



## Einführung zur Toolchain

Rechnerorganisation Praktikum WiSe 23/24 | Architektur eingebetteter Systeme |  
Installation, Verwendung, Beispiel

---



## Gliederung

### Allgemein

Was ist unsere Toolchain?

### Installation der Toolchain

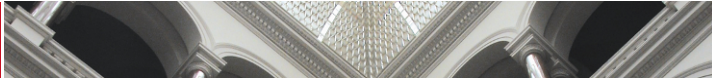
Toolchain auf den Fakultätsservern

Toolchain auf dem eigenen Rechner



## Was ist unsere Toolchain?

- Programme, die wir zur Entwicklung (also zum Kompilieren, Ausführen und Debuggen) benötigen
- bei uns:
  - `ghdl`: zum Kompilieren, Simulieren und Testen
  - `gtkwave`: zum Betrachten der Waveform und zum Debuggen
  - `git`: zur Abgabe und Versionsverwaltung
  - `make`: zur Automatisierung des Build Prozesses
  - *einen Code-Editor eurer Wahl*
  - Vivado: zur Synthese (optional und nur für das letzte Blatt)



## Wie kommen wir an die Toolchain?

### **zwei Möglichkeiten, die Toolchain zu nutzen:**

1. auf den Servern der Fakultät
  - entweder direkt vor Ort oder per Remote Desktop Verbindung/SSH (aus dem Uninetz!)
  - volle Unterstützung und Hilfe, wenn etwas nicht geht
2. auf dem eigenen PC
  - Installation je nach Betriebssystem unterschiedlich
  - eingeschränkte Unterstützung und Hilfe, wenn etwas nicht geht

### **vollständige Dokumentation in ISIS:**

<https://isis.tu-berlin.de/mod/resource/view.php?id=1679816>



## Toolchain auf den Fakultätsservern



## Remote Zugriff auf die Server der Fakultät

### Zugang nur im Uni-Netz möglich

Außerhalb des Uni-Netzes kann eine VPN-Verbindung verwendet werden, siehe:  
<https://www.tu.berlin/go59668/>

#### Windows: „Remote Desktop Connection“

- normalerweise vorinstalliert

#### Linux: „Remmina“

- Installation notwendig
- unter Ubuntu z.B.:  
`sudo apt install remmina`

#### Konfiguration (für Windows und Linux identisch):

- **Protokoll:** RDP Remote Desktop Protokoll
- **Server / Computer:** `ubu20.eecsit.tu-berlin.de`
- **Benutzername:** Euer tubit-Username
- **Benutzerpasswort:** Euer tubit-Passwort



## Aktivierung der Toolchain

1. Terminal öffnen
2. `source /afs/tu-berlin.de/units/Fak_IV/aes/scripts/hwptienv_basys` eingeben
3. Ausgabe „HWPTI tools loaded successfully“ abwarten
4. `ghdl --version` eingeben (es sollte Version GHDL 1.0.0 sein)

```
jcm3@cascada-ubu:~$ source /afs/tu-berlin.de/units/Fak_IV/aes/scripts/hwptienv_basys
HWPTI tools loaded successfully
jcm3@cascada-ubu:~$ ghdl --version
GHDL 1.0.0 (1.0.0.r0.g2fb2384.dirty) [Dunoon edition]
  Compiled with GNAT Version: Community 2021 (20210519-103)
  LLVM code generator
  Written by Tristan Gingold.

Copyright (C) 2003 - 2021 Tristan Gingold.
GHDL is free software, covered by the GNU General Public License.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```



## Toolchain auf dem eigenen Rechner



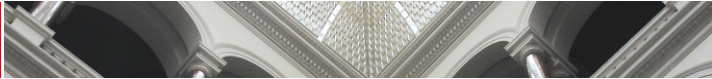


## Installation – allgemein

- nur unter Linux möglich
- unter Windows kann allerdings WSL verwendet werden
  - WSL (= Windows Subsystem for Linux) stellt unter Windows eine Linux-Umgebung bereit
  - Installation über Microsoft Store
  - „Ubuntu 22.04“ oder neuer verwenden

## Minimale Version

Es ist in jedem Fall mindestens GHDL **Version 1.0.0** erforderlich.



## Installation – unter Linux

- Ubuntu 22.04 „Jammy Jellyfish“ oder neuer empfohlen (Debian 11 „Bullseye“ oder neuer geht auch)
  - Installation über `sudo apt install ghdl-llvm`
- andere Distributionen können GHDL über der Paketmanager installieren, **wenn die Version 1.0.0 oder neuer verfügbar ist**
- andernfalls manuelle Installation notwendig (einfach beim ersten Praktikumstermin ansprechen!)
  - verschiedenste Downloads verfügbar:  
<https://github.com/ghdl/ghdl/releases/tag/v3.0.0>
- Version mit `ghdl --version` prüfen



## Beispiel