

# Instructions for installing RTDE (Real-Time Data Exchange)

Last Update: 04.02.2024

These instructions describe the installation and execution of the RTDE (Real-Time Data Exchange). There are several ways to make the RTDE work. In this guide we only deal with methods that use Ubuntu 22.04. This can be explained for permission reasons. The accessibility of various parameters that need to be changed for the RTDE are much easier to access and change on Ubuntu 22.04.

The RTDE can also be installed in other ways, but these are not described here. If you are interested in other methods, this [website](#) can help. Since the UR-RTDE was developed by independent programmers, there is no guarantee that the RTDE will be installed successfully, as it has often not been updated and may therefore contain errors or missing information.

This guide deals **exclusively** with two RTDE installation methods:

1. A virtual machine (with Ubuntu 22.04) on a computer with Windows as an operating system
2. A computer with Ubuntu 22.04 as operating system

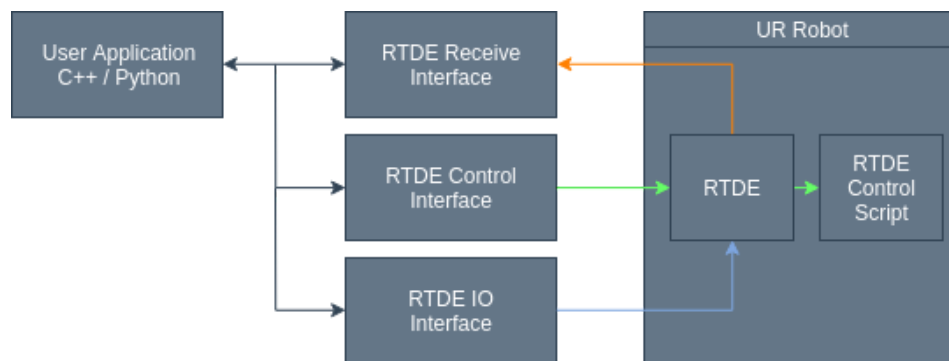


Illustration of how the RTDE Library works

The illustration above shows how the RTDE works. The RTDE makes it possible to establish real-time communication between the computer and the robot. For this purpose, either a Python or C++ library is installed, which interacts with the robot in three different ways. These interaction options are called interfaces.

- **RTDE Receive Interface** is used to receive data from the UR robot.
- **RTDE Control Interface** is used to send data to the robot from the computer or to forward commands to the robot.
- **RTDE IO Interface** is used to control external elements of the robot (e.g. the gripper) or to read their data.

These three interaction options are then forwarded directly from the robot through port 30004 in order to act on the robot's RTDE, which has an influence on the RTDE control script.

## Table of contents

<b>Compatibility.....</b>	<b>4</b>
Robots.....	4
Operating system.....	4
<b>Virtual machine.....</b>	<b>5</b>
Installation.....	5
Virtual machine configuration.....	5
Ubuntu 22.04 Installation.....	8
<b>Docker installation.....</b>	<b>12</b>
Activation of the Ubuntu real-time kernel.....	12
Docker configuration.....	13
<b>Python library installation.....</b>	<b>15</b>
<b>IP addresses.....</b>	<b>16</b>

# Compatibility

## Robots

Compatibility was only tested on the UR10 robot in the robotics laboratory at Munich University of Applied Sciences. However, according to the documentation, these robots should also be compatible:

- CB-Series from CB3/CB3.1 Software 3.3
- e-Series

On the official website of [Universal Robots](https://www.universal-robots.com) you can check whether a robot belongs to one of these series and is therefore compatible or not.

## Operating system

The compatibility of the operating system was only tested for Ubuntu 22.04 in the robotics laboratory at Munich University of Applied Sciences. The RTDE documentation states that the following operating systems are also compatible:

- Ubuntu 16.04 (Xenial Xerus)
- Ubuntu 18.04 (Bionic Beaver)
- Ubuntu 20.04 (Focal Fossa)
- macOS 10.14 (Mojave)
- Windows 10 Pro x64

As these have not been tested, no responsibility is accepted for testing on operating systems other than Ubuntu 22.04.

## Virtual machine

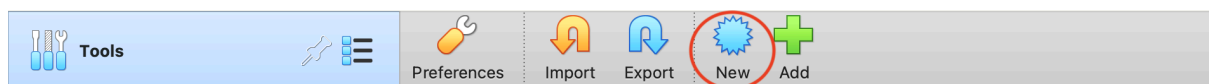
This part only concerns *1. a virtual machine (with Ubuntu 22.04) on a computer with Windows as an operating system.*

### Installation

Two elements are required for the installation:

- The virtual machine software. It is recommended to use this [virtual machine](#).
- The ISO file of the Ubuntu 22.04 operating system, which can be found [here](#).

### Virtual machine configuration



When you open the VirtualBox Manager software, you should see the bar. To create a virtual machine, press the *New* button circled in red.

**Virtual machine Name and Operating System**

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name:  ✓

Folder:  ✓

ISO Image:  ✓

Edition:

Type:  64

Version:

☒ Skip Unattended Installation

ⓘ You have selected to skip unattended guest OS install, the guest OS will need to be installed manually.

Expert Mode Back **Next** Cancel

The following parameters must now be entered:

- **Name**, this can be freely chosen. Here it is specified as *Ubuntu*.
- **Folder**, do not touch this link as it is specified by default.

- **ISO Image**, the ISO file of Ubuntu 22.04 must be indicated for this link.
- ***Skip Unattended Installation***, this field is not checked by default, but it must be checked so that the Ubuntu installation can be carried out correctly.

Once the various information has been taken into account, you can click *Next*.

**Hardware**

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 4 MB 16384 MB 4096 MB

Processors: 1 CPU 8 CPUs 4

☐ Enable EFI (special OSes only)

Back Next Cancel

The performance data of the virtual machine must be entered here. It is recommended to enter 4096 MB for the base memory and 4 processors.

## Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

☒ Create a Virtual Hard Disk Now

Disk Size:

4,00 MB

2,00 TB

50,00 GB

☐ Pre-allocate Full Size

☐ Use an Existing Virtual Hard Disk File

Empty

☐ Do Not Add a Virtual Hard Disk

Back




Next

Cancel

It is recommended to set the disk size to 50.00 GB.

## Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

 <b>Machine Name and OS Type</b>	
Machine Name	Ubuntu
Machine Folder	/Users/francois-melchiorletourneau/VirtualBox VMs/Ubuntu
ISO Image	/Users/francois-melchiorletourneau/Downloads/ubuntu-22.04.3-desktop-a...
Guest OS Type	Ubuntu (64-bit)
Skip Unattended Install	true
 <b>Hardware</b>	
Base Memory	4096
Processor(s)	4
EFI Enable	false
 <b>Disk</b>	
Disk Size	50,00 GB
Pre-allocate Full Size	false

Back

Finish

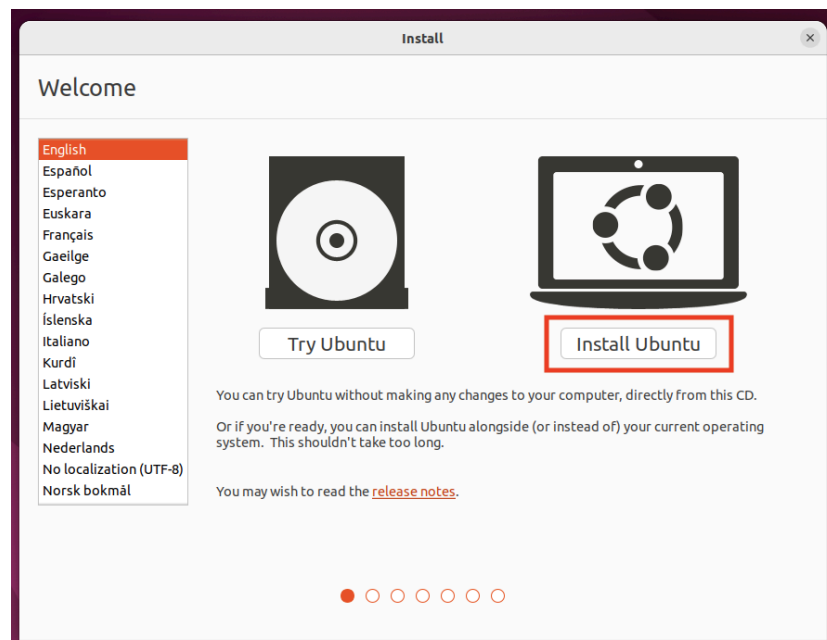
Cancel

Finally, press the *Finish* button. The virtual machine is now configured.

## Ubuntu 22.04 Installation

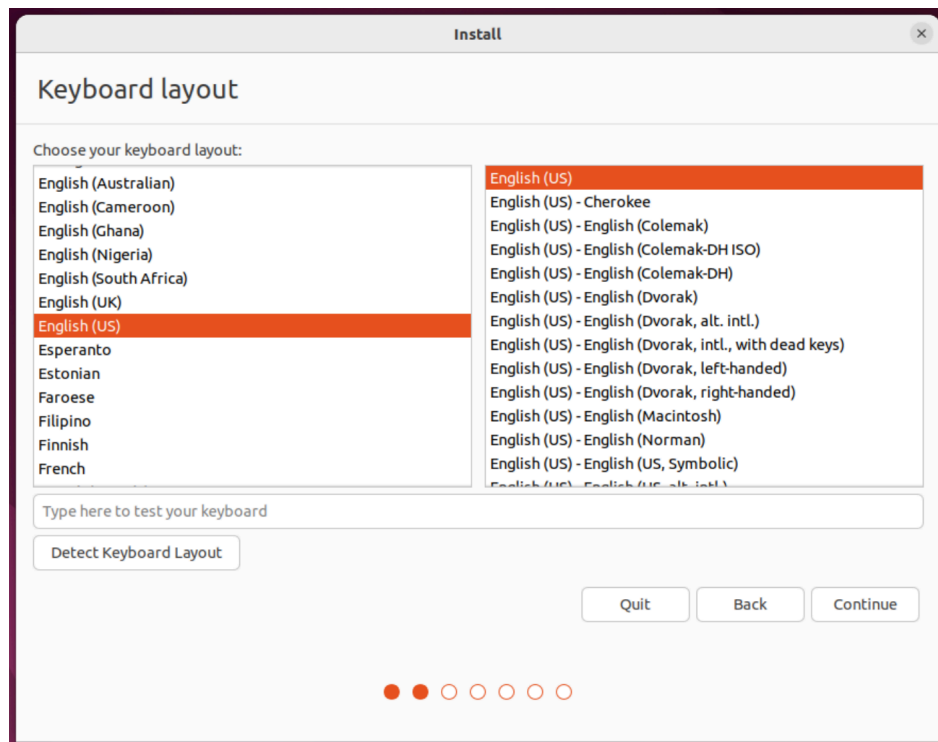


To start the virtual machine and begin the installation of Ubuntu 22.04, press the *Start* button marked in red.

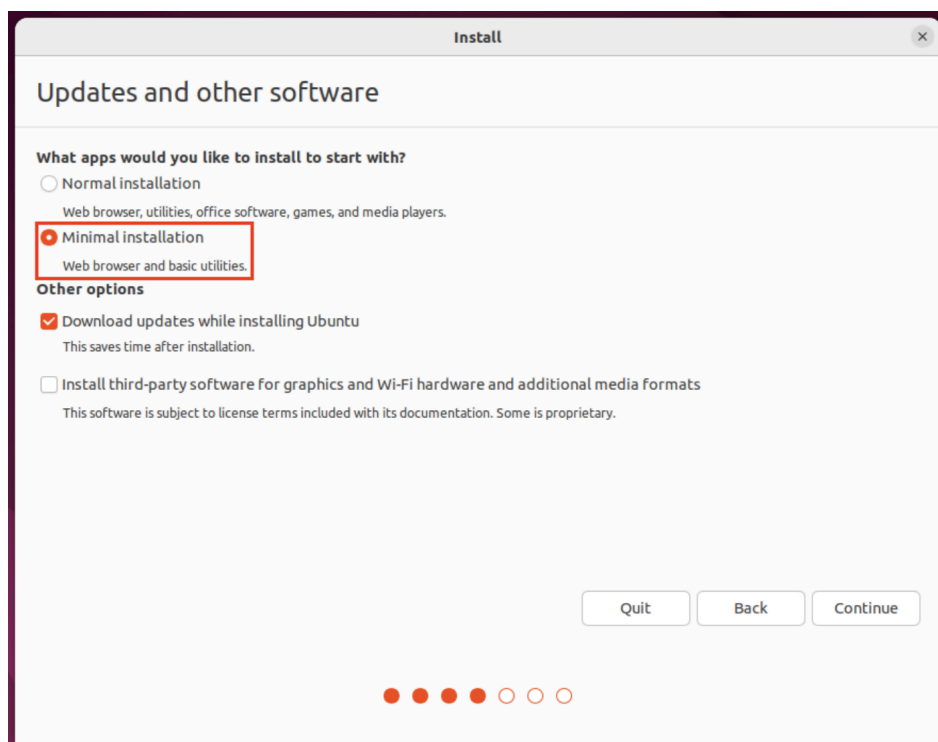


This interface follows after starting the virtual machine. *Install Ubuntu* must be clicked here. The language can be selected in the left-hand column. In this case, the language is set to English.

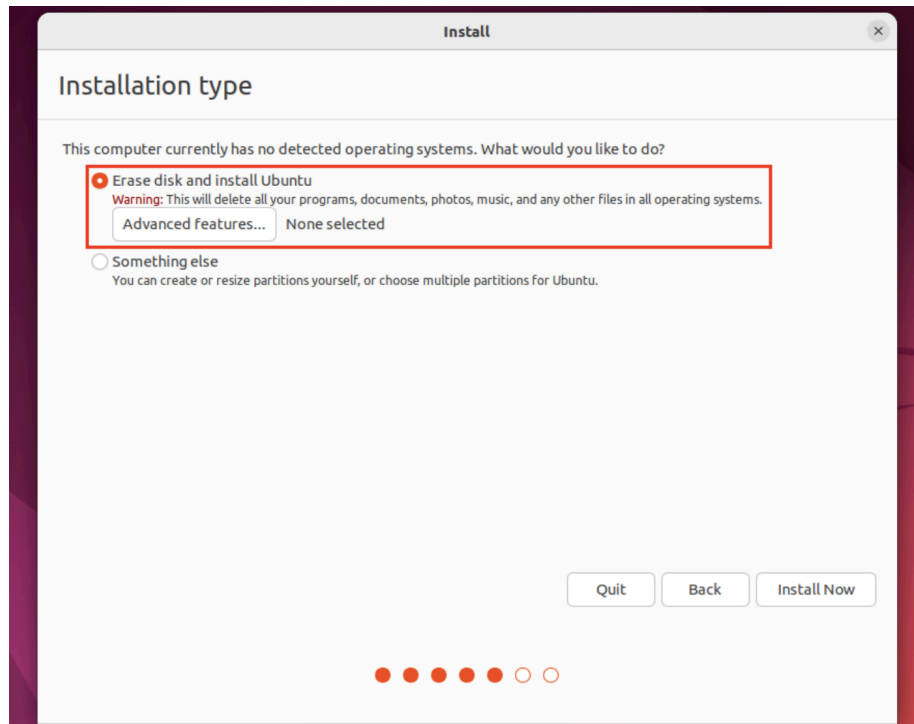




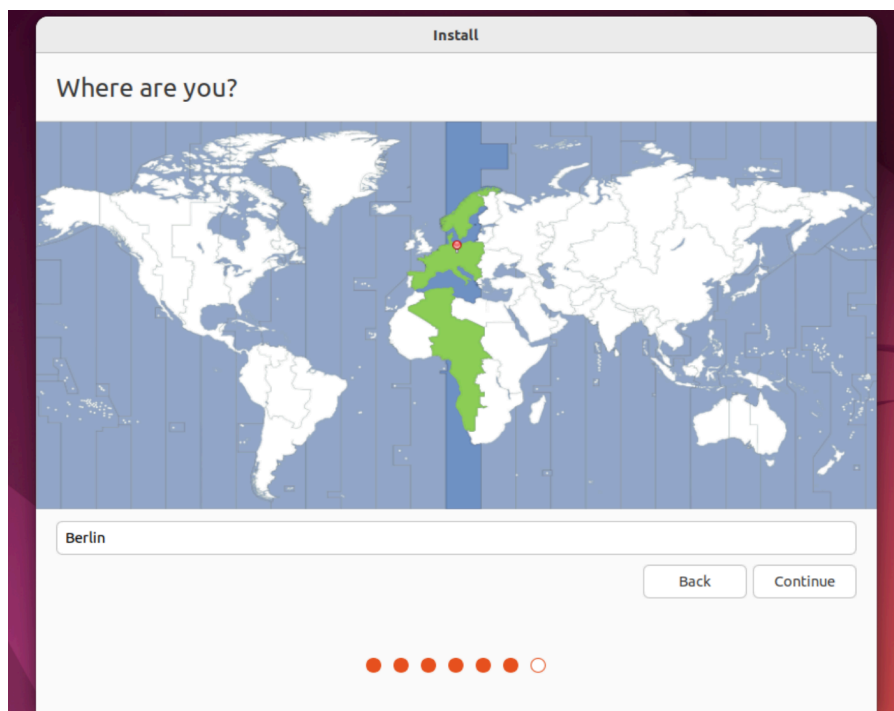
The keyboard setting can be selected here. Thanks to the *Detect Keyboard Layout* button, the keyboard of the device can be found.



Select *Minimal Installation*.



Select *Erase disk and install Ubuntu*.



Choose timezone.

Install

Who are you?

Your name:  ✓

Your computer's name:  ✓  
The name it uses when it talks to other computers.

Pick a username:  ✓

Choose a password:  Weak password

Confirm your password:  ✓

☐ Log in automatically

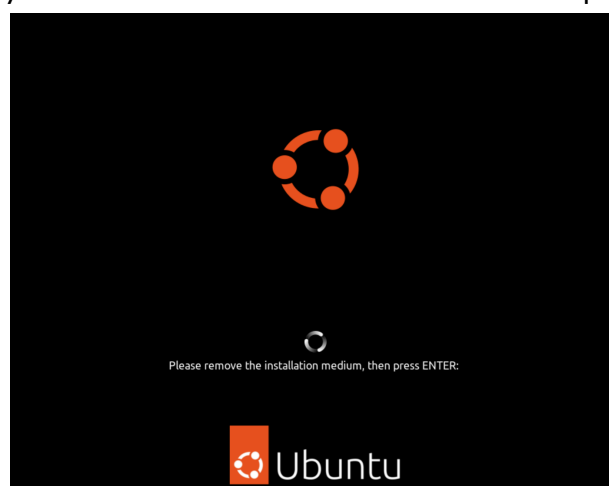
☒ Require my password to log in

☐ Use Active Directory  
You'll enter domain and other details in the next step.

Back Continue

Enter username and password. Then select the *Require my password to log in* button.

Finally, the virtual machine will restart and display the screen below. Then press the ENTER key. The Ubuntu 22.04 installation is complete.



## Docker installation

### Activation of the Ubuntu real-time kernel

**REMINDER: This method only works for Ubuntu 22.04**

In this part, all "commands" in the black boxes must be entered in the Ubuntu terminal.

Instead of <free token> you have to type in the Free Personal Token of your Ubuntu account. To create an Ubuntu account, follow this [link](#). To [create](#) or [copy](#) the Free Personal Token.

```
sudo ua attach <free token>
```

The version needs to be tested afterwards.

```
ua version
```

If the result is something lower than 27.8, then this command must be entered in the terminal:

```
sudo apt install ubuntu-advantage-tools=27.8~22.04.1
```

This command activates the real-time kernel, which enables constant communication between the robot and computer in the RTDE.

```
ua enable realtime-kernel --beta
```

The computer must then be restarted. This command can be entered or it can also be done manually.

```
reboot
```

## Docker configuration

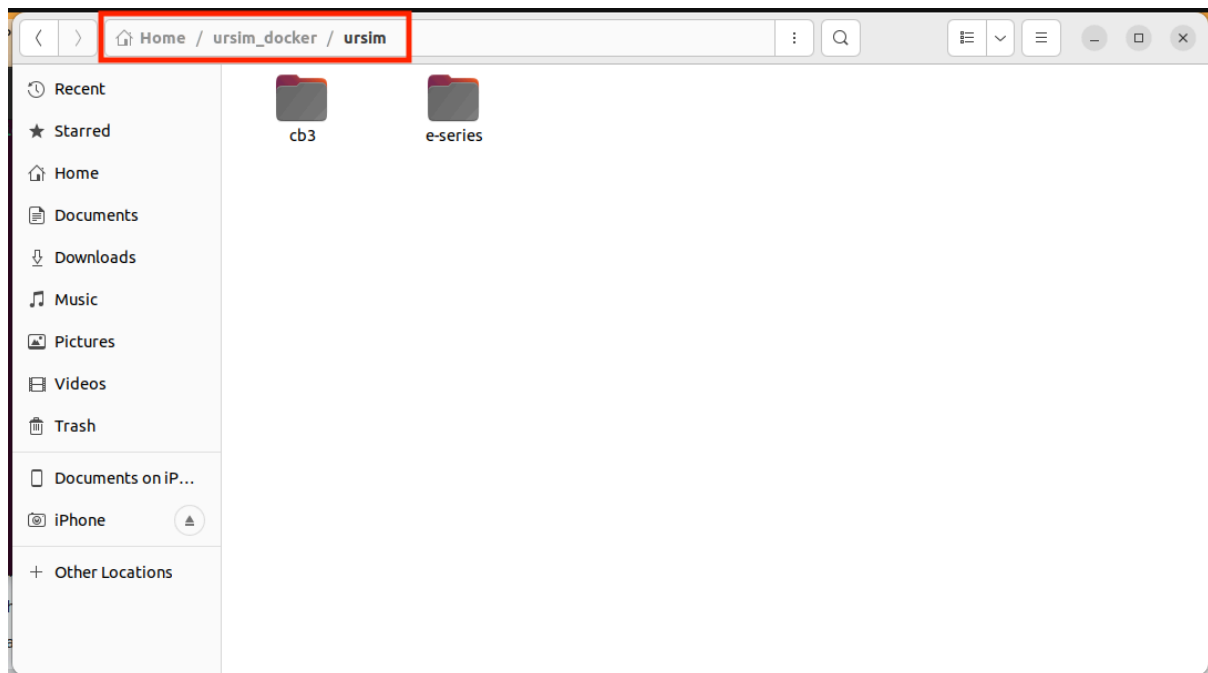
The terminal must be opened again.

```
git clone https://github.com/urnsk/ursim_docker.git
```

The following commands must be executed in sequence:

```
sudo apt update
sudo apt install docker.io
sudo systemctl start docker
sudo systemctl enable docker
sudo systemctl status docker
sudo usermod -aG docker $USER
su - $USER
```

Two documents are to be modified in the *Files* app. First, the *Home* folder must be opened. Then the *ursim\_docker* folder and then the *ursim* folder. The screen should look like this:



The red rectangle can help you to be sure that you are in the right folder.

The next step is a little more complicated. In both folders (*cb3* and *e-series*) there is a document *Dockerfile*. When you open it, you have to add the following line:

```
FROM ubuntu:latest
```

Replace with the following:

```
FROM ubuntu:20.04
```

**REMINDER: This action must be carried out in both folders.**

The following must then be entered in the terminal:

```
cd ursim_docker/
```

```
docker build ursim/e-series -t myursim --build-arg VERSION=5.11.1.108318 --build-arg  
URSIM="https://s3-eu-west-1.amazonaws.com/ur-support-site/118926/URSim_Linux-5.11.1.1083  
18.tar.gz"
```

After installation:

```
cd ~
```

```
docker run --rm -it -p 5900:5900 -p 29999:29999 -p 30001-30004:30001-30004 myursim
```

After the Docker image has been loaded once, the Ctrl+C key can be pressed to stop the Docker.

## Python library installation

The terminal must be opened and the following commands executed:

```
sudo add-apt-repository ppa:sdurobotics/ur_rtde  
sudo apt-get update  
sudo apt install librtde librtde-dev
```

```
pip install --user ur_rtde
```

```
sudo apt-get install libboost-all-dev
```

```
git clone https://gitlab.com/sdurobotics/ur_rtde.git  
cd ur_rtde  
git submodule update --init --recursive  
mkdir build  
cd build  
cmake ..  
make  
sudo make install
```

The terminal can then be closed.

## IP addresses

To be able to program the robot using the UR-RTDE Python Library, the IP addresses of the robot and computer must be set to enable them to communicate with each other.

For example: The computer has the IP address 169.254.217.10

→ The robot is given the following IP address so that both can communicate:  
169.254.217.11