

An emulated Raspberry Pi using QEMU

This gives you access to an emulated ARM based Raspberry Pi machine running the Raspbian operating system using the free emulator QEMU directly on your PC without virtualization.

1. Installation QEMU:

Download and install QEMU:

<https://www.qemu.org/download/#windows>

Edit the windows system environmental variable PATH and add the path to QEMU e.g.:

C:\Program Files\qemu

2. Download QEMU Raspberry Pi kernel:

Download QEMU Raspberry Pi kernel as a zip archive:

<https://github.com/dhruvvyas90/qemu-rpi-kernel>

3. Download Raspberry Pi OS Buster Lite:

https://downloads.raspberrypi.org/raspios_oldstable_lite_armhf/images/raspios_oldstable_lite_armhf-2023-02-22/2023-02-21-raspios-buster-armhf-lite.img.xz

Unzip both to a newly created directory e.g. qpi

Rename the folder qemu-rpi-kernel-master to qemu-rpi-kernel

4. Start QEMU:

Open „Windows Command Prompt“ or „Windows Power Shell“ change to your directory with the image (e.g. qpi) and enter:

```
qemu-system-arm -M versatilepb -cpu arm1176 -m 256 \  
-drive "file=./2023-02-21-raspios-buster-armhf-lite.img,if=none,index=0,media=disk,format=raw,id=disk0" \  
-device "virtio-blk-pci,drive=disk0,disable-modern=on,disable-legacy=off" \  
-net "user,hostfwd=tcp::5022-:22" \  
-dtb ./qemu-rpi-kernel/versatile-pb-buster-5.4.51.dtb \  
-kernel ./qemu-rpi-kernel/kernel-qemu-5.4.51-buster \  
-append 'root=/dev/vda2 panic=1' -no-reboot -net nic
```

4. Log in:

```
raspberrypi login: pi
Password: raspberry
```

Now we have to enable the ssh server on the Pi, type:

```
sudo systemctl enable ssh
sudo systemctl start ssh
```

Download lab1.zip from Moodle.

Open a new Power shell / Terminal and type:

```
scp -P 5022 C:\Users\username\Downloads\lab1.zip pi@127.0.0.1:/home/pi
```

Confirm with yes and use the Pi's password: raspbian
This will copy the file lab1.zip into the emulated Raspberry Pi's file system.

Now type:

```
ssh pi@127.0.0.1 -p 5022
```

Log in (user: pi Password:raspbian) and start using the emulated Raspberry Pi, see screenshots in this document.

To transfer any files back from your emulated Pi to your PC (e.g. file test.txt):

```
scp -P 5022 pi@127.0.0.1:/home/pi/test.txt ./
```

You only have to copy the assembly (.s) files back to submit them into Moodle. But if you want to zip a whole directory, you can use:

```
tar cvf Listings.zip ./Listings
```

To zip the directory „Listings“ into the zip file „Listings.zip“

To safely shut the Pi down, type:

```
sudo halt
```

5. Alternatively use Raspberry Pi OS with GUI

If your computer is fast enough, you can also try and download the Raspberry Pi OS Image with graphical user interface:

https://downloads.raspberrypi.org/raspios_oldstable_armhf/images/raspios_oldstable_armhf-2023-02-22/2023-02-21-raspios-buster-armhf.img.xz

Put it in the same directory as before, and start it:

```
qemu-system-arm -M versatilepb -cpu arm1176 -m 256 \
-drive "file=./2023-02-21-raspios-buster-armhf.img,if=none,index=0,media=disk,format=raw,id=disk0" \
-device "virtio-blk-pci,drive=disk0,disable-modern=on,disable-legacy=off" \
-net "user,hostfwd=tcp::5022-:22" \
-dtb ./qemu-rpi-kernel/versatile-pb-buster-5.4.51.dtb \
-kernel ./qemu-rpi-kernel/kernel-qemu-5.4.51-buster \
-append 'root=/dev/vda2 panic=1' -no-reboot -net nic
```

Use the unix commands `ls` and `cd` to navigate through the file system.

```
martin — pi@raspberrypi: ~/lab1/1_add — ssh pi@1...
[pi@raspberrypi:~ $ ls
lab1 lab1.zip
[pi@raspberrypi:~ $ cd lab1/
[pi@raspberrypi:~/lab1 $ ls
1_add 1_add_ref 2_absolute
[pi@raspberrypi:~/lab1 $ cd 1_add
[pi@raspberrypi:~/lab1/1_add $ ls
add_asm.s add.c Makefile
[pi@raspberrypi:~/lab1/1_add $ nano add_asm.s
```

Use the editor nano to edit the assembly program:
`nano add_asm.asm`

```
martin — pi@raspberrypi: ~/lab1/1_add — ssh pi@127.0.0.1 -p 5022 — 77x20
GNU nano 3.2 add_asm.s Modified

/* add_asm.s 1.4.2020 */
/* Calculate the sum of r0 and r1 */
/* The result is return by r0 */

.global add_asm @ Program Entry Point

add_asm:

    @Please Continue
    adds r0,r1
    mov PC,LR @ back to the calling function

.data

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell
```

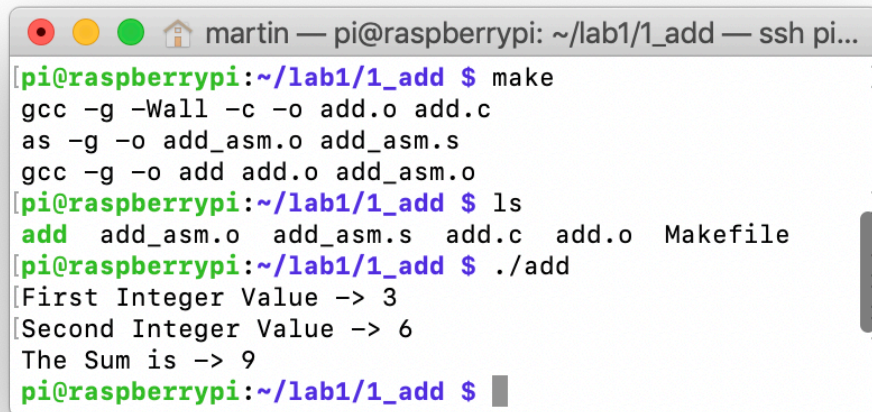
Use `ctrl-x` to exit nano

To compile the C program and the assembly program type:

`make`

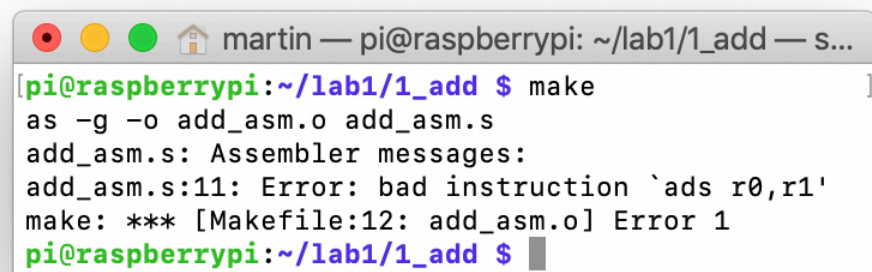
If there was no error compiling, you can run the program now by typing:

`./add`



```
martin — pi@raspberrypi: ~/lab1/1_add — ssh pi...
pi@raspberrypi:~/lab1/1_add $ make
gcc -g -Wall -c -o add.o add.c
as -g -o add_asm.o add_asm.s
gcc -g -o add add.o add_asm.o
pi@raspberrypi:~/lab1/1_add $ ls
add  add_asm.o  add_asm.s  add.c  add.o  Makefile
pi@raspberrypi:~/lab1/1_add $ ./add
First Integer Value -> 3
Second Integer Value -> 6
The Sum is -> 9
pi@raspberrypi:~/lab1/1_add $
```

If you get an error message while compiling, use nano again to fix it. In this example there is a typing error (ads instead of adds) in line 11 in the file `add_asm.s` (your assembly source code)



```
martin — pi@raspberrypi: ~/lab1/1_add — s...
pi@raspberrypi:~/lab1/1_add $ make
as -g -o add_asm.o add_asm.s
add_asm.s: Assembler messages:
add_asm.s:11: Error: bad instruction `ads r0,r1'
make: *** [Makefile:12: add_asm.o] Error 1
pi@raspberrypi:~/lab1/1_add $
```

To safely shut down your virtual Pi type:

`sudo halt`

Important Links:

eBook: Raspberry Pi Assembly Language Programming : ARM Processor Coding / by Stephen Smith:

This eBook can be downloaded inside the RWU Network (!):

[https://bsz.ibs-bw.de/aDISWeb/app?service=direct/0/Home/\\$DirectLink&sp=SOPAC16&sp=SAKSWB-IdNr1681717395](https://bsz.ibs-bw.de/aDISWeb/app?service=direct/0/Home/$DirectLink&sp=SOPAC16&sp=SAKSWB-IdNr1681717395)
<https://doi.org/10.1007/978-1-4842-5287-1>

Printed Book in the University Library:

Raspberry Pi assembly language : RASPBIAN [beginners]; hands on guide / Bruce Smith
[https://bsz.ibs-bw.de/aDISWeb/app?service=direct/0/Home/\\$DirectLink&sp=SOPAC16&sp=SAKSWB-IdNr770531830](https://bsz.ibs-bw.de/aDISWeb/app?service=direct/0/Home/$DirectLink&sp=SOPAC16&sp=SAKSWB-IdNr770531830)

ARM Instruction Set for Raspberry Pi 1 (ARMv6) and Pi 2 (ARMv7)

http://infocenter.arm.com/help/topic/com.arm.doc.100076_0200_00_en/xhv1530702676759.html

Raspberry Pi

<https://www.raspberrypi.org>

Wikipedia entry for Raspberry Pi

https://de.wikipedia.org/wiki/Raspberry_Pi
https://en.wikipedia.org/wiki/Raspberry_Pi

QEMU

<https://www.qemu.org/download/>

Homebrew for Mac:

<https://brew.sh>

To install QEMU on MacOS, install home-brew, then open Terminal and type:
brew install qemu

Kernels for QEMU and Raspberry Pi:

<https://github.com/dhruvvyas90/qemu-rpi-kernel>