|zzzz
\macro;xxxx|zzzz
\table;xxxx|zzzz
\b;xxxxxx
\in;xxxx
\out;xxxx
\mod;xxxx

```

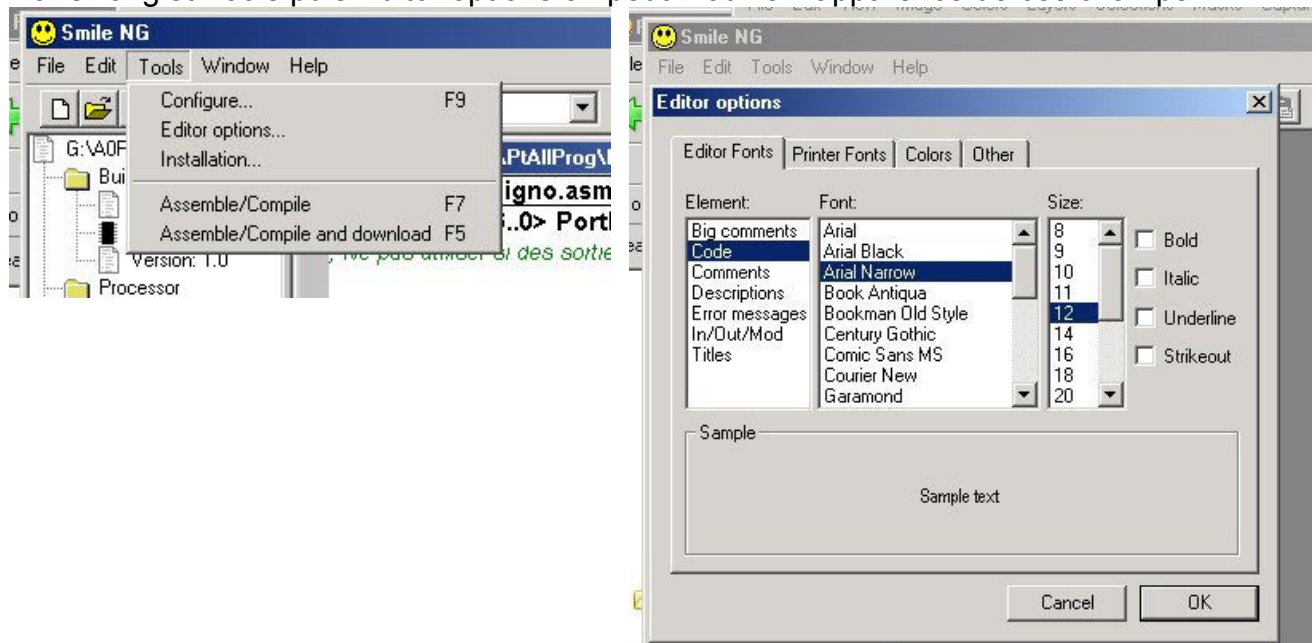
```

XXXXXXXX
Program xxxx|zzzzz
Module  xxxx|zzzzz
Routine  xxxx|zzzz
Constants xxxx|zzzz
Variables xxxx|zzzz
Macro   xxxx|zzzz
Table   xxxx|zzzz

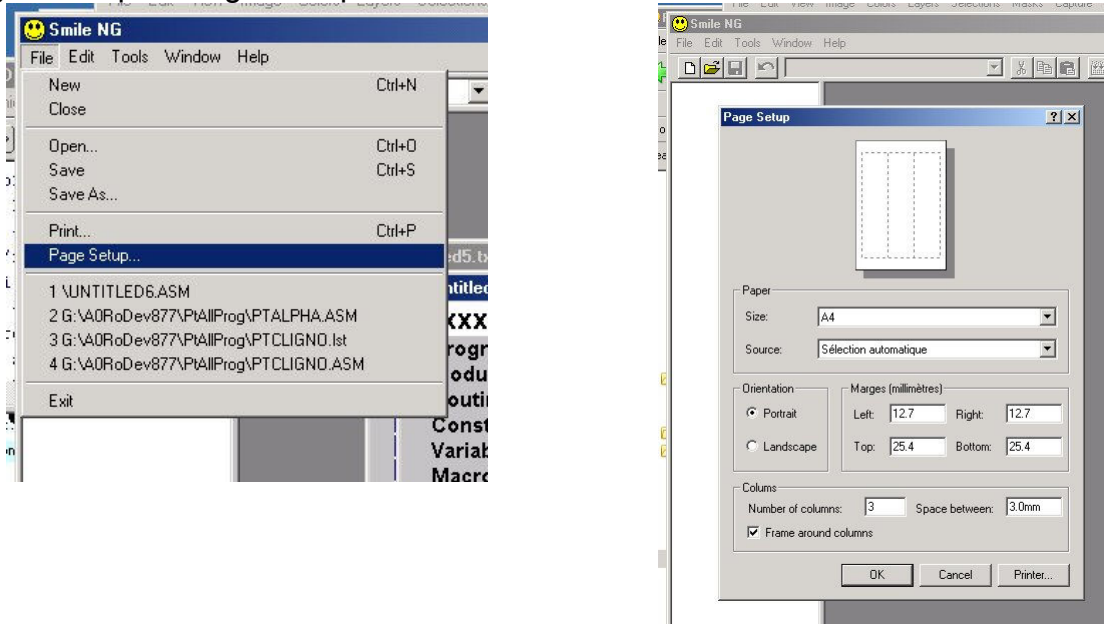
XXXXXXXX
In:  xxxx
Out: xxxx
Mod:xxxx

```

Dans l'onglet Tools puis Editor options on peut modifier l'apparence de ces champs.



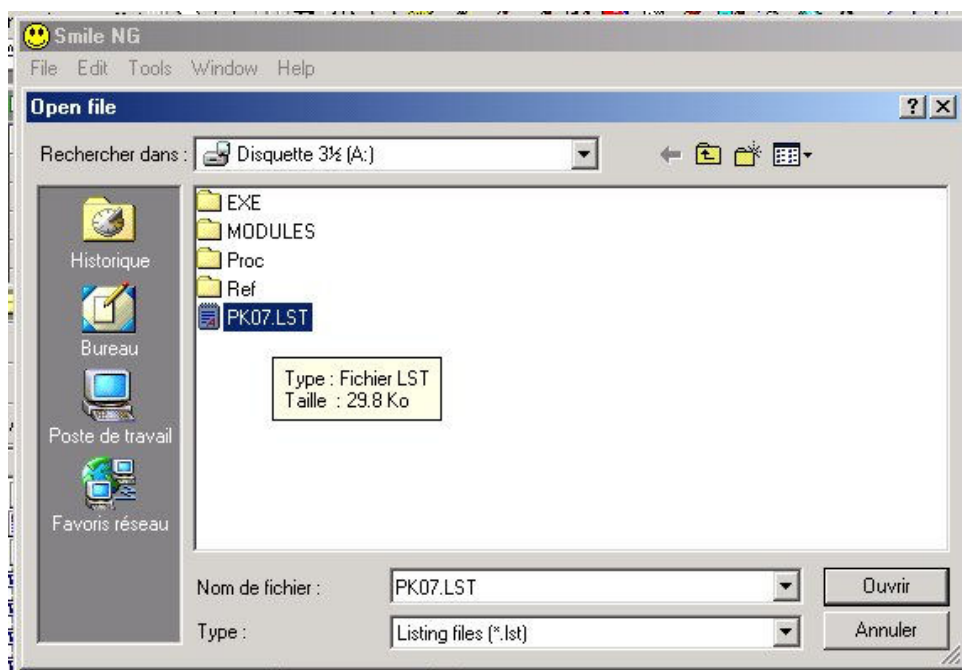
Encore plus intéressant, on obtient des listages sur n colonnes en spécifiant dans l'onglet File puis Page set-up



Il y a toutefois un problème avec les listages. Les ordres Lilas doivent être en première position dans la ligne. Il faut donc un peu d'édition manuelle (enlever ce qui a été rajouté par l'assembleur) pour obtenir les listages copiés dans notre documentation.



A noter que pour charger un listing, il faut sélectionner la rubrique (*.lst) ou "All files" et cliquer sur le fichier choisi pour l'ouvrir.

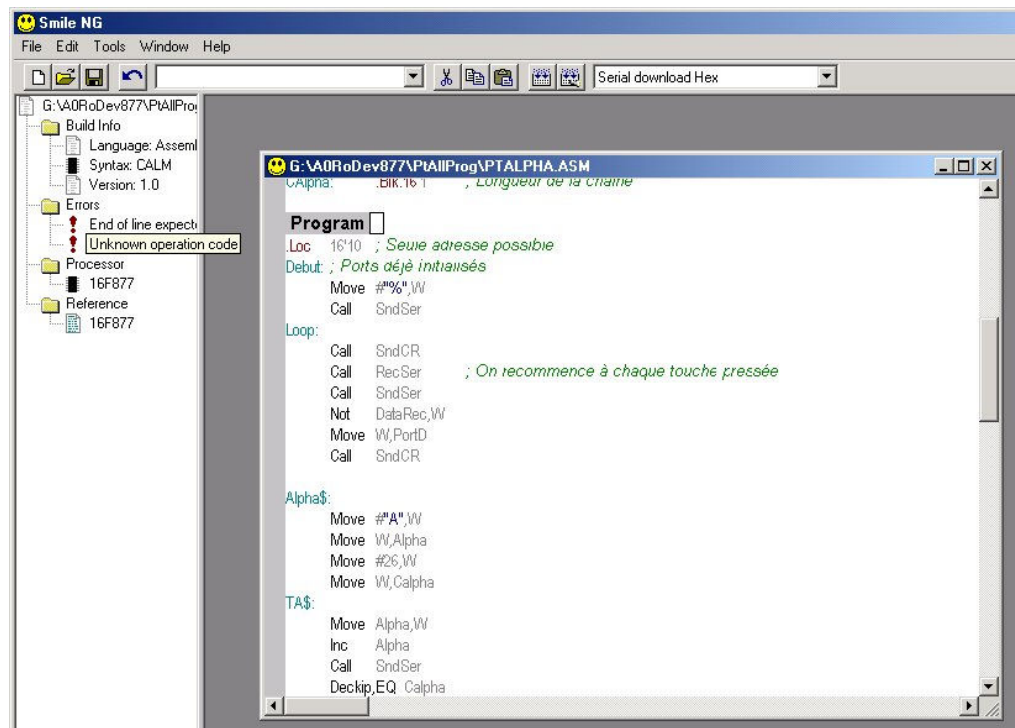
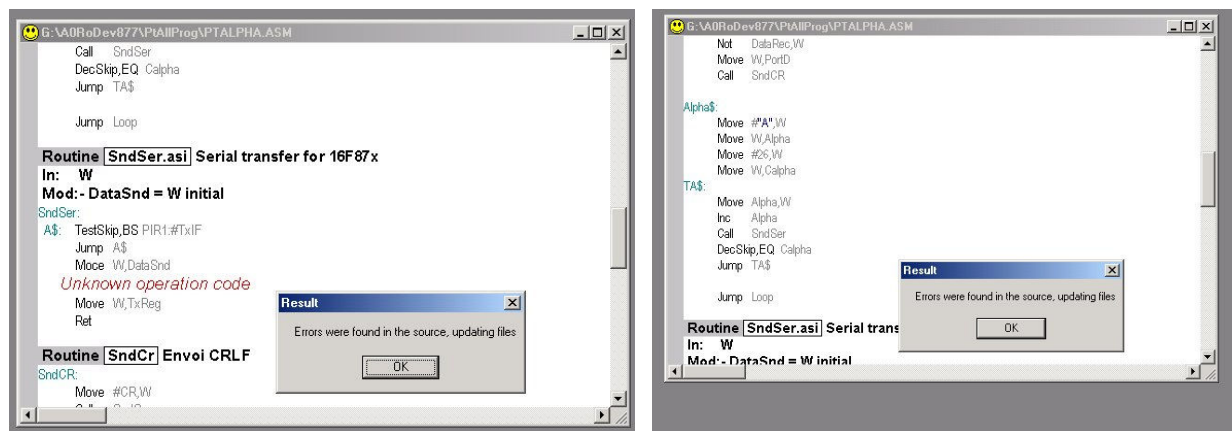


Assemblage

L'assemblage d'un programme se fait en pressant la touche F7 ou cliquant l'icône A à côté des icônes coupe/copy/colle. Avec le Pickit2, on s'intéresse à tout de suite charger le fichier .hex, (si l'assemblage est correct) la touche F5 (icône T) faire cela. Dans la fenêtre en haut, on doit avoir "Save Hex" et pas "Serial download" que l'on utilise avec un boot loader.



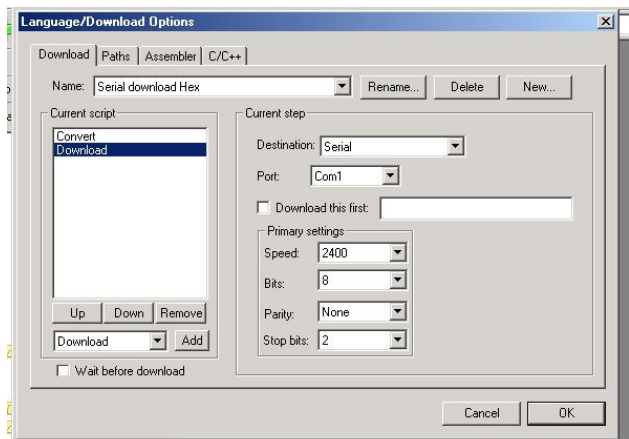
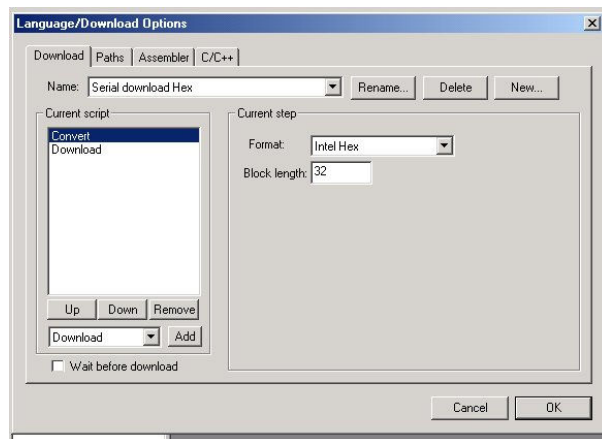
Si l'assemblage génère des erreurs, elles sont indiquées dans le source et listées à gauche. En cliquant sur le point d'exclamation on sélectionne la ligne correspondante, avec malheureusement un bug fréquent : la ligne d'erreur est juste cachée au bas de l'écran et il faut descendre l'ascenseur.



Si l'erreur est dans un fichier inséré, il est en général ouvert. Il faut alors corriger et sauver, et naturellement retourner au programme source pour retenter un assemblage. Après correction d'une erreurs, il n'est pas nécessaire de détruire le message d'erreur de la ligne suivante. Un nouvel assemblage nettoie le tout et resigneale les erreurs. Les itérations de correction sont donc très rapides.

Téléchargement

SmileNG permet différentes configurations pour créer des fichiers résultant d'un assemblage ou d'une compilation. Ce qui nous intéresse est de télécharger un programme. L'option Serial download Hex permet ceci. Elle résulte de deux choix dans le menu configuration (Tools – Configure).



Les possibilités de configuration sont nombreuses. Elles sont mémorisées dans le fichier Script.ini (dossier modules) et ce fichier peut être édité pour changer l'ordre ou récupérer dans le fichier ScriptComplet.ini des configurations utiles.

Par exemple, pour télécharger le programme PtCligno.asm dans la carte RoDev877 programmée avec le moniteur Pd08, il faut que le module soit sous tension et que la liaison série soit établie (voir PdSerie.doc).

Pour plus de détails sur la carte RoDev877, optimisée pour des applications avec 2 moteurs, 4 servos, 5 potentiomètres et des routines téléchargées et un mode de dialogue avec un programme "terminal" ou un visual Basic par exemple, voir <http://www.didel.com/dev877/>

Programmateurs

En programmant directement le PIC, on dispose de toute la place mémoire, mais on doit utiliser un programmeur qui génère des impulsions série sur 3 pins du processeur.. C'est nécessaire pour programmer des processeurs plus petit que le 16F877, après avoir mis au point le logiciel et les interfaces sur le RoDev877. C'est aussi nécessaire pour reprogrammer le RoDev877 si son programme a été corrompu ou si l'on veut charger PkLodHex.hex.

Le PicKit2 est la solution la plus économique et élégante. SmileNG crée un fichier au format .hex, repris par le logiciel du PicKit2 pour programmer le processeur choisi. Didel a abondamment documenté différente solution pour programmer avec le PicKit2, à différents niveaux de complexité. Voir <http://www.didel.com/pic/Programmer.pdf>