راهنمای نصب simplescaler

1) Get the following source files:

- simplesim-3v0d.tgz and simpletools-2v0.tgz from http://www.simplescalar.com/tools.html
- simpleutils-990811.tar.gz from http://www.eecs.umich.edu/mirv/
- gcc-2.7.2.3.ss.tar.gz from http://arch.cs.ucdavis.edu/RAD/gcc-2.7.2.3.ss.tar.gz
- flex-2.5.4a.tar.gz from http://flex.sourceforge.net/

2) type in terminal: flex -V . if you get “flex version 2.5.4” , **skip this step**.else:
Installing flex (I am not sure if this is really needed, but I did not try it without this step, and it does not harm as well):

```
cp flex-2.5.4a.tar.gz ~

cd ~

tar xvfz flex-2.5.4a.tar.gz

cd flex-2.5.4

./configure --prefix=/usr/local/flex-2.5.4

make

make install

export PATH=/usr/local/flex-2.5.4/bin/:$PATH
```

Now, test your flex:

```
flex –version
```

you should get “flex version 2.5.4”.

3) Setup some environment variables (assuming your home directory is /usr/local):

```
export IDIR=/usr/local/simplescalar

export HOST=i686-pc-linux

export TARGET=sslittle-na-sstrix
```

4) Create the directory “simplescalar” under your home directory, and copy all the four tarballs into it:

```
mkdir $IDIR

cp simplesim-3v0d.tgz $IDIR

cp simpletools-2v0.tgz $IDIR

cp simpleutils-990811.tar.gz $IDIR

cp gcc-2.7.2.3.ss.tar.gz $IDIR
```

5) Start installing packages in the following order:

a) simpletools:

Just un-tar the package file, and remove the old gcc folder:

```
cd $IDIR

tar xvfz simpletools-2v0.tgz

rm -rf gcc-2.6.3
```

b) simpleutils:

```
cd $IDIR

tar xvfz simpleutils-990811.tar.gz

./configure --host=$HOST --target=$TARGET --with-gnu-as --with-gnu-ld --prefix=$IDIR

make

make install
```

Problem: You may encounter the following error:

```
ldlex.l: In function `yy_input':
ldlex.l:589: error: `yy_current_buffer' undeclared (first use in
this function)
ldlex.l:589: error: (Each undeclared identifier is reported only
once
ldlex.l:589: error: for each function it appears in.)
```

How to solve? Change in "ld/ldlex.l": "if (yy_current_buffer->yy_input_file)" to "if (YY_CURRENT_BUFFER->yy_input_file)"

c) simplesim:

```
cd $IDIR

tar xvfz simplesim-3v0d.tgz
cd simplesim-3.0

make config-pisa

make
```

you may test the installation of simplesim by:

```
./sim-safe tests/bin.little/test-math
```

d) gcc-2.7.2.3:

```
export PATH=$PATH:/usr/local/simplescalar/sslittle-na-sstrix/bin

./configure --host=$HOST --target=$TARGET --with-gnu-as --with-gnu-ld --prefix=$IDIR

make
```

Here, I got several error messages one after the other. I found many fixes on the web, which did the job perfectly. However, the thing that was never mentioned anywhere was to clean up and re-run the configure command after applying the fixes. So, for you to avoid going into all the troubles I've went through, I suggest doing the following changes at this stage:

** Change the **Makefile** at line **130**, by appending **–I/usr/include** to the end of the line.

** Edit line 60 of **protoize.c**, and replace "#include <varargs.h>" with "#include <stdarg.h>".

** To fix an error message about decl.c, saying “invalid lvalue in increment" in decl.c, do the following: Edit **obstack.h** at line **341** and change :

```
            *((void **)__o->next_free)++=((void *)datum);\
```

to

```
            *((void **)__o->next_free++)=((void *)datum);\
```

** To fix an error message about objc/sendmsg.c, edit **objc/sendmsg.c**, and insert "#define INVISIBLE_STRUCT_RETURN 0" at line **35**, and delete lines **36 to 40** inclusive.

** Also run the following commands:

```
    cp ./patched/sys/cdefs.h ../sslittle-na-sstrix/include/sys/cdefs.h

    cp ../sslittle-na-sstrix/lib/libc.a ../lib/

    cp ../sslittle-na-sstrix/lib/crt0.o ../lib/
```

Next, re-run the configure command :

```
make clean; make distclean

./configure --host=$HOST --target=$TARGET --with-gnu-as --with-gnu-ld --prefix=$IDIR

make
```

That very last command will give you an error message, which requires you to :

** Edit insn-output.c :

- append ‘\’ at the end of at line 675

- append ‘\’ at the end of at line 750

- append ‘\’ at the end of at line 823

Now, run:

```
make

make enquire
```

```
../simplesim-3.0/sim-safe ./enquire -f > float.h-cross

make install
```

That should execute with no troubles.

With that, Simple Scalar has been installed, along with its tools, utils, and compilers. Let's now test compiling a new code and running it on top of simplescalar:

Create a new file, test.c, that has the following code:

```
#include<stdio.h>

main()

{

printf("Hello World!\n");

}
```

then compile it using the following command:

```
 $IDIR/bin/sslittle-na-sstrix-gcc hello.c
```

That should generate an a.out, which we will run over the simulator, copy this file to $IDIR/simplesim-s.3 and:

```
 $IDIR/simplesim-3.0/sim-safe a.out
```

In the output, you should be able to find the following:

```
sim: ** starting functional simulation **

Hello World!
```

References:

[1] http://fdemesmay.dyndns.org/cs/hacks/SimpleScalarInstallation

[2] http://www.comp.nus.edu.sg/~panyu/simplesim.htm

[3] http://research.iiit.ac.in/~abu_saad/simplescalarinstall.html