

## Basic Bascom mode d'emploi

### Avant Propos à lire absolument

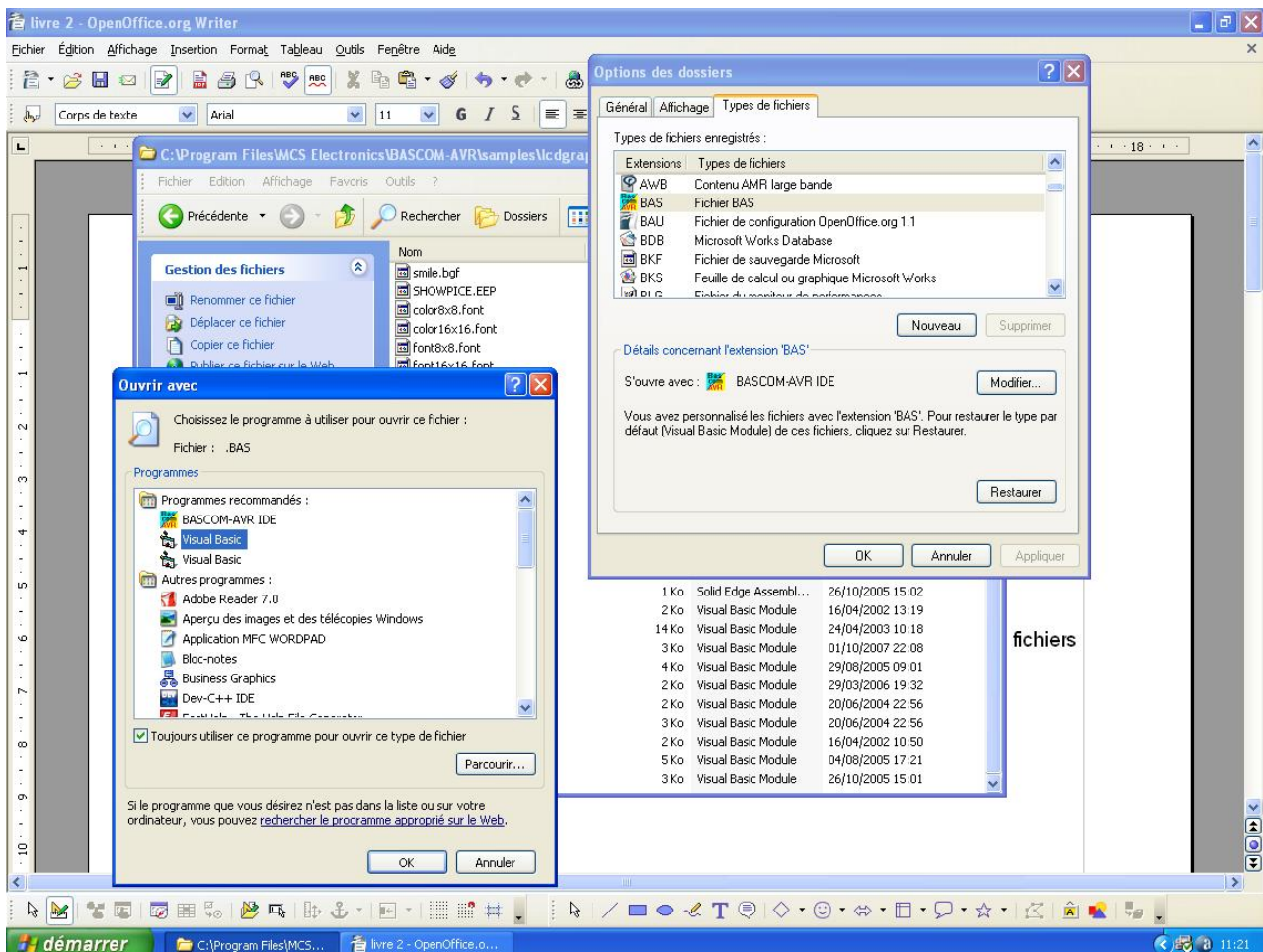
Une nouvelle version du E-livre sur le Basic Bascom est toujours un événement, surtout pour nous car c'est une masse de travail considérable !

Nous avons modifié l'utilisation de cet ouvrage en utilisant massivement les liens hypertextes pour naviguer dans l'ouvrage et pour les exemples, à la différence de l'aide PDF Bascom cette aide est modifiable par l'utilisateur.

Les exemples sont toujours ceux de la dernière version de Bascom sous Windows XP ils doivent être sous <C:\Program Files\MCS Electronics\BASCOM-AVR>

Elle est rédigé avec le logiciel libre OpenOffice.org Writer

Beaucoup d'entre nous utilisent le Visual-Basic pour développer des applications, donc les fichiers de type .bas s'ouvrent « naturellement » avec VB, il faut modifier cela



Dans la barre de titre des répertoires (Windows XP mode 98) extension visible.

cliquer sur outils/ option des dossiers/ choisir Bascom-AVR.exe pour ouvrir les dossiers .bas

Cette option est réversible si vous utiliser conjointement VB et Bascom, à vous de choisir !

## ***Un système intégré de développement***

Le logiciel BASCOM-AVR est disponible gratuitement en version démo (limité à 4Ko de développement, 4 Ko cela permet déjà de réaliser quelques applications. En revanche, malgré le simulateur très efficace par ailleurs, il faut se procurer rapidement une carte de développement, une alimentation et un câble avec le dongle parallèle ou l'interface USB.

Un système intégré de développement réuni plusieurs logiciels :

- l'éditeur qui permet d'écrire le programme et les remarques y afférentes.
- le simulateur qui permet de faire tourner sur le PC le programme écrit, il permet de visualiser l'état des ports, des registres etc..
- le compilateur qui traduit le code Basic en code binaire compatible avec le  $\mu$ -contrôleur il visualise les erreurs possibles.
- l'émulateur terminal qui permet de visualiser les caractères envoyés par le port série.
- Différents outils permettant de visualiser les broches des  $\mu$ -contrôleurs par exemple, de rechercher des données, de fixer des marques pages etc..

## ***Installation du logiciel***

Celle-ci ne pose aucun problème, on se laisse guider, et tout se passe bien.

Le classique SETUP.exe fait des merveilles. Nous vous conseillons d'accepter toutes les options d'installation. Vous devez disposer de VOTRE fichier bcsavr.dll (pas pour la version de démo) Nous vous conseillons de recopier les exemples dans un sous-répertoire, pour éviter des modifications désastreuses.

Une fois le programme chargé, démarrer en cliquant sur l'icône, Bascom démarre avec un écran vierge la première fois et avec le dernier programme ouvert ensuite. On retrouve les menus déroulant classiques de l'environnement Windows® .

La première chose à faire est le contrôle de la version fournie par le revendeur.

Si celle-ci n'est pas la dernière, à contrôler sur le site [www.mcselec.com](http://www.mcselec.com), une mise à jour s'impose.

## ***Où faire une mise à jour ?***

nous vous conseillons très fortement d'utiliser le dossier **C:\Program Files\MCS Electronics** Dans ce répertoire il doit y avoir un sous-répertoire **BASCOM-AVR** [Tout ce que vous allez télécharger devra être mis dans ce répertoire.](#)

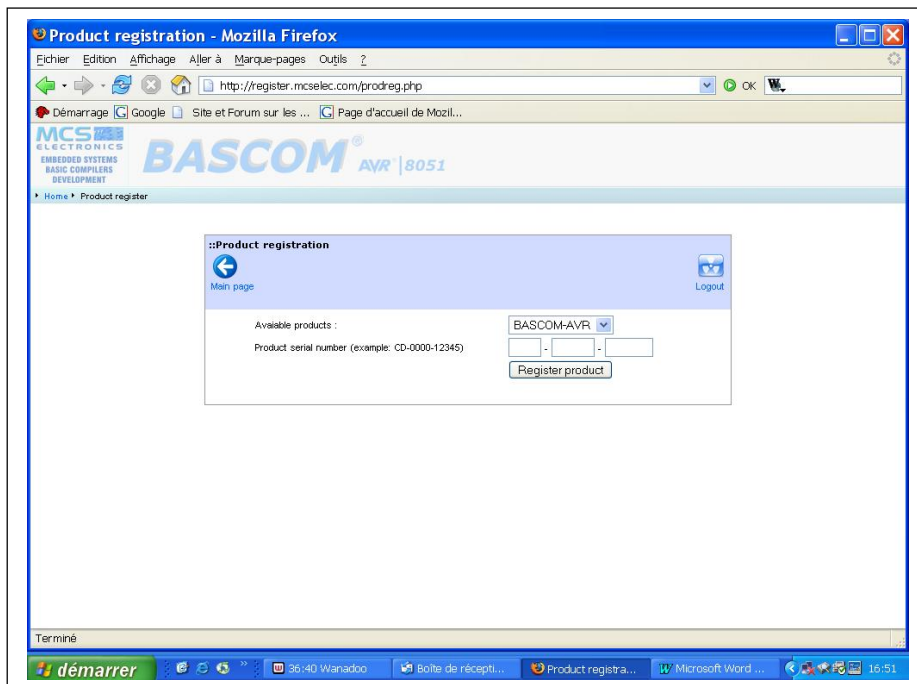
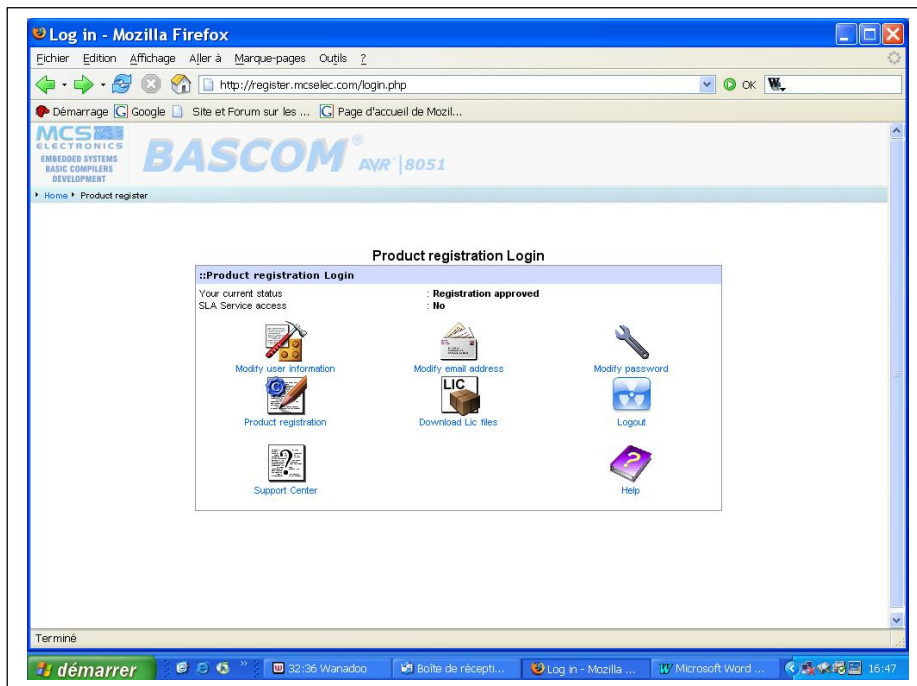
## ***Comment faire une mise à jour de programme a partir du site ?***

- 1) Durant toute la mise à jour, il faudra rester en ligne sur le site.
- 2) Vous devez connaître votre numéro de licence, sinon, ouvrez Bascom et allez dans **Help** puis **about** en rouge apparaît le numéro de série SERIAL XX-1234-56789
- 3) Allez dans Product registration puis create new account et répondre aux questions Vous

serez prévenu quelques heures ou quelques jours après de votre mot de passe.

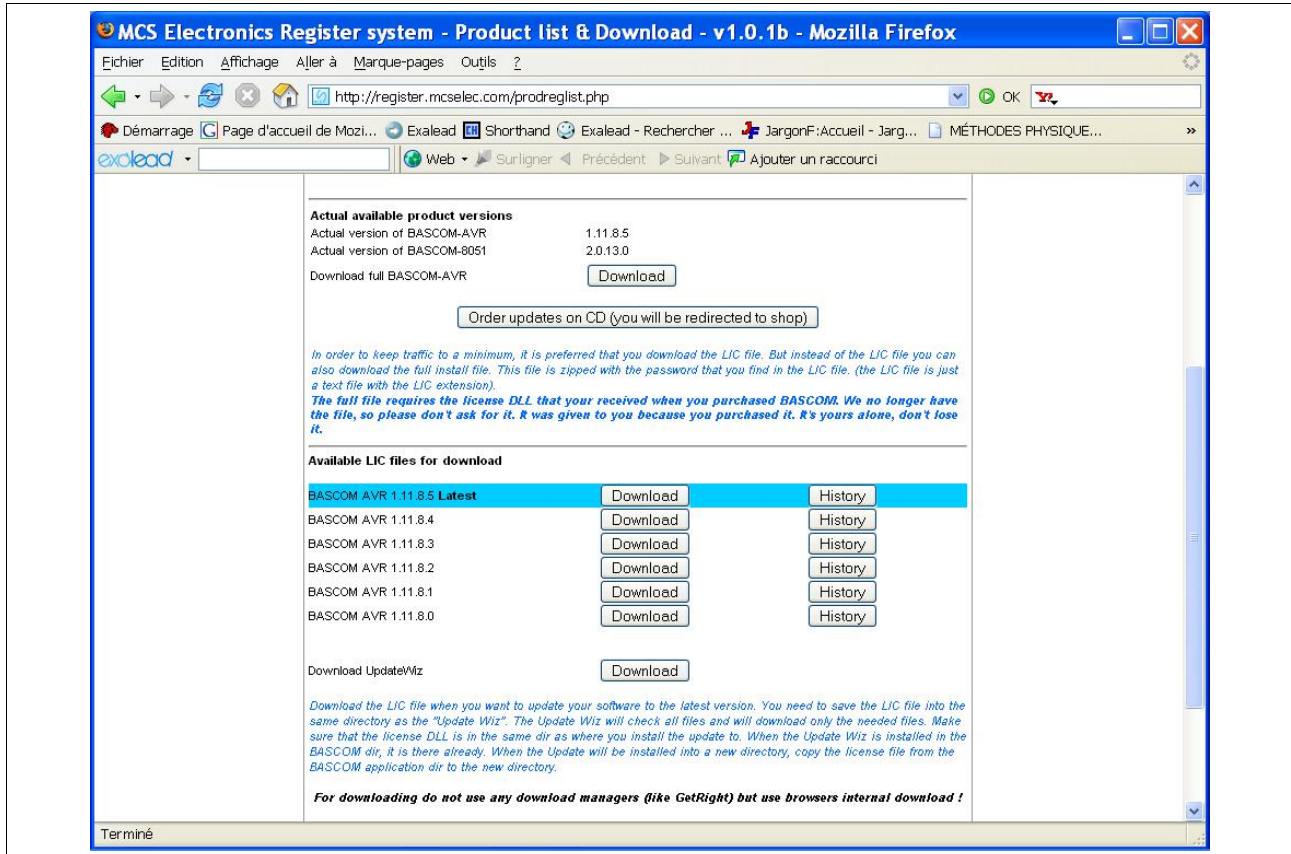
4) Avec votre user-name et votre mot de passe, vous pouvez entrer dans le centre de mise à jour.

5) Dans celui-ci on choisira d'abord l'onglet Product registration



Où on est invité à rentrer son numéro de licence. Il est impératif de faire une sauvegarde du numéro de licence.

A ce moment on est admis à retirer les mises à jours en cliquant sur l'onglet LIC. Commencez par celui-ci. Updatewiz il est souvent spécifique à la dernière version.



De même le fichier licence : bascom-avr-1.11.9.1.lic ou supérieur.

Lancer UPDATEWIZ.exe dans le fichier BASCOM-AVR tout doit se passer sans problème.

Pour mémoire: Il nous reste à désigner la carte de programmation choisie et le port parallèle de communication : Après quelques déboires, parce que je n'avais pas respecté tout ce qui précède, j'ai enfin réussi à configurer ma carte STK200 sur le port &HC000 (carte additionnelle parallèle) Ceci est valable pour Windows XP car Mr Gates à trouvé très intelligent de supprimer le support des ports // dans la version WISTA.

En général le port Parallèle est à l'adresse &H378 ou &h278  
 Pour connaître le numéro du port, aller dans : Démarrer de Windows®, paramètres, panneau de configuration, système, matériel, gestionnaire de périphériques, Port (Com et Lpt) choisir le port affecté au dongle, puis dans ressources la valeur de celui-ci...

### Les programmes de test (Samples)

Chaque programme permet d'évaluer certaines instructions, faites en une sauvegarde dans un autre répertoire, cela permet de les modifier pour constater de visu le rôle de chaque instruction. Dans le dictionnaire, les hyperliens vont chercher les fichiers dans les « samples » donc attention aux modifications.

Nous voici enfin prêt à découvrir le BASCOM-IDE.

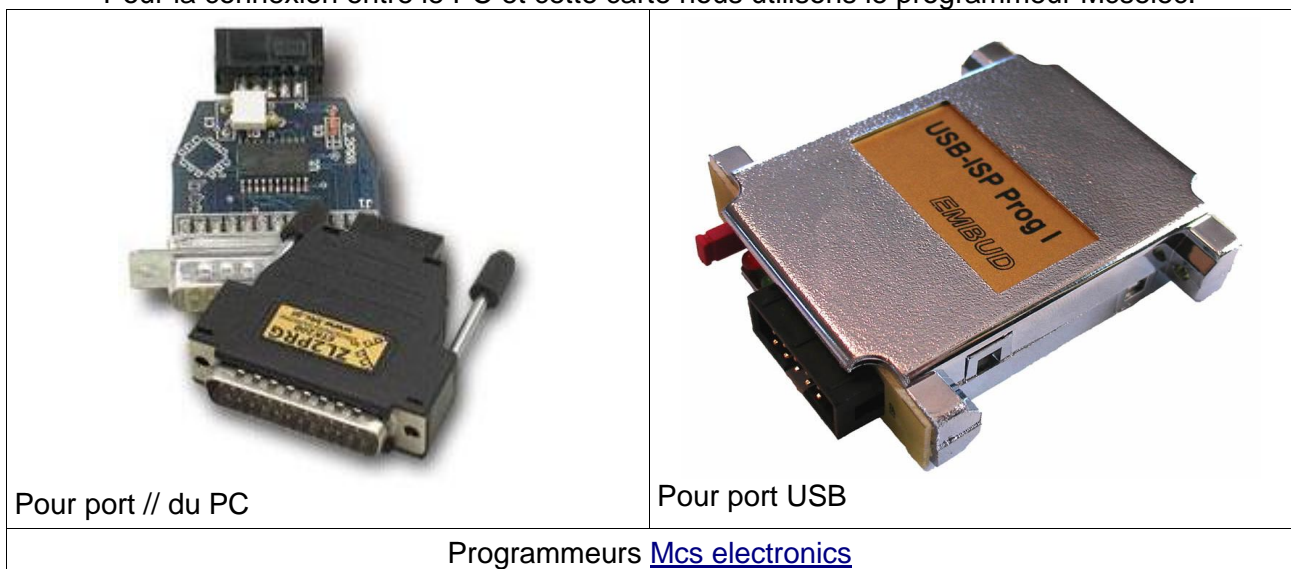
#### Mais avant quelques remarques importantes :

- Si vous avez installé une version démo, il faut la désinstaller avant d'installer la version définitive. Surtout le fichier bscavr1.dll
  - Quand vous recevez le logiciel de votre revendeur, inscrivez-vous sur [www.mcselec.com](http://www.mcselec.com) pour obtenir les droits de mise à jour.
  - Si vous utiliser NT vous devez avoir les droits d'Administrateur pour l'installation, vous devez aussi lancer Bascom avec ses droits ouverts pour configurer les drivers.
  - Pour d'autres problèmes lire les fichiers Readme.com, ou envoyer un Email à [support@mcselec.com](mailto:support@mcselec.com) avec tous les détails : version, versions windows, pb..
- Ou en utilisant le forum : [www.mcselec.com/forum/](http://www.mcselec.com/forum/), ou encore en passant par mon site : <http://perso.orange.fr/bils-instruments/>

### **La carte de développement**

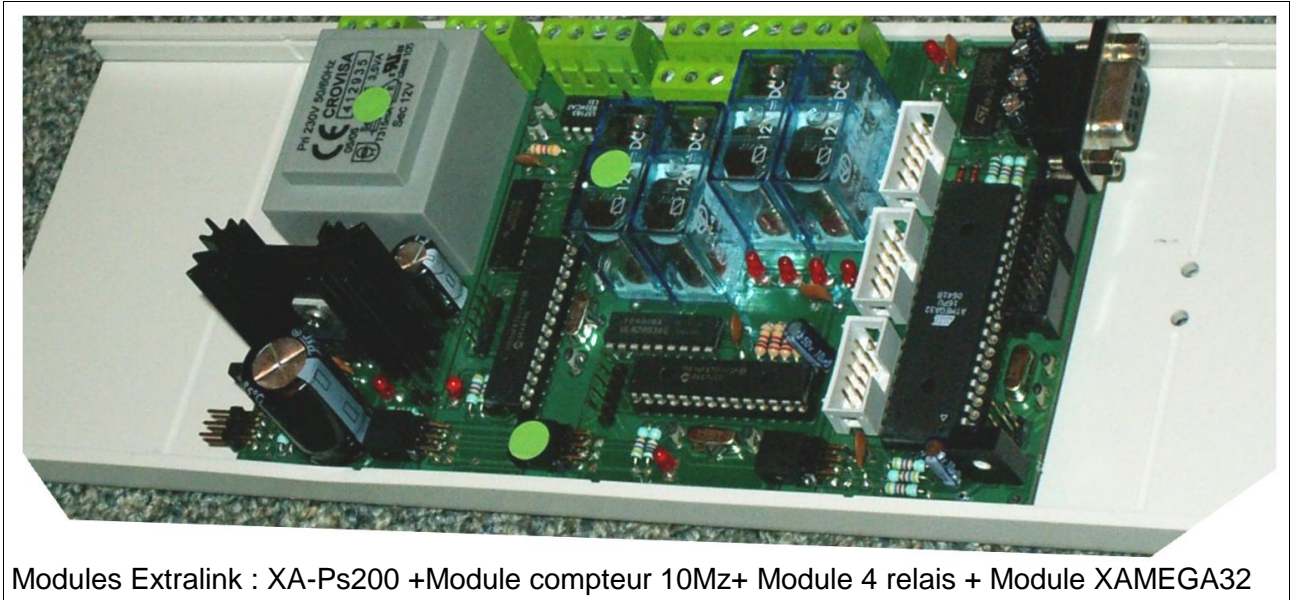
Il y a beaucoup de carte de développement fonctionnant avec Bascom, nous en utilisons trois:

- La carte XAMEGA32 de [Sidena](#) qui à l'avantage de pouvoir être utilisé dans une application puisqu'elle fait partie du système Extralink
- Pour la connexion entre le PC et cette carte nous utilisons le programmeur Mcselec:



- les cartes STK200 et STK300 sont bien détaillé sur le site [Kanda](#)

- En plus il faut ajouter une alimentation type PS200 ou PS600 pour Siden, elle a l'avantage de délivrer 5V et 12V et de sortir sur un petit connecteur Le bus Extralink (I2C + 5V, 12V et reset), il n'y a pas d'alimentation spécifique pour les STKxxx



Modules Extralink : XA-Ps200 +Module compteur 10Mz+ Module 4 relais + Module XAMEGA32

Sur le site Mcselec il y a d'autres propositions de cartes de compilation, cartes que nous n'avons pas essayés.

## L'IDE BASCOM

```

46 Dim Minute4(9) As Eram Byte , Seconde4(9) As Eram Word , Agitation4(9) As Eram Byte
47 Dim Minute5(9) As Eram Byte , Seconde5(9) As Eram Word , Agitation5(9) As Eram Byte
48 Dim Minute6(9) As Eram Byte , Seconde6(9) As Eram Word , Agitation6(9) As Eram Byte
49 Dim Pas_init_x As Eram Byte , Pas_init_z As Eram Byte
50 Dim Premier_bac As Eram Byte , Ecart_bac As Eram Word , Profondeur_bac As Eram Word
51 Dim Sechage(6) As Eram Byte , Temperature(6) As Eram Byte
52 Dim Flag_de_memoire As Eram Byte
53 Dim Temps_sechage(6) As Eram Byte
54 Dim Val_Temp As String * 4
55 Dim Btemps_sechage As Byte
56 Dim Minute(9) As Byte , Seconde(9) As Word , Agitation(9) As Byte , Decoloration(9) As Byte
57 Dim Sommetemps As Byte
58 Dim Bsechage As Byte , Blemper As Byte
59 Dim Bac As Byte
60 Dim Prog As Byte , Choix As Byte , J As Byte
61 Dim Valeur_string As String * 2
62 Dim Jbyte As Byte , Kword As Word , Bbyte As Byte , Dummy As Byte , Lbyte As Byte
63 Dim Valeur As Long 'Value.
64 Dim Btere(20) As Byte
65 Dim Statusmodule As Byte
66 Dim Adresse As Byte , Longueur As Byte , Commande As Byte
67 Dim Adresse_fonction As Byte
68 Dim Arg1 As Long , Arg2 As Long , Arg3 As Long , Arg4 As Long , Arg5 As Long
69 Dim Dsi As Byte , Sens As Byte , Periode As Word
70 Dim Rampe As Byte , Tacc As Byte
71 Dim Course As Long , Courselu As Long , Courselu1 As Long
72 Dim Longueur_écrit As Byte
73 Dim Message_tableau(40) As String * 1
74 Dim Vartest As Word
75 Dim Temps_seconde As Integer , Temps_timer As Long
76 Dim Y As Byte
77 Dim Nb_test As Byte
78
79 '---[les constantes]-----
80 Const Modif = 100
81 Const Menu = 200
82 Const Moteur_z = 20
83 Const Moteur_x = 40
84 Const Ds1621 = 72
85 Periode = 2000
86 Wait 1
87 'le petit pb d'écriture de l'adresse 0 en eeprom
88 Bac = 100
89 If Meronyzero <> Bac Then
90 Meronyzero = Bac
91 End If

```

IDE ou Integrated Development Environment : éditeur, programmeur, simulateur, compilateur, émulateur terminal

A partir de la version 1.11.9.0 L'IDE est multilingue, nous y reviendrons plus tard, mais il faut savoir que lors de l'installation la langue maternelle est retenue.

Si ce n'est pas le cas ouvrir l'onglet Options→environnement→IDE→langage et choisir le votre !  
La traduction n'est pas parfaite, car nous avons voulu respecter les raccourcis clavier et l'emplacement dévolu ! Nous allons essayer de nous améliorer en écoutant vos remarques.

Voici le descriptif des menus.

Ils existent des raccourcies claviers voir leur descriptions dans les menus, ils respectent la version anglaise.

➤ **Fichier**

- Nouveau pour démarrer un nouveau programme
- Ouvrir pour ouvrir un programme existant
- Fermer pour fermer un programme avec un enregistrement ou non
- Sauvegarder pour enregistrer le programme
- Sauvegarder comme pour sauvegarder sous un autre nom
- Aperçu avant impression (en noir et blanc)
- Imprimer en noir et blanc, pour avoir les couleur il est possible de sortir le fichier .Bas en fichier Rtf (voir outils)
- Quitter pour quitter l'IDE





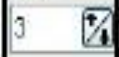




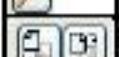


➤ **Editer**

- Défaire annule la dernière frappe
- Refaire restaure la dernière frappe
- Couper un bloc surligner
- Copier un bloc surligner
- Coller un bloc  
-----
- Trouver un mot
- Trouver le suivant
- Remplacer le mot par un autre
- Aller (au numéro de ligne)  
-----
- Echanger les signets, pour y accéder rapidement, dans les grands programmes c'est très utile. Ils sont sauvés à la fermeture de Bascom. Un petit signe s'inscrit en face de la ligne marquée.
- Aller au Signet permettent de noter des lignes et de supprimer les signets en redonnant la valeur du signet à la ligne de ce signet.  
-----
- Indenter un bloc
- supprimer l'indentation
- Marquer /demarquer un bloc pour mettre tout un bloc en REM

- Insérer Char. Ascii permet d'insérer un caractère Ascii (voir explications tome 1)

➤ **Vue**

- Plan des broches suivant le micro-contrôleur, soit défini par \$regfile soit en suivant l'onglet Options → Compiler→chip
  - c'est un outil très utile, il présente le µP sous ses différents packages, les différents usages des broches (cliquer sur les broches pour les connaître)
  - un résumé du µP en cliquant sur le boîtier lui-même.
  - on peut rechercher les broches par leur nom :Fenêtre rechercher
  - chercher un autre µP en cliquant sur Chercher chip puis sur Search
- Visualiseur PDF permet de voir les data-sheets des µP ou l'aide Bascom PDF par ordre alphabétique. Par ordre d'icône en haut et de gauche à droite:

	ouvrir un fichier PDF
	copier une partie du texte que vous aurez mis en surbrillance dans le presse-papier
	première page
	page précédente
	la fenêtre indique le numéro de page courante, on peut aussi aller à une page choisie.
	page suivante
	dernière page
	Trouver du texte dans le fichier PDF
	Zoom +
	Zoom -
	tourne les pages vers la gauche ou la droite ?.
	<ul style="list-style-type: none"> <li>■ Imprime.</li> </ul>

Voir l'onglet options →environnement →PDF pour d'autre information sur l'aide PDF.

- Fenêtre des erreurs, affiche ou cache la fenêtre des erreurs

### ➤ **Programme**

- Compiler le compilateur créera les fichiers défini dans Options→compiler →sortie dans un premier temps il faut savoir que ERR, DBG, BIN sont obligatoires , mais nous vous conseillons de laisser les choix par défaut.
  - Contrôler la syntaxe recherche les erreurs sans créer de fichier .err
  - Montrer le résultat cet onglet va permettre de voir les informations concernant la compilation, les registres utilisés, les paramètres de dates et heures, du Lcd, des piles des variables déclarées mais non utilisées etc..
- 
- Simulation pour essayer le programme avec le simulateur
  - Envoyer au chip pour programmer la flash EEPROM
    - programme
    - programme manuel
  - Reset chip pour relancer le programme

### ➤ **Outils**

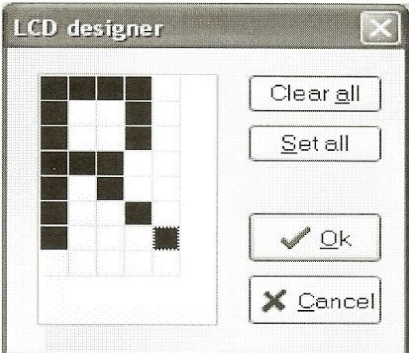
On attaque un gros morceau, mais que le lecteur se rassure, les outils ne sont pas utilisables à chaque instant.

- Terminal Emulateur

Avec cette option vous pouvez communiquer via l'interface RS-232 vers et depuis le microcontrôleur.

- les données envoyées et reçues apparaissent dans la même fenêtre.
- vous devez utiliser la même vitesse de transmission (le baud rate)
- on retrouve le réglage de la vitesse de transmission dans le fichier « [Report](#) »
- le focus DOIT être sur cette fenêtre pour voir les données. Prévoir un délai ( wait 5 par exemple) pour vous laisser le temps de passer le focus à cette fenêtre après une compilation/ transfert.

- LCD designer

	<p>La matrice LCD est de 7X5, la ligne du bas est réservé pour le curseur, mais peut être quand même utilisée. On clique sur les carrés blancs pour les rendre noirs et inversement. Clear All remet tout à blanc, Set all tout à noir.</p> <p>En cliquant sur OK, une ligne de programme : apparaît: par exemple un « é » Deflcdchar ?, 2 , 4 , 14 , 17 , 17 , 14 , 8 , 6 Vous devez remplacer le « ? » par une valeur allant de 0 à 7 Les octets qui suivent le « ? » correspondent au points noir et blancs.</p>
---	---

Voir l'exemple :[une jauge pour ULM](#)

---

- Lib manager

Permet d'écrire ou de modifier des librairies, les librairies fournies avec Bascom sont très complètes et très complexes, il est nécessaire d'avoir une connaissance approfondie de l'assembleur pour « jouer » avec, cela est hors de propos du présent ouvrage.

- Exporter au format RTF le fichier sur lequel on travaille est recopier en fichier xx.RTF ce qui permet de l'imprimer en utilisant un éditeur comme writer de OpenOffice.org qui tient compte des couleurs.
- Convertisseur Graphique
  - Le Convertisseur Graphique permet d'utiliser les images BMP converties en BGF( Bascom graphic files)
  - les images.BGF sont utilisables avec les LCD graphiques
  - Pour utiliser une image cliquer sur le bouton Load
  - l'image peut avoir un maximum de 128 x 240 pixels, si l'image est plus grande elle sera réduite.
  - Les images sont converties en noir et blanc
- Analyseur de pile permet de déterminer la taille optimale des piles soft , voir \$DBG pour une utilisation correcte de cette option.

- EasyTCP/IP

Pour l'instant, cet outils n'est pas documenté dans l'aide Bascom, il permet de faire communiquer le monde internet avec les microcontrôleurs.

Voir Config TCPIP dans le dictionnaire et l'exemple [tcpip.bas](#) et les autres fichiers se trouvant dans ce répertoire.

- Plugin Manager

- quand on ouvre cet outils pour la première fois il n'y a qu'une fenêtre vide. Aller dans le repertoire :

C:\Program Files\MCS Electronics\BASCOM-AVR\Plugins

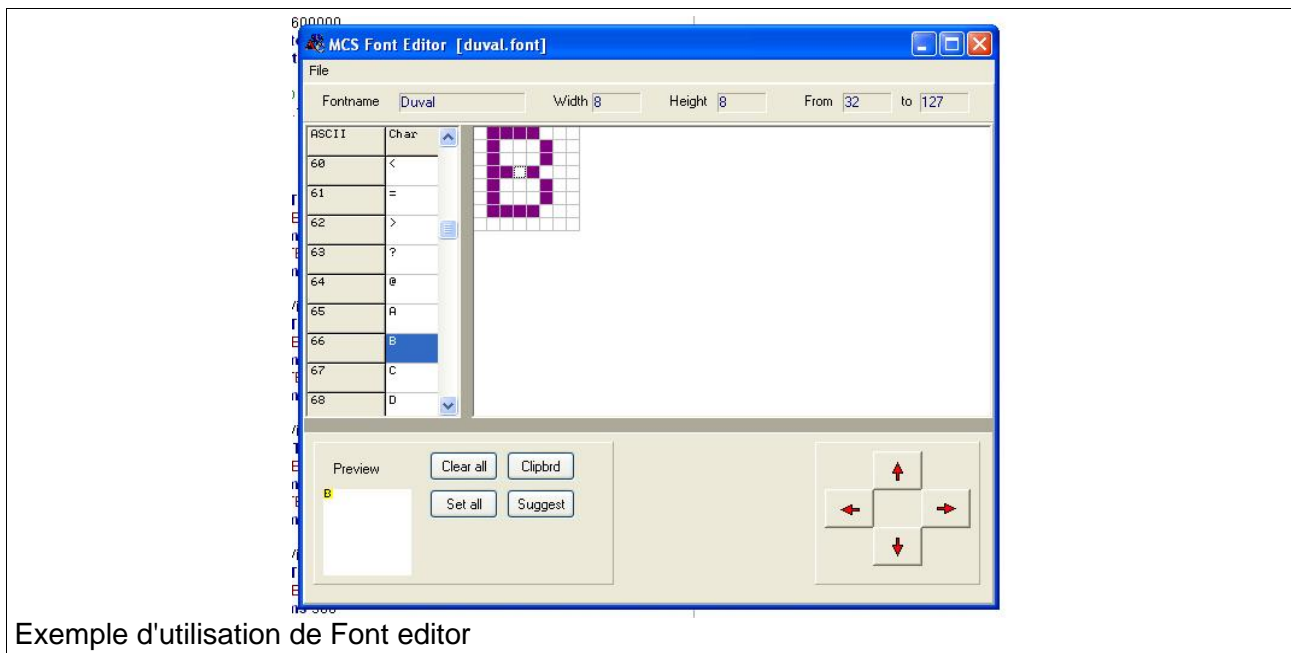
vous ne trouverez qu'un plugin : le Fonteditor utiliser avec la fonction setfont pour les afficheur graphiques ne possédant pas de polices de caractères résidentes.

Le fonteditor est un petit programme proche du LCDdesigner.

On doit copier ce plugin dans le répertoire principal (où se trouve Bascom) pour le voir s'afficher au redémarrage.

Les fichiers .Bpl peuvent avoir une autre signification (ouverts par Winamp par exemple) Il faut supprimer cette ouverture automatique, si le fichier fonteditor.bpl ne s'affiche pas.

- Plugins Permet le lancement du Plugin choisit



### ➤ Options

Nous n'allons pas décrire l'utilisation de tous les réglages, certains parlent d'eux même, d'autres sont redondants avec les directives de compilation, il est préférable, dans ce dernier cas d'utiliser les directives car le programme comporte alors toutes les données nécessaires à sont fonctionnent. Exemple:

```
$regfile = "m32def.dat"
```

```
$crystal = 11059200
```

```
$hwstack = 128
```

```
$swstack = 128
```

```
$framesize = 128
```

```
$baud = 9600
```

#### ○ Compiler

- Chip pour choisir le microcontrôleur (90at2313 par défaut) avec ses valeurs dEEPROM de flash et de Ram, et aussi l'espace laissé pour les piles.
- Sortie affiche les fichiers sorties par défauts pour ce programme, les fichiers :  
.binaire, debug, error, sont obligatoires,  
.report vivement conseillé en cas de problème pour se faire aider par Bascom  
.Hex et .Avr studio suivant la carte de programmation utilisée. Pas nécessaire pour les cartes Siden ou Mcselec ou encore Kanda STK200
- Communication Affiche le baudrate la fréquence du quartz et le taux d'erreur de transmission pour la paire baudrate # fréquence, par exemple un quartz de 8 Mhz

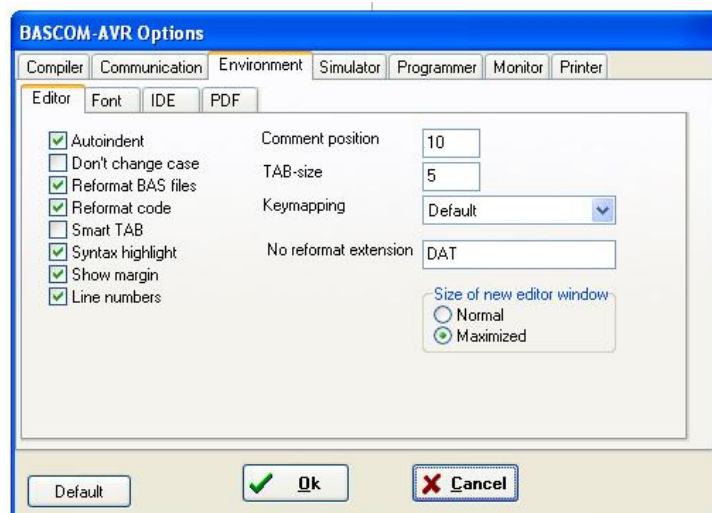
présente un taux d'erreur de 0.16% pour 9600 baud et un taux d'erreur de 0% pour une fréquence de 11059200 Hz (nombre magique pour la plupart des fréquences ! Est-ce aussi pour cela que nos fournisseurs habituels le vendent si cher?)

- I2C permet d'assigner les broches qui seront affectées aux ports I2C et 1wire (si utilisés) et au port SPI pour la programmation.
  - LCD (en mode texte) permet d'assigner les broches qui seront affectées aux ports LCD et la manière de l'utiliser. Non-utilisé si configuré par soft (recommandé)
  - Divers pour s'amuser...
- Communication
- Permet de régler les paramètres du port série (le cas échéant):



Une nouvelle option est apparue Keep « terminal emulator open » qui peut nous aider dans bien des cas.

- Environnement
- tout ceci est fait pour votre confort et une utilisation agréable du compilateur



Il y a 4 onglets dans celui-ci :

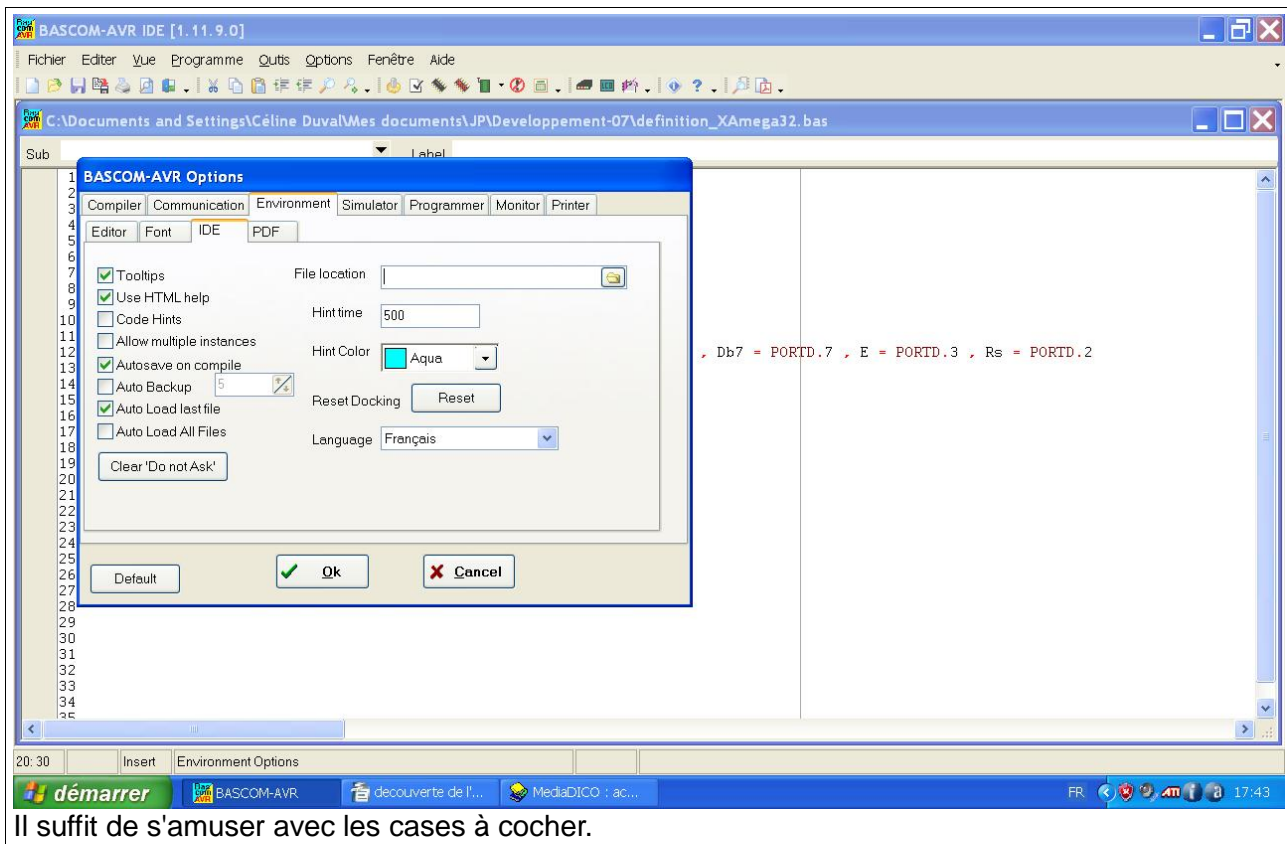
- Editeur

autoindent	Quand on tape return, le curseur est placé à la ligne suivante à la même colonne.
Don't change case	Par défaut, chaque mot commence par une majuscule sauf si cette case est cochée
Reformat BAS file	Pour supprimer les espaces créés par d'autres éditeurs
Reformat code	Corrige les petites erreurs de frappes endif= End If par exemple
Smart tab	Place une tabulation qui suit la précédente
Syntax highlighting	Met et surbillance les fonctions Bascom
Show margin	Place une marge sur le côté droit de l'éditeur
Comment position	La position où se place les commentaires
Tab Size	Taille d'une tabulation
Key mapping	? je n'ai vu aucune différence entre les modes proposés sur mon clavier AZERTY
No reformat	Se rapporte au fichiers qui ne doivent pas être reformatés

- Font

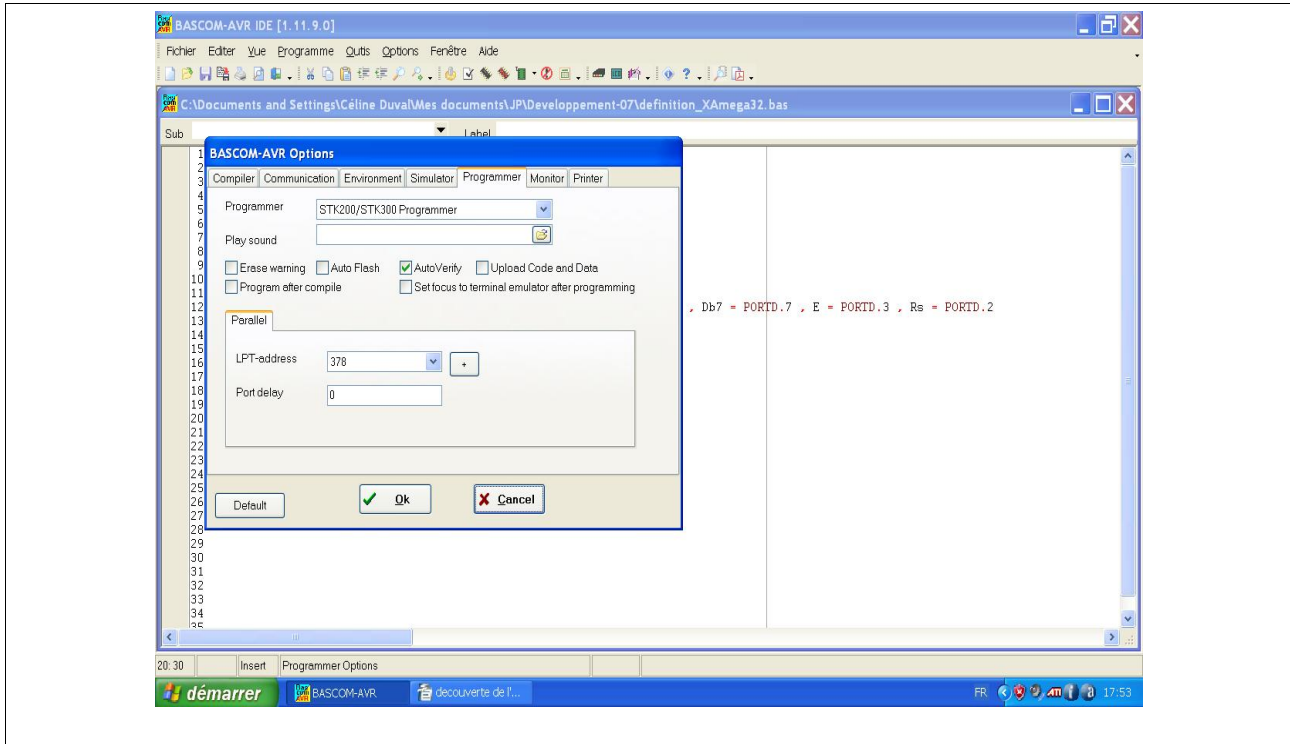
En cliquant sur la fenêtre Editor font on a accès aux choix des police de caractères. Il faut savoir que les polices type courier sont à écartement fixe (comme les afficheurs LCD)

- IDE cet onglet permet de regler les paramètres de travail de L'IDE



- PDF cet onglet règle la façon dont l'IDE gère les fichiers PDF.
- Simulateur permet de choisir le simulateur utilisé (par défaut le simulateur BASCOM) nous reviendrons sur l'usage du simulateur au chapitre qui lui est consacré.

- Programmeur (programmer) il s'agit du type de carte et du système de transfert du programme compilé vers le micro-ordinateur.



### Les différents types de programmeurs

- Programmeur par le port parallèle : AVR ISP Programmer, STK200, STK300, SIDENA, Sample Electronics, MCS Universal Interface Programmer

Le AVRISP programmer est basé sur la note d'application AVR910.ASM ce vieux ICP910 ne supporte pas les Microcontrôleur de type MEGA et Tiny, seule une version modifiée de AVR910 est universelle.

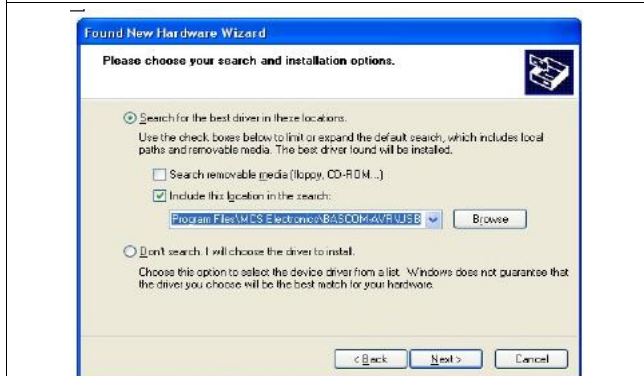
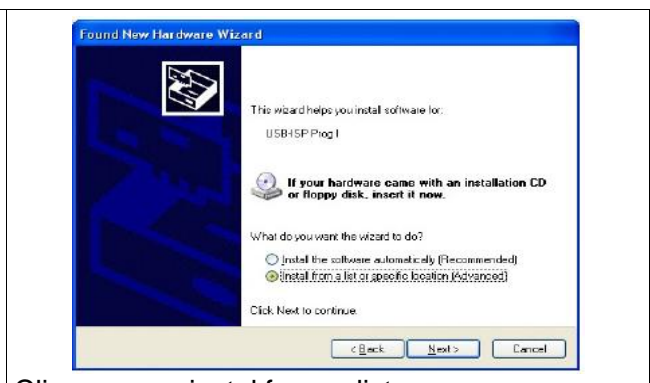
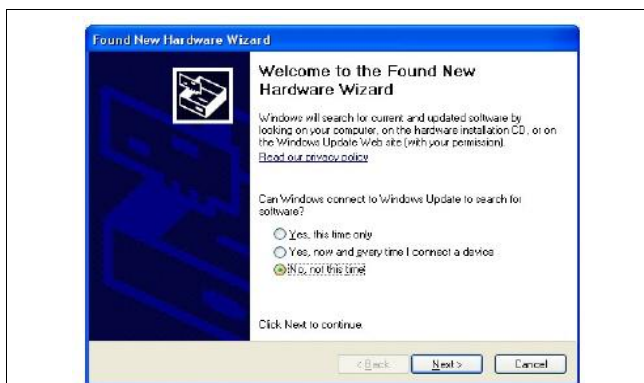
Le nouvel AVRISP de chez Atmel qui peut utiliser AVR Studio, n'est pas compatible ! Vous devez utiliser le protocole STK500.

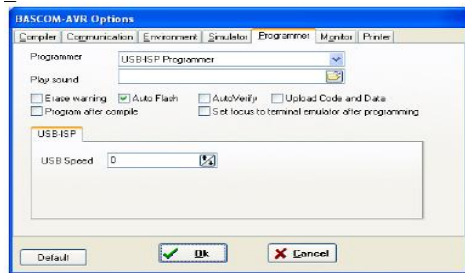
Il est possible de modifier le baud rate dans le bascavr.ini dans le windows directory

Ajouter la section [AVRISP]

puis: COM=19200,n,8,1

- Programmeur par le port série : KITSRUS K122, PG302 (Iguana labs), STK500 (serial), Lawicel Bootloader
- Programmeur par le port USB : MCS a choisi l'USB-ISP programmer from EMBUD, Proggy, STK500(USB) ces programmeurs nécessitent des drivers USB, par exemple pour le programmeur MCS:





Et voilà !

- MCS bootloader

Le Chargeur de « boot » est un programme qui doit être utilisé avec l'exemple utilisant [\\$Loader](#)

Il utilise le protocole X-modem Checksum pour charger le fichier Bin, ce qui est très rapide.

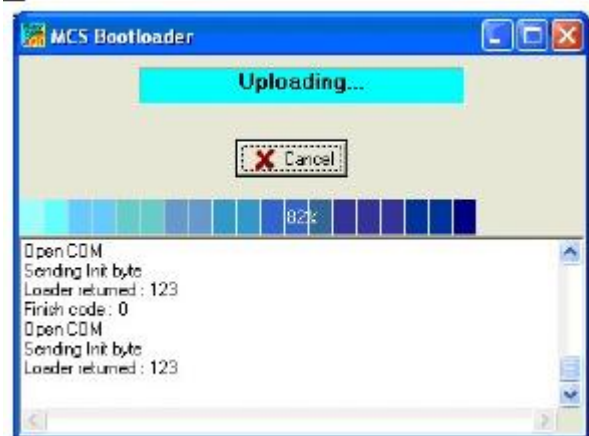
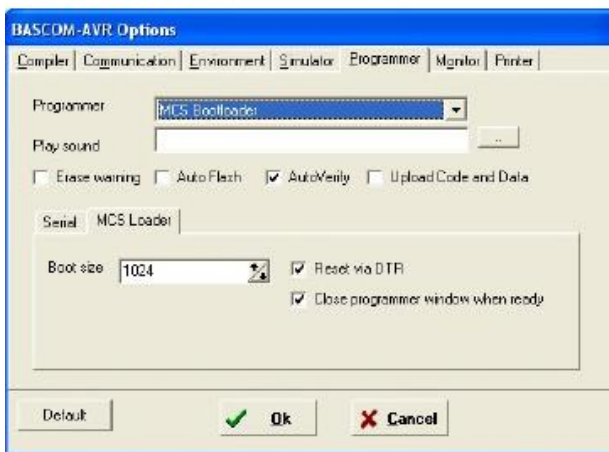
Il autorise le chargement de la partie programme (Flash) ou de la partie EEPROM

Le bootloader envoie un bit d'une valeur de 123 au AVR boot loader, celui-ci charge alors le le boot loader ou saute au vecteur reset (0000) pour charger le programme.

Si le bit vaut 124 il charge l'EEPROM.

Quand on choisit un fichier BIN on charge la flash, pour un fichier EEP l'EEPROM est chargée.

le bootloader a des options spécifiques:



La taille du boot est de 1024, attention vous perdez 1Ko de programme.

Le boot loader démarre quand le chip est reseté, aussi vous devez le reseter après avoir taper F4(Program)

Si la ligne DTR est connecté via un MAX232 vous pouvez choisir un reset automatique: choisir Reset via DTR

la second option permet de fermer la fenetre en fin de programmation.

Après avoir taper F4 la fenetre suivante apparaît : -->

Comme vous pouvez voir le Boot loader envoie un bit 123.

vous devez reseter et voir le Bit 123 retourné ce qui veut dire que tout c'est bien passé.

Ensuite il lancera le chargement et affichera une barre de progression, puis un code de Fin « 0 » Tout autre code démontre une erreur.

- Moniteur: Règle les paramètres du moniteur RS232
- Imprimante: Règle les paramètres de l'imprimante

Option	Description
Font	Police d'impression
setup	Pour changer l'imprimante à utiliser
Color	Pour imprimer en couleur
Print header	Pour imprimer un haut de page
Line number	Pour imprimer le numéro de ligne
Syntax	Pour utiliser la même syntaxe que l'éditeur (conseillé)
Margin (s)	Pour régler les marges

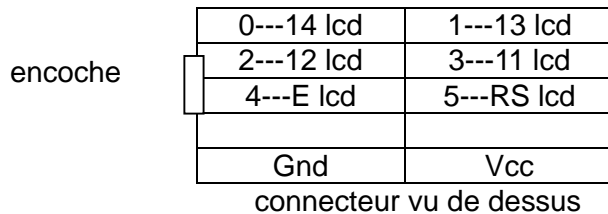
### ➤ Fenêtre

- Cascade, quadrillage, arranger les icônes, tout réduire  
sont les options de présentations des divers programmes ouverts.
- Liste des programmes ouverts  
un coche indique le programme en cours.

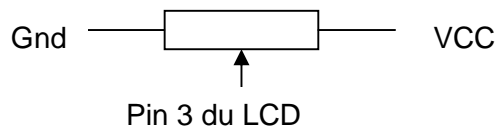
## L'aventure commence...

Il est bon de se procurer un afficheur 2 X 16 caractères.

Une petite plaquette de test (des bandes au pas de 2.54mm font très bien l'affaire), un connecteur HE10 femelle 2 x 5, nous allons donc le connecter à la carte de développement, en suivant le schéma ci-dessus :



Sur le LCD les broches data (0 à 3) 10-9-8-7 et (R/W) 5 sont mises à la masse. On connecte le contraste à un potentiomètre de 10K $\Omega$  de la façon suivante :



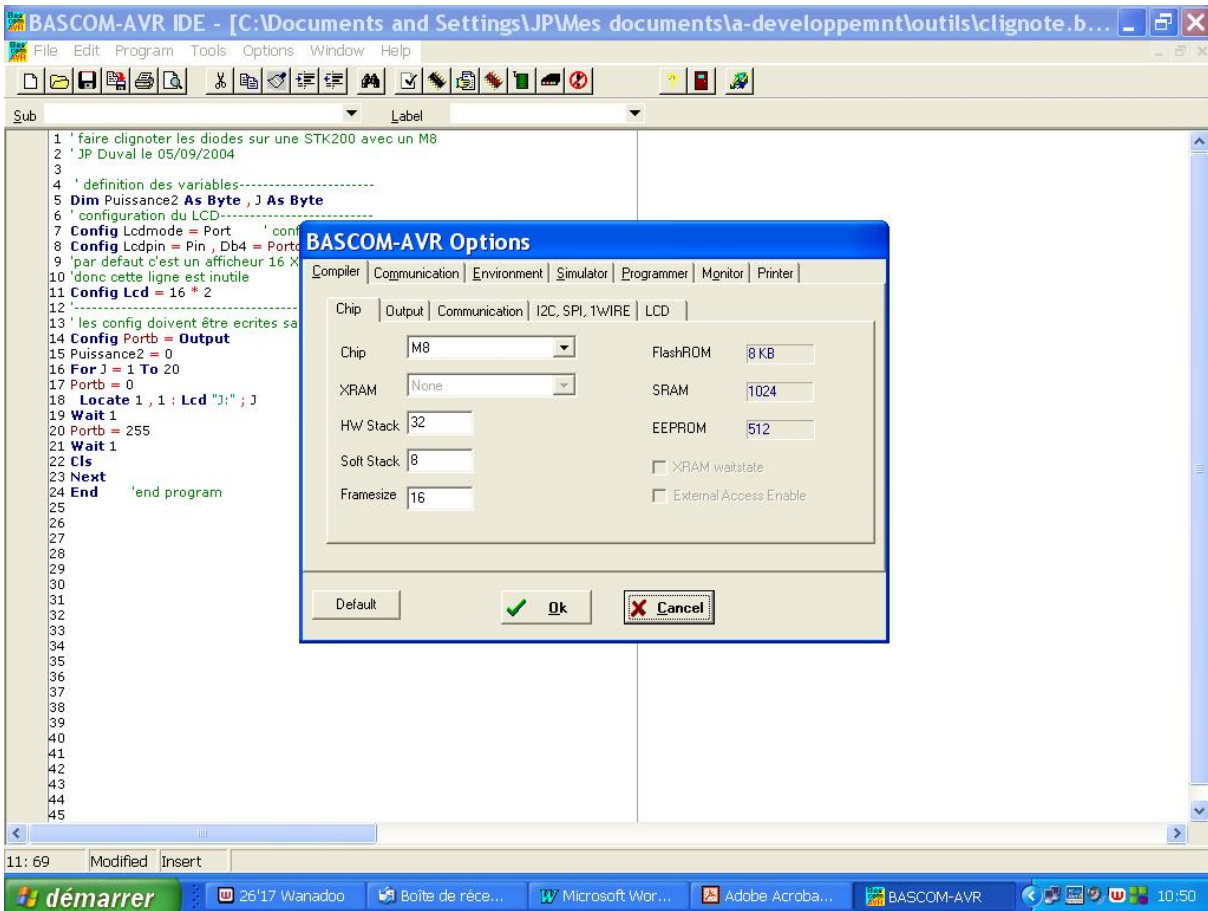
L'auteur a fait un vilain bricolage ....bien utile, il n'utilise que 6 broches de  $\mu$ P et le potentiomètre est sur la carte interface.



## Préparation de Bascom

Il faut préparer Bascom pour utiliser le  $\mu$ -contrôleur choisi et la configuration LCD proposée :

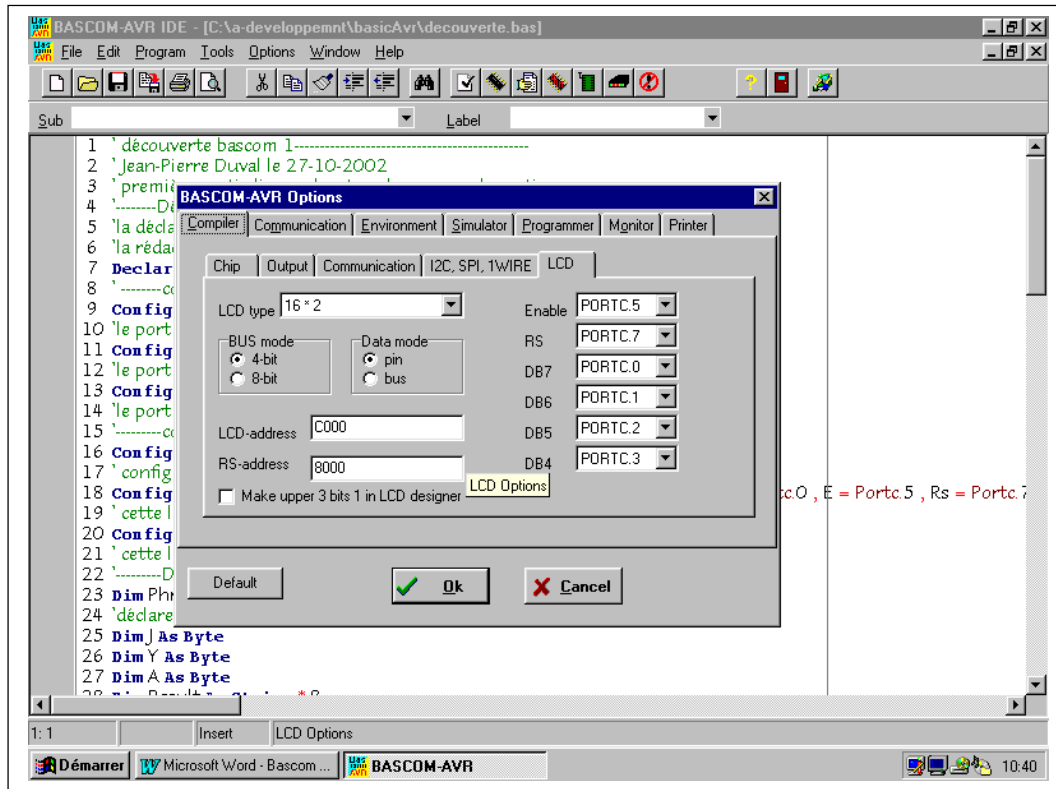
Pour ce faire, on va ouvrir le menu **OPTIONS** puis **Compiler**, puis on va choisir le  $\mu$ -contrôleur dans le sous-menu **Chip**, on en profitera pour configurer les ports du **LCD**.



OPTION	DESCRIPTION
<b>Chip</b>	le choix du $\mu$ P, à chaque $\mu$ P est associé un fichier xx.DAT. A chaque évolution de $\mu$ P ces fichiers sont mis à jour.
XRAM	Si nécessaire choisir la taille de la RAM statique.
<b>HW Stack</b>	La Pile Harware = 2 bytes à chaque GOSUB 2 subs imbriquées utilisent (2*2 bytes) 1 interruption = 32 Bytes.
<b>Soft Stack</b>	La pile soft chaque variable LOCAL = 2 bytes, chaque variable passée dans une SUB =2 bytes
<b>Frame size</b>	Taille de la "Frame" chaque variable locale, chaque String, chaque conversion STR( ), VAL( ), INPUT consomme de la frame le maximum est de 50
XRAM waitstate	Cocher pour insérer un état d'attente pour la RAM
External Access Enable	Cocher pour permettre l'accès externe, EX: le 8515 peut utiliser le portA ou le port C pour contrôler la RAM
<b>Default</b>	Cliquer pour lancer les options de base de chaque nouveau projet

Les options en gras sont importantes. (A surveiller les éventuelles dépassements des piles et frame dans **SIMULATE**\mathbf{\mu}P et **Stack analyser** du menu Options).

## Le LCD



OPTION	DESCRIPTION
<b>LCD type</b>	Le type d'afficheur utilisé
<b>Bus mode</b>	L'afficheur peut être connecté sur les lignes data (ex : mémoire PortA), il travaille en Bus mode, dans ce cas il peut travailler sur 4bits ou sur 8bits (DB4-DB7)
<b>Data mode</b>	En Mode Pin chacune des broches du µP peut être sélectionnée individuellement, en bus mode c'est le bus de donnée.
<b>LCD adress</b>	En mode Bus l'adresse de Enable doit être spécifiée :&hCOOO pour le STK200
<b>RS Adress</b>	Idem pour RS :&h8000 pour le STK200
<b>Enable</b>	En mode Pin le port du µP branché à Enable doit être spécifié
<b>RS</b>	En mode Pin le port du µP branché à RS doit être spécifié
<b>DB4-DB7</b>	En mode Pin les ports du µP branchés à DB4-DB7 doivent être spécifiés
<b>Make upper 3bits 1 in LCD designer</b>	Quelques afficheurs demandent que lors de l'affichage des caractères définis par l'utilisateur, les 3 bits supérieurs soient mis à 1. Non coché par défaut.

Les options en gras sont importantes.

L'instruction **Config LCD** est prioritaire sur le réglage ci-dessus.

## Rédaction du programme

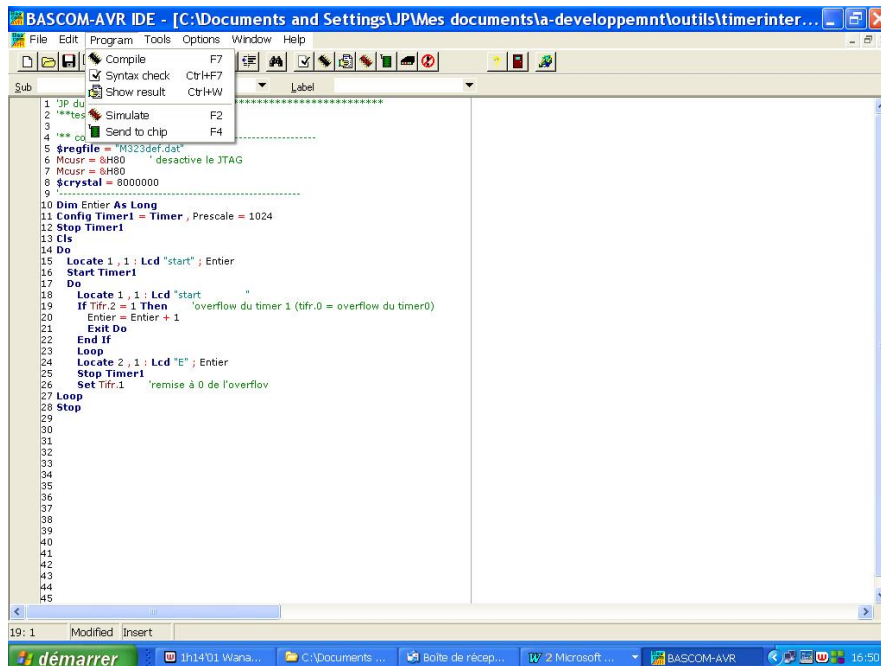
' faire clignoter les diodes sur une STK200 avec un M8 et afficher le nombre de fois sur l'afficheur.  
' JP Duval le 05/09/2004

```
' ----definition des variables-----
Dim J As Byte
'----les config -----
' les config doivent être écrites sans retour à la ligne.
Config Portb = Output
' configuration du LCD-----
Config Lcdmode = Port      ' config 4 bit
Config Lcdpin = Pin , Db4 = Portc.3 , Db5 = Portc.2 , Db6 = Portc.1 , Db7 = Portc.0 , E = Portc.4 , Rs =
Portc.5
'par défaut c'est un afficheur 16 X 2 qui est utilisé dans Bascom
'donc cette ligne est inutile
Config Lcd = 16 * 2

For J = 1 To 255
    Portb = 0
    Locate 1 , 1 : Lcd "J:" ; J
    Wait 1
    Portb = 255
    Wait 1
    Cls
Next
End
```

## Enregistrement, Compilation, Contrôle.

On commence par une sauvegarde, en tapant **Ctrl+S** ou en choisissant l'icône représentant une disquette ; sauf si la coche autosave dans environment /IDE a été utilisé puis, dans le menu **Program**



## F7 Compilation

Si le compilateur ne trouve pas d'erreur on passe au choix suivant, sinon en tapant sur l'erreur, la (es) ligne(s) en faute est surlignée.

Le compilateur génère les fichiers demandés dans le menu **Options\Compiler\ Output**, parmi ces fichiers nous trouverons celui ou ceux qui seront transmis au  $\mu$ -contrôleur.

## CTRL+F7 Syntax Check

Permet de contrôler l'ensemble du programme sans le compiler.

## CTRL+W Show result

Donne le résultat de la compilation, certaines notions expliquées dans le tableau ci-dessous sont un peu ardues, le lecteur peut passer et revenir plus tard.

Informations données dans le rapport :

INFO	DESCRIPTION
Report	Nom du programme
Date et heure	La date de compilation

Compiler	La version du compilateur
Processor	Le $\mu$ -processeur utilisé
SRAM	Taille de la SRAM (RAM interne)
EEPROM	Taille de EEPROM
<b>INFO</b>	<b>DESCRIPTION</b>
<b>ROMSIZE</b>	Taille de la FLASH ROM
ROMIMAGE	Taille du programme compiler
BAUD	Vitesse de transfert série UART
<b>XTAL</b>	Fréquence du Quartz
BAUD error	% d'erreur de la transmission
XRAM	Taille de la mémoire extérieure si disponible
Stack start	L'adresse-mémoire où démarre le Pointeur de pile. Hardware. Ce pointeur descend
S-Stacksize	La taille de la pile « soft »
S-Stackstart	L'adresse-mémoire où démarre le pointeur de pile soft. Ce pointeur descend.
Framesize	La taille de la zone réservée aux variables
Framestart	L'adresse-mémoire où démarre la zone « Frame »
Space left	Espace restant en décimal pour la pile
LCD adress	Si le LCD est en mode Bus , adresse de l'Enable
LCD RS	« « « « adresse de RS
LCD Mode	Mode de fonctionnement du LCD 8 ou 4 bits
LCD DB7-DB4	Les broches où sont branchés les ports LCD en mode Pin
LCD E	La broche où est branché le port Enable du LCD en mode Pin
LCD RS	La broche où est branché le port RS du LCD en mode Pin
Variables	Le nom des variables et leurs adresses
Constants	Le nom des constantes et leurs valeurs

### Quelques points importants :

- La vitesse du quartz, elle reprend l'information donnée dans le menu **Options\Compiler\Communication** ; elle doit, bien entendu, correspondre à la vitesse du Quartz utilisé.
  - La taille du programme compilé.
  - La taille des piles Soft et hard que l'on retrouve dans **Options\Compiler\ chip**
- A noter dans celui-ci les « flag<sup>1</sup> » indiquant les valeurs à donner en fonction du nombre de variables, des locales, de gosub...
- Les warning et remarques de fin de rapport.
  - L'espace restant pour la pile.

## Simulateur

Pour lancer le simulateur (voir le chapitre qui lui est consacré)

## transfert

Pour transférer le programme vers le  $\mu$ P (voir le chapitre qui lui est consacré)

<sup>1</sup> Aide qui s'affiche spontanément quand le pointeur s'approche de la case.

## LE SIMULATEUR

Dans le menu **options\simulator**

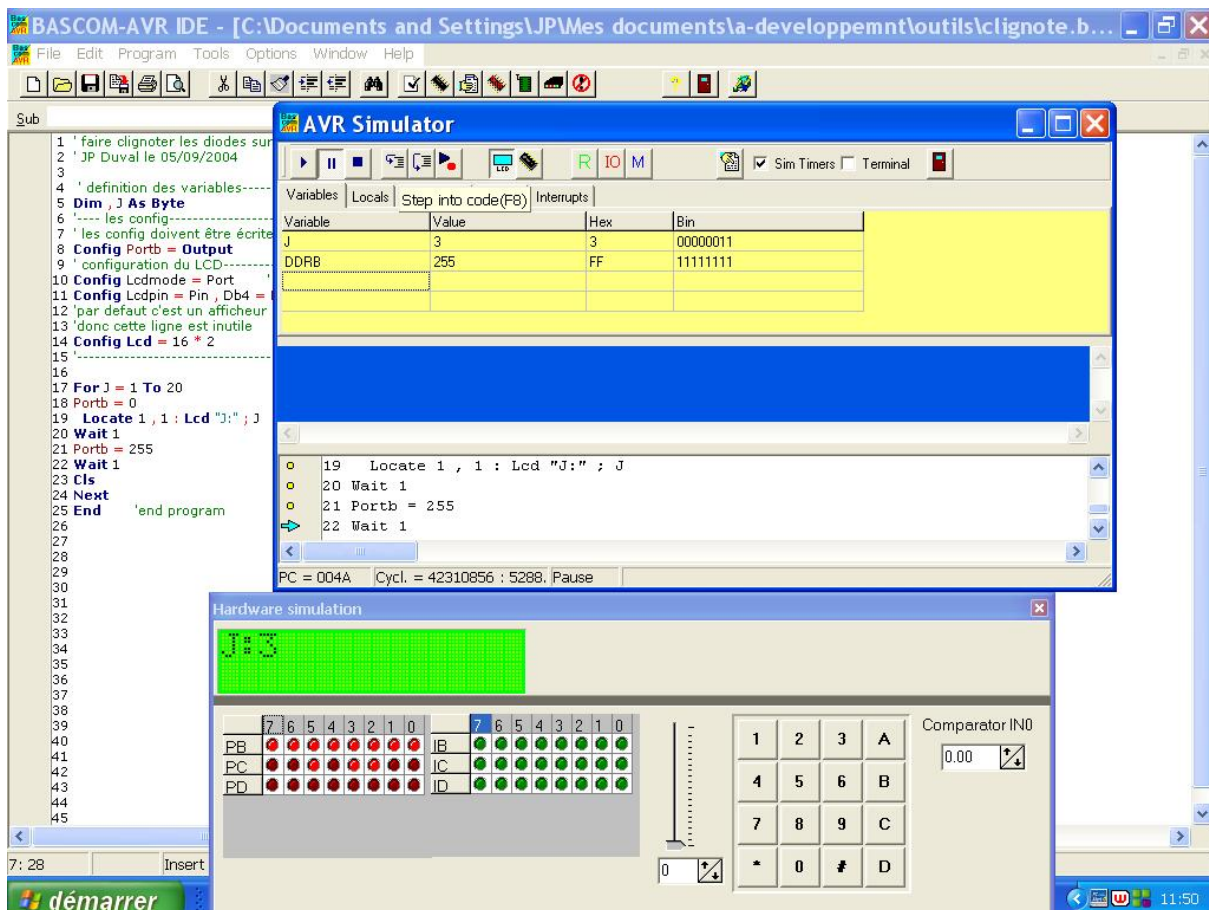
Il est possible d'utiliser un autre simulateur que le Bascom, dans ce cas il suffit de donner le titre du programme à utiliser et son emplacement sur le disque.

Dans ce menu, une case à cocher est intéressante pour la mise au point des programmes: Elle permet de lancer le simulateur dans la foulée du compilateur !

Le programme est compilé, sans erreur ?

On peut simuler le fonctionnement avec le menu **Simulateur** ou avec la touche **F2** ou encore avec la petite icône.

Le Simulateur est un programme à lui seul :



## Description du simulateur

### La barre d'outils

De gauche à droite, trois outils classiques : Démarrage (F5), Pause, Arrêt.

Puis :

- **PAS-à-PAS (F8)** passe les REM et exécute chaque ligne de code.
- **Exécution des SUB (Shift+F8)** : on exécute la Sub en entier.
- **Run To** : exécute jusqu'à la ligne exécutable.
- **LCD** permet d'afficher la fenêtre Hardware simulation
- **Le µP** permet de lancer une simulation hard (l'auteur n'a jamais utilisé cette option)

Dans l'exemple ci-dessus, on vient de passer l'instruction `Portb=255`

- Les Poids PB7---PB0 sont à 1 (ce qui éteint les diodes sur la STK200 !)
- PC représente les bits de contrôle du LCD (dans notre programme)
- IB, IC, ID... correspondent aux input, non utilisé ici
- Le curseur peut être utilisé pour simuler l'ADC (avec les différents canaux)
- Le clavier pour simuler GETKBD
- Enfin avec le comparateur on, peut spécifier le niveau logique de IN0

On travaille en pas-à-pas pour suivre le déroulement du programme. ( 4° icône)

- Variable permet de suivre l'état des variables utilisées, mais aussi des registres (ici DDRB)
- **R** permet de visualiser en temps « Reel » les registres
- **I/O** idem pour les registres d'entrée/sortie
- **M** idem pour la mémoire Sram et EEPROM.
- Une touche permet de rafraîchir les variables.
- Une autre pour simuler les timers (pas les Wait !)
- Enfin une dernière est utilisée pour activer un simulateur réel avec le logiciel Basmon.bas (voir les samples et l'aide Bascom)

## **La barre d'onglets**

De gauche à droite :

- **Variables** permet de visualiser les variables, en double-cliquant sur une case, le choix de variables disponibles s'affiche. Il est possible de changer la valeur d'une variable dans la colonne « value ». **CTRL+DEL** Pour effacer un choix
- **Locals** permet de visualiser les variables locales des Sub ou Fonctions, fonctionne comme précédemment.
- **Watch** pour entrer une expression qui sera évaluée pendant la simulation, quand l'expression sera vraie, le programme stoppera.
- **µP** permet de visualiser et de contrôler le registre d'état (SREG), la taille des piles, de la FRAME et leurs dépassements éventuels.
- **Interrupts** montre les sources d'interruptions, et permet de les exécuter.

Notre programme tourne, nous le transférons dans le micro-contrôleur :

## Preparation au transfert vers le $\mu$ p

Suivant la carte de programmation choisie (parallèle ou USB ou Bootloader) ici les exemples démontre le port // pour une carte STK200 ou un module Sidena ATMEGA32

- Le microprocesseur doit être installé sur la carte de transfert, cela va sans dire.
- Le câble de programmation (connecteur d'imprimante sur le Dongle d'un côté et HE10 de l'autre) doit être connecté.
- Une imprimante doit être déclarée dans le Logiciel ! ( voir installation Bascom-IDE, ne cherchez pas à comprendre C'EST COMME CA ! j'ai cherché longtemps....)

Mon Pc de programmation est équipé d'une 2<sup>eme</sup> carte parallèle pour plus de facilité, un switch d'imprimante fait l'affaire aussi.

- Connecter l'alimentation.

Et maintenant ...

## Transfert

Avant de transférer le programme vers le  $\mu$ -contrôleur, il faut configurer le programmeur avec l'**Option Programmer** comme nous l'avons déjà vu au programme installation Bascom-IDE

dont voici les explications des choix En **gras** les options importantes.

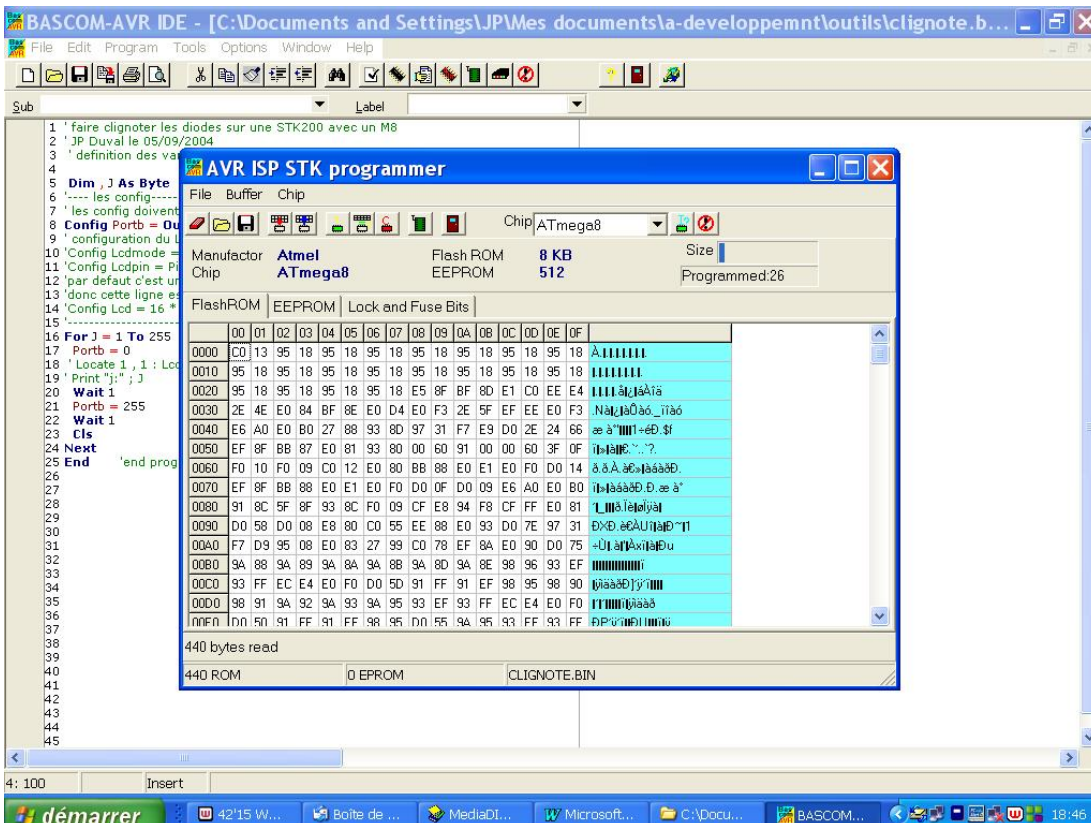
OPTION	Description
<b>Programmer</b>	<b>Choisir un programmeur dans la liste</b>
Play sound	Nom du fichier .WAV à jouer en fin de programmation
Erase warning	Si cocher confirmation de l'effacement
Auto-flash	Si le programmeur le supporte, en tapant F4 le transfert se fera sans passer par la fenêtre de transfert.
Auto-verify	Si le programmeur le supporte le $\mu$ P sera vérifié après transfert.
Upload Code and DATA	Choisir cette option pour transférer le programme en FLASH et en EEPROM.
<b>Port Parallèle</b>	
<b>LP address</b>	<b>Adresse du port parallèle (voir dans le menu démarrer\paramètres\Systeme\gestion des périphériques\propriétés\ressources du PC</b>

OPTION	Description
<b>Port Série</b>	
Comm port	Le n° du port série
STK500.EXE	Le répertoire où se trouve le programme STK500 pour ce programmeur.
<b>Other</b>	
Use HEX	A cocher pour l'utilisation d'un fichier xxx.HEX
Program	Le programme du programmeur si différent du BASCOM
Parameter	Les éventuels paramètres à transmettre.
<b>Universal</b>	
Programmer	Donner le nom de la carte de programmation universelle.(le cas échéant)

On transfère le programme compilé par l'intermédiaire du sous menu **Program \ send to chip** avec le raccourci clavier F4 ou par l'icône de transfert.

Là encore c'est un programme à lui tout seul.

Il est aussi possible de transférer le programme par les outils de transfert proposés avec la carte programmeur. Sinon voir plus haut.



Le menu de ce programme se retrouve comme d'habitude avec BASCOM, dans les icônes de la ligne suivante, en revanche, il n'y a pas de raccourci-clavier.

<b>MENU</b>	<b>Descriptions</b>
<b>FILE</b>	
Exit	Pour sortir
Test	seulement pour le programmeur <b>Sample electronic</b>
<b>BUFFER</b>	
Clear	Effacer le buffer
Load from file	Charger le buffer avec un fichier
Save to file	Sauver le buffer vers un fichier
<b>CHIP</b>	
Identify	Pour identifier le $\mu$ P
Write buffer into Chip	Programme le $\mu$ P vers la ROM ou l'EEPROM
Read Chip code into Buffer	Lit et transmet les data du $\mu$ P dans le buffer.
Blank check	Contrôle si le $\mu$ P est remis à "FF"
<b>Erase</b>	Pour effacer le $\mu$ P
<b>Verify</b>	Vérifie si le contenu du $\mu$ P est identique au buffer
Set lockbits	Ecrit les bits de blocage LB1 ou/et LB2 ces bits ne sont resettés que par un Effacement.
<b>Auto-program</b>	Efface, programme et vérifie le buffer par rapport au $\mu$ P
RCEN	Ecrit un bit pour autorisé l'oscillation interne. Pour les $\mu$ P AVR disposant de cette possibilité.

- Dès l'alimentation de la carte, le clic sur l'icône de programmation ou la touche F4 déclenche une alerte si le  $\mu$ P n'est pas reconnu.
- Le transfert se déclenche par un clic sur l'icône vert (autoprogram chip) qui commence par un effacement de la flash.
- Quand on clique sur l'icône la plus à droite, on relance le programme.
- Quand on double-clique sur « programmed », on efface le compteur de programmation
- L'étude de l'onglet Fuse-bits est approfondi dans le livre 1
- Quand tout est correct, **Verify Ok** s'affiche en bas à gauche.

## Adaptation du programme

Mais le programme risque de mal fonctionner sur le  $\mu$ P :

En effet si le programme présenté plus haut tourne sans problème sur le simulateur, sur la carte STK200, il peut ne pas tourner correctement, les diodes 7 et 6 sont bloquées, une allumée, l'autre éteinte, ou la vitesse d'exécution est trop lente (1seconde = 8 secondes) pourquoi ?

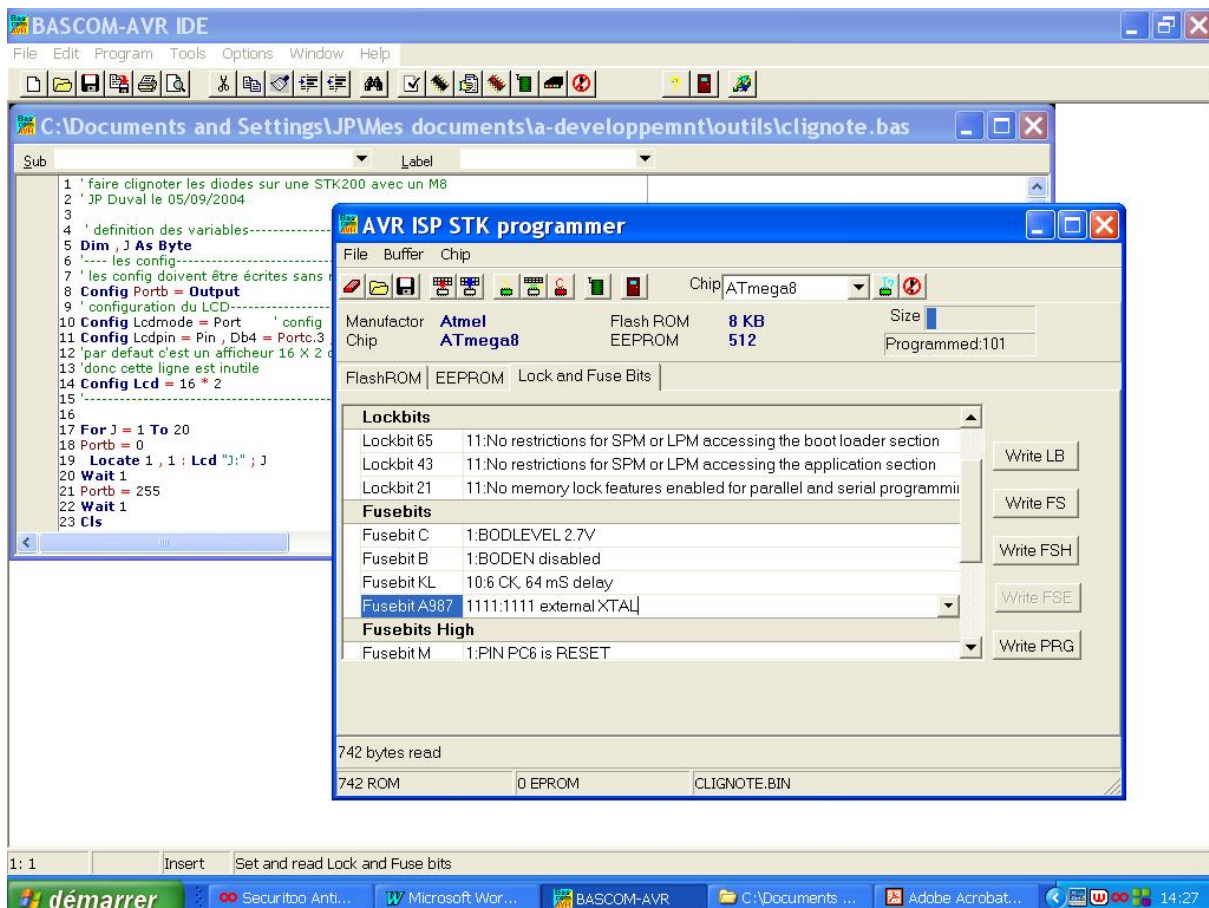
Revenons à la page 3 à la remarque sur les jumpers :

*CP1 Clock ou PortB.6 pour les CI 28 Broches* cela veut dire que si le  $\mu$ P est utilisé avec le quartz comme horloge : Port.B7 et Port.B6 pour l'ATMEGA8. On ne peut pas utiliser ces ports comme des ports I/O (entrée/sortie)

Si on le change de côté, alors notre programme pourra soit tourner correctement soit toujours présenter le même défaut, soit ne plus tourner du tout.

Donc laissons le jumper en Clock et retournons à notre programme de transfert.

En cliquant sur l'onglet Fuse bits on constate que les fuses bits A987 sont réglés sur 1111 :1111 ou sur Internal oscillator 1MHz



nous allons choisir l'option Internal RC oscillator 8MHz pour ces mêmes fuses bits, les écrire : **Write FB**, puis nous pourrons changer notre jumper.

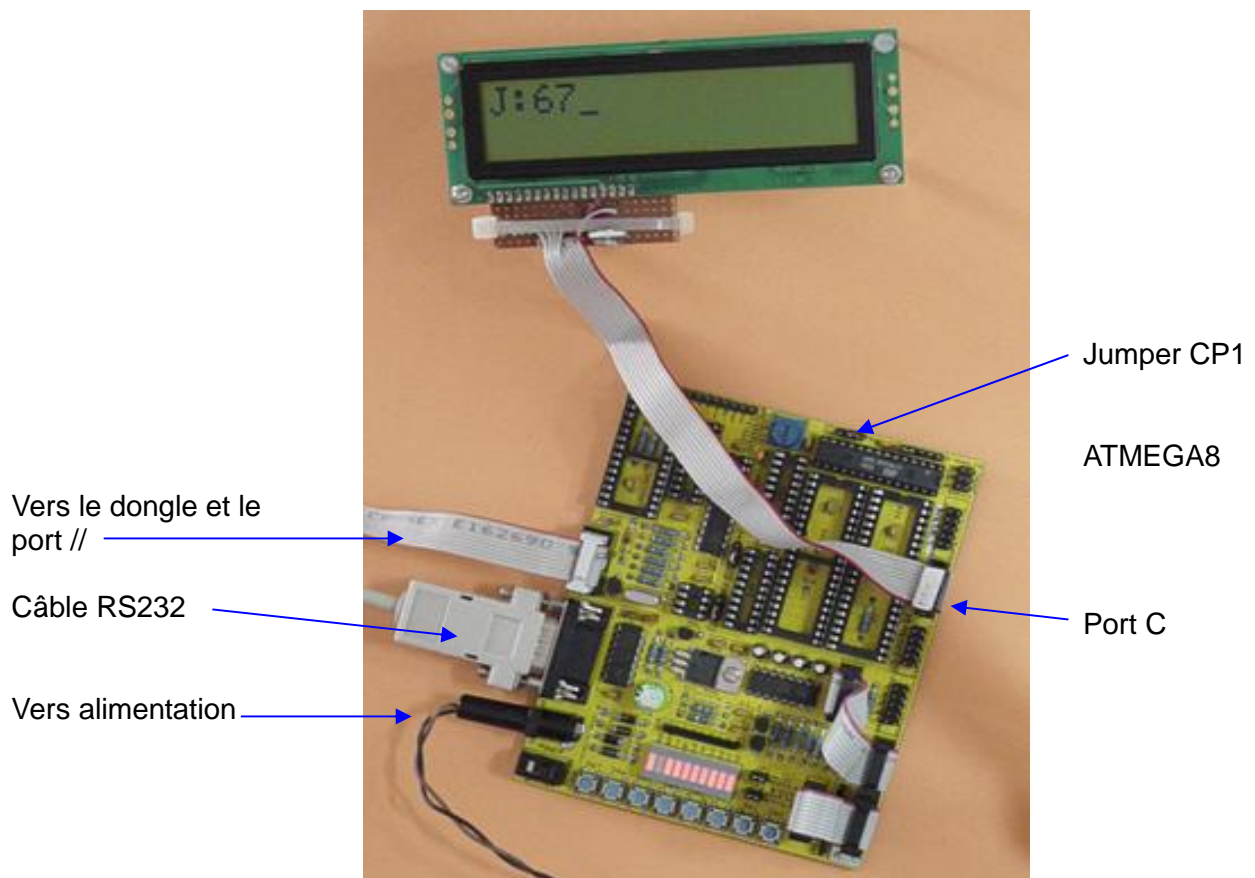
## LES ATMEGA SONT EN GENERAL LIVRES AVEC LE FUSEBIT A987 REGLE A : INTERNAL RC OSCILLATOR 1MHZ

Un autre problème peut se rencontrer sur l'ATMEGA8 utilisé avec d'autres cartes de développement, il concerne l'utilisation du port C.

Celui-ci est un port I/O mais aussi le port d'entrée analogique. Dans le cas d'une application logique AVCC doit être connecté à VCC. Si AVCC n'est pas connecté, la référence interne de 2.56V est prise en compte, ce qui est insuffisant pour piloter le LCD.

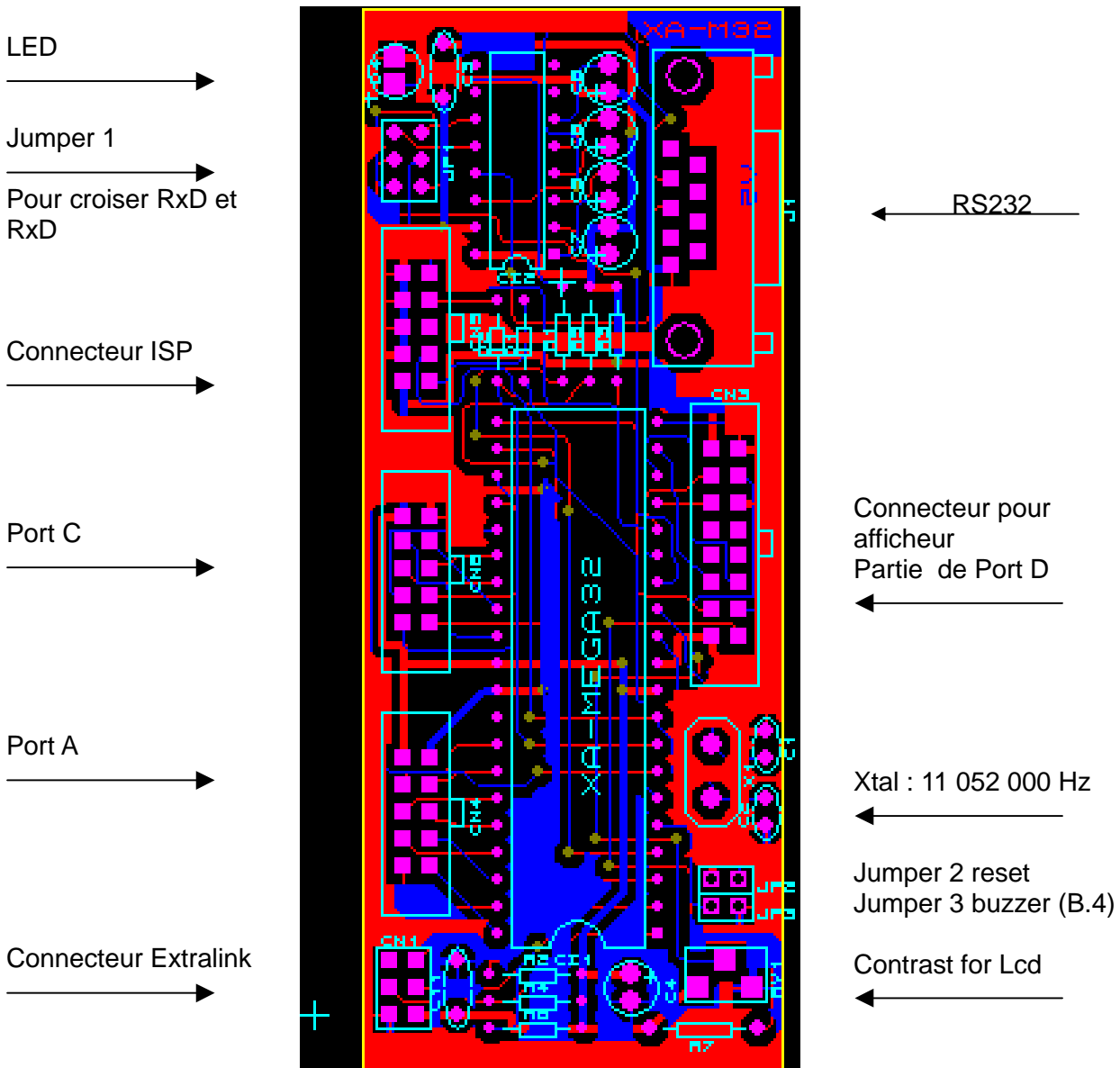
La carte STK200 utilise le câblage recommandé par ATMEL : AVCC à VCC à travers une résistance et filtré par un condensateur.

Les options « Write » des fuses-bits ne sont accessibles que si on change une valeur à partir de la version 1.11.8.0 du compilateur.



- Le câble RS232 n'est pas nécessaire ici, c'est un câble fil à fil, non croisé.
- La led à l'extrême gauche de la barre indique la présence de l'alimentation.

### Le module Xamega32



Jumper 1	
1 RXD Max 232	6 TXD Max 232
2 RXD micro	5 TXD micro
3	4
1	6
2 RXD micro 232	5 TXD micro 232
3 TXD Max 232	4 RXD Max 232

Display Port	
2 VDD	1 VSS (Gnd)
4 RS(D.2)	3 VO (contrast)
6 E (D.3)	5 R/W (Gnd)
8	7
10	9
12 DB5 (D.5)	11 DB4 (D.4)
14 DB7 (D.7)	13 DB6 (D.6)
16 - led (Gnd)	15 +led (390Ω)

Port A or C	
10 Vcc	9 Gnd
8 P.7	7 P.6
6 P.5	5 P.4
4 P.3	3 P.2
2 P.1	1 P.0

Programming C.	
10 Gnd	9 miso
8 Gnd	7 sck
6 Gnd	5 rst
4 Gnd	3
2 Vcc	1 Mosi

la led est en b.3 à travers un résistance de 270 ohms