

Course IN2075: Microprocessors
Winter Course 16/17

Exercise 4: Architecture and Emulation

In these exercises, we will install an ARM/MIPS emulator and write some code for those architectures.

1. Install the GNU **gcc/g++** cross compilers for ARM(64) and MIPS(64) on your computer in order to generate code for those architectures.
2. Install **qemu** on your computer (unless you have already done so):

```
sudo apt-get install qemu
```

Create a hard disk using

```
qemu-img create -f raw armdisk.img 2G
```

and

```
qemu-img create -f raw mipsdisk.img 2G ,  
respectively.
```

In order to check which ARM architectures can be emulated, type

```
qemu-system-<architecture> -machine help
```

This also works for other architectures, e.g. PPC, SPARC., etc. .

3. Look for an appropriate Linux installation media on the Internet and generate on installation for ARM and one for MIPS, respectively.
4. Connect to your virtual ARM/MIPS box:

If you start your virtual machine by appending

```
... -redir tcp:5555::22 &
```

you can connect to it through

```
ssh -p 5555 localhost
```

File transfer works accordingly.

Exercise 5: Toupper on ARM or other Achitectures

Use file **ex1.tar.gz** from exercise 1 and extract it into a subfolder:

```
tar xzvf ex1.tar.gz
```

Continue as in exercise 1 using the previously installed cross-compiler and transfer the binary. Run the binary on the virtual machine and adjust **toupper**'s command line parameters accordingly.

Try to optimise your **toupper**'s performance through whatever looks appropriate. Use your favourite architecture (apart from x86 :)).