

1. Source du projet
2. Installation des outils
 1. SystemCass
 2. or1k - binutils
 3. or1k - gcc
 4. Newlib
 5. Xilinx ISE WebPACK
3. Finalisation de l'installation de Morpheo
4. Test de l'installation

L'installation de la plate forme Morpheo se fait en trois étapes.

1. La première consiste à rapatrier les sources du projet
2. Ensuite d'installer les logiciels tiers utilisés par Morpheo
3. Enfin finaliser l'installation de Morpheo

L'interpréteur de commandes utilisé est le *bash*.

Source du projet

Voici les étapes à suivre afin de rapatrier les sources du projet :

1. Placez vous dans le répertoire qui va contenir les sources de Morpheo
2. Vous avez deux manières d'acquérir les sources de morpheo :
 1. En téléchargeant la version en développement du projet. Cette version n'est pas forcément dans un état stable. Pour cela, exécutez la commande suivante :

```
svn co https://www-asim.lip6.fr/svn/morpheo/trunk/ Morpheo
```

Cette commande va placer dans le répertoire courant, un répertoire *Morpheo* contenant la version en développement du projet.

2. Soit en récupérant une archive du projet (disponible sur ce [lien](#)). La nomenclature de l'archive est disponible sur ce [lien](#). Pour décompresser l'archive, exécutez la commande suivante :

```
tar xvjf morpheo-x.y.z.tar.bz2
```

3. Le répertoire courant contient maintenant un répertoire *Morpheo*. Pour utiliser Morpheo, il faut positionner quelques variables d'environnement. Pour ce faire, placez vous dans le répertoire *Morpheo* nouvellement créé et lancez le script suivant :

```
./environment.sh
```

Parmi toutes les variables d'environnement, il y a celle qui indique la racine du projet morpheo :
MORPHEO_TOPLEVEL

Installation des outils

Nous allons maintenant installer les outils réalisés par des tiers, utilisés par la librairie Morpheo. Ces applications peuvent soit être récupérées sur le site de ces outils, soit dans le répertoire `${MORPHEO_TOPLEVEL}/Tools`

SystemCass

Site [?https://www-asim.lip6.fr/trac/systemcass](https://www-asim.lip6.fr/trac/systemcass)

Version 11_05_2007

Fichiers systemcass_11_05_2007.tar.bz

systemc_tests-2.1.v1.tgz

SystemCass est un simulateur compatible SystemC ([?http://www.systemc.org/](http://www.systemc.org/)). Pour l'installer, effectuez les étapes suivantes :

1. Décompressez l'archive

```
tar xvjf systemcass_11_05_2007.tar.bz
```

2. Positionner la variable d'environnement SYSTEMC vers le répertoire cible

```
export SYSTEMC=/opt/systemcass  
export SYSTEMCASS=${SYSTEMC}
```

3. Dans le répertoire crée (en l'occurrence systemcass_11_05_2007), allez dans le répertoire src

4. Lancez la compilation

```
make all
```

or1k - binutils

Site [?http://www.beyondsemi.com/page/products/processor_cores/openrisc](http://www.beyondsemi.com/page/products/processor_cores/openrisc)

Version 2.16.1

Fichiers or1k_binutils.tar.gz

Il s'agit d'un ensemble d'outils de développement logiciel. Pour les installer, effectuez les étapes suivantes :

1. Décompressez l'archive

```
tar xzvf or1k_binutils.tar.gz
```

2. Créez un répertoire qui va contenir les sources compilées.

```
cd or1k  
mkdir or32-elf-binutils  
cd or32-elf-binutils
```

3. Lancez le script de configuration. L'option --prefix cible le répertoire qui va contenir les binaires et les libraries.

```
../binutils-2.16.1/configure --target=or32-elf --prefix=/opt/or32-elf
```

4. Lancez la compilation

```
make  
make install
```

or1k - gcc

Site [?http://www.beyondsemi.com/page/products/processor_cores/openrisc](http://www.beyondsemi.com/page/products/processor_cores/openrisc)

Version 3.4.4

Fichiers or1k_gcc.tar.gz

Il s'agit d'une collection de logiciels libres intégrés capables de compiler divers langages de programmation. Pour les installer, effectuez les étapes suivantes :

1. Modifiez la variables d'environnement PATH pour inclure les binaires des binutils

```
export PATH=/opt/or32-elf/bin:$PATH
```

2. Décompressez l'archive

```
tar xzvf or1k_gcc.tar.gz
```

3. Créez un répertoire qui va contenir les sources compilées.

```
cd or1k
mkdir or32-elf-gcc
cd or32-elf-gcc
```

4. Lancez le script de configuration. L'option --prefix cible le répertoire qui va contenir les binaires et les libraries.

```
../gcc-3.4.4/configure --target=or32-elf --prefix=/opt/or32-elf --with-gnu-as --with-gnu-
```

5. Lancez la compilation

```
make
make install
```

Newlib

Site [?http://sourceware.org/newlib/](http://sourceware.org/newlib/)

Version 1.17.0

Fichiers newlib-1.17.0.tar.gz

Il s'agit d'une implémentation de la librairie standard C pour systèmes embarqués.

1. Modifiez la variables d'environnement PATH pour inclure les binaires des binutils

```
export PATH=/opt/or32-elf/bin:$PATH
```

2. Décompressez l'archive

```
tar xvzf newlib-1.17.0.tar.gz
```

3. Créez un répertoire qui va contenir les sources compilées.

```
mkdir newlib-or32
cd newlib-or32
```

4. Lancez le script de configuration. L'option --prefix cible le répertoire qui va contenir les binaires et les libraries.

```
../newlib-1.17.0/configure --prefix=/opt/newlib --host=or32 --target=or32-elf
```

5. Lancez la compilation

```
make
make install
```

Xilinx ISE WebPACK

Site [?http://www.xilinx.com/ise/logic_design_prod/webpack.htm](http://www.xilinx.com/ise/logic_design_prod/webpack.htm)

Version 10.1

Fichiers ?

C'est environnement de développement intégré. Il contient un simulateur ainsi qu'un synthétiseur pour les FPGA xilinx.

Finalisation de l'installation de Morpheo

Voici les dernières étapes pour finaliser l'installation de Morpheo

1. Editez le fichier `${MORPHEO_TOPLEVEL}/Makefile.tools`. Ce fichier contient la liste des commandes unix ainsi que les outils utilisés. Vérifiez l'ensemble des informations présentes.
2. Editez également le fichier `${MORPHEO_TOPLEVEL}/Makefile.tools_path`. Il contient les chemins d'accès vers les différents outils.
3. Vous êtes fin prêt pour faire quelques tests

Test de l'installation