

# 0 Hilfsmittel

## Student Group

First Name	Surname	Matrikel Nr.

## Table of Contents

- 0 Hilfsmittel** ..... 2
- 0.1 Einarbeiten in die Umgebung** ..... 2
- 0.2 KiCAD** ..... 2
- Installation von KiCad ..... 2
- Erster Start von KiCad ..... 2
- 0.2.1 Erstellen des ersten Schaltplans ..... 3
- Ziele ..... 3
- Video ..... 3
- Ziele ..... 3
- Video ..... 3
- Ziele ..... 4
- Video ..... 4
- 0.2.2 Entwickeln des Platinenlayouts ..... 4
- Ziele ..... 4
- Video ..... 4
- 0.2.3 Anlegen ein neues Bauteils ..... 4
- Ziele ..... 4
- Video ..... 4
- 0.3 MEXLE GitLab** ..... 5
- Installation der Software auf Ihrem Rechner** ..... 5
- Anmeldung beim MEXLE GitLab** ..... 5
- Installation der Software auf Ihrem Rechner** ..... 5
- Upload / Commit von Dateien** ..... 6
- Hinweise und Mögliche Fehler** ..... 8

# 0 Hilfsmittel

## 0.1 Einarbeiten in die Umgebung

- Beginnen Sie frühzeitig mit der Dokumentation. Dazu habe ich die [Randbedingungen für die Dokumentation](#) zusammengestellt.
- Für die Lötübungen im D041 ist die entsprechende [Laborordnung](#) zu beachten.
- Falls Sie Unklarheiten zu den Laboreinrichtungen für die Lötübungen haben, so finden Sie hier die [Anleitungen der Laborgeräte](#).

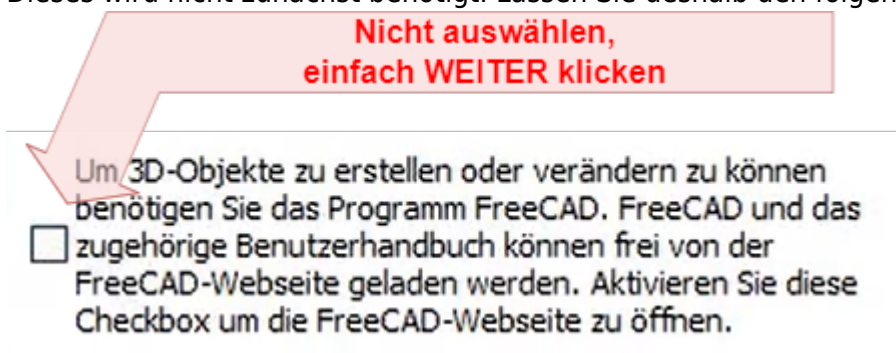
## 0.2 KiCAD

KiCad ist eine eine kostenlose, plattformübergreifende Open-Source-Software zur Erstellung von Schaltplänen und Leiterplatten-Layouts.

Es ist eine umfassende Suite von Tools, die von Hobbyisten und professionellen Entwicklern gleichermaßen verwendet wird.

### Installation von KiCad

1. Sie benötigen ca. 7 GB auf ihrer Festplatte / SSD Card.
2. KiCad kann von der offiziellen Website heruntergeladen werden:  
<https://www.kicad.org/download/>.  
Dort finden Sie Installationsanweisungen für Windows, Mac und Linux. Im Kurs wird KiCAD 9 verwendet.
3. Bitte installieren Sie KiCAD mit den voreingestellten Paketen (alle Bibliotheken, Demo Projekte). Im Notfall können die 3D Modelle nicht mit installiert werden. Dies reduziert den benötigten Speicher auf ca 1.5 GB. ABER: damit wird die Ansicht der Platinen beeinträchtigt.
4. Bei der Installation wird unter Umständen auch Microsoft Visual C++ installiert / aktualisiert. Auch dies kann abgebrochen werden, falls der Speicher nicht ausreicht.
5. Nach der Installation wird gefragt, ob das Programm FreeCAD installiert werden soll. Dieses wird nicht zunächst benötigt. Lassen Sie deshalb den folgenden Haken weg.



### Erster Start von KiCad

Nach der Installation können Sie KiCad starten. Die Benutzeroberfläche von KiCad besteht aus mehreren Tools, die im Projektmanager rechts angeordnet sind.

Eine gute Einführung dazu finden Sie in folgendem Video:



Daneben gibt es eine ausführliches Tutorial, sowie eine Beschreibung der einzelnen Werkzeuge auf der [KiCAD Webseite](#).

Es empfiehlt sich einige Einstellungen im Menu Einstellungen » Einstellungen ... vorzunehmen

1. Allgemein: Hier sollte der Pfad zu einem Texteditor hinterlegt werden, z.B.  
`C:\Windows\System32\notepad.exe`  
 Generell empfehle ich das Herunterladen und die Verwendung des kostenlosen Programms [Notepad++](#), welches für die Verarbeitung von Codedaiten optimiert ist: `C:\Program Files\Notepad++\notepad++.exe`
2. Maus und Touchpad: Hier können die Scroll-Gesten je nach der Verwendung von Maus oder Touchpad ausgewählt werden.
3. Symboleditor, Schaltplanneditor, Footprinteditor, Leiterplatteneditor
  1. Hier bietet es sich an unter Rasteroptionen » Rasterdarstellung kleine Kreuze zu wählen.
  2. Ebenso ist unter Cursor Anzeige » Cursorform Fensterfüllendes Fadenkreuz sinnvoll

## 0.2.1 Erstellen des ersten Schaltplans

### Ziele

Nach dieser Lektion sollten Sie:

1. wissen, wie man einen Schaltplan erstellt und Libraries benutzt

### Video



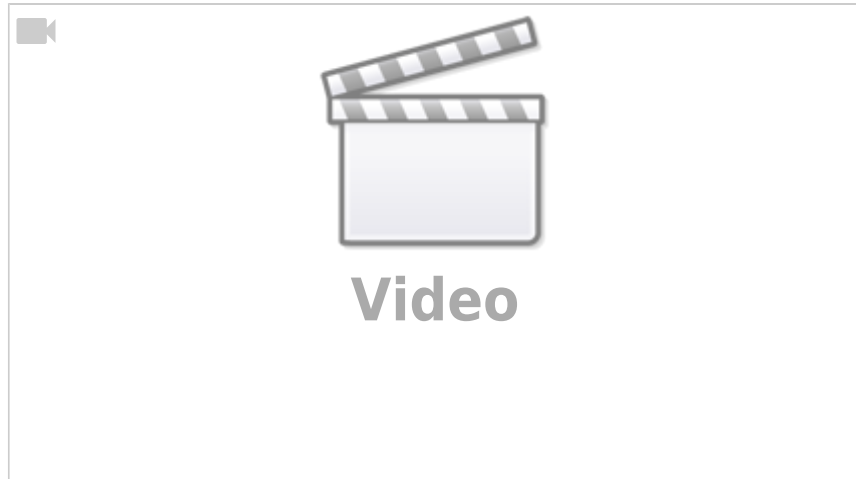
### Ziele

### Video

Nach dieser Lektion sollten

Sie:

1. wissen, wie die Funktionen Zoom, Suchen, Ersetzen, Drehen, Spiegeln nutzt.
2. wissen, wie man viele Werte von Komponenten (z.B. Footprints) auf einmal ändert.

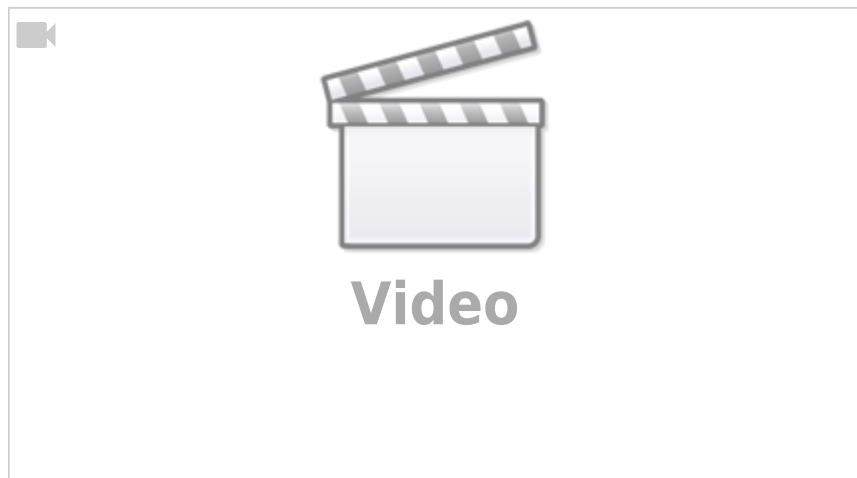


## Ziele

Nach dieser Lektion sollten Sie:

1. wie man ein erstes Schaltbild erzeugt
2. wie man die Komponenten annotiert ("nummeriert")

## Video



## 0.2.2 Entwickeln des Platinenlayouts

### Ziele

Nach dieser Lektion sollten Sie:

1. wissen, wie man aus einer Schaltung ein Platinenlayout erstellt

### Video

TBD

## 0.2.3 Anlegen ein neues Bauteils

### Ziele

Nach dieser Lektion sollten Sie:

1. wissen, wie man eine neue Komponente in einer Bibliothek anlegt

### Video

TBD

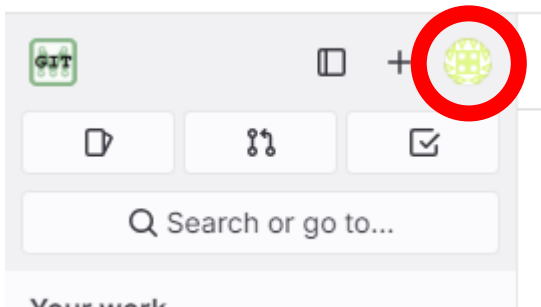
## 0.3 MEXLE GitLab

### Installation der Software auf Ihrem Rechner

1. Laden Sie Git von folgender Seite herunter: <https://git-scm.com/download/win> » "Standalone Installer"  
Git bietet die Möglichkeit mit dem GitLab Server der Hochschule oder mit GitHub in Kontakt zu treten
2. Installieren Sie das Git (alle mit "Ok" bzw "Weiter" bestätigen)
3. Laden Sie TortoiseGit von folgender Seite herunter: <https://tortoisegit.org/download/>  
TortoiseGit bindet den Explorer direkt an die Services des Git an. Damit ist Git direkt in den Explorer eingebunden.
4. Installieren Sie das TortoiseGit (alle Hinweise mit "Ok" bestätigen)

### Anmeldung beim MEXLE GitLab

1. Melden Sie sich bei GitLab mit den Hochschul-Credentials an: <https://git.mexle.org/>
2. Gehen Sie zu User settings » Preferences » Password:  
z.B. über folgenden Link: [https://git.mexle.org/-/user\\_settings/password/edit](https://git.mexle.org/-/user_settings/password/edit)



3. Geben Sie ein Passwort mit mindestens 16 Zeichen ein und merken Sie sich dieses Passwort. Dieses Passwort ermöglicht die Authentifizierung auf GitLab. Bitte nutzen Sie ein neues Passwort!

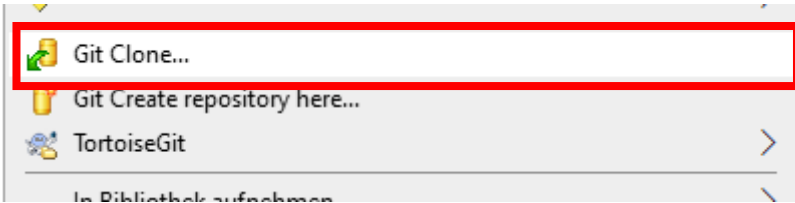
### Installation der Software auf Ihrem Rechner

Die Abfolge ist wie folgt:

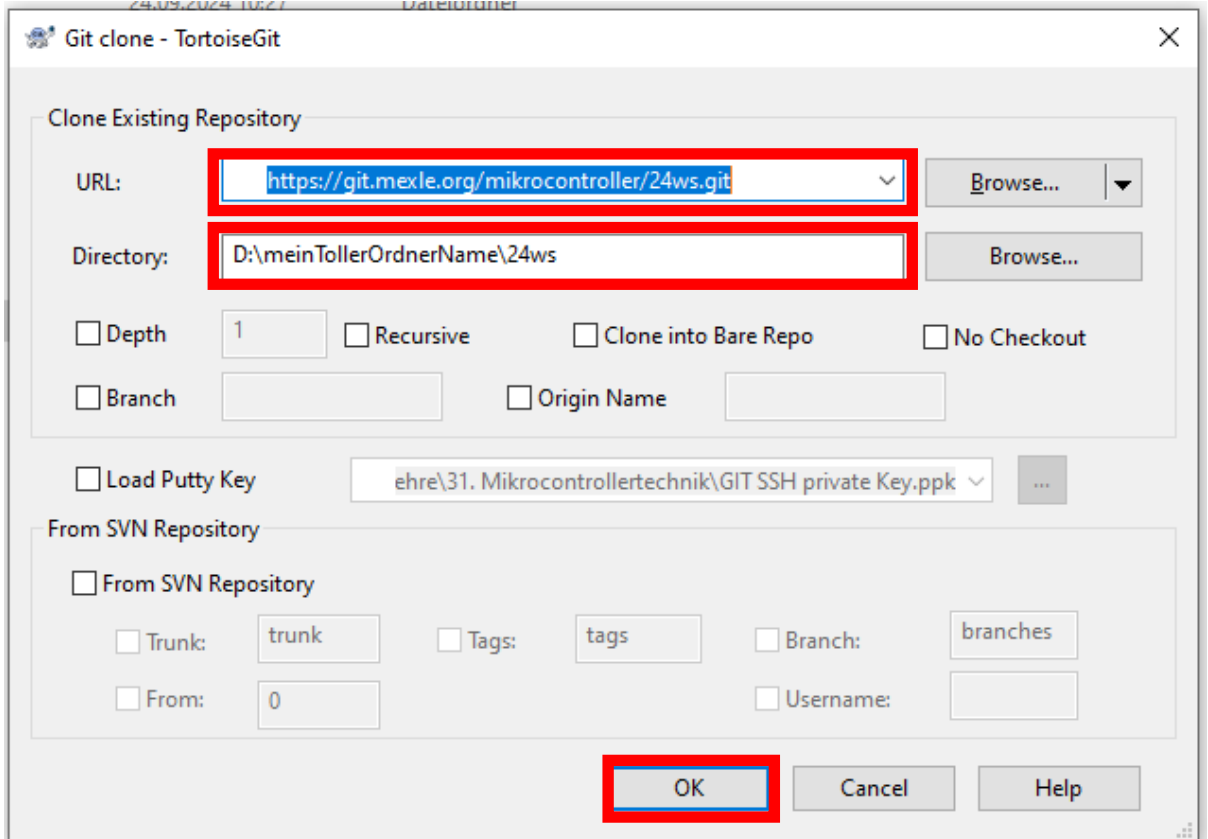


1. Erst müssen Sie sich in **GitLab anmelden**
2. Dann müssen Sie einem **Betreuer per Mail bescheid geben** (Prof. Fischer, Ralf Ziegler) , damit dieser Sie zum Projekt zuordnet
3. **Erst dann sind einem Projekt zugewiesen** und können die folgenden Punkte durchführen.

1. Gehen Sie in den (Windows) Explorer und legen Sie einen neuen Ordner für die Vorlesung an, z.B. Mikrocontroller oder Elektronik Labor
2. Klicken Sie mit der rechten Maustaste auf den Ordner, um in das Kontextmenü zu kommen. Wählen Sie dort Git Clone... aus.



3. Im erscheinenden GitClone Fenster sollten Sie Folgendes eingeben:
  1. als URL: `https://git.mexle.org/[Fach]/[Semester].git`  
also z.B. `https://git.mexle.org/mikrocontroller/26ss.git` oder `https://git.mexle.org/elektronik/26ss.git` für das Sommersemester 2026
  2. als Directory sollte der ausgewählte Ordner eingetragen sein
  3. Klicken Sie nun auf Ok



4. Im Anschluss sollten Sie eine Fehlermeldung erhalten, da noch das Passwort fehlt. Hier ist nun das vorher gewählte Passwort einzugeben. Dies müssen Sie auch nur einmalig machen.
5. Der Download sollte nun klappen und es sollten alle Ordner heruntergeladen werden

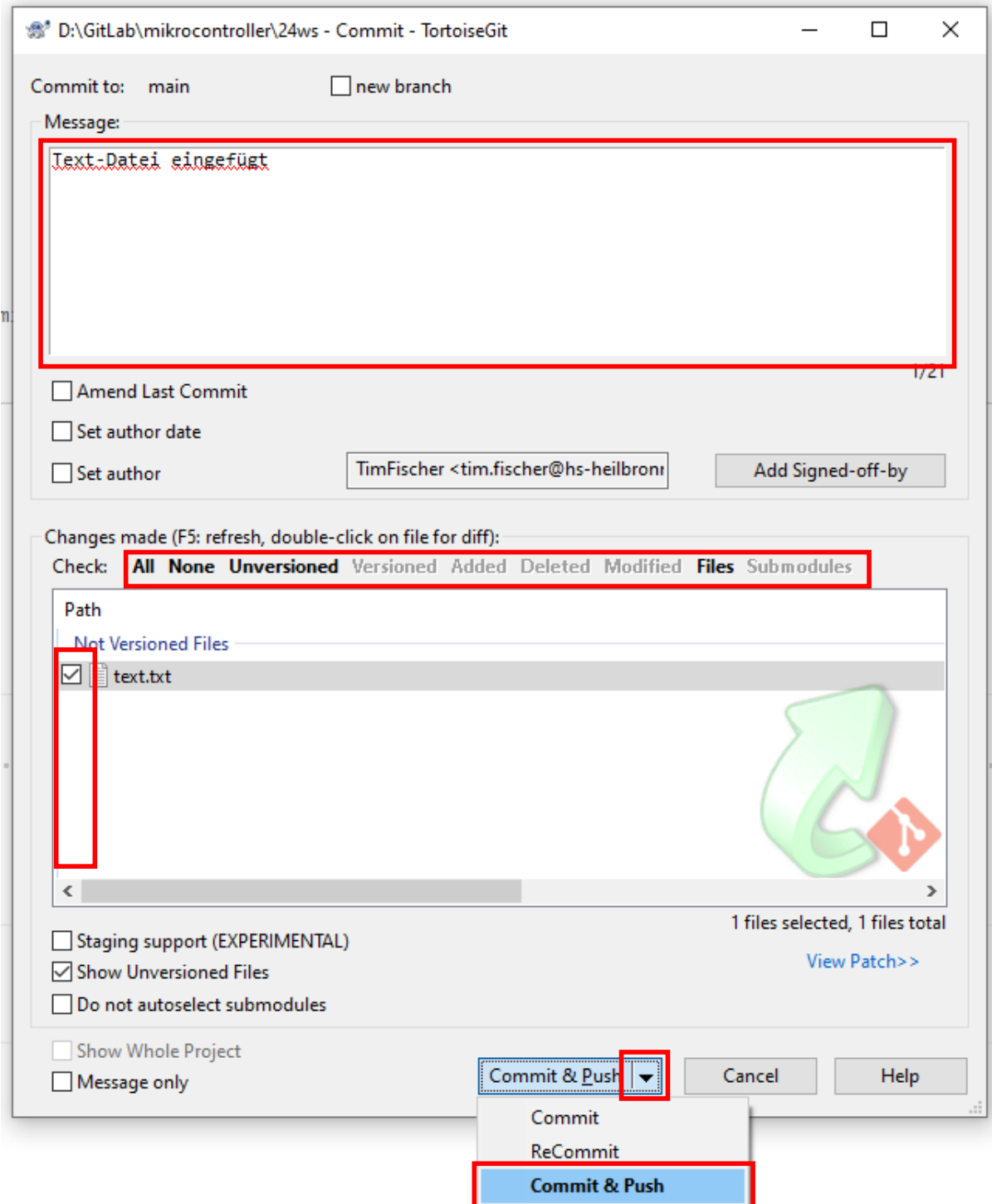
## Upload / Commit von Dateien

Das Hochladen und Ändern von Dateien bei Git wird als "Commit" bezeichnet. Diese Nomenklatur wird auch im Folgenden genutzt. Im Folgenden wird ein Upload beschrieben; über einen Commit können aber auch Dateien gelöscht werden.

1. Rechtsklick auf den übergeordneten Ordner » Git Commit -> "main" ...



2. Falls eine Fehlermeldung erscheint, siehe nächstes Kapitel
3. Im folgenden Fenster ist nun einiges einzutragen:
  1. Tragen Sie unbedingt einen Text unter Message ein, ansonsten ist kein Upload möglich!  
Schreiben sie einen Text, welcher die Änderungen beschreibt.
  2. Im unteren Teil sind die zu ändernden Dateien zu markieren.  
Entweder Sie wählen die Dateien einzeln aus, oder Sie wählen z.B. All .
  3. Zum Commit an den Server wählen Sie am Button unten ▼ aus und dort Commit & Push



4. Bestätigen Sie nun den Commit über Druck auf den Button Commit & Push
4. Überprüfen Sie, ob der Commit erfolgreich war durch einen Blick auf die Homepage des Projekts.

# Hinweise und Mögliche Fehler

1. Für Abschlussarbeiten und studentischen Projekte:
  1. Da diese häufig mehr als 100 MB hochladen, sollten Sie **nicht** [https://git.mexle.org/...](https://git.mexle.org/) nutzen, sondern: **[http://git.mexle.te.hs-heilbronn.de/....](http://git.mexle.te.hs-heilbronn.de/)**
  2. In diesem Fall müssen Sie im Hochschulnetz befinden (z.B. per eduVPN).
  3. Fragen Sie bei mir (Tim Fischer) nach, welches Git Repository für Sie das passende ist.
2. fatal: detected dubious ownership in repository at [...] is on a filesystem that does not record ownership  
Das Problem ist, dass der Ordner auf einem Laufwerk liegt, welches keine Benutzerzuordnung erlaubt (z.B. ein USB-Stick).  
Die Lösung wird gleich mitgeliefert:
  1. Rechtsklick im Explorer auf den entsprechenden Ordner (z.B. 24WS) » Open Git Bash here » Es öffnet sich eine Text-Konsole
  2. Fügen Sie Folgendes ein `git config --global --add safe.directory D:/GitLab/elektronik/25WS` (ändern Sie ggf. elektronik in mikrocontroller und das Semester) und bestätigen Sie mit Return
3. git did not exit cleanly (exit code 1)  
Das Problem ist, dass ihr lokale Datenbank nicht mehr aktuell ist und sie zunächst die Datenbank vom Server herunterladen müssen ("Pull").
  1. Generell hilft hier erst zu Pull'en dann zu Commit&Push'en

From:

<https://first.mexle.te.hs-heilbronn.de/> - **MEXLE Wiki**

Permanent link:

[https://first.mexle.te.hs-heilbronn.de/elektronik\\_labor/0\\_hilfsmittel](https://first.mexle.te.hs-heilbronn.de/elektronik_labor/0_hilfsmittel)

Last update: **2025/02/20 21:04**

