

COMP212 Computer Architecture

How to generate Verilog testvector from MIPS assembly code

To verify your CPU design, you need to write testvectors. For that, you should write some test programs and use those programs as testvectors. You will write the test programs in MIPS assembly language. The test programs are stored in Verilog memory model.

MIPS assembler and linker convert your assembly programs into machine codes. Thus, you should have MIPS assembler and linker installed first.

This note explains how to install MIPS assembler and linker using GNU bin utilities under Linux and how to generate Verilog testvector from MIPS assembly code

There are 2 options to install MIPS assembler and linker: The first option is to download the GNU binutils and build the MIPS assembler and linker. The second option is to simply download the MIPS assembler and linker pre-built under x86 (32-bit Fedora).

Option 1: Build the MIPS assembler, linker and Cross-Compiler Yourself

Build the MIPS assembler and linker

1. Download the latest GNU bin utility, binutils-2.19.1.tar.bz2, either from <http://ftp.gnu.org/gnu/binutils/> or from the class website at http://comedu.korea.ac.kr/~suhtw/teaching/comp212_CA/binutils-2.19.1.tar.bz2
2. Create directories where the downloaded file is copied and where the compiled binaries are located.
 - “mkdir mips-elf” // this is a directory where compiled binaries are located
 - “mkdir classes”
 - “cd classes”
 - “mkdir comparch”
 - “cd comparch”
 - “cp ~/Downloads/binutils-2.19.1.tar.bz2 .”
3. Untar (Uncompress) the file
 - “tar jxvf binutils-2.19.1.tar.bz2 “
4. Move to the bin utility directory
 - “cd binutils-2.19.1”
5. Set environment for compilation
 - “export TARGET=mips-elf”
 - “export PREFIX=~/\$TARGET”
 - “export PATH=\$PATH:\$PREFIX/bin” // include the new path, so you can run executable anywhere
6. Check if your environment set correctly
 - “set” // scroll up and down to check HOST, TARGET, and PATH
7. Compile the bin utilities
 - “./configure --target=\$TARGET --prefix=\$PREFIX”
 - “make”

- “make install”
8. Check if the bin utilities are located in ~/mips-elf’
 - “cd ~/mips-elf”
 - “ls -al” // The MIPS assembler and linker are “mips-elf-as” and “mips-elf-ld, respectively”

Build the MIPS cross-compiler

1. Download the GCC source, gcc-4.1.1.tar.bz2, either from <http://ftp.gnu.org/gnu/gcc/gcc-4.1.1> or from the class website at http://comedu.korea.ac.kr/~suhtw/teaching/comp212_CA/gcc-4.1.1.tar.bz2
2. Go to the comparch directory and copy the downloaded file
 - “cd ~/classes/comparch”
 - “cp ~/Download/gcc-4.1.1.tar.bz2 .”
3. Untar (Uncompress) the file
 - “tar jxvf gcc-4.1.1.tar.bz2 “
4. Move to the gcc-4.1.1 directory
 - “cd gcc-4.1.1”
5. Set environment for compilation // skip this if you are working in the same shell (you already did this in the first page)
 - “export TARGET=mips-elf”
 - “export PREFIX=~/\$TARGET”
 - “export PATH=\$PATH:\$PREFIX/bin” // include the new path, so you can run executable anywhere
6. Compile the gcc cross-compiler
 - “./configure --target=\$TARGET --prefix=\$PREFIX --without-headers --with-newlib --with-gnu-as --with-gnu-ld”
 - “make all-gcc”
 - “make install-gcc”
7. Check if the cross-compiler is located in ~/mips-elf’
 - “cd ~/mips-elf”
 - “ls -al” // The MIPS cross-complier is “mips-elf-gcc”

```

[Taeveon@localhost bin]$ pwd
/home/Taeveon/mips-elf/bin
[Taeveon@localhost bin]$ ls -al
total 41908
drwxrwxr-x 2 Taeveon Taeveon 4096 2009-08-13 10:55 .
drwxrwxr-x 10 Taeveon Taeveon 4096 2009-08-12 18:03 ..
-rwxr-xr-x 1 Taeveon Taeveon 2800802 2009-08-12 17:58 mips-elf-addr2line
-rwxr-xr-x 2 Taeveon Taeveon 2929715 2009-08-12 17:58 mips-elf-ar
-rwxr-xr-x 2 Taeveon Taeveon 4298826 2009-08-12 17:58 mips-elf-as
-rwxr-xr-x 2 Taeveon Taeveon 317618 2009-08-13 10:55 mips-elf-c++
-rwxr-xr-x 1 Taeveon Taeveon 2778046 2009-08-12 17:58 mips-elf-c++filt
-rwxr-xr-x 1 Taeveon Taeveon 317275 2009-08-13 10:55 mips-elf-cpp
-rwxr-xr-x 2 Taeveon Taeveon 317618 2009-08-13 10:55 mips-elf-g++
-rwxr-xr-x 2 Taeveon Taeveon 312712 2009-08-13 10:55 mips-elf-gcc
-rwxr-xr-x 2 Taeveon Taeveon 312712 2009-08-13 10:55 mips-elf-gcc-4.1.1
-rwxr-xr-x 1 Taeveon Taeveon 438699 2009-08-12 18:21 mips-elf-gcc-4.3.0
-rwxr-xr-x 1 Taeveon Taeveon 15931 2009-08-13 10:55 mips-elf-gccbug
-rwxr-xr-x 1 Taeveon Taeveon 325340 2009-08-13 10:55 mips-elf-gcj
-rwxr-xr-x 1 Taeveon Taeveon 299071 2009-08-13 10:55 mips-elf-gcjh
-rwxr-xr-x 1 Taeveon Taeveon 82685 2009-08-13 10:55 mips-elf-gcov
-rwxr-xr-x 1 Taeveon Taeveon 319950 2009-08-13 10:55 mips-elf-gfortran
-rwxr-xr-x 1 Taeveon Taeveon 299107 2009-08-13 10:55 mips-elf-gjnih
-rwxr-xr-x 1 Taeveon Taeveon 330491 2009-08-13 10:55 mips-elf-jcf-dump
-rwxr-xr-x 1 Taeveon Taeveon 147517 2009-08-13 10:55 mips-elf-jv-scan
-rwxr-xr-x 2 Taeveon Taeveon 3702546 2009-08-12 17:58 mips-elf-ld
-rwxr-xr-x 2 Taeveon Taeveon 2834107 2009-08-12 17:58 mips-elf-nm
-rwxr-xr-x 2 Taeveon Taeveon 3367687 2009-08-12 17:58 mips-elf-objcopy
-rwxr-xr-x 2 Taeveon Taeveon 3674644 2009-08-12 17:58 mips-elf-objdump
-rwxr-xr-x 2 Taeveon Taeveon 2929710 2009-08-12 17:58 mips-elf-ranlib
-rwxr-xr-x 1 Taeveon Taeveon 570851 2009-08-12 17:58 mips-elf-readelf
-rwxr-xr-x 1 Taeveon Taeveon 2826335 2009-08-12 17:58 mips-elf-size
-rwxr-xr-x 1 Taeveon Taeveon 2802728 2009-08-12 17:58 mips-elf-strings
-rwxr-xr-x 2 Taeveon Taeveon 3367678 2009-08-12 17:58 mips-elf-strip
[Taeveon@localhost bin]$

```

Option 2: Simply download pre-built (under x86 with 32-bit Fedora) MIPS assembler, linker, and cross-compiler

1. Download the pre-built (under x86 with 32-bit Fedora) MIPS assembler and linker from http://comedu.korea.ac.kr/~suhtw/teaching/comp212_CA/mips-elf.tar.bz2
2. Copy the download file
 - “cp ~/Download/mips-elf.tar.bz2 .”
3. Untar (Uncompress) it
 - “tar jxvf mips-elf.tar.bz2”
4. Change the directory and check if you have necessary files as shown in the above figure.
 - “cd mips-elf/bin”
 - “ls -al” // The MIPS assembler and linker are “mips-elf-as” and “mips-elf-ld”

Option 3: Simply download pre-built (Cygwin under x86) MIPS assembler, linker, and cross-compiler

1. Download the pre-built (under x86 with 32-bit Fedora) MIPS assembler and linker from http://comedu.korea.ac.kr/~suhtw/teaching/comp212_CA/mips-elf-cygwin.tar.bz2
2. Copy the download file
 - “cp ~/Download/mips-elf-cygwin.tar.bz2 .”
3. Untar (Uncompress) it
 - “tar jxvf mips-elf-cygwin.tar.bz2”
4. Change the directory and check if you have necessary files as shown in the above figure.
 - “cd mips-elf/bin”
 - “ls -al” // The MIPS assembler and linker are “mips-elf-as” and “mips-elf-ld”

Download example assembly code and generate testvector

1. Download an example from
http://comedu.korea.ac.kr/~suhtw/teaching/comp212_CA/example.tar.bz2
2. Untar (uncompress) it
 - “cd ~/classes/comparch/”
 - “cp ~/Download/example.tar.bz2 .”
 - “tar jxvf example.tar.bz2”
3. Generate a Verilog testvector
 - “cd example”
 - “more testvec.s” // Check out what kind of assembly program you are assembling
 - “make” // you should be able to see the following files

```

[Taeueon@localhost comparch]$ cd example_assembly/
[Taeueon@localhost example_assembly]$ ll
total 20
-rwxrwxr-x 1 Taeueon Taeueon 409 2009-08-06 00:49 bin2hex.perl
-rwxr-xr-x 1 Taeueon Taeueon 823 2009-08-12 17:33 Makefile
-rw-rw-r-- 1 Taeueon Taeueon 162 2009-08-05 19:47 test.lds
-rwxr-xr-x 1 Taeueon Taeueon 151 2009-08-05 19:47 testvec.lds
-rw-rw-r-- 1 Taeueon Taeueon 294 2009-08-13 11:46 testvec.s
[Taeueon@localhost example_assembly]$ make
"/mips-elf/bin/mips-elf-as -g testvec.s -o testvec.o
"/mips-elf/bin/mips-elf-ld -N -X -Itestvec.lds testvec.o -o testvec
"/mips-elf/bin/mips-elf-objdump -xS testvec > testvec.dump
"/mips-elf/bin/mips-elf-objcopy -O binary testvec testvec.bin
./bin2hex.perl > testvec.hex
[Taeueon@localhost example_assembly]$ ll
total 56
-rwxrwxr-x 1 Taeueon Taeueon 409 2009-08-06 00:49 bin2hex.perl
-rwxr-xr-x 1 Taeueon Taeueon 823 2009-08-12 17:33 Makefile
-rw-rw-r-- 1 Taeueon Taeueon 162 2009-08-05 19:47 test.lds
-rwxrwxr-x 1 Taeueon Taeueon 4752 2009-08-13 11:56 testvec
-rwxrwxr-x 1 Taeueon Taeueon 4144 2009-08-13 11:56 testvec.bin
-rw-rw-r-- 1 Taeueon Taeueon 1630 2009-08-13 11:56 testvec.dump
-rw-rw-r-- 1 Taeueon Taeueon 9324 2009-08-13 11:56 testvec.hex
-rwxr-xr-x 1 Taeueon Taeueon 151 2009-08-05 19:47 testvec.lds
-rw-rw-r-- 1 Taeueon Taeueon 928 2009-08-13 11:56 testvec.o
-rw-rw-r-- 1 Taeueon Taeueon 294 2009-08-13 11:46 testvec.s
[Taeueon@localhost example_assembly]$

```

- Check out “testvec.hex” // You can use testvec.hex as a Verilog testvector for MIPS

[illegible]