



---

# Handbuch

---

Software Engineering Praktikum  
Wintersemester **2011/2012**

**Auftraggeber Universität Passau**

Dr. Christian Bachmaier, Innstraße 33, 94032 Passau  
Fakultät für Informatik und Mathematik  
Betreuer: Andreas Gleißner

**Team 3** (*team3@sep2011.de*)  
Peter Dahlberg (*dahlberg@fim.uni-passau.de*)  
Lukas Elsner (*friends@ccube.de*)  
Thomas Poxrucker (*poxrucker\_t@web.de*)  
Julian Stier (*mail@julian-stier.de*)  
Alexander Tremel (*alex.tremel@directbox.com*)  
Sebastian Vetter (*basti-vetter@gmx.de*)

# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>4</b>
<b>2. Technische Informationen</b>	<b>5</b>
2.1. Systemvoraussetzungen . . . . .	5
2.1.1. Software-Anforderungen . . . . .	5
2.1.2. Hardware-Anforderungen . . . . .	5
2.2. Installation . . . . .	5
2.2.1. Linux . . . . .	5
2.2.2. Mac OS . . . . .	5
2.2.3. Windows . . . . .	6
<b>3. Bedienungsanleitung</b>	<b>6</b>
3.1. Benutzeroberfläche - eine Übersicht . . . . .	7
3.2. Die Menübar . . . . .	9
3.3. Editier-Werkzeuge . . . . .	12
3.4. Neue-Baustein-Liste . . . . .	13
3.5. Simulationsbedienung . . . . .	13
3.6. Editieren . . . . .	14
3.6.1. Neues Projekt . . . . .	14
3.6.2. Bausteine erstellen . . . . .	15
3.6.3. Bausteine verbinden . . . . .	15
3.6.4. Bearbeiten der Schaltung . . . . .	16
3.7. Komponenten . . . . .	17
3.7.1. Systemkomponenten . . . . .	17
3.7.2. Komponenten erstellen . . . . .	17
3.7.3. Komponenten bearbeiten . . . . .	17
3.7.4. in Komponenten hineinschauen . . . . .	17
3.7.5. Komponenten als neues Projekt öffnen . . . . .	17
3.8. Simulieren . . . . .	18
3.9. Speichern und Öffnen . . . . .	19
3.9.1. Speichern . . . . .	19
3.9.2. Öffnen . . . . .	19
<b>4. Features im Detail</b>	<b>20</b>
4.1. Tools . . . . .	20
4.2. Der Projektbaum . . . . .	20
4.2.1. Verwaltung von Projekten . . . . .	21
4.3. Navigation innerhalb von Projekten . . . . .	21
4.4. Einstellungen . . . . .	22
4.5. Shortcuts . . . . .	22
<b>A. Dateiformat</b>	<b>24</b>
A.1. Anforderungen . . . . .	24
A.2. Beschreibung . . . . .	25
A.3. Unterstützte Bausteine . . . . .	25

<b>B. Haftungsausschluss</b>	<b>26</b>
<b>C. Externe Bibliotheken</b>	<b>26</b>
<b>Glossar</b>	<b>28</b>

# 1. Einleitung

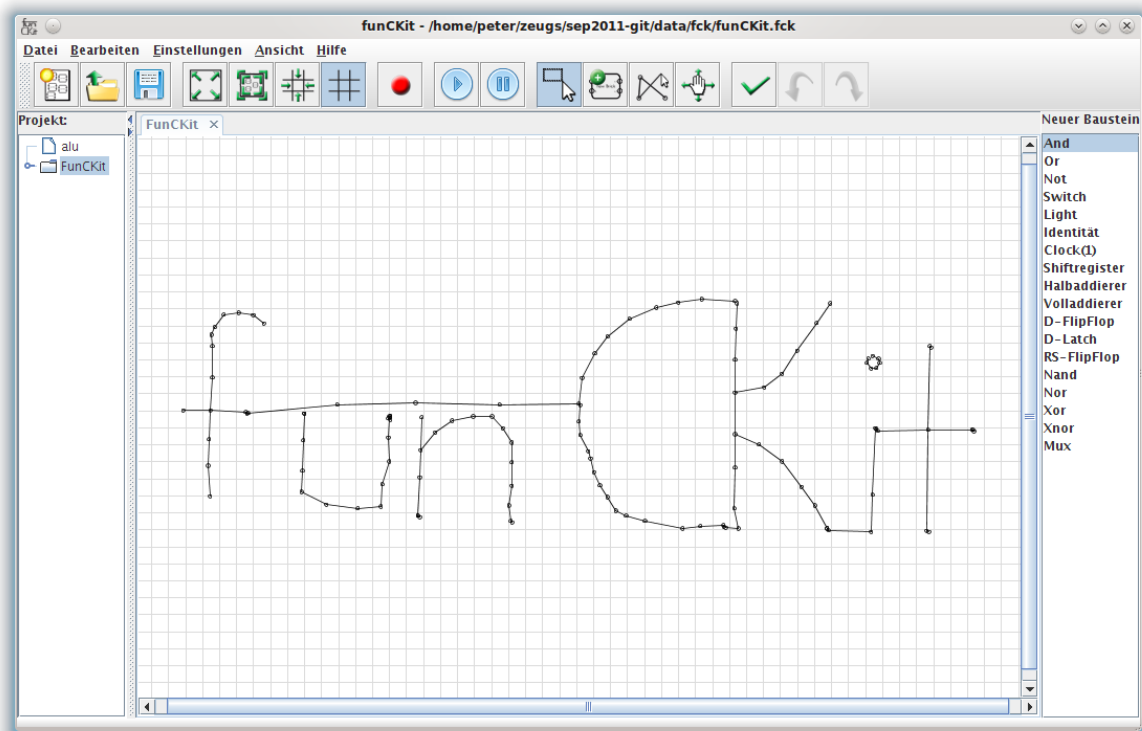


Abbildung 1: funCKit

funCKit (*Functional Circuit Kit*; englisch für *Funktionales / fachliches Schaltungswerkzeug*) ist eine Modellierungs- und Simulationssoftware für logische Schaltungen. Sie ermöglicht - aufbauend auf grundlegenden Logikgattern - das Erstellen, Editieren, Simulieren, Austauschen und Untersuchen von Logikschaltungen, die auf der Booleschen Algebra basieren.

## 2. Technische Informationen

### 2.1. Systemvoraussetzungen

Dieses Kapitel beschreibt die Hard- und Softwarevoraussetzungen für die Installation von funCKit. Bitte vergewissern Sie sich vor Durchführung der Installation, dass die genannten Vorbedingungen in Ihrer Arbeitsumgebung gegeben sind.

#### 2.1.1. Software-Anforderungen

Zur Nutzung von funCKit benötigen Sie ein Java Runtime Environment der folgenden Version: Java 2 Runtime Environment, Standard Edition (J2SE JRE), mindestens Version 6.0.

#### 2.1.2. Hardware-Anforderungen

Prinzipiell stellt funCKit keine besonderen Anforderungen an die Hardware. Empfohlen wird eine Umgebung, die mindestens folgende Kriterien erfüllt:

- Intel Pentium 4 oder vergleichbarer Prozessor
- Arbeitsspeicher (RAM) mind. 512 MB
- Monitor wenigstens 1024x768 Pixel
- Maus und Tastatur

### 2.2. Installation

Öffnen Sie die Software (\*.jar-Datei) mittels Doppelklick. Sollte das Programm nicht automatisch ausgeführt werden, benötigen Sie die aktuelle Version von Java auf Ihrem Computer. Nähere betriebssystemspezifische Informationen dazu finden Sie im Folgenden:

#### 2.2.1. Linux

In den neuesten Versionen ist die JDK6 schon eingebunden. Linux bietet viele verschiedene Distributionen - eine eindeutige Lösung um die neueste Version zu erhalten, gibt es demnach nicht. Am einfachsten wird es sein, den unter "Windows" eingetragenen Link zu verwenden und auf der Homepage von Sun die Version für die Linux Distribution zu installieren.

#### 2.2.2. Mac OS

Die neuesten Versionen vom Mac OS unterstützen Java 6. Sollte ihre Mac Version trotzdem Probleme haben, führen Sie im Menü "Apple" eine Softwareaktualisierung aus. Näheres dazu unter [http://support.apple.com/kb/HT1338?viewlocale=de\\_DE&locale=de\\_DE](http://support.apple.com/kb/HT1338?viewlocale=de_DE&locale=de_DE).

### 2.2.3. Windows

Unter <http://java.sun.com/javase/downloads/index.jsp> können Sie die virtuelle Umgebung für Java herunterladen. Diese wird benötigt, um das Programm ausführen zu können. Wir empfehlen JRE (Java runtime environment).

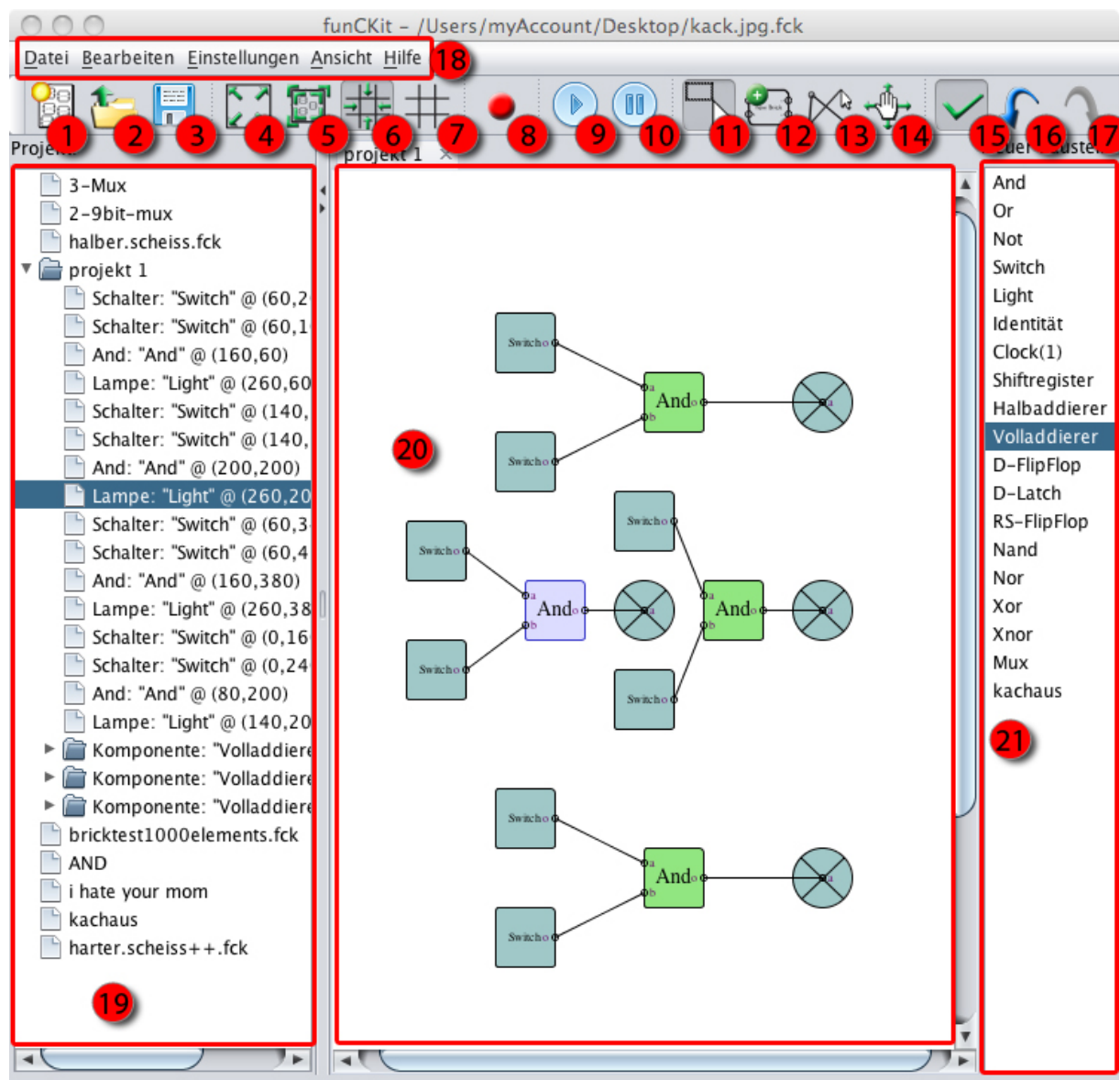
## 3. Bedienungsanleitung

Beim Erstellen von Schaltungen hängt vieles von der richtigen Verwendung von Verbindungen, Gattern und Komponenten ab. Die Voraussetzung dafür ist die genaue Kenntnis der geltenden Vorschriften und Bestimmungen. Diese Anleitung soll als Stütze zur korrekten und fehlerfreien Entwicklung von Schaltnetzen und Schaltwerken dienen.

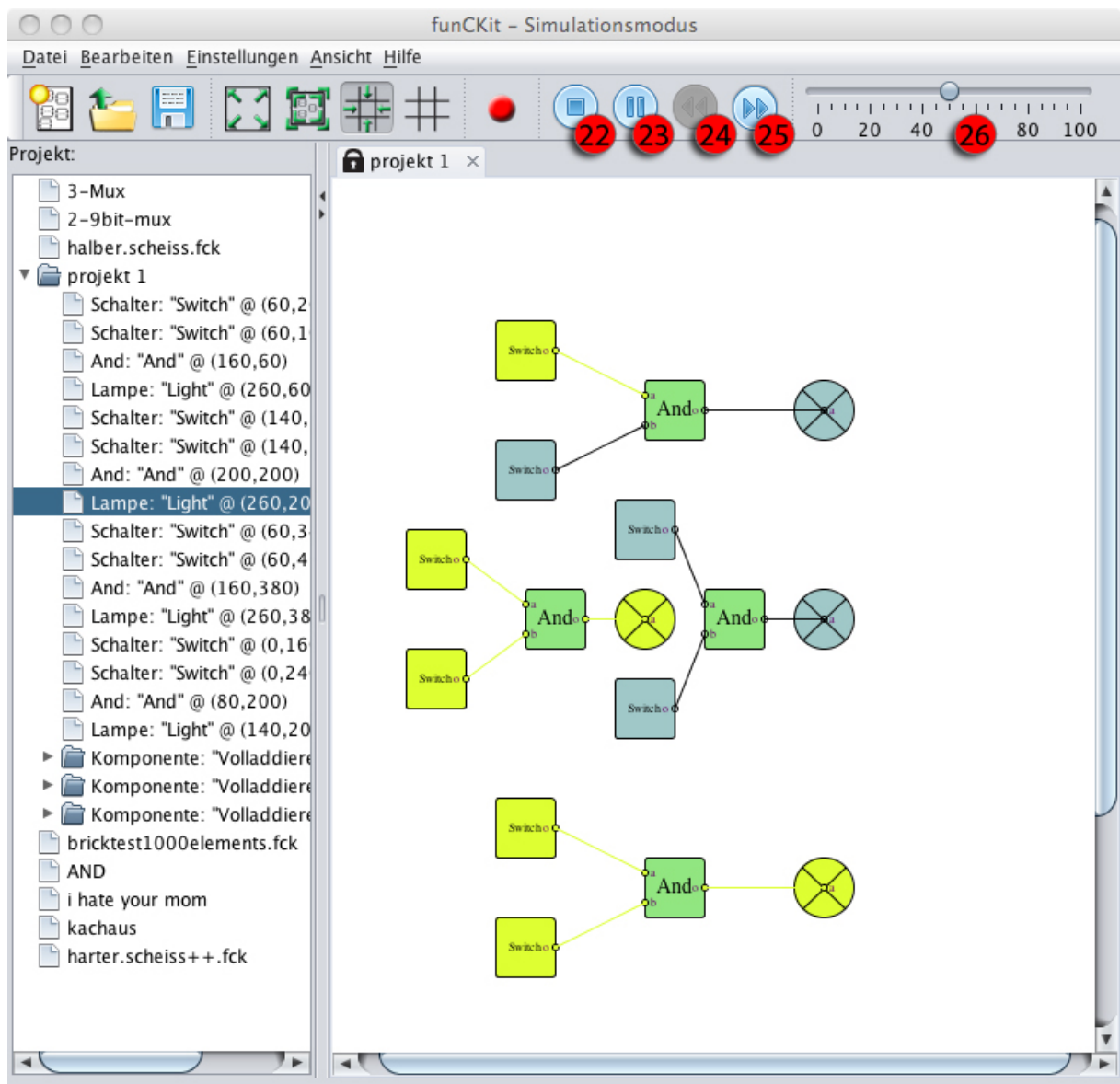
Die Tastenkürzel funktionieren funktionieren unter Windows und Linux gleichermaßen. Bei Mac OS X wird die *Cmd* Taste anstelle der *Strg* Taste verwendet. Einzige Ausnahme: um mehr Elemente zu selektieren, verwendet man auch bei Mac OS X die *Strg* benutzt (siehe dazu das Kapitel über Auswahl - Werkzeug).

Im Nachfolgenden findet sich eine geführte Einleitung in die grundlegende Funktionalität der Anwendung. Für ausführlichere Beschreibungen befindet sich im Anschluss eine detaillierte Einführung in einzelne Werkzeuge und ihr Funktionsumfang.

### 3.1. Benutzeroberfläche - eine Übersicht



- |                       |                               |                         |
|-----------------------|-------------------------------|-------------------------|
| 1 Neues Projekt       | 8 Simulationshistorie anlegen | 15 Echtzeitvalidierung  |
| 2 Datei Öffnen        | 9 Simulation starten          | 16 Rückgängig           |
| 3 Projekt Speichern   | 10 Simulation pausieren       | 17 Wiederherstellen     |
| 4 Vollbild            | 11 Auswählen/Bewegen          | 18 Menüleiste           |
| 5 Ansicht einpassen   | 12 Neues Bauteil              | 19 Projektbaum          |
| 6 Gitter einrasten    | 13 Kabelwerkzeug              | 20 Editierbereich       |
| 7 Gitternetz anzeigen | 14 Ansicht verschieben        | 21 Neuer Baustein Liste |



22 Simulation beenden

23 Simulation Pausieren

34 Einen Simulationsschritt zurück

25 Einen Simulationsschritt vorwärts

26 Simulationsgeschwindigkeit anpassen



## 3.2. Die Menübar

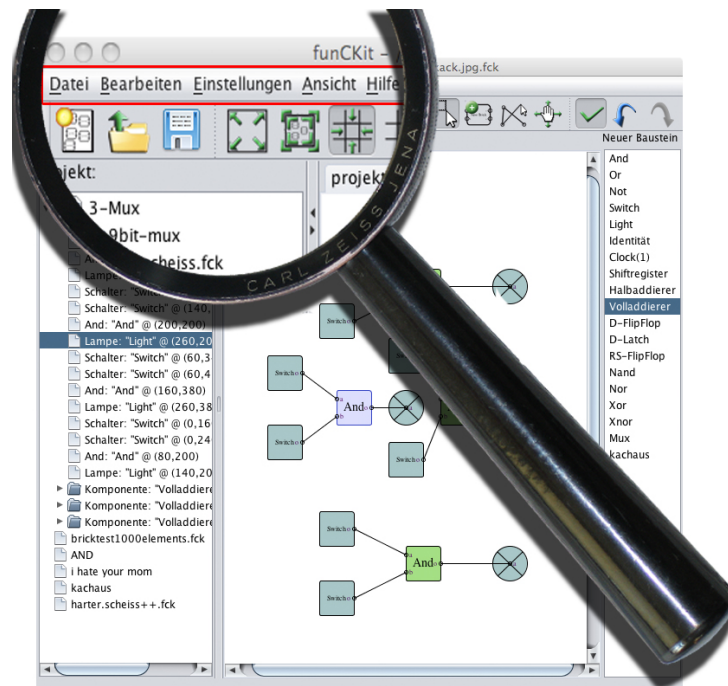


Abbildung 2: Übersicht Menüleiste

Grundsätzlich teilt sich die Benutzeroberfläche in eine Menüleiste, ein Editierfenster, einen Projektbaum und verschiedene so genannte Werkzeugsets- Sammlungen zusammengehörender Werkzeuge.

Über die [Menüleiste](#) erreicht man alle wesentlichen Funktionen der Anwendung. Sie teilt sich auf in folgende Punkte:

### 1. **Datei**

Menüs mit Verbindung zum Dateisystem oder anwendungsnahe Funktionen. Beispielsweise Anlegen eines neuen Projekts, Speichern und Öffnen von existierenden Projekten oder Import und Export von Komponenten.

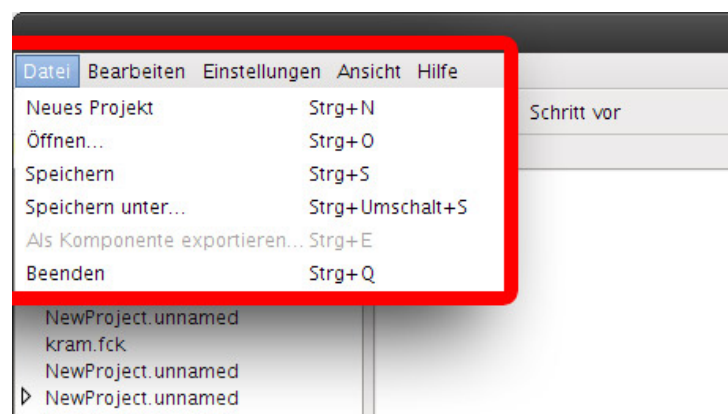


Abbildung 3: Datei Menu

## 2. Bearbeiten

Auswahl von Funktionen, die in Verbindung mit dem Bearbeiten von Schaltungen stehen. Beispielsweise Löschen, Kopieren, Ausschneiden von Bausteinen und Rückgängigmachen von Änderungen.

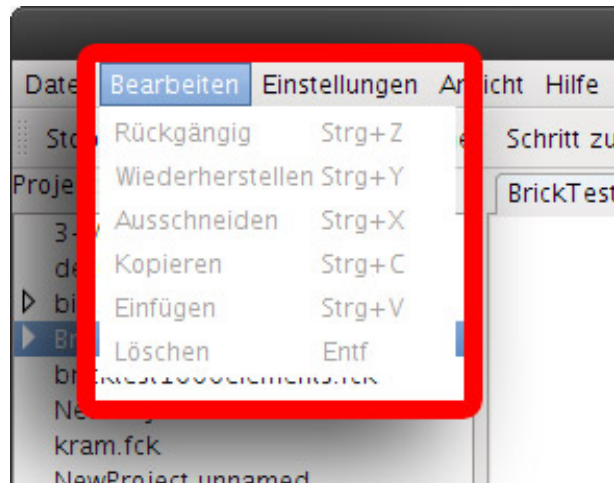


Abbildung 4: Bearbeiten Menu

## 3. Einstellungen

Zugriff auf verschiedene Einstellungen wie Echtzeitvalidierung, Sprache oder Anzeige des Gitternetzes.

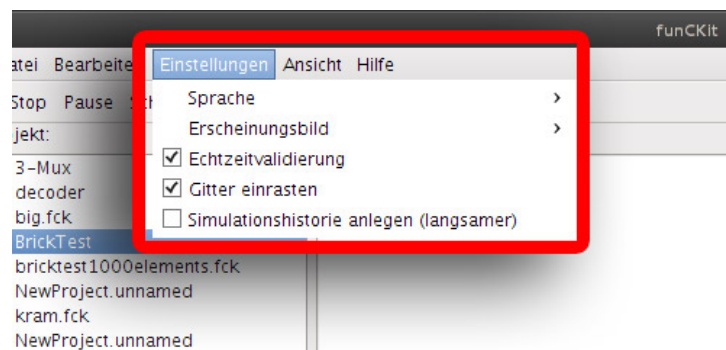


Abbildung 5: Einstellungen Menu

## 4. Ansicht

Menüpunkt zum Verändern der Ansicht; beispielsweise durch Zoom oder Vollbild.

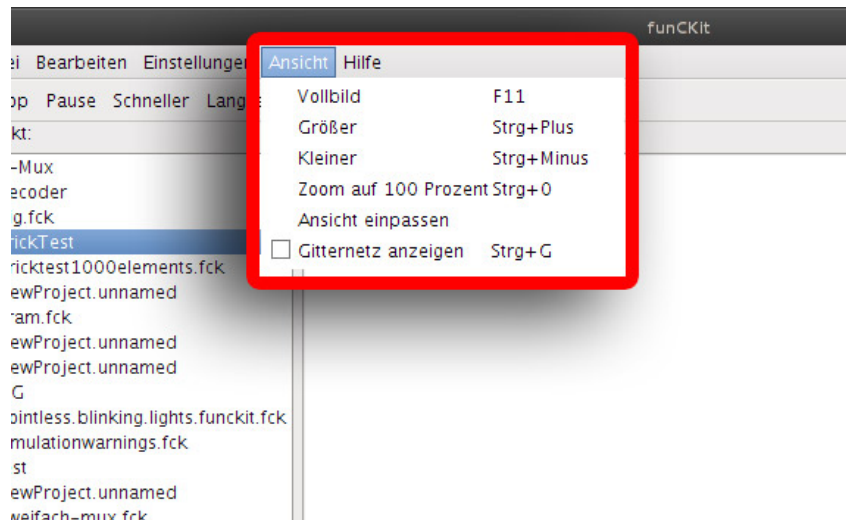


Abbildung 6: Ansicht Menu

## 5. Hilfe

Informationen über und zur Verwendung der Anwendung.

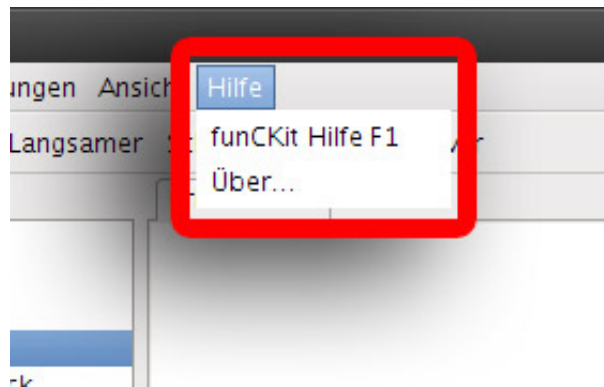


Abbildung 7: Hilfe Menu

Das [Editierfenster](#) ist der zentrale Platz zur Modellierung und Simulation. Es ist je nach Projekt in mehrere Reiter (verschiedene Ansichten) angeordnet. Wählt man aus dem [Projektbaum](#) ein anderes Projekt aus, wechselt man in das Editierfenster des entsprechenden Projekts.

Über *Strg+T* lässt sich ein neuer Reiter der aktuellen Schaltung öffnen. Damit lässt sich die Schaltung aus mehreren Perspektiven (Zoomstufen, Teilansichten) betrachten. Mit *Strg+W* oder mit der mittleren Maustaste auf den Tabnamen lässt er sich wieder schließen.

Je nach momentan ausgewähltem [Tool](#) kann man verschiedenste Änderungen auf dem Modell vornehmen. Beispielsweise lässt sich über das [Neues Bauteil](#) - Tool ein neuer Baustein (Gatter, System- oder eigens definierte Komponente) platzieren oder mit dem [Auswählen / Bewegen](#) - Tool bestimmte Bausteine auswählen und verschieben.

Um Übersicht über die verschiedenen Projekte zu behalten, gibt es links neben dem Editier-

fenster angeordnet einen so genannten [Projektbaum](#). Wird die Anwendung neu gestartet, sind noch keine Projekte vorhanden. Mit zunehmender Verwendung neuer Projekte wächst der Projektbaum und man hat schnellen Zugriff auf die jeweiligen Schaltungen. Neben dem Öffnen und der Übersicht vorhandener Projekte, bietet der Projektbaum außerdem Einsicht in die verwendeten Bauteile des Projekts.

Neben der Menüleiste, dem Editierfenster und dem Projektbaum existieren außerdem zwei bzw. drei [Werkzeugsets](#):

### 3.3. Editier-Werkzeuge

Unter der Menüleiste existiert eine Leiste mit verschiedenen Editier-Werkzeugen. Häufig kann man mit einfachen Tastenkürzeln auf andere Editier-Werkzeuge umstellen oder ähnliche Funktionalitäten in verschiedenen Werkzeugen verwenden, so dass eine intuitivere Bedienung möglich ist. Die 4 Editier-Werkzeuge:

- **Auswahl - Werkzeug**



- Strg+1 • Shift (gedrückt)

Dieses Werkzeug dient dazu, um bereits vorhandene Bausteine und Kabel zu selektieren um diese dann zu verschieben oder zu löschen. Falls zu einer bereits vorhanden Selektion neue hinzugefügt (oder sich bereits in einer Selektion Befindenden entfernt) werden sollen muss die *Strg* Taste gedrückt sein. Natürlich gibt es auch das übliche Selektionsviereck, das Sie aus den verschiedensten Programmen kennen. Gelöscht wird mit *Entf*, im Menü oder im Kontextmenu. (Rechtsklick)

- **Neuer-Baustein - Werkzeug**



- Strg+2 • Neue-Baustein-Liste

Dieses Werkzeug dient dazu, neue Bausteine zu erstellen und schnell unkomplizierte Kabel zu verlegen. Sobald sich in diesem Modus befinden erscheint ein sogenannter "Ghostbrick" im Editpanel, falls der Baustein gezeichnet werden kann. Befindet sich die Maus über einem Accesspoint, so kann mit gedrückter Maustaste die Kabelzeichnung eingeleitet werden. Lässt man über einem anderen Accesspoint die Maustaste wieder aus, so werden diese beiden Punkte mit einem Kabel verbunden. Macht man das Selbe jedoch an "freier Stelle", so springt das Programm automatisch ins Wiretool.

- **Kabel - Werkzeug**



- Strg+3

(Erreichbarkeit fehlt: button, alle kürzel) Mit diesem Werkzeug können ganz frei Kabel gezeichnet werden. Befindet man sich in diesem Modus, so kann an beliebiger

Stelle mit dem Zeichnen eines Kabels begonnen werden, indem an beliebiger Stelle geklickt wird. Man kann auch an bestehende Kabel und Accesspoints anknüpfen. Es muss beachtet werden, dass sich anfangs erst um sogenannte "GhostWires" handelt. Mit *ESC* oder Rechtsklick kann das "GhostWires" wieder entfernt werden. Mit einem einfachen Klick auf einen Accesspoint oder Doppelklick auf eine freie Fläche wird das Kabel befestigt und funktionsfähig.

- **Ansicht-verschieben - Werkzeug**



- **Strg+4** • **Leertaste (gedrückt)**

Mit diesem Werkzeug können sie sich leicht sehr schnell über die gesamte Schaltung (und darüber hinaus) bewegen.

Mit *Strg+1*, *Strg+2*, *Strg+3* und *Strg+4* lässt sich ganz einfach zwischen den wichtigsten Tools (*Auswahl - Werkzeug*, *Neuer-Baustein - Werkzeug*, *Kabel - Werkzeug* und *Ansicht-verschieben - Werkzeug* ) hin- und herwechseln.

### 3.4. Neue-Baustein-Liste

Die Neue-Baustein-Liste dient zur Übersicht von existierenden Bausteinen, die in der aktuellen Schaltung hinzugefügt werden können. Sie beinhaltet sowohl grundlegende Gatter wie *And*, *Or* und *Not*, vorgegebene Systemkomponenten (*XOR* oder *NOR*), als auch eigens über den Export definierte Komponenten - worüber man eine ganze Reihe komplexerer Bausteine leicht hinzufügen kann (beispielsweise selbst gestaltete arithmetisch-logische Einheiten). Durch Klicken auf eines dieser Bricks wechseln sie in das "New-Brick-Tool".



### 3.5. Simulationsbedienung



Startet man aus dem Editieren heraus die Simulation, verschwinden die Editier-Werkzeuge und die Steuerungsleiste zur Simulation tritt an ihre Stelle. Über diese Werkzeugset lässt sich die Simulation pausieren, beschleunigen, verlangsamen oder schrittweise voranschreiten. Wenn die Simulationshistorie auch aktiviert ist, lässt sich die Simulation sogar schrittweise zurücksetzen.

## 3.6. Editieren

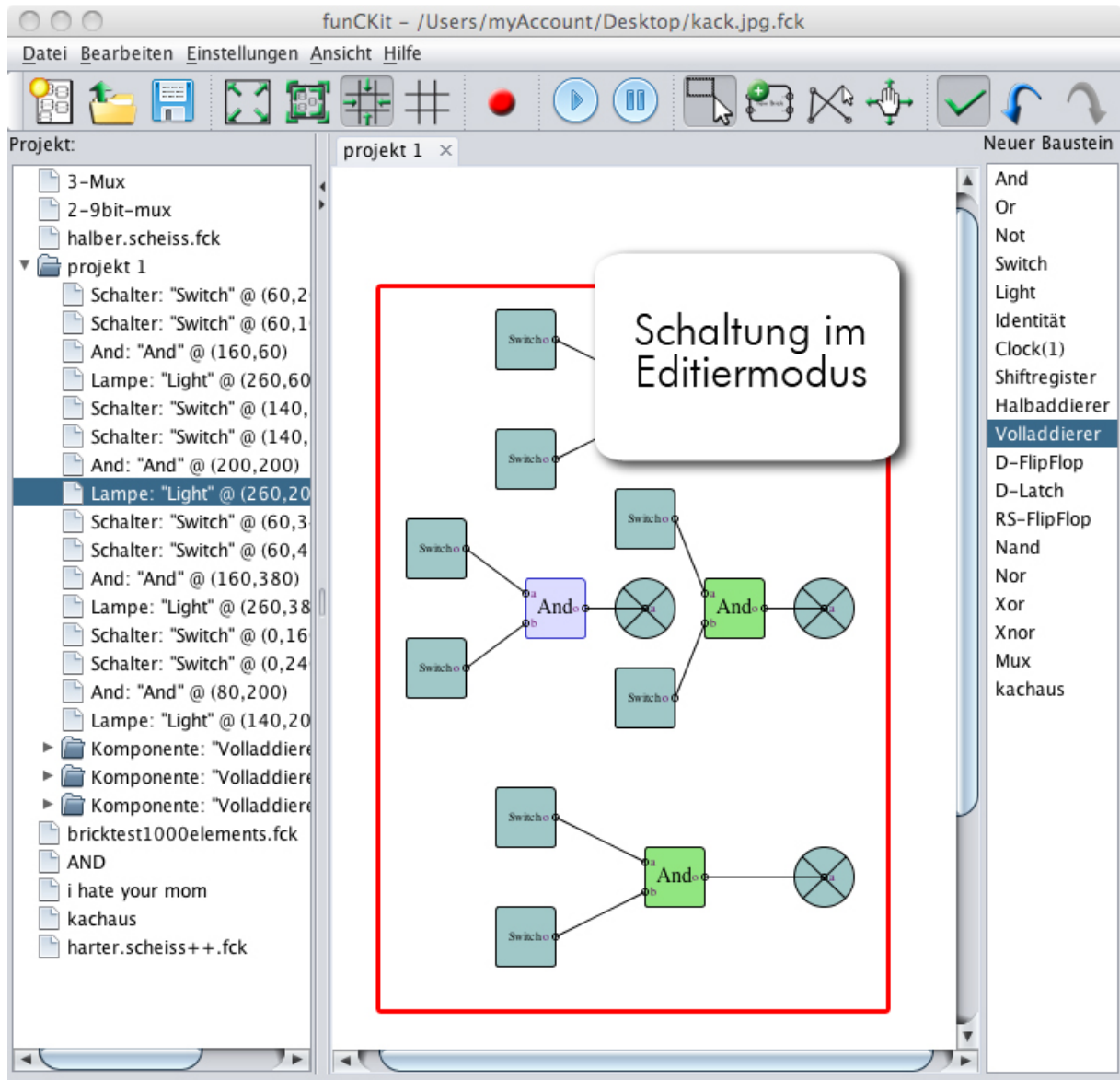


Abbildung 8: Editiermodus

In diesem Abschnitt wird das wesentliche Bearbeiten einer Schaltung erläutert - beginnend beim Anlegen eines Projekts über Hinzufügen und Verknüpfen und bis hin zum Entfernen und Bearbeiten von Bausteinen. Daneben werden einige praktische Funktionalitäten beim Editieren erwähnt, auf die im Abschnitt »*Features im Detail*« näher eingegangen wird.

### 3.6.1. Neues Projekt



- Strg+N

Nach dem ersten Start der Anwendung existieren noch keinerlei Projekte im Projektbaum. Man kann also entweder ein neues (leeres) Projekt anlegen oder eine bestehende Schaltung öffnen bzw. importieren. Auf Import und Export wird später detaillierter eingegangen. Für ein neues Projekt reicht zu Beginn ein Projektname aus. So lange nicht gespeichert wurde,

existiert das Projekt nur temporär.

Mit den Tastenkombinationen *Strg+S* bzw. *Strg+Umschalt+S* lässt sich ein Projekt schnell speichern bzw. unter einem neuen Ort speichern.

Nach dem Erstellen des Projekts, erscheinen nun die Editier-Werkzeuge und eine Neue-Baustein-Liste. Außerdem hält der Projektbaum nun das neue Projekt vor und lässt sich nach dem Speichern darüber immer wieder neu laden. Über »Ansicht →« Gitter anzeigen (oder *Strg+G*) lässt sich übrigens ein Hilfsgitter im Hintergrund anzeigen.

### 3.6.2. Bausteine erstellen

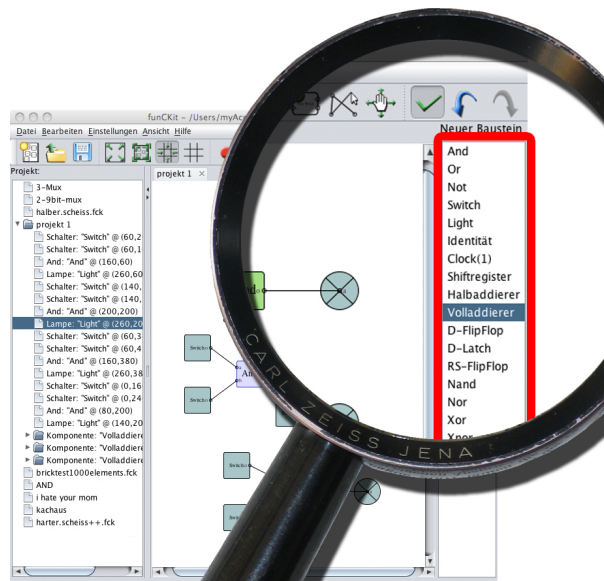


Abbildung 9: Neuer Baustein Liste

Über die [Neue Baustein - Liste](#) lässt sich nun ein Bausteintyp auswählen und auf dem Editierfenster platzieren. Überschneidungen werden grundsätzlich vermieden, auch wenn die Anwendung grundsätzlich damit arbeiten kann (einige Exportformate unterstützen keine Überschneidung von Bausteinen). Über die Einstellungen kann man zusätzlich "Gitter einrasten" aktivieren und damit sauberere Entwürfe anlegen.

### 3.6.3. Bausteine verbinden

Kabelverbindungen lassen sich entweder aus dem *Neuer-Baustein - Werkzeug* heraus direkt oder ausführlicher mit dem *Kabel - Werkzeug* erstellen. Eine Verbindung kann von einem beliebigen Verbindungspunkt an einem Baustein zu einem anderen Verbindungspunkt verlaufen. Dabei ist es zunächst nicht relevant ob es sich dabei um Ein- oder Ausgänge handelt.

Unter »Einstellungen → Echtzeitvalidierung« lässt sich eine automatische Validierung der aktuellen Schaltung aktivieren. Damit werden sofort Bereiche der Schaltungen markiert, die für einen fehlerhaften Graphen verantwortlich sind und damit nicht simuliert werden können.

Um eine korrekte (= simulierbare) Schaltung zu modellieren, müssen Verbindungen von



Ausgängen zu Eingängen führen. Bei welchem Verbindungspunkt man startet, ist dabei aber egal. Aus dem *Neuer-Baustein - Werkzeug* kann man mit Hilfe eines haltenden Klicks auf einen Verbindungspunkt und Ziehen des angezeigten Kabels eine Verbindung zu einem anderen Baustein herstellen. Lässt man das Kabel an einer freien Stelle im Modell los, wechselt man automatisch in das *Kabel - Werkzeug* und kann mit einer Verzweigung die Verbindung an dieser Stelle fortsetzen.

Im *Kabel - Werkzeug* reicht ein einfacher Klick auf den Verbindungspunkt, um das Anlegen einer Verbindung zu starten. Klicks auf freie Flächen erzeugen Verbindungspunkte, an denen man die Verbindung verzweigen oder umknicken lassen kann. Doppelklicks ermöglichen es, die Verbindung an dieser Stelle zu beenden und das Kabel an einem Verbindungspunkt enden zu lassen. Mit einem Klick auf einen anderen Verbindungspunkt wird das Erstellen der Verbindung abgeschlossen.

Hat man aus dem *Neuer-Baustein - Werkzeug* heraus begonnen eine Verbindung zu erstellen, wechselt man nach Abschließen der Verbindung zurück in diesen Modus. Andernfalls kann man direkt im *Kabel - Werkzeug* weitere Verbindungen zwischen Bausteinen erstellen.

Hat man einmal einen falschen Verbindungspunkt ausgewählt, kann man die aktuelle Verbindung mit *Esc* abbrechen. Außerdem lässt sich über *Strg+Z* eine Änderung rückgängig machen und mit Hilfe von *Strg+Y* wiederherstellen.

#### 3.6.4. Bearbeiten der Schaltung

Mit dem Erstellen und Verbinden von Bausteinen kann man bereits eine größere Schaltung entwerfen. Oft fällt während dem Editieren aber auf, dass sich Fehler eingeschlichen haben, oder Umordnungen sinnvoll wären. Dazu kann man die erstellen Elemente mit Hilfe verschiedener Möglichkeiten wieder bearbeiten.

Über den **Selektionsmodus** (*Strg+1*) lassen sich ein oder mehrere Elemente auf der Schaltung auswählen. Ausgewählte Elemente können verschoben, gelöscht, kopiert und sogar zwischen verschiedenen Projekten ausgetauscht werden. Dazu dienen verschiedene Funktionen, die sich über die Menüleiste oder durch Tastenkombinationen erreichen lassen.

Am einfachsten lassen sich bereits erstellte Bausteine über die *Entf*-Taste wieder löschen. Der entsprechende Menüpunkt dazu befindet sich unter »Bearbeiten → Löschen«.

Ausgewählte Elemente können durch Ziehen und wieder Loslassen der Maus verschoben werden. So genannte *Geist-Elemente* helfen bei der exakten Positionierung der zu verschiebenden Elemente.

**Komfort vs. Einbußen:** Verbindungspunkte haben beim Erstellen von Verbindungen einen so genannten *Toleranzbereich*. Dieser Toleranzbereich dient dazu, dass man Verbindungspunkte nicht exakt treffen muss, um eine neue Verbindung von einem Baustein zu einem anderen zu erstellen (oder gar eine längere Verbindungsreihe mit vielen weiteren Verbindungspunkten). Der Komfort hat aber einige Einbußen, so dass man keine Verbindungen ganz nah an einen Verbindungspunkt setzen kann, ohne die beiden zu verknüpfen.

**Die Lösung:** Man setzt eine Verbindung etwas daneben und kann mit dem Selektionswerkzeug Verbindungspunkte eines Kabels ganz einfach verschieben.



## 3.7. Komponenten

Komponenten sind selbst definierte Bausteine, die wiederum aus Bausteinen bestehen und sich somit auf die grundlegenden Gatter *And*, *Or* und *Not* runterbrechen lassen. Sobald man eigene Komponenten definiert hat, kann man so sehr einfach Übersicht in seiner Schaltung schaffen.

### 3.7.1. Systemkomponenten

Systemkomponenten unterscheiden sich von den restlichen Komponenten insoweit, dass sie bereits mit der Anwendung mitgeliefert werden. Je nach Version können das unterschiedlich viele sein. Beispielhafte Systemkomponenten sind *Nand*-, *Nor*-, *Xor*-Gatter oder *Clocks*.

### 3.7.2. Komponenten erstellen

In funCKit können Schaltungen als Komponenten exportiert werden, indem alle Schalter als Eingänge und alle Lichter als Ausgänge der Komponente verwendet werden. Die Reihenfolge der generierten Ein- und Ausgänge richtet sich hierbei nach der vertikalen Position der Schalter bzw. Lichter. Diesem Schema entsprechend werden die Namen der Schalter und Lichter als Namen der Ein- bzw. Ausgänge übernommen.

Zum exportieren dieser so erstellten Schaltung, wählt man im Menü *Datei* den Punkt *Als Komponente Exportieren*. Wird die Komponente im vorgegebenen Pfad gespeichert taucht sie in der Neue-Baustein-Liste auf und kann verwendet werden.

### 3.7.3. Komponenten bearbeiten

Komponenten können über *Datei* → *Öffnen...* importiert und dann auf gewöhnliche Art und Weise bearbeitet werden.

### 3.7.4. in Komponenten hineinschauen

Um besser nachvollziehen zu können, was im Inneren einer Komponente passiert öffnet sich bei Doppelklick darauf einer neuer Tab mit dem Aufbau der Komponente. Hier kann nichts editiert werden, deshalb befindet sich am Register auch das Schloss.

### 3.7.5. Komponenten als neues Projekt öffnen

„Damit schnell und einfach bestehende Komponenten als Vorlage für eine neue Schaltung oder für eine neue Komponente zur Verfügung stehen, kann man diese durch Rechtsklick auf der Komponente oder im Komponentenbaum als neues Projekt öffnen.“

### 3.8. Simulieren

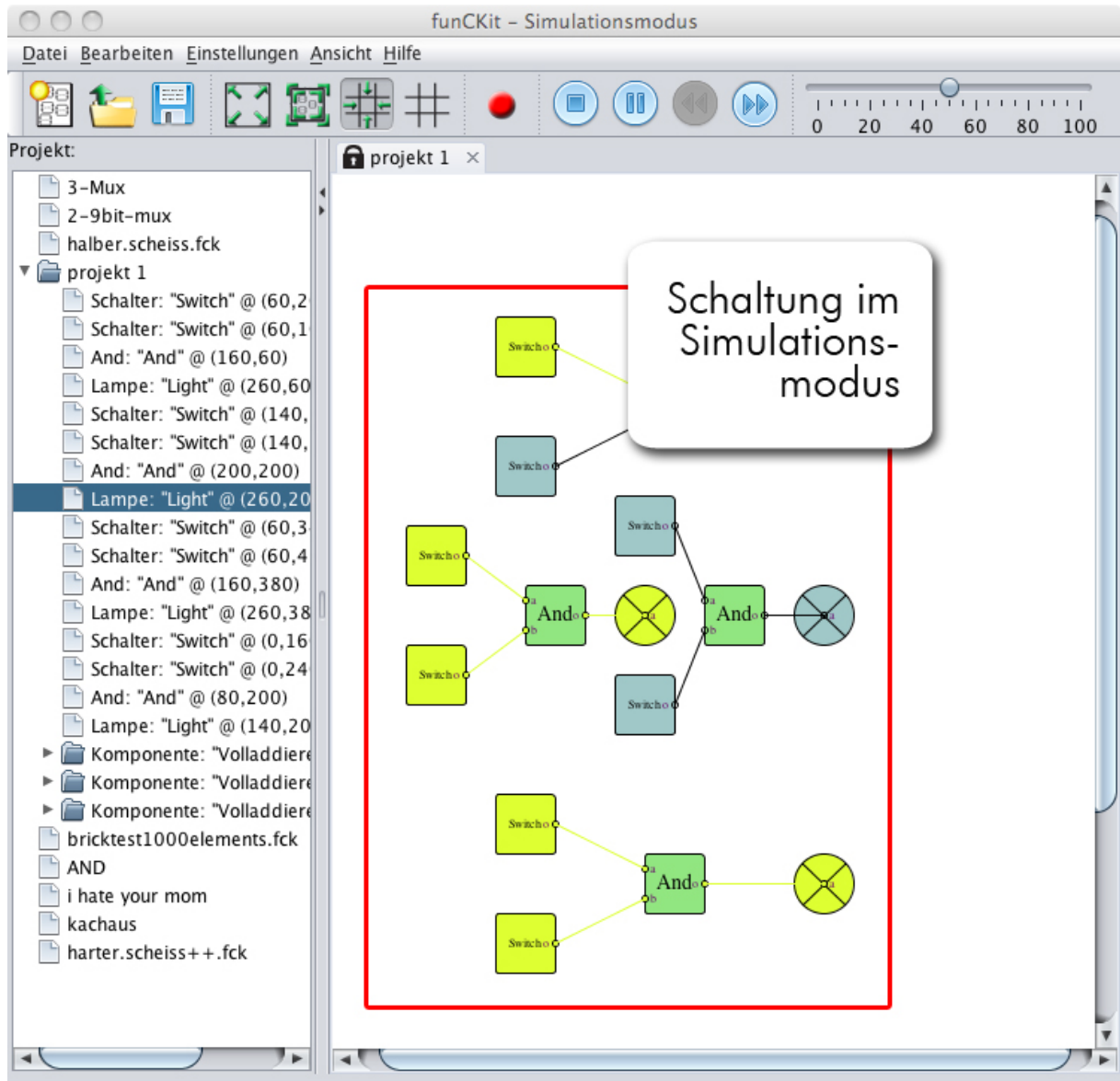


Abbildung 10: Simulation

Aus dem Editiermodus heraus lässt sich mit dem Simulations-Startknopf eine Simulation der aktuell geöffneten Schaltung starten. Während der Simulation kann man auf ein anderes Projekt (und damit eine andere Schaltung) wechseln, wodurch die aktuelle Simulation pausiert. So können parallel auch mehrere Simulationen gestartet werden.

Die Simulation startet nur, wenn die Validierung zur Simulation erfolgreich war. Diese Validierung prüft auf notwendige Eigenschaften, die die Schaltung erfüllen muss. Beispielsweise dürfen keine Rückkopplungen (Schleifen) ohne Verzögerung stattfinden. Eine entsprechende Fehlermeldung wird angezeigt.

Echtzeitvalidierung kann während der Modellierung bereits dabei helfen, Fehler in der Schaltung zu finden und sie so simulierbar zu halten, um manuell den logischen Ablauf der modellierten Schaltung zu überprüfen.

In der Simulation selbst werden - im näheren Zoom auf die Schaltung - Leitungen und Lampen als aktiv oder inaktiv markiert (*logisch-0* und *logisch-1*). Die drei Grundgatter *And*, *Or* und *Not* können auch je nach Verzögerungszahl ihre anliegende Queue für jeden Ausgang anzeigen (wird bei aktivem Hervorheben mit der Maus über dem Baustein sichtbar).

Über die Werkzeuge in der Simulations-Werkzeugleiste kann man die Geschwindigkeit der Simulation verändern oder schrittweise die Simulation voranschreiten lassen, um einzelne Zustandsveränderungen detaillierter beobachten zu können.

Wenn »Einstellungen → Simulationshistorie anlegen« aktiviert ist, kann die Simulation auch schrittweise rückwärts gegangen werden (sofern bereits entsprechende Simulations-schritte ausgeführt wurden). Die Simulationshistorie wird sowohl in der automatischen (geschwindigkeitsgeregelten), als auch in der schrittweisen Simulation angelegt.

**Komfort vs. Einbußen:** Zwar ist es durchaus vorteilhaft zum debuggen, aber während der Simulation eine Historie mitlaufen zu lassen, kostet deutlich mehr Rechenleistung, als wenn ohne Zustandskopie simuliert wird. Das hat den Grund, dass für jeden Simulations-schritt der Gesamtzustand der Simulation gespeichert werden muss. Somit ist eine flüssige Darstellung der Simulation ggf. nicht mehr möglich.

**Die Lösung:** Sinnvoll ist es bis zu bestimmten Zuständen "hin" zu simulieren, um dann erst schrittweise mit Simulationshistorie fehlerhaftes Verhalten der Schaltung zu untersuchen. Durch regelmäßiges Aktivieren & Deaktivieren verhindert man verzögerte Reaktionen.

## 3.9. Speichern und Öffnen

### 3.9.1. Speichern



● Strg+S

Zum Speichern des aktuell geöffneten Projektes wählen Sie »Datei → Speichern« oder drücken Sie Strg+S. Sollten Sie ein Windows oder Mac Betriebssystem verwenden, so wird der Dateityp über die eingegebene Dateiergung bestimmt. Andernfalls können Sie diesen über den Öffnen Dialog wählen. Zur Speicherung wird das funCKit Dateiformat empfohlen. Zur Auswahl stehen zum einen einige Bildformate (PDF, PNG, GIF, SVG, JPG), zum anderen das SEP-Austauschformat. Die Speicherung im SEP-Austauschformat wird ausdrücklich nicht empfohlen, da aufgrund der Einschränkungen des Formates nicht der volle Funktionsumfang gewährleistet werden kann. So ist es nicht möglich Schaltungen zu speichern, in denen Bausteine an der selben Position platziert wurden.

### 3.9.2. Öffnen



● Strg+O

Zum Öffnen eines Projektes wählen Sie »Datei → Öffnen« oder drücken Sie Strg+O. Wählen Sie nun die gewünschte Datei aus und drücken Sie *Öffnen*. Während die Datei geladen wird, präsentiert sich Ihnen eine Fortschrittsanzeige und Sie haben die Möglichkeit, den Vorgang abubrechen. War das Laden erfolgreich, so wird das Projekt in den Projektbaum aufgenommen und eine Ansicht der Hauptschaltung geöffnet.

## 4. Features im Detail

### 4.1. Tools

Die Software stellt viele nützliche Tools zur leichteren Bedienung des Programms zur Verfügung, damit schnell und einfach Schaltungen erzeugt werden können. Diese Tools werden im Folgenden aufgelistet:

- **Auswahl - Werkzeug**

Mit Hilfe dieses Tools lassen sich Linien, Gatter und Komponenten markieren und diese zu dem gewünschten Platz im Gitternetz ziehen. Dabei wird das markierte Element beim Verschieben zur besseren Orientierung im Fenster grau gezeichnet.

Über *Strg+A* werden alle Elemente im Editierfenster markiert. Des Weiteren ist es möglich, markierte Elemente mit Hilfe von *Strg+C* zu kopieren und diese über *Strg+V* wieder am gewünschten Ort einzufügen.

Kopierte Elemente können bei jedem geöffnetem Projekt eingefügt werden.

- **Neuer-Baustein - Werkzeug**

Durch das *Neuer-Baustein - Werkzeug* wählt man aus der *Neue-Baustein-Liste* das gewünschte Gatter aus, z.B. *And-Gatter* oder vordefinierte Komponenten(u.a. *Xor*, *Nand*), und platziert diese im Gitternetz. In diesem Modus ist es außerdem möglich, einzelne Gatter untereinander durch Linien zu verbinden.

- **Kabel - Werkzeug**

Im *Neuen-Baustein-Modus* lassen sich Gatter untereinander verbinden. Mit Hilfe des Kabel - Werkzeug lassen sich jedoch auch Gatter verbinden, aber in diesem Modus können zur Übersichtlichkeit der Schaltung Verbindungen um Gatter herum platzieren werden. Klickt man auf einen Eingang bzw. Ausgang eines Gatters, erscheint der Anfangspunkt der Verbindung. Um jetzt zwei Gatter miteinander zu verknüpfen, können Wegpunkte für die Verbindung bei Mausklick im Gitternetz gesetzt werden. Schließlich werden die gewünschten Gatter miteinander verbunden.

Mit diesem Tool ist es des Weiteren möglich, Verzweigungen von Verbindungen zu machen. Dazu klickt man einfach auf die Verbindung (es wird ein Verzweigungspunkt gesetzt), und verfolgt den gleichen Schritt wie zu Beginn.

- **Ansicht-verschieben - Werkzeug**

Mit gedrückter Maustaste lässt sich in diesem Modus das Editierfenster zum gewünschten Bereich verschieben.

Unter »*Ansicht* → *Ansicht anpassen*« wird das Editierfenster der Schaltung angepasst, damit alle Elemente darin sichtbar werden.

Mit *Leertaste (gedrückt)* + linke Maustaste lässt sich die Anzeige aus jedem Tool heraus bequem verschieben ohne in den *Ansicht verschieben - Modus* wechseln zu müssen.

### 4.2. Der Projektbaum

Der Projektbaum dient der Verwaltung der bekannten Projekte sowie zur Navigation zwischen den geöffneten Projekten.

### 4.2.1. Verwaltung von Projekten

Auf der obersten Ebene des Baumes sind die der Anwendung bekannten Projekte aufgelistet. Hierbei sind geöffnete (ausklappbar) und geschlossene Projekte zu unterscheiden.

Geschlossene Projekte können durch einen Doppelklick geöffnet werden.

Durch markieren eines einzelnen (geöffneten) Projektes wird in dieses gewechselt. Beim Rechtsklick auf ein (geöffnetes) Projekte erscheint ein Kontextmenü mit folgenden Optionen:

- **In einem neuen Reiter öffnen**  
Öffnet eine neue Ansicht der Hauptschaltung des Projektes in einem neuen Reiter.
- **Projekt schließen**  
Schließt das Projekt.
- **Projekt löschen**  
Löscht das Projekt aus dem Projektbaum. Eine eventuell zugehörige Datei bleibt erhalten so dass das Projekt über den Öffnen Dialog wieder geladen werden kann.
- **Projektnamen ändern**  
Ermöglicht es den Projektnamen zu ändern.

### 4.3. Navigation innerhalb von Projekten

Durch aufklappen eines Projektes im Baum erhält man eine Auflistung aller Bausteine innerhalb der Hauptschaltung mit Typ, Name und Position innerhalb der Schaltung. Ein markieren eines dieser Baustein-Einträge bewirkt, dass der entsprechende Baustein innerhalb des Anzeigebereichs der Schaltung markiert und dieser innerhalb des Anzeigebereichs zentriert wird. Das Kontextmenü bietet hier folgende Operationen:

- **Baustein löschen**  
Löscht den zugehörigen Baustein aus der Schaltung.
- **Eigenschaften**  
Zeigt den Einstellungsdialog des zugehörigen Bausteins an in dem dessen Name, Größe und ähnliches verändert werden können.

Eine besondere Rolle nehmen die Einträge für Bausteine vom Typ "Komponente" ein. Diese lassen sich wiederum aufklappen. Hier wird dann der Typ der Komponente angezeigt (z.B. "SHIFTREGISTER"), der wiederum aufgeklappt werden kann und den Inhalt der Schaltung der Komponente auflistet. Bei Komponenten bietet das Kontextmenü folgende weitere Optionen:

- **In einem neuen Reiter öffnen**  
Öffnet eine Ansicht der Komponente in einem neuen Reiter.
- **Als neues Projekt öffnen**  
Erstellt ein neues Projekt mit der Schaltung des Komponententyps als Hauptschaltung.

## 4.4. Einstellungen

Im Menüpunkt *Einstellungen* können zahlreiche Einstellungen am Programm vorgenommen werden. Diese Einstellungen werden in einer Konfigurationsdatei abgespeichert und bleiben auch nach dem Neustart des Programmes erhalten. Im folgenden sind dies:

- **Sprache**

Hier kann eine der Sprachen ausgewählt werden, in die das Programm übersetzt ist. Diese Einstellung wird erst nach einem Neustart des Programmes wirksam, in dem alle ungespeicherten Änderungen verloren gehen.

- **Erscheinungsbild**

Hier kann das Erscheinungsbild der Anwendung verändert werden. Diese Einstellung wird erst nach einem Neustart des Programmes wirksam, in dem alle ungespeicherten Änderungen verloren gehen.

- **Echtzeitvalidierung**



Hier kann die Echtzeitvalidierung ein bzw. ausgeschaltet werden, da diese sehr aufwendig ist und beim Bearbeiten von größeren Schaltungen zu unerwünschten Verzögerungen führen kann.

- **Gitter Einrasten**



Hier kann man einstellen, ob beim Erstellen oder Verschieben von Bausteinen diese am Gitternetz einrasten sollen oder nicht.

- **Simulationshistorie anlegen**



Hier kann man festlegen, ob eine Historie der Simulation angelegt werden soll. Dies ermöglicht ein Rückwärtslaufen der Simulation, verlangsamt diese aber.

- **Niedrige Qualität**

Hier kann auf eine niedrigere Anzeigequalität gewechselt werden. Die kann gerade bei größeren Schaltungen ein flüssigeres Arbeiten ermöglichen.

- **Tooltips Ein/Aus**

Stellt ein, ob die Tooltips für Bauteile der Schaltung angezeigt werden sollen oder nicht.

- **Standardeinstellungen wiederherstellen**

Stellt die Standardeinstellungen wieder her.

## 4.5. Shortcuts

Die Tastaturbelegungen zur leichteren Bedienung der Software in Übersicht:

- **Datei**

- *Strg+N*: Neues Projekt anlegen
- *Strg+O*: Gespeichertes Projekt öffnen
- *Strg+S*: Projekt speichern
- *Strg+Umschalt+S*: Projekt speichern unter

- *Strg+E*: Komponente exportieren
- *Strg+Q*: Programm beenden
- **Bearbeiten**
  - *Strg+Z*: Rückgängig machen
  - *Strg+Y*: Wiederherstellen
  - *Strg+X*: Ausschneiden
  - *Strg+C*: Kopieren
  - *Strg+V*: Einfügen
  - *Entf*: Löschen
- **Ansicht**
  - *F11*: Vollbild
  - *Strg+Plus*: Größer (nicht bei Mac OS X)
  - *Strg+Minus*: Kleiner (nicht bei Mac OS X)
  - *Strg+0*: Zoom auf 100-Prozent
  - *Strg+G*: Gitternetz anzeigen
- **Hilfe**
  - *F1*: funCKit Hilfe
- **Editieren**
  - *Strg+1*: Auswahl - Werkzeug
  - *Strg+2*: Neuer-Baustein - Werkzeug
  - *Strg+3*: Kabel - Werkzeug
  - *Strg+4* oder *Leertaste (gedrückt)* + linke Maustaste: Ansicht-verschieben - Werkzeug
  - *Strg+T*: Neuer Tab
  - *Strg+W* oder mittlere Maustaste: Tab schließen
- **Bedienung**
  - *Scrollrad*: scrollen
  - *Strg+Scroll*: zoomen
  - *Alt+Scroll*: horizontal scrollen

## A. Dateiformat

Das von funCKit verwendete Dateiformat stellt eine Abwandlung des SEP-Austauschformates dar und baut auf der Extensible Stylesheet Language<sup>1</sup> auf.

### A.1. Anforderungen

Um eine Datei im SEP-Austauschformat als funCKit-Datei zu Kennzeichnen, muss der zugehörige Namespace "*fck*" verwendet werden. Dazu ist im Tag "*circuits*"

**xmlns:fck="http://git.sep2011.de/funckit"**

zu setzen. Jede so markierte Datei kann nur geladen werden wenn sie sämtliche der nachfolgenden Anforderungen erfüllt:

- **circuits**  
Das Element *circuits* muss das Attribut *fck:projectname* enthalten.
- **circuit**  
Sofern das Attribut *fck:type* auf *component* gesetzt ist müssen die Attribute *fck:height*, *fck:name*, *fck:orientation* und *fck:width* vorhanden sein.
- **component**  
Eine *component* muss die Attribute *fck:height*, *fck:name*, *fck:orientation* und *fck:width* enthalten. Desweiteren müssen, falls es sich um den *type circuit* handelt, die Elemente *accesspoint* und *mapping* jeweils paarweise enthalten sein. Dabei müssen die *fck:id* und *fck:outerPoint* Attribute übereinstimmen. Das *fck:innerPoint* Attribut muss dabei das *name* Attribut einer *component* in der durch *type2* referenzierten *circuit* deren Attribut *fck:componentTypePoint* auf *true* gesetzt ist referenzieren.  
Hat eine *component* das Attribut *fck:componentTypePoint* auf *true* gesetzt, so darf es nur die Attribute *fck:name*, *fck:posx* und *fck:posy* enthalten.  
Eine *component* kann das Attribut *fck:delay* enthalten, sowie Elemente vom Typ *accesspoint*.
- **connection**  
Eine *connection* muss, sofern nicht das Attribut *fck:ignore* auf *true* gesetzt ist, die Attribute *fck:name*, *fck:source* und *fck:target* enthalten. Dabei müssen *fck:source* und *fck:target* jeweils einen *accesspoint* im selben *circuit* durch seine *fck:id* referenzieren.
- **uuid**  
Alle Attribute vom Typ *fck:uuid*, sowie das Attribut *name* müssen UUIDs enthalten und innerhalb des gesamten Dokuments eindeutig sein.

Im Unterschied zum SEP-Austauschformat müssen Dateien im funCKit Dateiformat die beiden Restriktionen der Eindeutigkeit von Positionen und des Verbots von negativen Positionen nicht einhalten. Deshalb steht beim Speichern als Alternative das SEP-Austauschformat zur Verfügung. Wird dieses gewählt werden zusätzlich die beiden genannten Restriktionen sichergestellt.

---

<sup>1</sup><http://www.w3.org/TR/2008/REC-xml-20081126/>



## A.2. Beschreibung

Wird eine im Programm erstellte Schaltung abgespeichert, so ist eine Transformation in das Dateiformat notwendig. Dabei wird mit der Hauptschaltung begonnen. Für diese wird ein *circuit*-Tag angelegt. Anschließend werden alle Bausteine der Schaltung als *component*-Tag angelegt und deren Eigenschaften - wie Position, Breite, Höhe, Orientierung und Verzögerung - entsprechend als Attribute gespeichert. Als Name wird hier zur eindeutigen Referenzierung eine UUID vergeben. Zusätzlich wird für jeden Anschlusspunkt auf dem Baustein ein *AccessPoint*-Tag erstellt, welcher seine relative Position auf dem Baustein, seinen Namen und seinen Typen (Ein- oder Ausgang) speichert, sowie eine eindeutige UUID erhält. Sofern es sich bei dem Baustein um eine Komponente handelt werden zusätzlich *mapping*-Tags für jeden Anschlusspunkt eingefügt. Dabei wird als äußerer Punkt der Anschlusspunkt auf dem Baustein referenziert und als innerer zum einen ein Anschlusspunkt eines Bausteins in der Schaltung des zugehörigen Types, zum anderen ein Baustein der als Komponententyp-Anschlusspunkt markiert ist und zusätzliche Informationen zur Verfügung stellt.

Im nächsten Schritt werden sämtliche Kabelverbindungen gespeichert, indem ein *connection*-Tag angelegt wird. Dabei werden die *source*, *sourcePort*, *target*, *targetPort* Attribute aus Kompatibilitätsgründen angelegt, jedoch ansonsten von der Anwendung ignoriert. Die eigentlichen Verbindungsinformationen werden durch eigene Attribute *fck:source* und *fck:target* vorgehalten, in welchen direkt die Anschlusspunkte referenziert werden.

Nun werden für alle eben gespeicherten Komponenten die Komponententypen und deren Schaltungen als neuer *circuit*-Tag gespeichert. Dazu werden dessen Eigenschaften wie Name, Höhe, Breite, Orientierung in den entsprechenden Attributen gespeichert und als Name wiederum eine UUID verwendet. Für die Anschlusspunkte des Types werden jeweils Dummy-Komponenten angelegt, die zum einen die Kompatibilität mit dem SEP-Austauschformat sicherstellen, zum anderen Zusatzinformationen wie Name und die relative Position des Anschlusspunktes auf dem Typen speichern. Hierbei erhalten diese Komponenten den selben Namen wie die zugehörigen Anschlusspunkte zur einfachen Referenzierung.

Nun wird wie schon oben beschrieben die Schaltung des Komponententypen gespeichert. Zum Schluss werden noch für die Dummy-Komponenten entsprechende Dummy-Kabelverbindungen angelegt, welche wiederum allein die Kompatibilität mit dem SEP-Austauschformat sicherstellen und ansonsten von der Anwendung ignoriert werden.

Damit ist nun die Schaltung komplett persistiert und kann zur erneuten Bearbeitung später wieder geladen werden.

## A.3. Unterstützte Bausteine

Zusätzlich zu den im SEP-Austauschformat festgelegten Bausteinen wird noch ein Verzögerungselement unterstützt. Dieses wird durch eine *component* mit dem Attribut *type=other* und dem Attribut *type2=delay* festgelegt. Dabei verzögert dieser Baustein seinen Eingangswert genau um eine diskrete Zeiteinheit. Die Benennung der ein und Ausgänge entspricht dem SEP-Austauschformat.

## B. Haftungsausschluss

Die Software wurde mit größter Sorgfalt entwickelt und auf verschiedenen Rechnersystemen sorgfältig getestet. Dabei waren für die freigegebenen Produktversionen keine Fehler festzustellen. Es kann aber nicht garantiert werden, dass die Software auf jedem Zielsystem hundertprozentig fehlerfrei läuft. Deshalb übernehmen wir keine Haftung für Unverträglichkeiten mit Hardwarekomponenten und anderen Softwareprodukten oder deren Komponenten. Die Software wird wie sie ist ("as is") zur Verfügung gestellt, ohne jede Garantie für die Brauchbarkeit für einen bestimmten Anwendungsfall. Das gesamte Risiko, das aus der Nutzung der Software entsteht, liegt beim Anwender der Software. Für Schäden, die direkt oder indirekt aus der Nutzung der Software resultieren, sind wir unter keinen Umständen haftbar zu machen, es sei denn, es liegt ein vorsätzliches oder grob fahrlässiges Verhalten unsererseits vor. Sollten Fehler auftreten, so sind wir bemüht, diese im Rahmen der gegebenen Möglichkeiten zu beheben und eine fehlerbereinigte Version anzubieten. Die beiliegende Dokumentation/Hilfe der Software erhebt keinen Anspruch auf Richtigkeit und Vollständigkeit.

## C. Externe Bibliotheken

Zusätzlich der zum Lieferumfang des Java Development Kit gehörenden Bibliotheken werden zusätzliche Bibliotheken benutzt. Eine kurze Übersicht findet sich hier:<sup>2</sup>

- **Apache Log4j**
  - **Homepage:** <http://logging.apache.org/log4j/1.2/>
  - **Version:** 1.2
  - **Lizenz:** The Apache Software License, Version 2.0
- **MiGLayout - Java Layout Manager for Swing, SWT and JavaFX**
  - **Homepage:** <http://www.miglayout.com/>
  - **Version:** 3.7.4
  - **Lizenz:** BSD
- **Guava: Google Core Libraries for Java**
  - **Homepage:** <http://code.google.com/p/guava-libraries/>
  - **Version:** 10.0.1
  - **Lizenz:** Apache License, Version 2.0
- **Batik SVG transcoder classes**
  - **Homepage:** <http://xmlgraphics.apache.org/batik/>
  - **Version:** 1.7
  - **Lizenz:** The Apache Software License, Version 2.0

---

<sup>2</sup>Die Bibliotheken benötigen zum Teil noch weitere Bibliotheken, die hier nicht aufgeführt sind

- **Batik Java2D SVG generator**

- **Homepage:** <http://xmlgraphics.apache.org/batik/>
- **Version:** 1.7
- **Lizenz:** The Apache Software License, Version 2.0

- **SwingX**

- **Homepage:** <http://www.swinglabs.org/>, <https://swingx.dev.java.net>
- **Version:** 1.6.2-2
- **Lizenz:** Lesser General Public License (LGPL)

- **JIDE Common Layer**

- **Homepage:** <http://java.net/projects/jide-oss/>, <http://www.jidesoft.com/products/oss.htm>
- **Version:** 2.10.2
- **Lizenz:** GPL Version 3 with classpath exception

- **MRJ Toolkit Stubs**

- **Version:** 1.0

- **base64**

- **Homepage:** <http://iharder.net/base64/>
- **Version:** 2.3.8
- **Lizenz:** Public domain

- **Napkin Look and Feel**

- **Homepage:** <http://napkinlaf.sourceforge.net/>
- **Version:** 1.2
- **Lizenz:** BSD

## Glossar

**Alt+Scroll** Bei gedrückter Taste lässt sich bei Betätigung des Scrollrades vertikal scrollen. 23

**Cmd** Die Command-Taste bei Mac OS X. Wechselt in den Zoom-Modus und tritt in Verbindung mit anderen Tasten oft in anderen Kombinationen auf. 6

**Entf** Löschen eines markierten Bereichs. 12, 23

**ESC** Die ESC-Taste 13

**F1** Anzeige des Benutzerhandbuches. 23

**F11** Anzeigen des Programms in Vollbild. 23

**Komponente** Eine Komponente ist eine Schaltung mit benannten Ein- und Ausgängen, die als Baustein in anderen Schaltungen verwendet werden kann. 28

**Leertaste (gedrückt)** Wechselt in den Ansicht verschieben - Modus. 20, 23

**Leertaste (gedrückt)** Wechselt temporär während die Taste gedrückt ist in den Ansicht-verschieben-Modus. 13

**Neue-Baustein-Liste** Bietet eine Auswahl an Bausteinen die zum Erstellen ausgewählt werden können. 12

**Projekt** Ein Projekt besteht aus einer Hauptschaltung und bestimmten Zusatzinformationen wie einem Namen. Ein Projekt kann sowohl in eine Datei gespeichert werden als auch aus einer Datei geladen werden. 20, 21

**Schaltung** Eine Schaltung ist eine Menge von Bausteinen oder Komponenten, die mit Leitungen verbunden sind. 21, 28

**Scrollrad** Das Scrollrad der Maus. Bei Betätigung lässt es sich horizontal scrollen. 23

**Shift (gedrückt)** Wechselt temporär während die Taste gedrückt ist in den Selektions-Modus. 12

**Strg** Die Steuerung-Taste. Wechselt in den Zoom-Modus und tritt in Verbindung mit anderen Tasten oft in anderen Kombinationen auf. 6, 12

**Strg+0** Zoom auf 100-Prozent. 23

**Strg+1** Wechselt in den Selektions-Modus. 12, 13, 16, 23

**Strg+2** Wechselt in den Hinzufügen-Modus. 12, 13, 23

**Strg+3** Wechselt zum Kabelwerkzeug. 12, 13, 23

**Strg+4** Wechselt in den Ansicht verschieben - Modus. 13, 23

- Strg+A** Markiert alle Elemente im Gitternetz. 20
- Strg+C** Kopieren eines markiertes Bereichs. 20, 23
- Strg+E** Komponente exportieren. 23
- Strg+G** Anzeige des Gitternetzes. 15, 23
- Strg+Minus** Verkleinert die Ansicht des Editierpanels. 23
- Strg+N** Neues Projekt öffnen. 14, 22
- Strg+O** Gespeichertes Projekt öffnen. 19, 22
- Strg+Plus** Vergrößert die Ansicht des Editierpanels. 23
- Strg+Q** Programm beenden. 23
- Strg+S** Projekt speichern. 15, 19, 22
- Strg+Scroll** Bei gedrückter Taste lässt sich bei Betätigung des Scrollrades zoomen. 23
- Strg+T** Öffnet einen neuen Tab der aktuell geöffneten Schaltung. 11, 23
- Strg+Umschalt+S** Projekt speichern unter. 15, 22
- Strg+V** Einfügen eines davor kopierten Bereichs. 20, 23
- Strg+W** Schließt den aktuellen Tab. 11, 23
- Strg+X** Ausschneiden eines markiertes Bereichs. 23
- Strg+Y** Wiederherstellen von Editieraktionen. 16, 23
- Strg+Z** Rückgängigmachen von Editieraktionen. 16, 23
- Tooltip** Kästchen mit Informationen, die beim Überfahren eines Elements mit der Maus innerhalb der Benutzeroberfläche angezeigt werden. 22