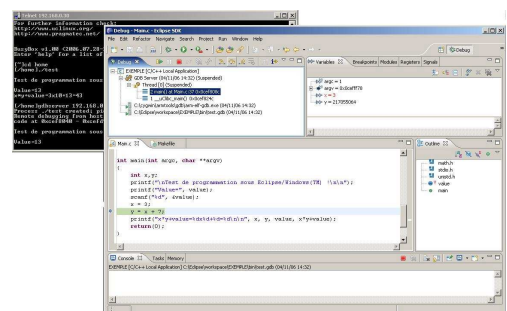


Pragmatec

Produits et services dédiés aux systèmes embarqués

uClinux - WINUX

Développement uClinux depuis Windows™





Bâtiment EARHART
ZAC Grenoble Air Parc
38590 St Etienne de St Geoirs - France
www.pragmatec.net

uClinux – WINUX

TABLE DES MATIERES

1	<i>Installation de la chaîne de compilation</i>	5
	Répertoire d'installation.....	5
	Installation de Cygwin et de la chaîne de compilation.....	5
	Paramétrage de cygwin.....	6
2	<i>Installation d'Eclipse</i>	8
	Installation de la JRE.....	8
	Installation d'Eclipse	8
3	<i>Création du projet EXEMPLE</i>	10
	Installation des sources	10
	Création d'un projet	11
4	<i>Mise à jour des applications</i>	16
	Transferts par FileZilla	16
	Communication par TELNET	17
5	<i>Débug de l'application</i>	18
	Lancement de gdbserver sur la cible	18
	Paramétrage d'Eclipse.....	19
	Lancement du débog réseau.....	21



Bâtiment EARHART
ZAC Grenoble Air Parc
38590 St Etienne de St Geoirs - France
www.pragmatec.net

uClinux – WINUX

Les kits de développement ARM7 sont des kits réalisés par la société PRAGMATEC S.A.R.L., société située à Grenoble (www.pragmatec.net). Ils sont basés sur une carte de développement ARM7, importé d'Asie ou directement produite par Pragmatec. Ces cartes sont largement utilisées de par le monde depuis de nombreuses années. Il s'agit donc d'un produit efficace, fiable et disponible.

Pragmatec s'est attaché à faire de ces kits un environnement de développement complet et immédiatement opérationnel, avec une introduction en français et le reste des documents et exemples en langue anglaise. En cas de difficultés techniques vous pourrez retrouver le support de l'équipe Pragmatec sur www.pragmatux.net et partager votre expérience avec les nombreux internautes qui fréquentent ce forum.

Ce document a pour but de démontrer la simplicité d'utilisation d'une telle plate-forme depuis un environnement Windows™. Nous allons montrer comment installer les outils de développement et comment réaliser, compiler puis déboguer une application type pour uClinux/S3C44B0X.

Ce document est la propriété de la société PRAGMATEC S.A.R.L. Il ne peut être reproduit et distribué sans l'accord de cette société.



1 Installation de la chaîne de compilation

La chaîne de compilation utilisée pour construire les applications de EXEMPLE est basée sur la version Windows™ de GCC 2.95.3.

Toutefois, afin de pouvoir fonctionner correctement avec la plate-forme de développement Eclipse, il importe d'installer « cygwin » un système de simulation d'un shell UNIX sous Windows™.

Répertoire d'installation

Pour installer les outils nécessaires au développement Eclipse sous Windows™ il est nécessaire d'utiliser 2 fichiers zip et un exécutable. Vous trouverez ces fichiers dans le répertoire « uClinux Development / Winux » de votre CDROM.



Un zip est dédié à l'installation d'Eclipse et un autre à celle de cygwin.

L'exécutable est destiné à installer une machine JAVA sur la station de travail Windows™.

Un dernier fichier zippé contient l'exemple utilisé plus loin dans le cadre du debug distant.

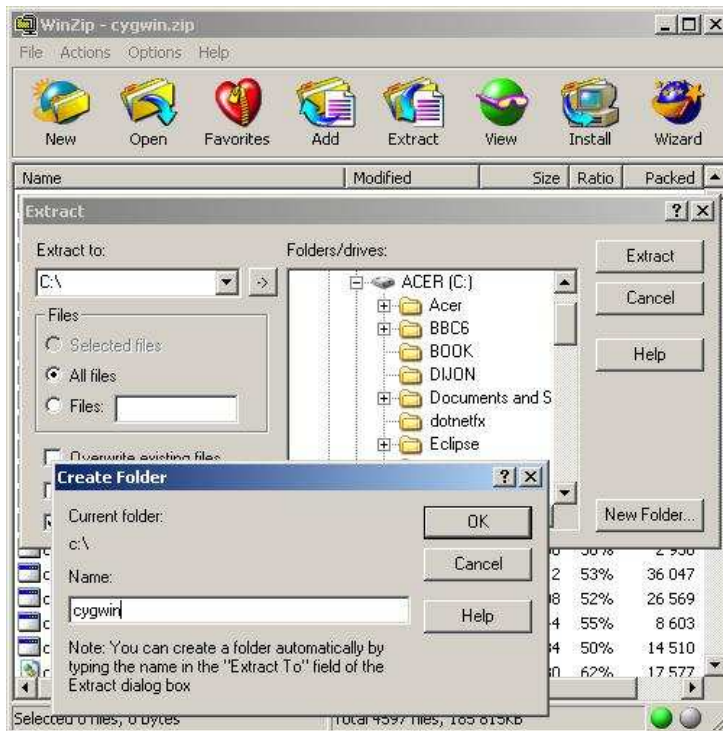
Installation de Cygwin et de la chaîne de compilation

Afin d'installer cygwin et la chaîne de compilation GCC pour processeur ARM™, double cliquez sur le fichier « cygwin.zip ».

Une fenêtre s'ouvre. Cliquez sur le bouton « Extract » puis sélectionnez le disque C. Enfin cliquez sur le bouton « New Folder ». Une nouvelle fenêtre vous demandera alors le nom répertoire de destination : nommé le « cygwin ».

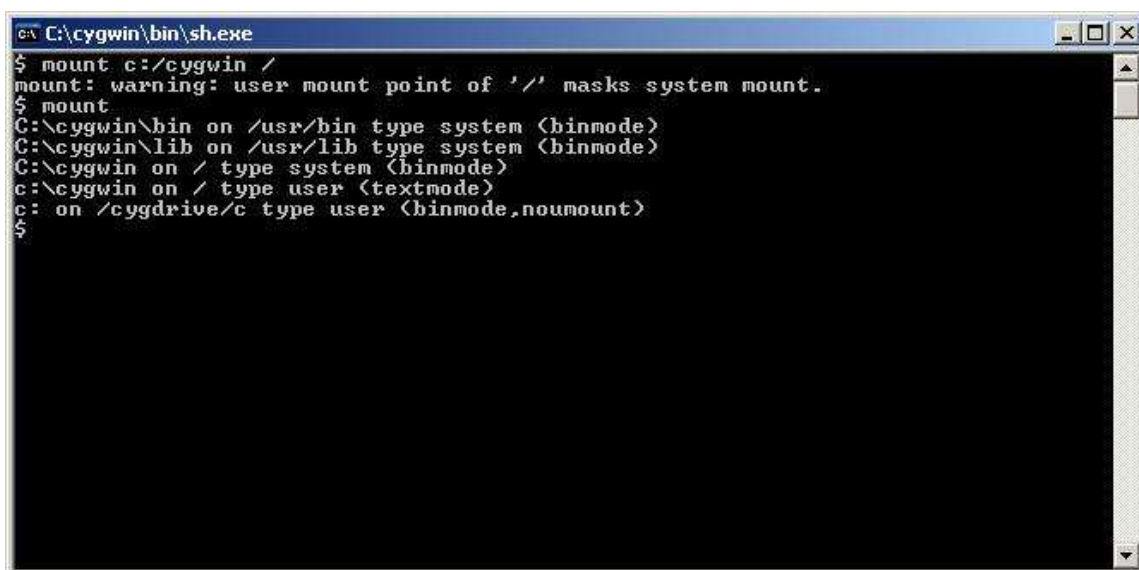
Débutez alors l'installation en cliquant sur « Extract ».

uClinux – WINUX



Paramétrage de cygwin

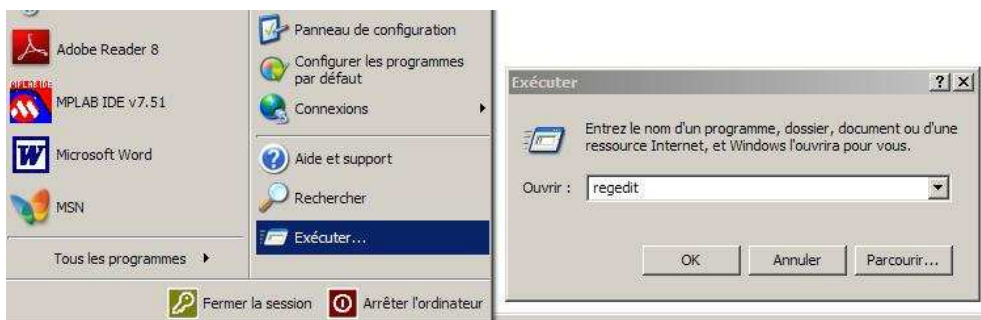
Dans le répertoire « C:\cygwin\bin », double cliquez sur le programme « sh.exe » afin d'ouvrir une session shell cygwin. Dans ce shell tapez alors la commande « mount c:/cygwin / » :



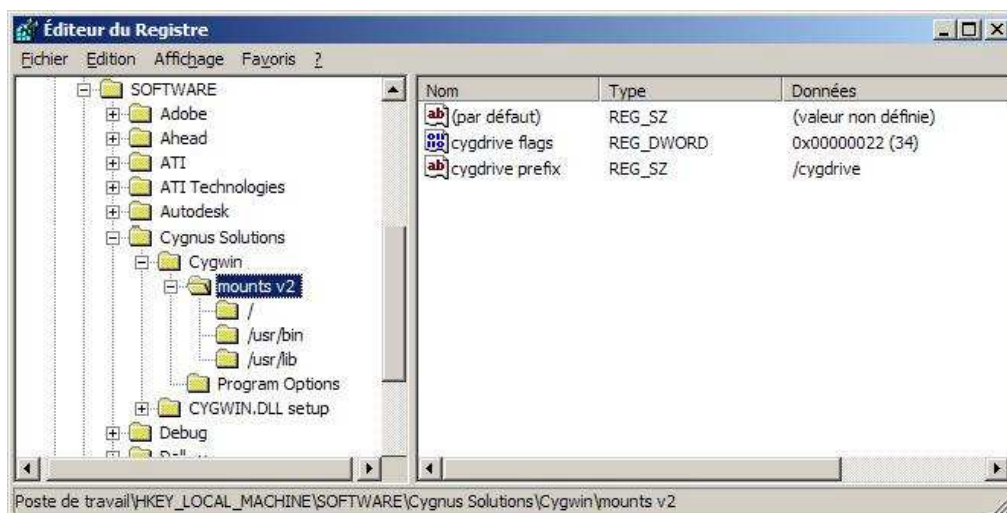
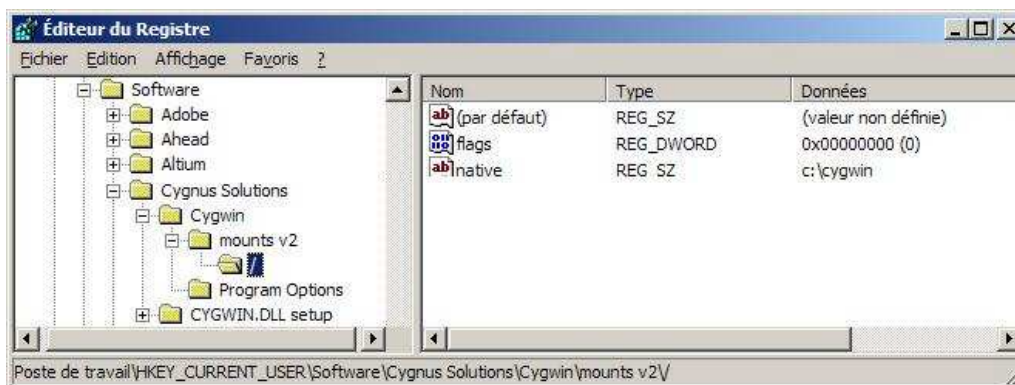
Vous pouvez retaper la commande « mount » afin de bien vérifier que « c:\cygwin » a bien été monté sous la racine du shell cygwin « / ».

uClinux – WINUX

Si vous ne parvenez pas à obtenir exactement ce qui indiqué ci-dessus à l'aide de la commande « mount », il vous faudra sans doute directement modifier les points de montage de cygwin dans la base de registre de Windows™ :



Enfin modifier les clefs comme suit sous HKEY_CURRENT_USER et HKEY_LOCAL_MACHINE :



La fenêtre la plus importante est sans doute la seconde fenêtre où n'apparaît peut-être pas sur votre machine les valeurs « DWORD » et « chaîne » nommées respectivement « cygdrive flags » et « cygdrive prefix ».

2 Installation d'Eclipse



Eclipse est une plate-forme de développement largement utilisée dans le monde entier y compris par d'illustres sociétés telles Windriver. Eclipse est à l'origine destinée à des développements en langage JAVA, l'outil étant lui-même réalisé en JAVA. Il est donc nécessaire d'installer sur votre station le JRE de SUN™ si vous ne l'avez jamais installé sur votre machine.

La version d'Eclipse que nous vous fournissons est une version patchée qui permet de programmer en langage C pour une cible ARM™.

Installation de la JRE

Commencez l'installation de la JRE de SUN™ en double cliquant sur l'exécutable :



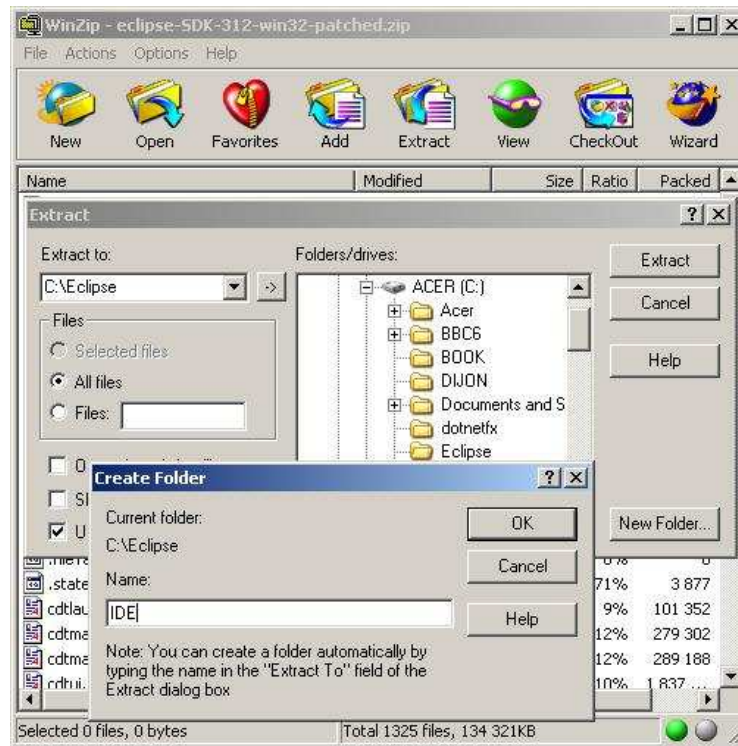
Installation d'Eclipse

Pour installer Eclipse, commencez par créer un répertoire « Eclipse » sous le répertoire C.

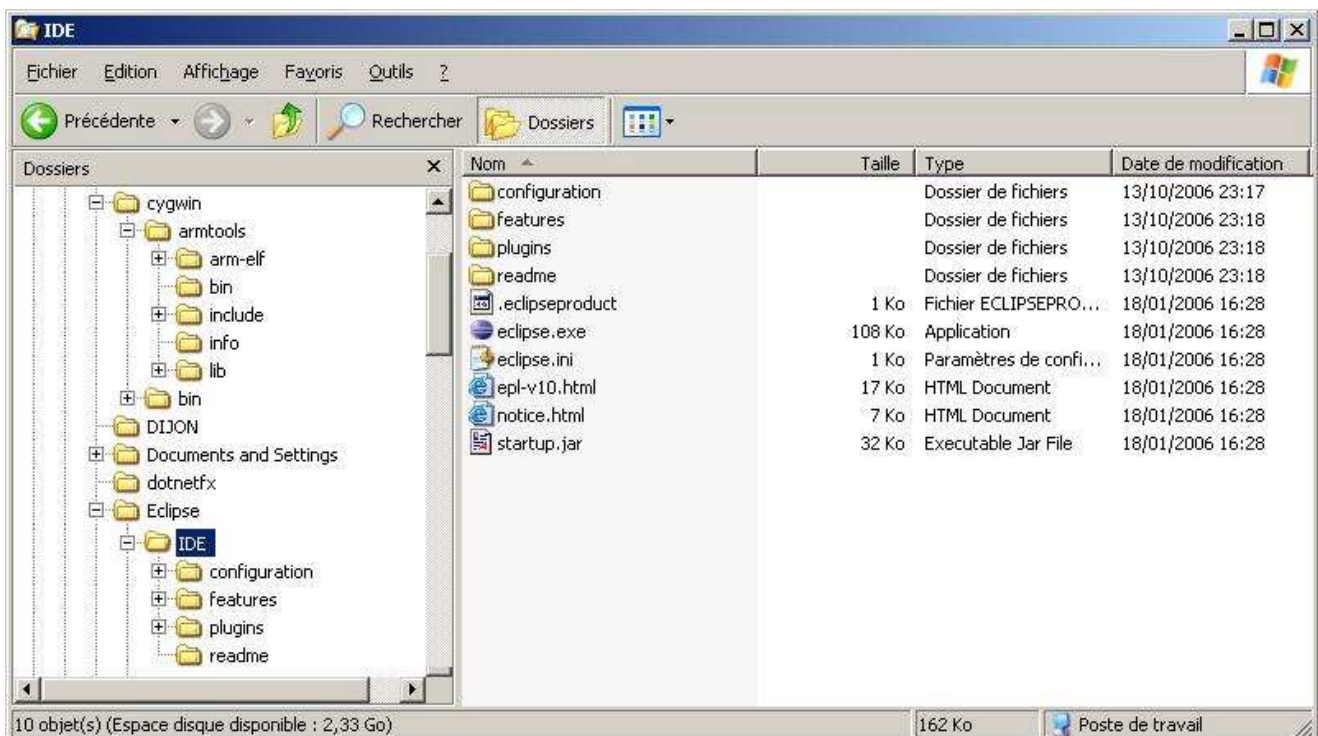
Double-cliquez ensuite sur le zip d'Eclipse. Une fenêtre s'ouvre. Cliquez sur le bouton « Extract » puis sélectionnez le répertoire « c:/Eclipse ». Enfin cliquez sur le bouton « New Folder ». Une nouvelle fenêtre vous demandera alors le nom répertoire de destination : nommé le « IDE ».

Débutez alors l'installation en cliquant sur « Extract ».

uClinux – WINUX



Une fois l'installation terminée, vous devriez obtenir l'organisation des répertoires d'installation comme suit :



L'exécutable « eclipse.exe » se trouve alors sous « c:/Eclipse/IDE ».

3 Création du projet EXEMPLE



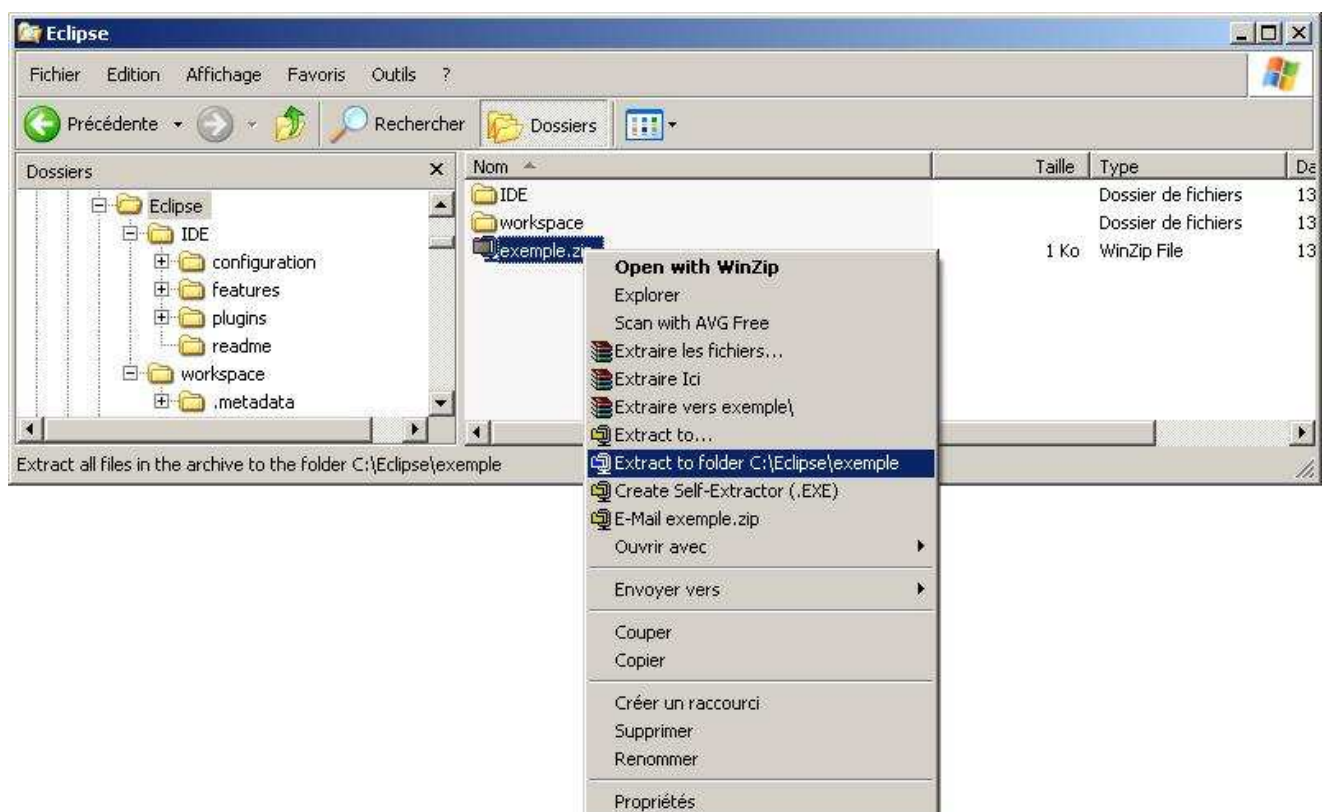
Les outils de développement installé sur votre station de travail. Lors du premier lancement d'Eclipse, un répertoire « workspace » sera créé sous « c:/Eclipse ».

Par la suite les répertoires de travail de chaque projet devront être situés sous « c:/Eclipse/workspace ».

Installation des sources

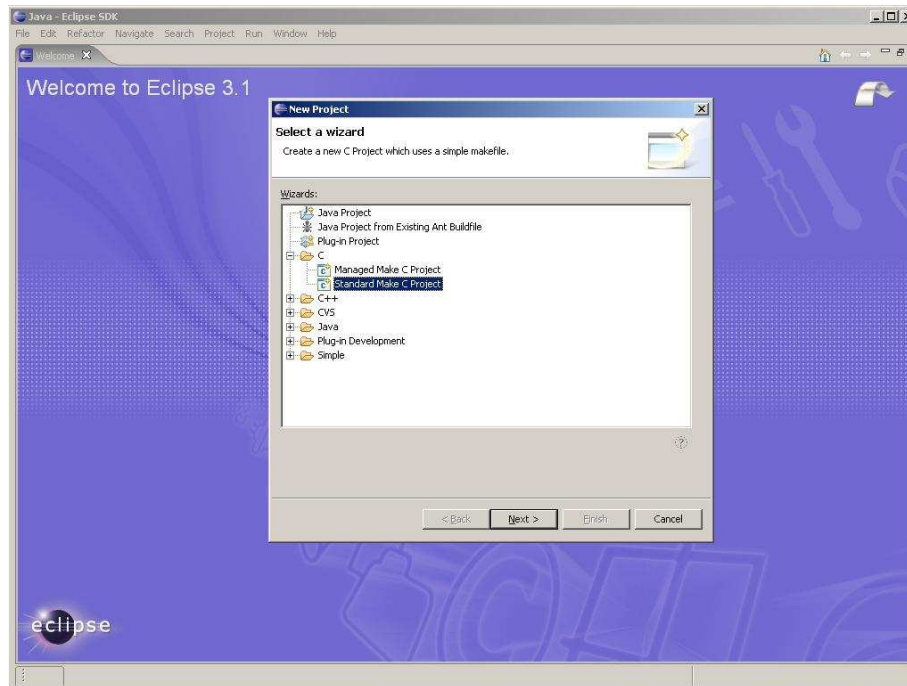
Avant toutes choses il est nécessaire de décompresser les sources de l'application EXEMPLE sous le répertoire « C:\Eclipse\EXEMPLE ».

Copiez le zip des sources EXEMPLE sous « C:\Eclipse » et décompressez les sources comme suit :



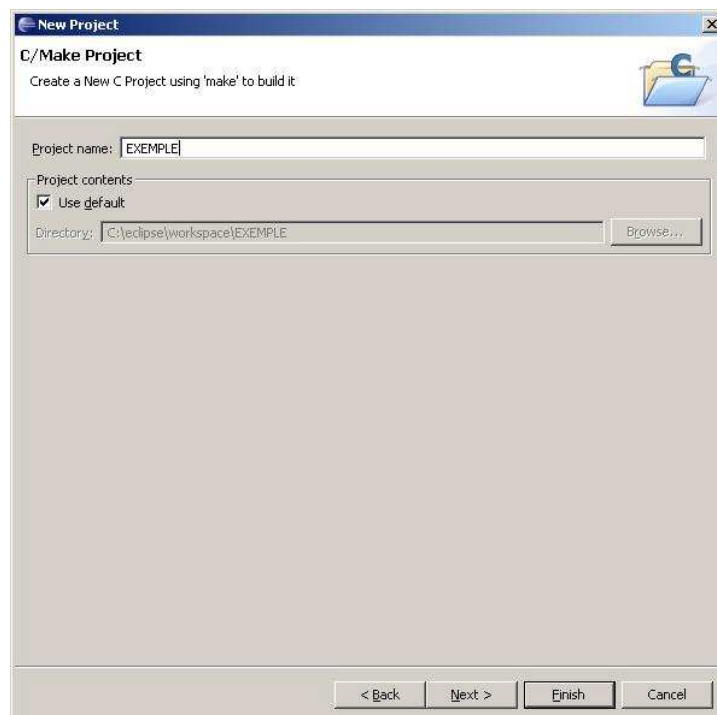
Création d'un projet

Une fois Eclipse lancé, cliquez sur « File/New/Project... ». La fenêtre suivante apparaît :



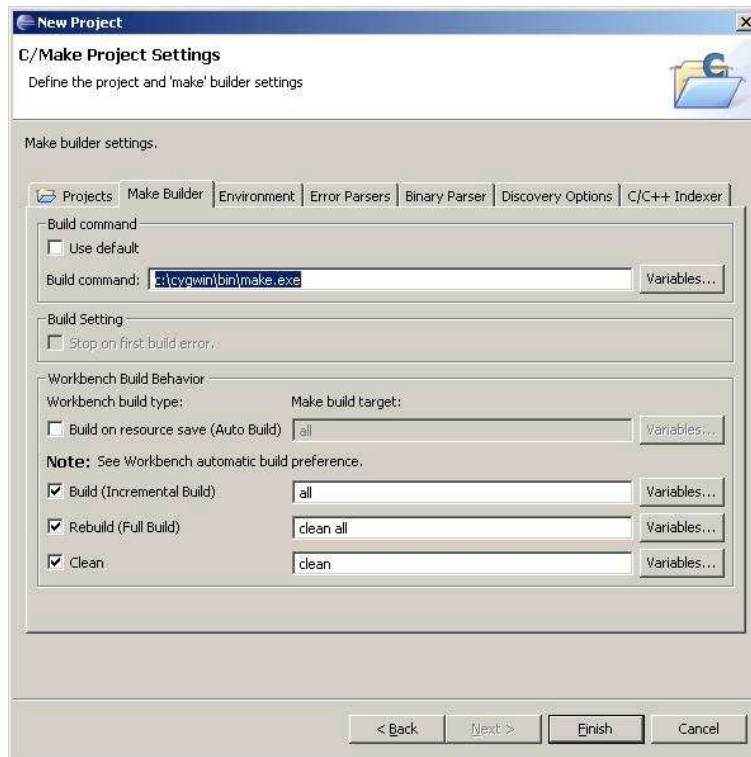
Comme type de projet, choisissez « Standard Make C project ».

Dans la fenêtre qui suit nommez le nouveau projet : EXEMPLE. Le nom du projet sera donné au répertoire. Par défaut il sera situé sous « c:\Eclipse\workspace\EXEMPLE ». Cliquez ensuite sur Next :



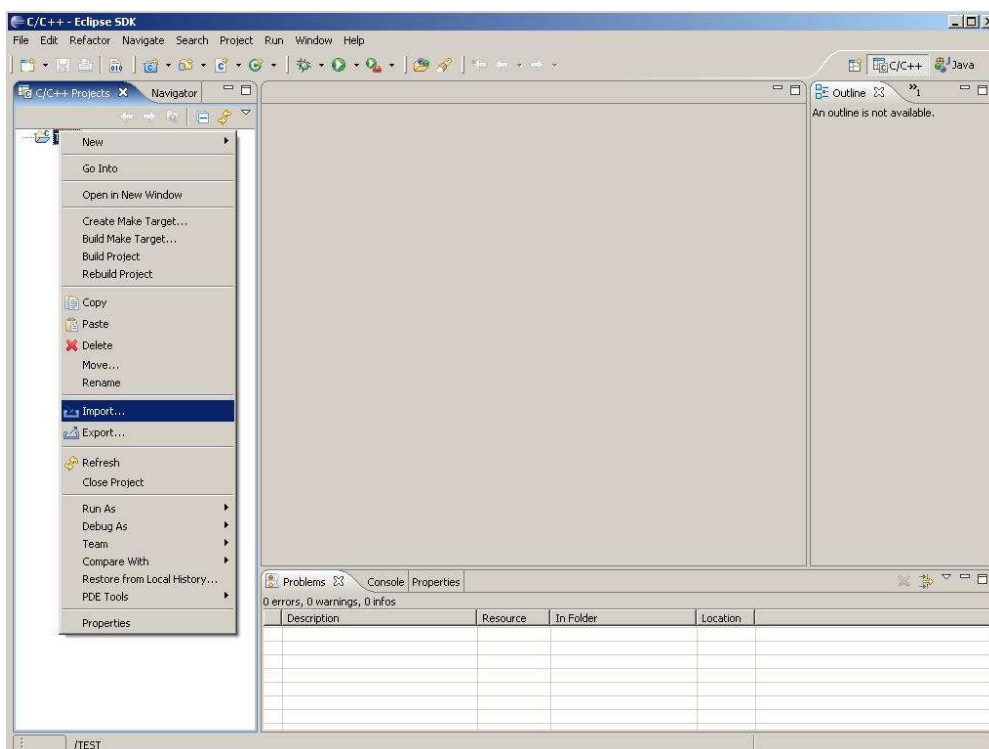
uClinux – WINUX

Il s'agit à présent de préciser que l'application « make.exe » doit être celle de cygwin :



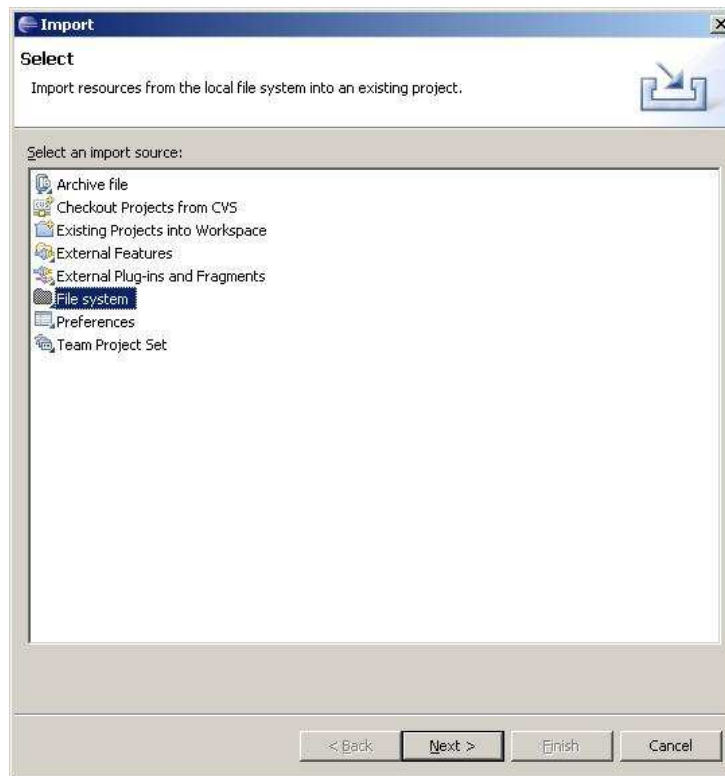
Cliquez finalement sur le bouton Finish pour terminer la création du projet.

Importez les fichiers C que vous avez dézippé précédemment (projet EXEMPLE) :



uClinux – WINUX

Dans la nouvelle fenêtre choisissez « File system » pour aller sélectionner le répertoire « C:\Eclipse\EXEMPLE » :

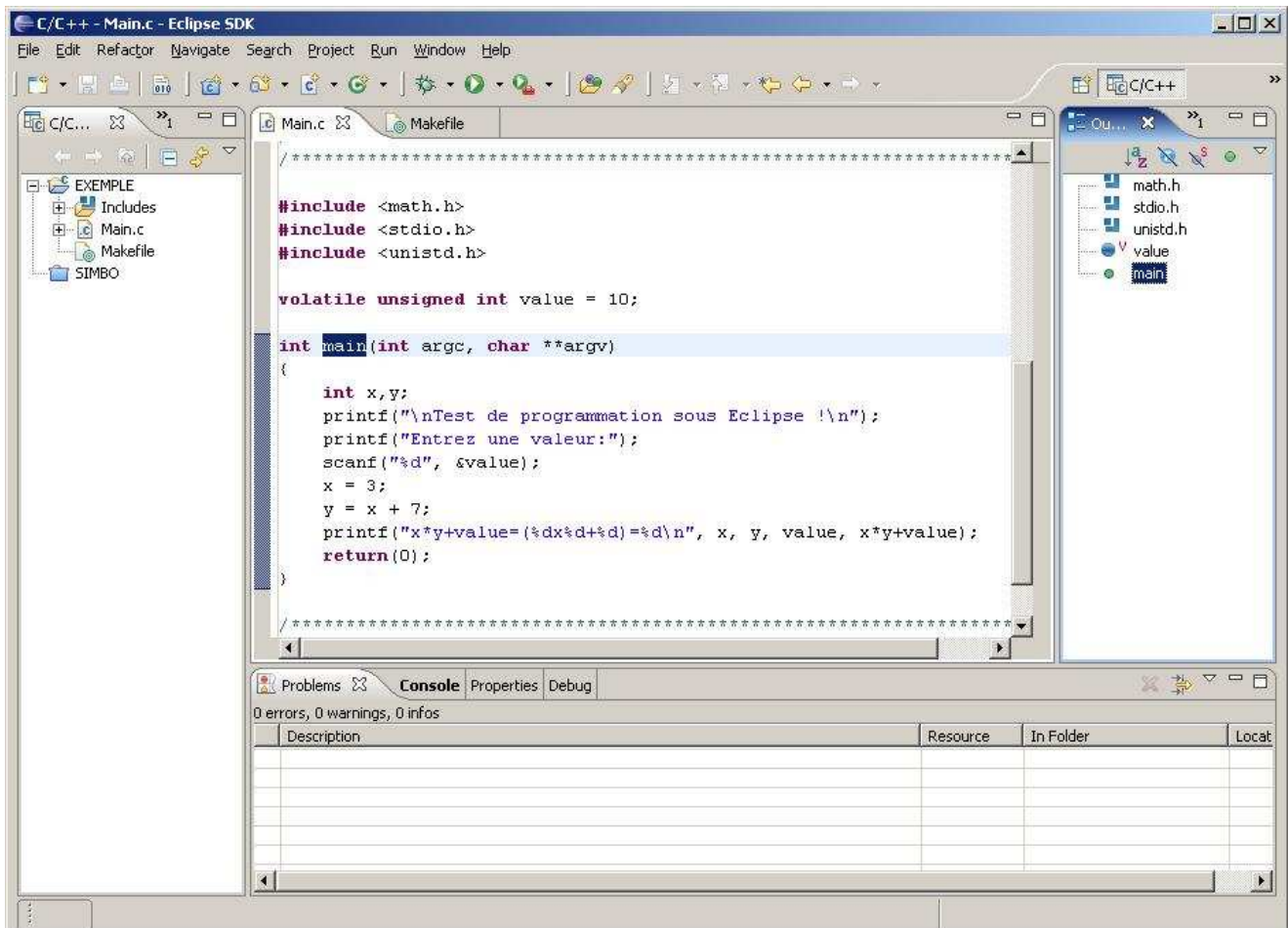


Sélectionnez l'ensemble des fichiers à l'aide du bouton « Select All ».



uClinux – WINUX

Vous pouvez désormais ouvrir les fichiers du projet en cliquant sur un des fichiers du projet EXEMPLE (partie gauche), et utiliser l'inspecteur d'objet pour visionner directement les fonctions, les variables globales ou les définitions.



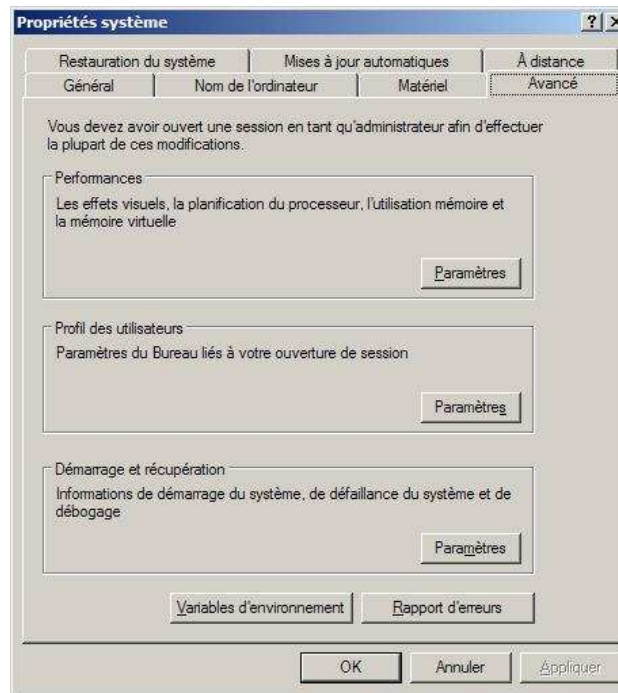
Enfin cliquer à l'aide du bouton droit de la souris sur le projet EXEMPLE, et sélectionner « New/Folder » afin de créer comme suit un sous-répertoire « obj » et « bin » (répétez donc l'opération une fois) :



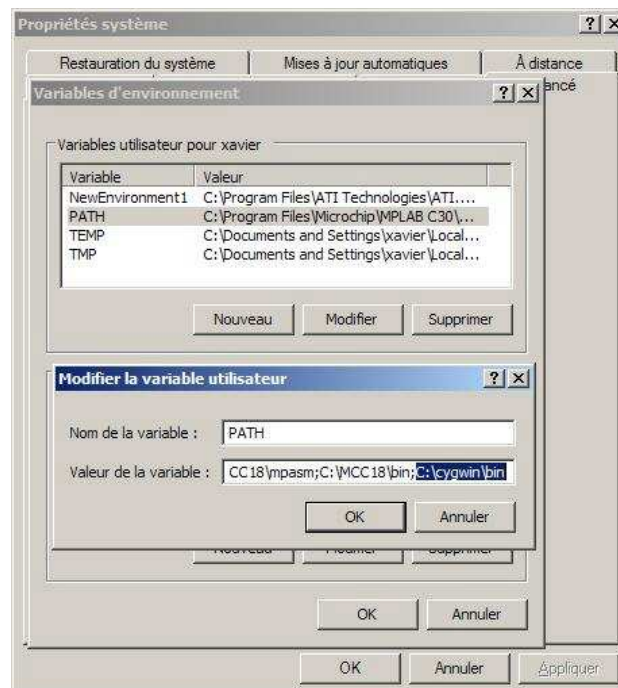
Pour compiler le projet EXEMPLE, cliquez sur « Project/Build All » (les programmes sont sous bin).

uClinux – WINUX

Si toutefois la compilation ne s'effectue pas correctement, il est possible que le chemin indiquant l'emplacement des programmes tels que « rm » ne soit pas mis à jour. Pour cela, modifier la variable d'environnement PATH à l'aide des panneaux suivants :



Cliquez sur le bouton « Variables d'environnement » :



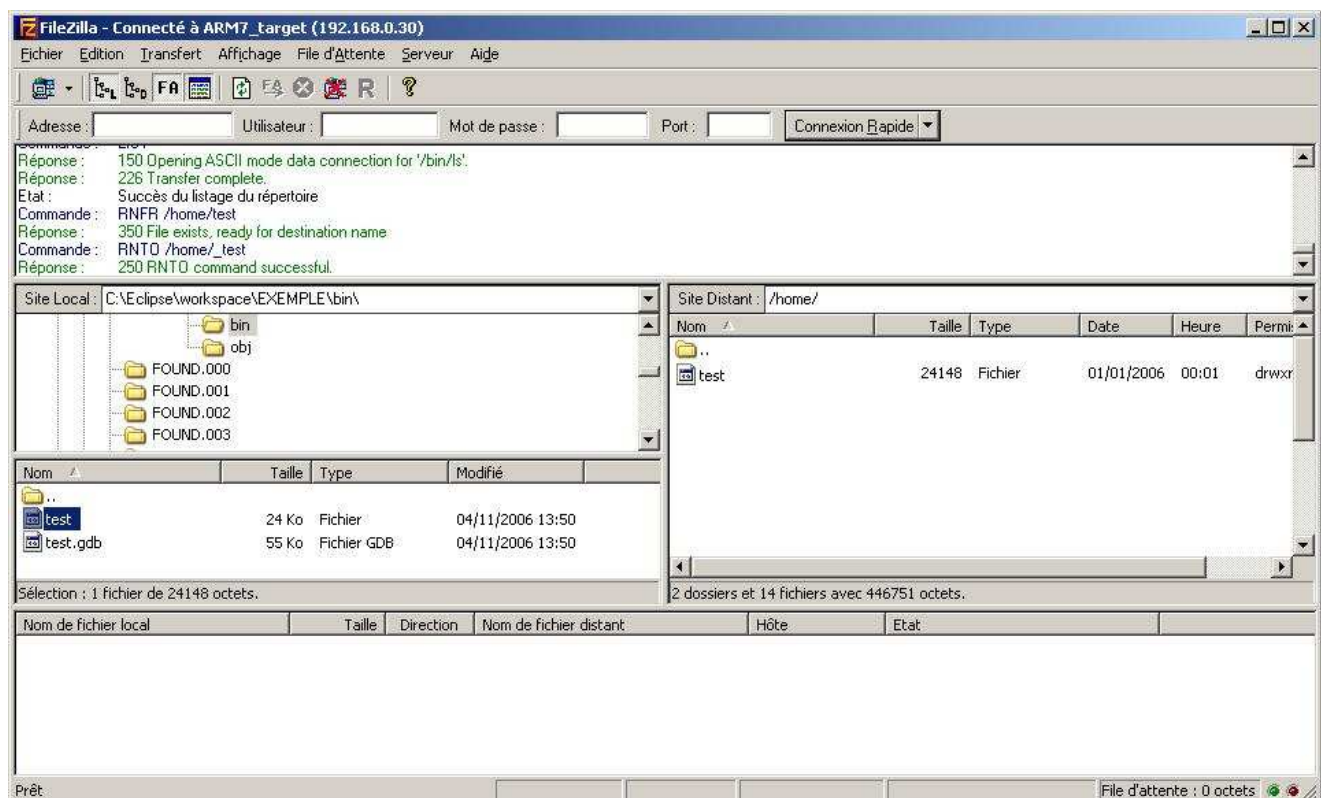
4 Mise à jour des applications



Votre carte de développement et son noyau pré-chargé disposent d'une couche réseau et des applications standards permettant les transferts de données ou la communication avec un PC distant : FTP, TELNET et TFTP.

Transferts par FileZilla

Il est possible de se connecter à votre carte ARM7 en FTP dans le but de télécharger des fichiers audio dans l'équipement ou bien pour lire des résultats de tests. Pour cela vous pouvez par exemple utiliser des logiciels gratuits de transferts FTP tel que FileZilla :



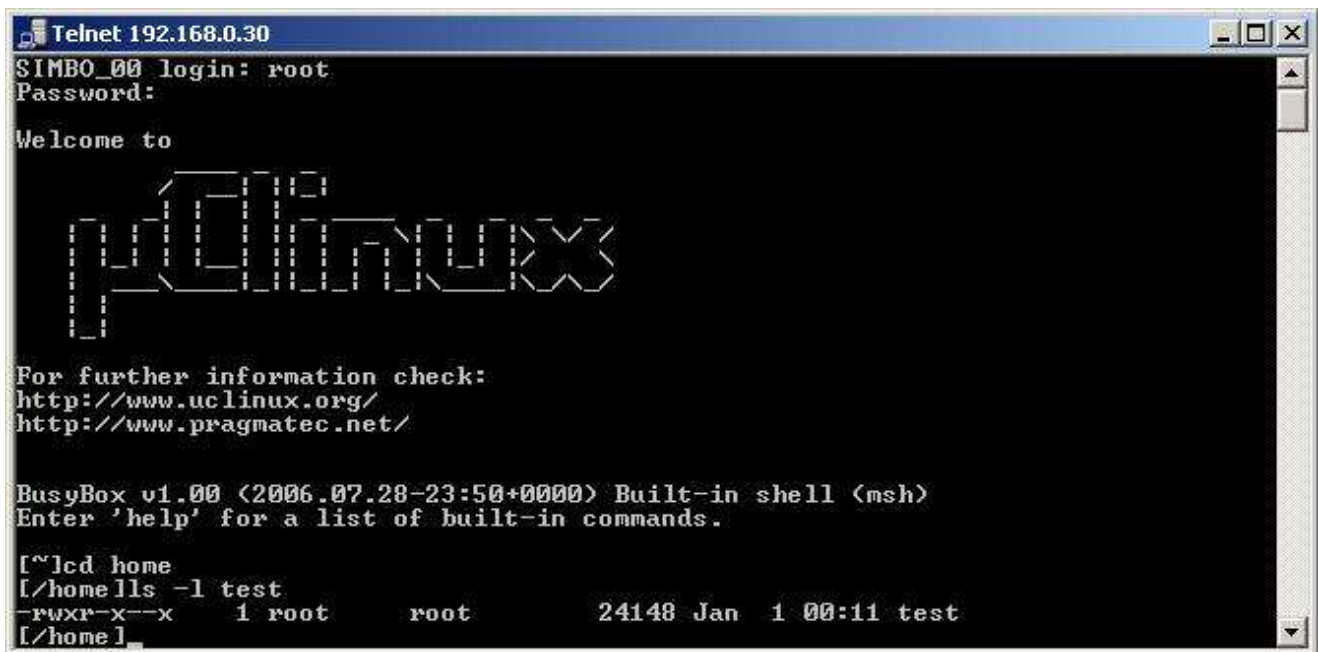
Utilisez le login « **root** » (password « **root** ») pour vous connecter. Pour transférer une application vers la cible, double cliquez sur le fichier choisi ou sélectionnez le et appuyez sur ENTER. Une compilation complète du projet et son transfert sur la cible se fait donc en quelques appuis de touches.

Transférer uniquement le fichier « test » destiné à être exécuté sur la cible. Le fichier « test.gdb » est destiné quant à lui à la station hôte pour le debug (contient la table des symboles).

uClinux – WINUX

Communication par TELNET

Il est possible de se connecter à votre carte ARM7 en TELNET afin d'ouvrir un shell distant et ainsi de contrôler votre cible. Vous pourrez ainsi lancer les programmes télécharger ou débiter une séquence de debug.

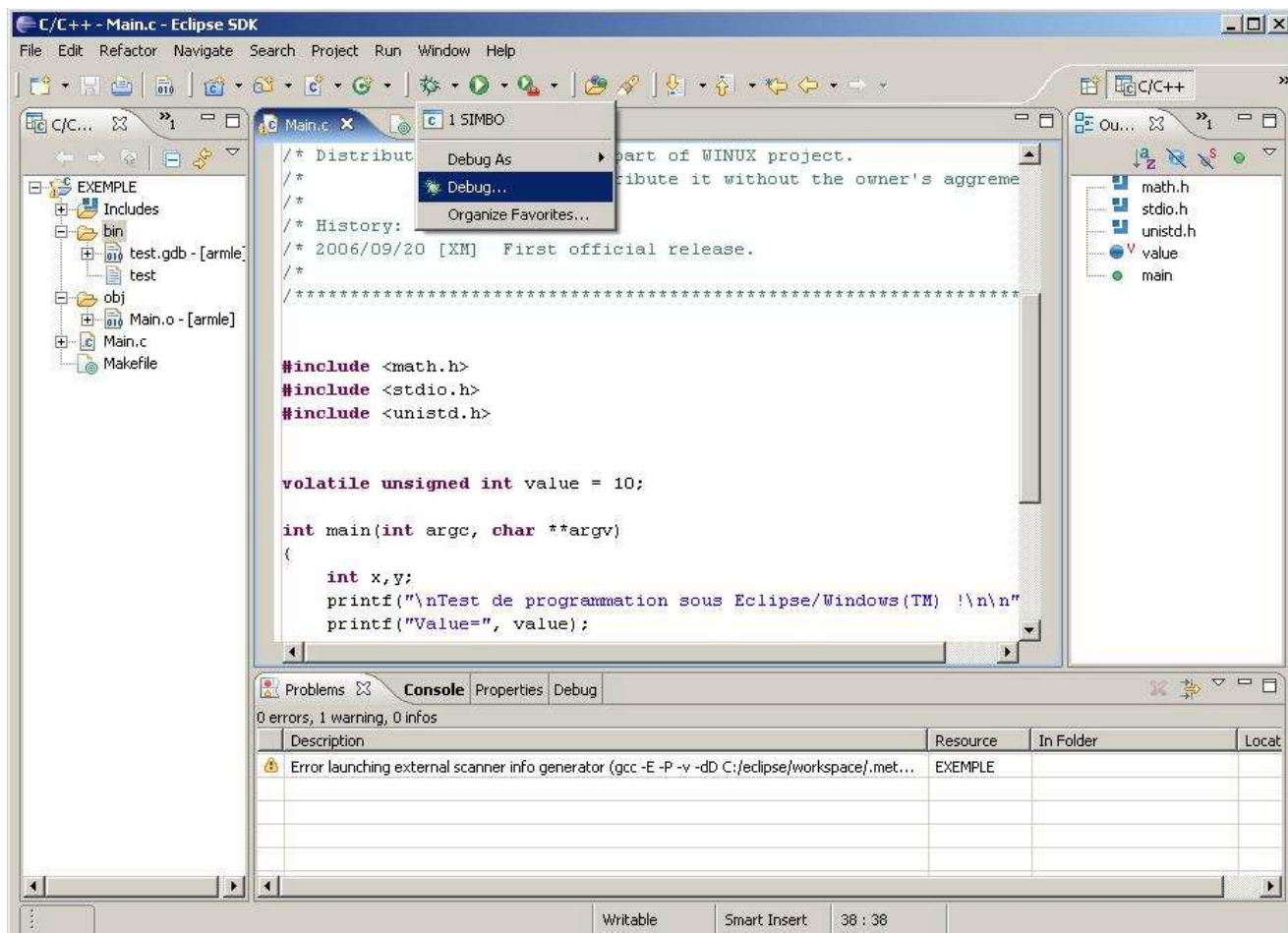


```
Telnet 192.168.0.30
SIMBO_00 login: root
Password:
Welcome to
uClinux
For further information check:
http://www.uclinux.org/
http://www.pragmatec.net/
BusyBox v1.00 (2006.07.28-23:50+0000) Built-in shell (msh)
Enter 'help' for a list of built-in commands.

[~]cd home
[/home]ls -l test
-rwxr-x--x  1 root  root    24148 Jan  1 00:11 test
[/home]
```


Paramétrage d'Eclipse

Pour créer une perspective de débog, cliquez sur la flèche à droite du bouton représentant un « cafard », utilisé comme symbolique pour traquer les « bugs » :



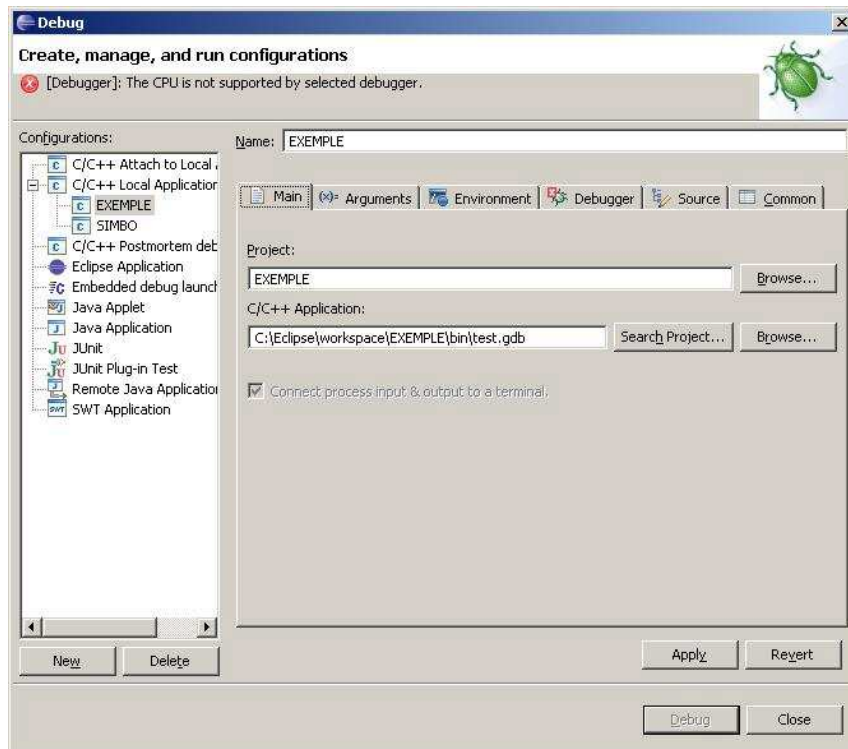
Puis cliquez sur « Debug »...

Une nouvelle fenêtre apparaîtra. Vous devez sélectionner « C/C++ Local Applications » et cliquer sur le bouton « NEW » afin de créer cette nouvelle perspective.

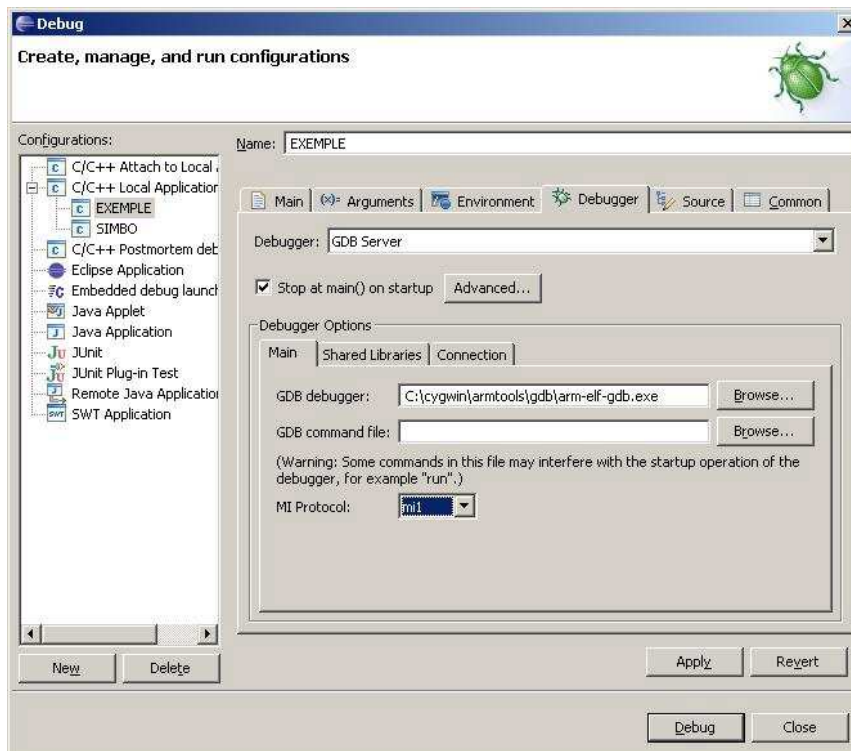
Ensuite dans la fenêtre de texte « C/C++ Application », cliquez sur le bouton « Browse... » pour sélectionner le fichier « test.gdb » contenant la table des symboles de notre application.

Un message apparaît en haut de la fenêtre indiquant que le débogueur choisi ne permet pas de déboguer pour la cible ARM7. Nous allons donc devoir sélectionner un débogueur approprié.

uClinux – WINUX

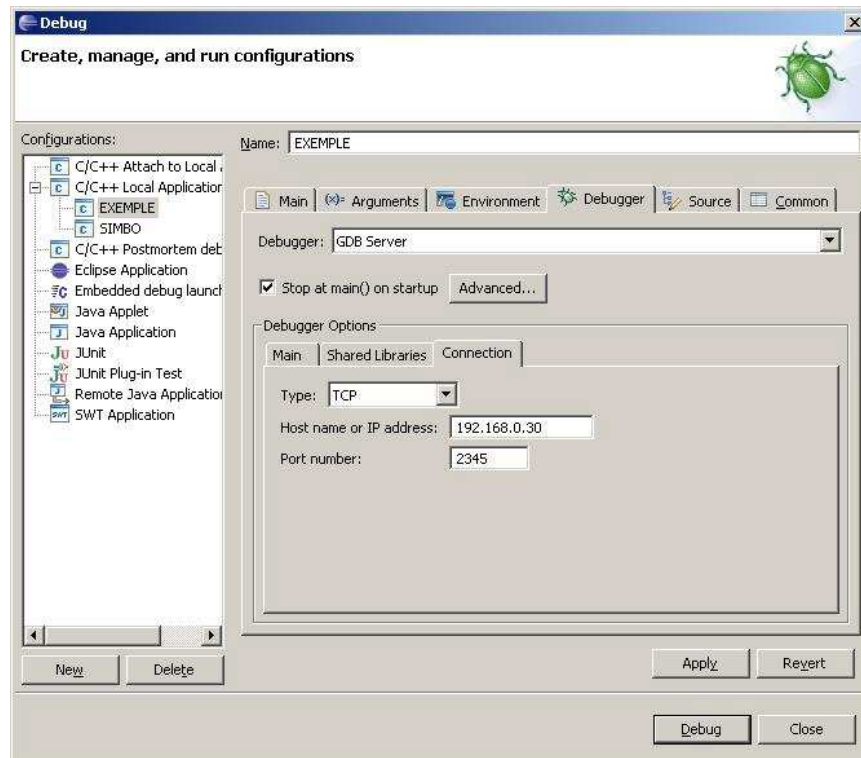


Cliquez donc par la suite sur l'onglet « Debugger » afin de choisir le bon débogueur :



N'oubliez pas de sélectionner « GDB Server » comme type de « Debugger ».

Enfin il faut renseigner les différents paramètres propres au réseau :



Vous pouvez ensuite cliquer sur les boutons APPLY et CLOSE.

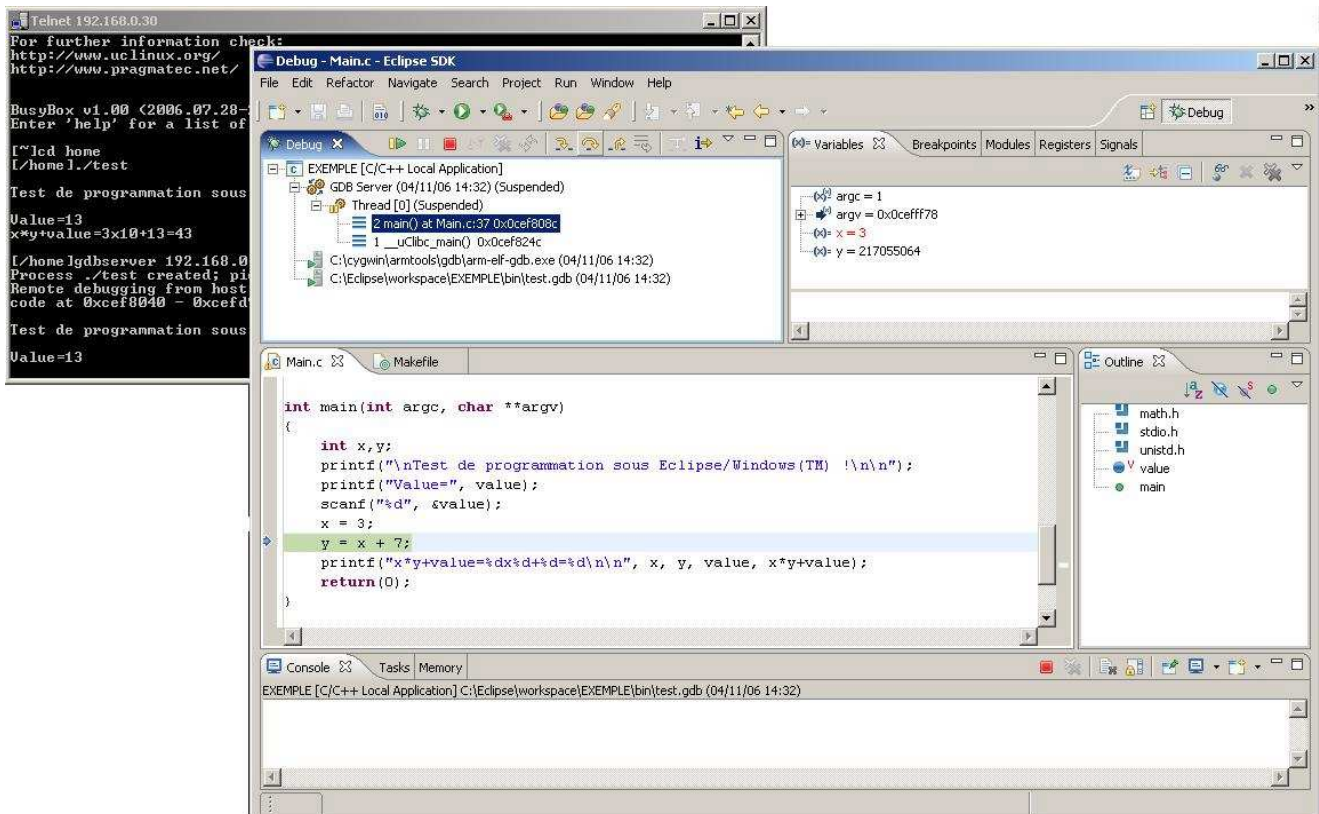
Lancement du debug réseau

De retour dans votre environnement de développement Eclipse, vous pouvez alors lancer le debug de l'application en cliquant sur le bouton de debug. S'il ne se passe rien il est possible que votre perspective ne soit pas apparente, il faut donc cliquer sur la flèche associée au debug et choisir l'option « Organize favorite... » :



uClinux – WINUX

Ceci étant fait vous pouvez alors lancer le débog de votre application en cliquant sur le bouton de débog. Désormais c'est la seule chose que vous aurez à faire pour commencer une session de débog :



Vous pouvez alors utiliser Eclipse comme n'importe quel autre débogueur : breakpoints, inspection de variables, modification de variables,...