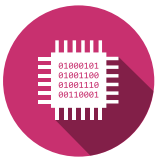
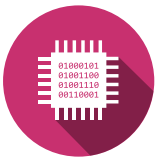



# Einleitung zur Benutzung der VM

# Inhalt




# 1 Ziel

Dieses Dokument ist es aufzuzeigen was eine Virtuelle Maschine genau ist, wie man VirtualBox installiert und zusätzlich wie man die Virtuelle Maschine für die ElN Labors importiert und benutzt.

# 2 Einleitung

Als virtuelle Maschine (VM) wird in der Informatik die Software-technische Kapselung eines Rechnersystems innerhalb eines lauffähigen Rechnersystems bezeichnet. Die virtuelle Maschine bildet die Rechnerarchitektur eines real in Hardware existierenden oder eines hypothetischen Rechners nach.

Die abstrahierende Schicht zwischen realem oder Host- bzw. Wirt-Rechner, auf dem die virtuelle Maschine ausgeführt wird, und der virtuellen Maschine wird Hypervisor oder Virtual Machine Monitor genannt und ihre Implementierung erfolgt rein hardwarebasiert, rein softwarebasiert oder durch eine Kombination aus beidem. Der Hypervisor erlaubt in der Regel den Betrieb mehrerer virtueller Maschinen gleichzeitig auf einem physischen Rechner.

Im Gegensatz zu Emulatoren werden virtuelle Maschinen direkt auf der CPU des Gastgeberrechners ausgeführt und nutzen üblicherweise die Virtualisierungsfunktionen der CPU. Bei Emulatoren wird die Ausführung rein als Software realisiert, wodurch auch eine andere Rechnerarchitektur als die des Gastgeberrechners nachgebildet werden kann.

Kurz gesagt ist eine Virtuelle Maschine ein kompletter Computer welche in einem Fenster des Host Rechners läuft

Für unseren Kurs benötigen wir dazu das Program VirtualBox von Oracle. Es gibt auf dem Markt noch weitere Hersteller welche Lösungen für Virtuelle Maschinen anbieten. Hierzu ein unvollständige Liste:

- VMWare
- Parallels
- Windows WSL
- Docker

# 3 Installation Virtual Box

Zuerst muss das Hypervisor program auf dem Host Rechner (eurem Computer) installiert werden.

## 3.1 Download

Laden Sie die neuste Version von VirtualBox auf deren Webseite https://www.virtualbox.org/ herunter


Abbildung 1 - VirtualBox Hauptseite

- Wählen Sie die Datei abhängig von eurem Betriebssystem aus

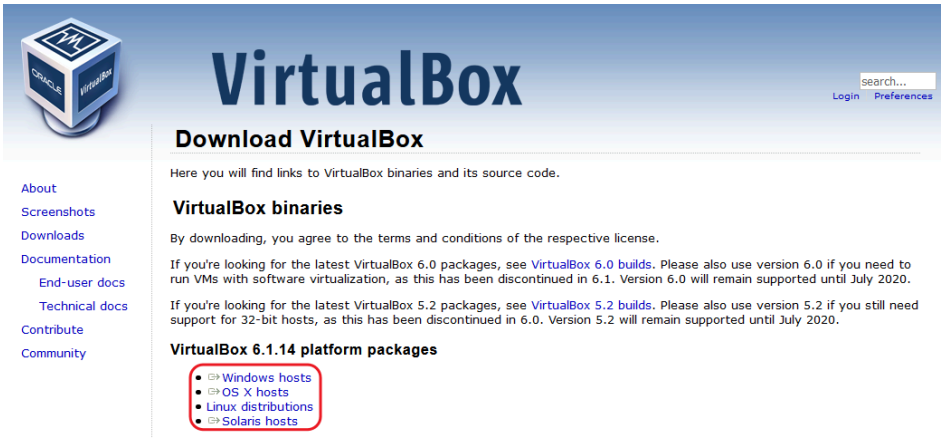



Abbildung 2 - VirtualBox Download Seite

- Speichert die Datei lokal auf dem Computer

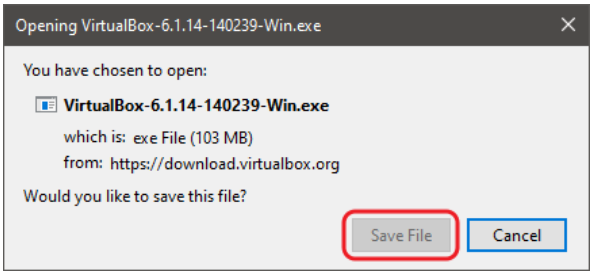



Abbildung 3 - Dialog zum speichern der Datei

## 3.2 Installation

Startet die soeben heruntergeladene Datei **`VirtualBox-<Version>-<Betriebssystem>.exe`**

z.B. **`VirtualBox-6.1.15-140239-Win.exe`**

- Drücken Sie 3 mal auf **`Next`**

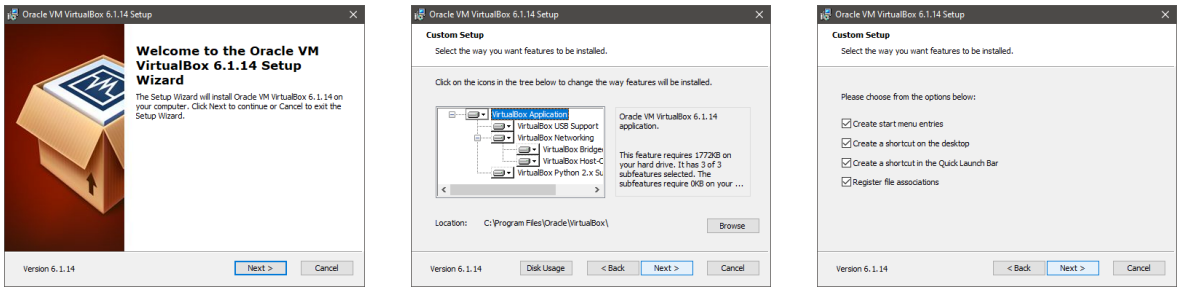



Abbildung 4 - VirtualBox Installationsschritt 1-3

- Danach werden automatisch ein Netzwerkinterface installiert. Dies wird benötigt damit die Virtuelle Maschine Zugriff auf das Netzwerk des Host Rechners erhält. Drücken Sie hier auf **`Yes`**.
- Danach wurde das Installationsprogram konfiguriert und Ihr könnt auf **`Install`** drücken

> Die Internetverbindung wird kurzzeitig unterbrochen

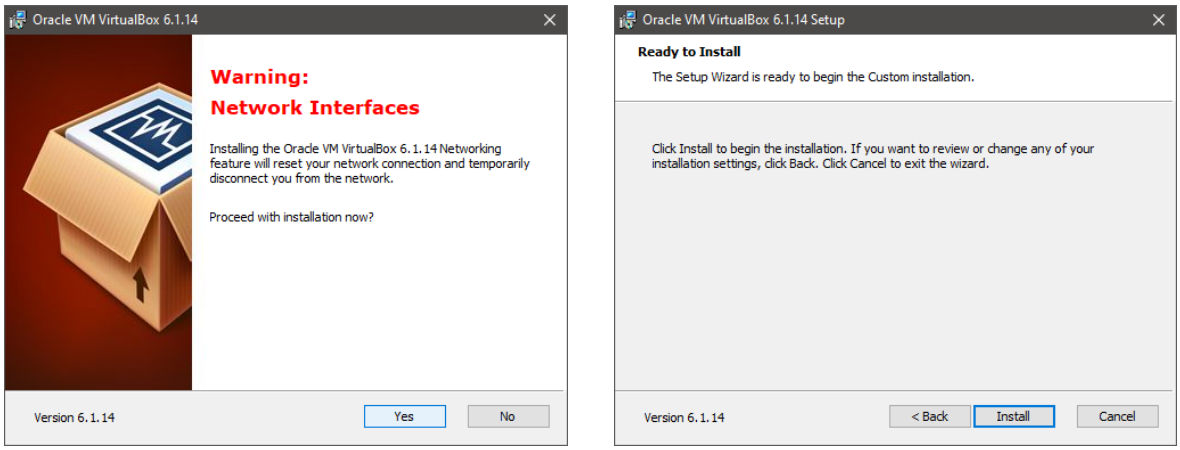
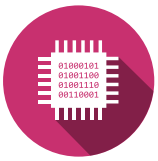



Abbildung 5 - VirtualBox Installationsschritt 4-5



Sie haben soeben erfolgreich VirtualBox installiert

## 3.3 Benutzung

Über das Desktopsymbol kann das Program gestartet werden.

Abbildung 6 - VirtualBox Icon

Das Hauptfenster erlaubt es Virtuelle Maschinen zu erstellen, importieren, löschen und zu starten.

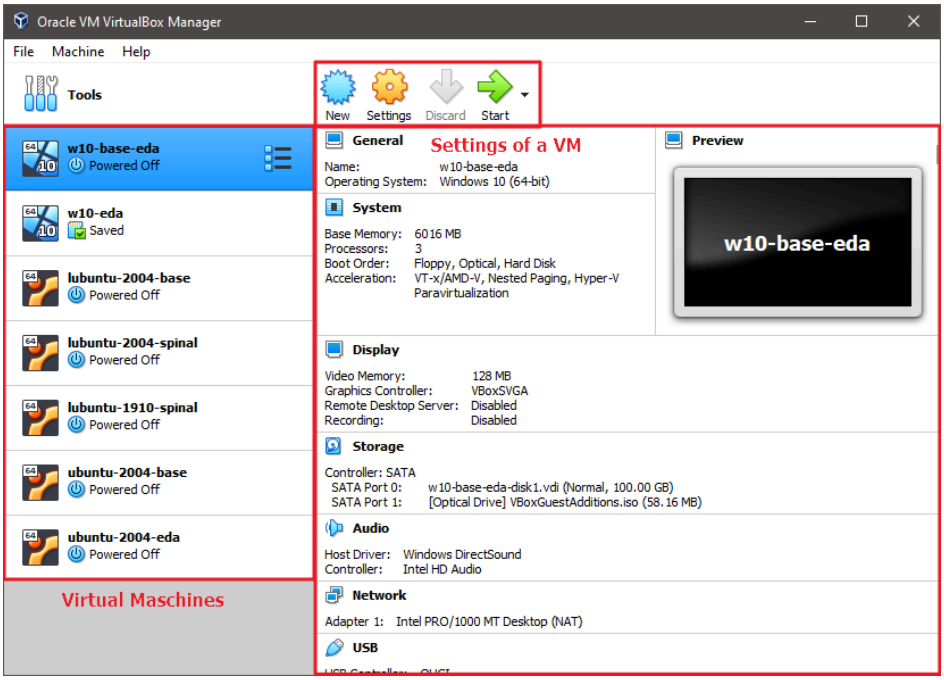



Abbildung 7 - VirtualBox GUI

# 4 Importieren einer Virtuellen Maschine

## 4.1 Importieren einer OVA Datei

Das Open Virtualization Format (OVF) ist ein offener Standard, um Virtual Appliances oder allgemeiner Software, die in virtuellen Maschinen läuft, zu verpacken und zu verteilen. Entwickelt wurde dieser Standard von der Distributed Management Task Force (DMTF).

Der Standard beschreibt ein „offenes, sicheres, portables, effizientes und erweiterbares Format für die Verpackung und Verteilung von Software, die in virtuellen Maschinen läuft“. Der OVF Standard ist nicht auf bestimmte Hypervisoren oder Prozessorarchitekturen beschränkt. Die Einheit, die in der Verpackung und Verteilung stattfindet, wird OVF Package genannt. Ein OVF Package kann ein oder mehrere virtuelle Systeme enthalten, von denen jedes in eine virtuelle Maschine eingespielt werden kann.

- Um das OVA File zu importieren drücken Sie auf **`File`** und **`Import Appliance`**

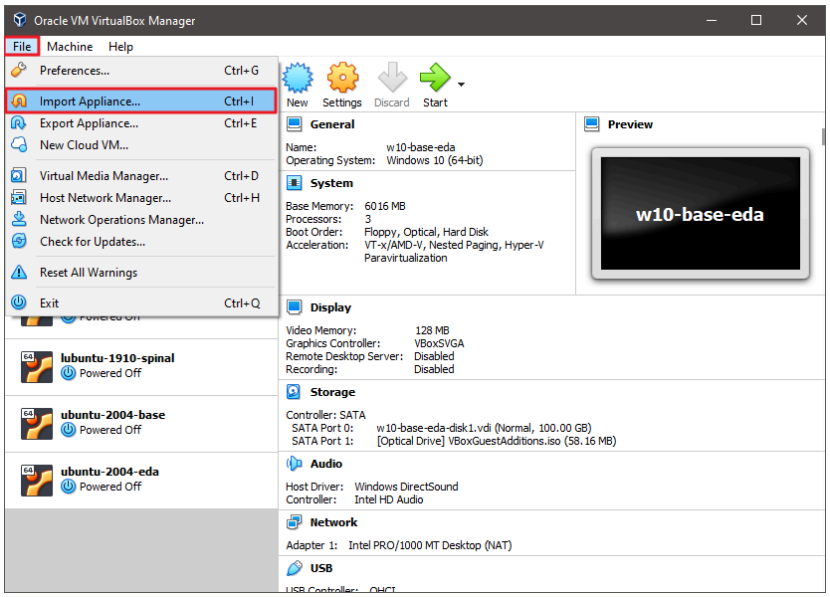



Abbildung 8 - Import Appliance

- Suchen Sie die **`*.ova`** Datei welche Ihnen übermittelt wurde aus
- Wahlen Sie den Speicherort der Virtuellen Maschinen aus und drücken Sie auf **`Import`**

> ⚠ Beachten Sie das die Virtuelle Maschine viel Speicherplatz einnehmen kann. Für das Standard DiD Labor image werden mindestens 26GB Speicherplatz benötigt.

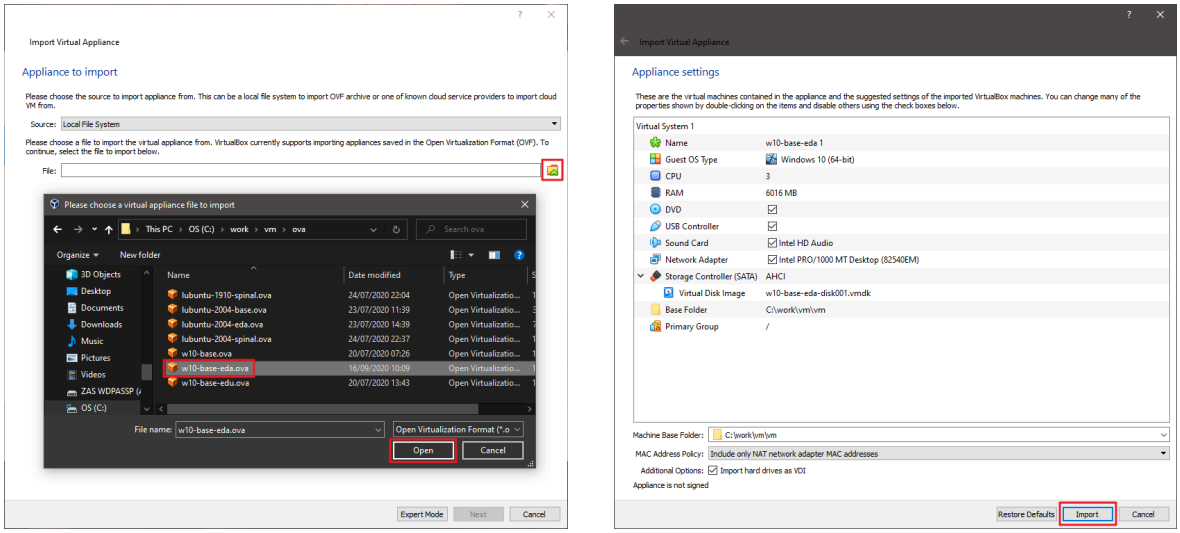



Abbildung 9 - Importieren einer existierenden OVA Datei

Sie haben soeben erfolgreich eine Virtuelle Maschine importiert

## 4.2 Konfiguration einer Virtuellen Maschine

Eine Virtuelle Maschine benutzt die Ressourcen (CPU, RAM, Speicherplatz) das Host Rechners. Je mehr Resourcen man der Virtuellen Maschinen zur Verfügung stellt desto weniger steht für den Host Rechner bereit. Wählen Sie eine Konfiguration von CPU-Kernen und RAM-Speicherplatz abhängig von eurem Computer aus.

- Öffnen Sie die Einstellungen Ihrer Virtuellen Maschine

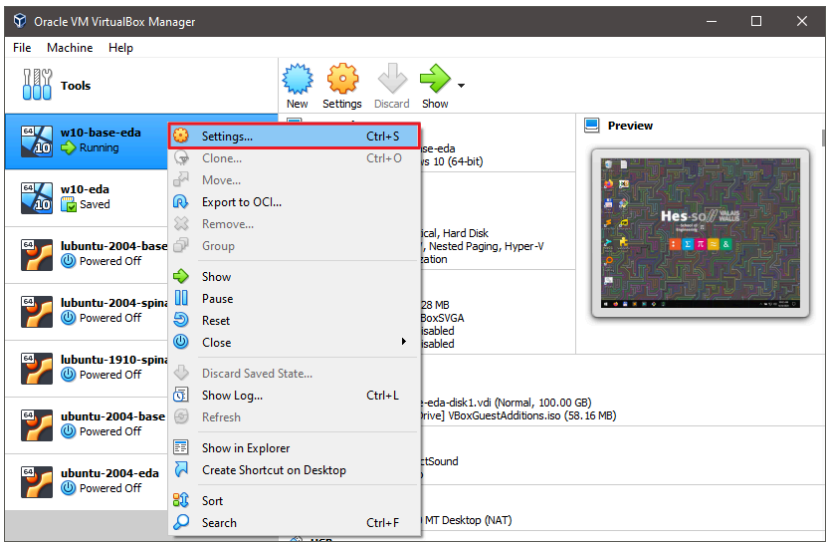



Abbildung 10 - Konfiguration einer Virtuellen Maschine

Eine empfohlen Konfiguration ist:

- >=2 CPU Kerne

- >=4 GB RAM

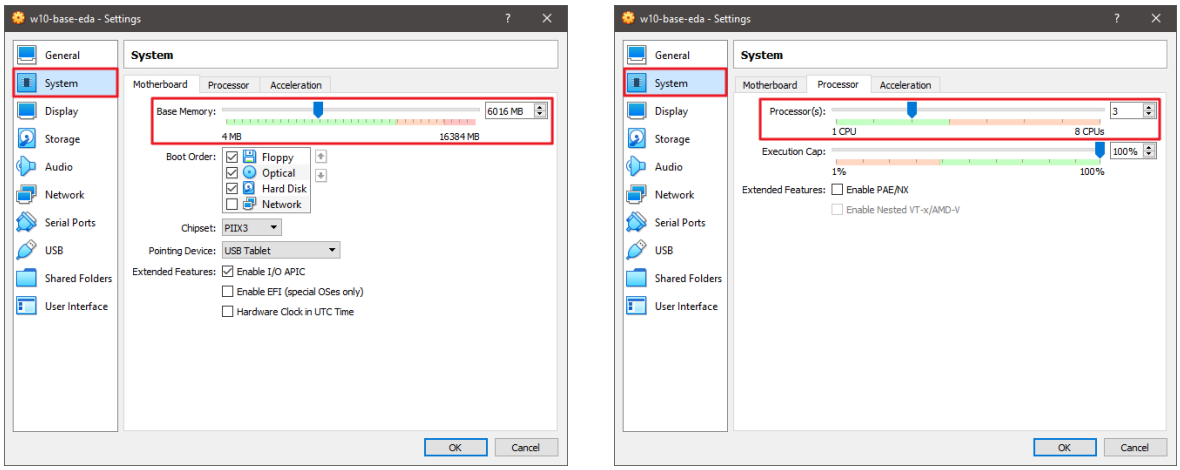
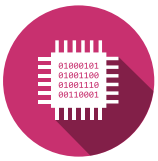



Abbildung 11 - Konfiguration von CPU-Kernen und RAM-Speicher




# 5 Benutzung der DiD Virtuellen Maschine

## 5.1 Login

Der Benutzter heisst **`uadmin`** und besitzt das Passwort **`eln`**.

Der Benutzer besitzt Administrator Rechte.
„With great power comes great responsibility“

## 5.2 Zur Verfügung stehende Programme

Das Image stellt viele unterschiedliche Programme zur Verfügung, nicht alle werden für das Labor benutzt.

### 5.2.1 Hauptprogramme

-  Mentor HDL-Designer - Konzeptionsprogram für Digitale Elektronik
-  Mentor Modelsim - Simulationsprogram für Digitale Elektronik

### 5.2.2 Optionale Programme

-  HESSO VPN Software um eine VPN Verbindung mit HEI herzustellen
-  Mozilla Firefox - Webbrowser
-  Hex Viewer und Editor
-  Dateibrowser
-  Software um schnell Dateien auf der Harddisk zu lokalisieren
-  Texteditor
-  Grafisches GIT Tool
-  PDF Viewer
-  Software um zu analysieren wo der Speicherplatz benutzt wird
-  Kommandozeile für git Befehle

## 5.3 Verbinden mit VPN

- Start PulseSecure
- Verbinden mit dem AAI Login **<vorname>.<nachname>**

> Das AAI-Login besteht aus den ersten 8 Zeichen Ihres Vor- und Nachnamens.

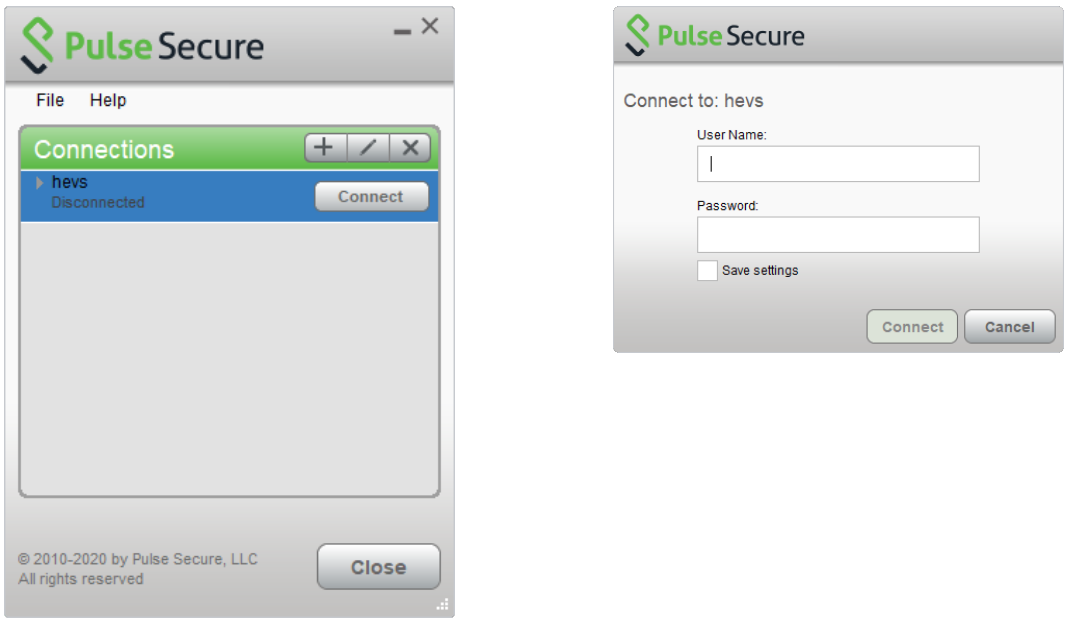



Abbildung 12 - Pulse Secure VPN Verbindung

## 5.4 Zugriff auf HEI Netzlaufwerke

- Sofern man nicht im Netz der Schule ist muss man zuerst die VPN Verbindung herstellen mit Pulse Secure Abschnitt 5.3
- Öffnen eines Dateiexplorers und auf das gewünschte Laufwerk klicken. Danach mit dem AAI Login einloggen **<vorname>.<nachname>**

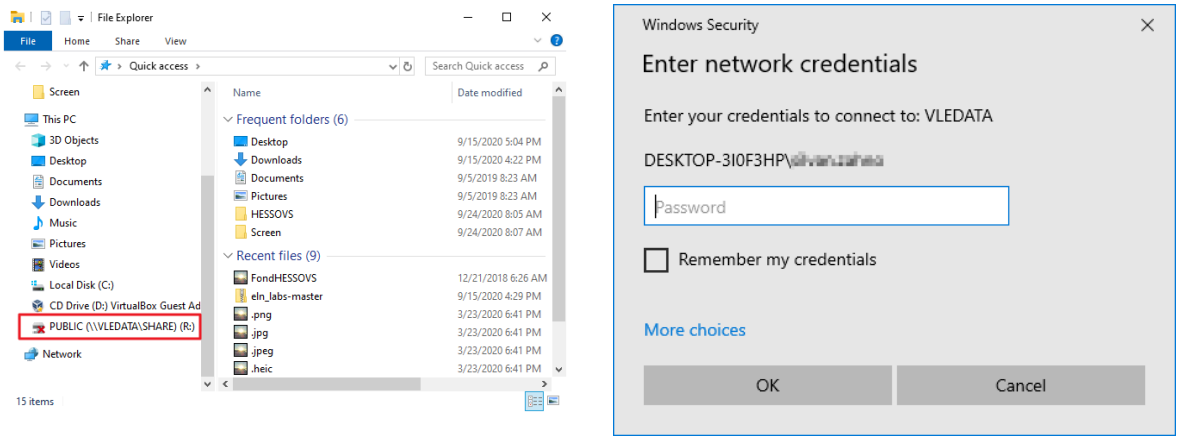



Abbildung 13 - Zugang zu den Lesern des HEI-Netzwerks

## 5.5 Benutzung des DiD Labors

Eine Kopie der DiD Labor Dateien findet Ihr bereits im Ordner **`C:\work\DiD_labs`**.

Der Link zum starten des Programs findet Ihr auch auf dem Desktop **`C:\Users\uadmin\Desktop\DiD_labs.lnk`**.


Abbildung 14 - Icon um starten von DiD Labs

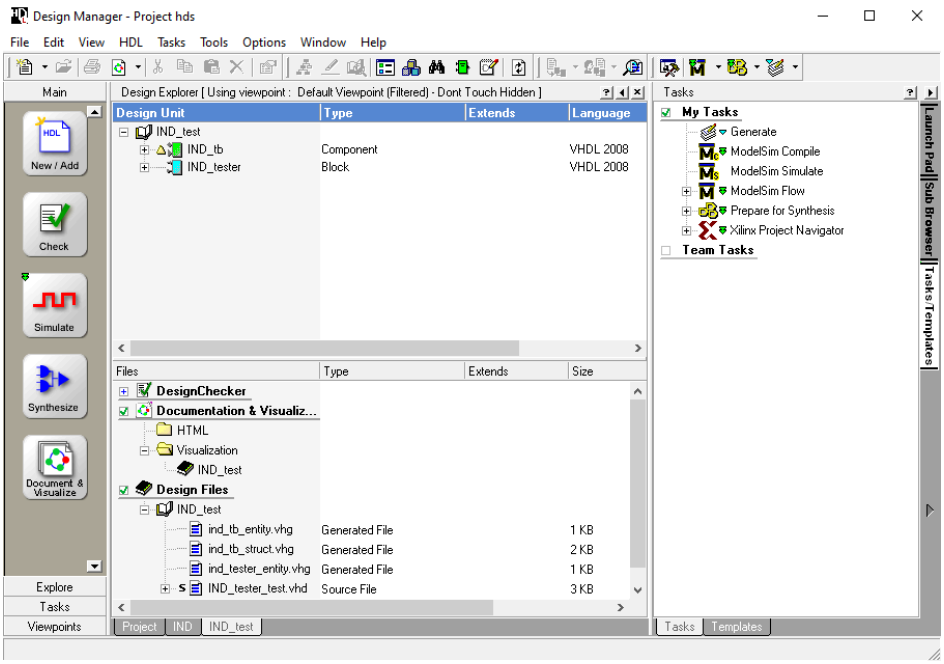
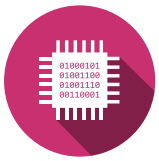



Abbildung 15 - HDL-Designer DiD Labs




# A GIT Befehle

Github git cheatsheet [1], [2]

## AA Änderungen überprüfen und eine Commit-Transaktion anfertigen

```
git status
```

Listet alle zum Commit bereiten neuen oder geänderten Dateien auf.

```
git diff
```

Zeigt noch nicht indizierte Dateiänderungen an.

```
git add [file]
```

Indiziert den derzeitigen Stand der Datei für die Versionierung.

```
git diff --staged
```

Zeigt die Unterschiede zwischen dem Index (“staging area”) und der aktuellen Dateiversion.

```
git reset [file]
```

Nimmt die Datei vom Index, erhält jedoch ihren Inhalt.

```
git commit -m "[descriptive message]"
```

Nimmt alle derzeit indizierten Dateien permanent in die Versionshistorie auf.

## AB Änderungen synchronisieren

Registrieren eines externen Repositories (URL) und Tauschen der Repository-Historie.

```
git fetch [remote]
```

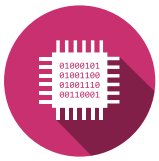
Lädt die gesamte Historie eines externen Repositories herunter.

```
git merge [remote]/[branch]
```

Integriert den externen Branch in den aktuell lokal ausgecheckten Branch.

```
git push [remote] [branch]
```

Pusht alle Commits auf dem lokalen Branch zu GitHub.




```
git pull
```

Pullt die Historie vom externen Repository und integriert die Änderungen.

# B Meistgebrauchten Git Befehle

## BA Start a working area

- `clone` - Clone a repository into a new directory
- `init` - Create an empty Git repository or reinitialize an existing one

## BB Work on the current change

- `add` - Add file contents to the index
- `mv` - Move or rename a file, a directory, or a symlink
- `reset` - Reset current HEAD to the specified state
- `rm` - Remove files from the working tree and from the index

## BC Examine the history and state

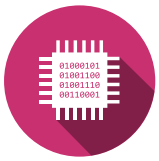
- `log` - Show commit logs
- `show` - Show various types of objects
- `status` - Show the working tree status

## BD Grow, mark and tweak your common history

- `branch` - List, create, or delete branches
- `checkout` - Switch branches or restore working tree files
- `commit` - Record changes to the repository
- `diff` - Show changes between commits, commit and working tree, etc
- `merge` - Join two or more development histories together
- `rebase` - Reapply commits on top of another base tip
- `tag` - Create, list, delete or verify a tag object signed with GPG

## BE Collaborate

- `fetch` - Download objects and refs from another repository
- `pull` - Fetch from and integrate with another repository or a local branch
- `push` - Update remote refs along with associated objects


01000101
01001100
01001110
00110001