

# Testdrehbuch

---



## Schaltsysteme-Arbeitsblätter im Netz

*Softwareprojekt 2018*

Dr.-Ing. Heinz-Dietrich Wuttke  
René Hutschenreuter

David Sukiennik

Alexander Beyer  
Lars Hinneburg  
Marcel Burkhardt  
Sarah Löcklin

Eike Krieg  
Linda Oppermann  
Peter Klein  
Stephan Enseleit

## Referenzen

Dieses Testdrehbuch wird von der Projektdokumentation unterstützt und bezieht sich auf die funktionalen Anforderungen (SANE\_FA\_XX) und spezifizierten Produktfunktionen aus dem Pflichtenheft.

## Einleitung

Dieses Testdrehbuch umfasst folgende Ziele:

- Festlegung der Testobjekte
- Festlegung grundsätzlicher Rahmenbedingungen für das Testen sowie Festlegungen zum Zeitplan und den Umweltbedürfnissen
- Festlegungen noch zu erledigender Aufgaben und Verantwortlichkeiten

## Testgegenstände

Das zu testende System ist die Frontend-Anwendung, so, wie der Nutzer sie wahrnimmt. Das System soll in Google Chrome auf einem Desktop-PC und einem mobilen Endgerät, wie z. B. einem Smartphone, getestet werden.

## Zu testende Features

Die Features, die getestet werden sollen, sind die Folgenden:

- Laden der Seite,
- Navigieren der Anwendung bzw. Auswahl der Views,
- Aufrufen der Settings,
- Ändern des Erscheinungsbildes der Anwendung,
- Export des aktuellen Anwendungszustandes,
- Import eines Anwendungszustandes,
- Export im BEAST-Format,
- Wechseln der Sprache,
- Hinzufügen/Löschen von Ein- bzw. Ausgangsvariablen in der Wertetabelle,
- Verändern der einzelnen Bits in der Wertetabelle und dem KV-Diagramm,
- Ändern der Lötunkte in den programmierbaren Strukturen,
- Gruppierung von Bits im KV-Diagramm,
- Minimierung mit Hilfe des Quine-McCluskey-Algorithmus,
- Automatische Bildung von maximalen Blöcken im KV-Diagramm,
- Anzeigen aller Minimierungsschritte (Implikanten und Kürzungstabellen) in der View des Quine-McCluskey-Algorithmus,
- Darstellung und Eingabe der zentralen booleschen Funktion als Funktionsindizes,
- Darstellen der zentralen booleschen Funktion im Mengendiagramm,
- Verschieben der Werte zwischen den Mengen im Mengendiagramm,
- Anzeigen der Funktions-Hasards in der gleichnamigen View,

- Anzeigen der Struktur-Hasards und deren Vermeidung in der gleichnamigen View,
- Auswahl der gewünschten programmierbaren Struktur,
- Anzeigen der programmierbaren Struktur als Ansicht der booleschen Funktion,
- Installation als offline-verwendbare Single-Page Applikation auf Smartphones und Tablet,
- Übergabe einer Startkonfiguration,
- Teacher-Modus,
- Automatische Aktivierung des Dual-View-Mode bei geeigneter Fenstergröße,
- Darstellung im Dual-View-Mode und
- Auswahl der anzuzeigenden Ausgangsvariablen in allen Views, die dieses Feature unterstützen

Diese Liste kann bei Fortschreiten des Projektes erweitert werden.

### **Ansatz**

Die einzelnen Tests werden entsprechend den dokumentierten Testfällen durchgeführt. Der Tester wird die einzelnen Tests ausführen und diese mit „bestanden“ oder „durchgefallen“ markieren. Außerdem sollte der Tester, wenn möglich, weitere Beschreibungen über das tatsächliche Resultat des Tests und andere relevante Informationen hinterlassen.

Wenn ein Test als durchgefallen markiert wurde, so soll ein Fehlerbericht erstellt werden.

### **Kriterien: bestanden/durchgefallen**

Alle Kernfunktionalitäten des Systems sollen wie erwartet und in den Testfällen spezifiziert funktionieren. Kein nicht bestandener Testfall darf sich entscheidend auf die Benutzbarkeit der Anwendung auswirken.

### **Testergebnisse**

Wenn ein Test abgeschlossen ist, sind die Ergebnisse nach einem klar definierten Rahmen festzuhalten, welche später vom Testmanager in einen einzigen Bericht zusammengefügt werden.

## **Testaufgaben**

Die folgenden Aufgaben müssen erledigt werden:

- Für alle zu testenden Features müssen Testfälle geschrieben werden und an die testenden Personen weitergeleitet werden
- Die Testfälle sollten im Laufe des Projekts ergänzt und angepasst werden
- Die Umweltbedürfnisse müssen gegeben sein
- Die Tests müssen durchgeführt werden
- Vorbereiten des zusammenfassenden Testberichts

## **Umweltbedürfnisse**

Für viele Tests, vor allem für die Minimierung mit Hilfe des Quine-McCluskey-Algorithmus, müssen boolesche Ausdrücke verwendet werden, deren minimierte Formen bekannt sind.

## **Verantwortlichkeiten**

Der Testmanager ist dafür verantwortlich den Testprozess zu koordinieren, dem Team mögliche Risiken mitzuteilen und den zusammenfassenden Testbericht zu schreiben. Jeder Tester sollte die Erwartungen bezüglich Fertigstellungstermin und Qualitätsniveau verstehen.

## **Bedarf an Personal und Ausbildung**

Es sollte mindestens zwei Tester geben. Alle Tester sollten fortgeschrittenes Wissen in den Themengebieten Rechnerorganisation und Schaltsysteme besitzen.

## **Zeitplan**

Das Testen soll bereits während der Implementierungsphase immer bei Fertigstellung einer Funktionalität bzw. eines Teilmodules einer solchen Funktionalität, spezifiziert in den Testfällen, durchgeführt werden. Außerdem soll ein allumfassender Test nach Abschluss der Implementierungsphase des Softwareprojektes am 14. Juni 2018 durchgeführt werden. Ist das Team zu diesem Zeitpunkt noch nicht fertig mit der Implementierung, verzögert sich der abschließende Test.

## Testfälle

---

In diesem Abschnitt werden konkrete Testfälle SANE\_T\_XX\_YY der einzelnen Produktfunktionen bzw. Anforderungen definiert, gegen welche die Applikation „SANE“ iterativ im Verlauf der Implementierungs- und Validierungsphase getestet werden soll. Die Bezeichnung „XX“ steht dabei für die Identifikationsnummer der funktionalen Anforderungen und „YY“ für einen Testfall dieser Anforderung. Bezieht sich ein Testfall auf ein Modul ohne Identifikationsnummer, so lautet die Bezeichnung SANE\_T\_00\_YY. Dieser Abschnitt wird im Laufe des Softwareprojekts erweitert. Die Datensätze, welche für manche Tests benötigt werden, sollen als Anlagen in einem externen Dokument beigefügt werden.

Version	Datum	Autor	Änderungsgrund / Bemerkung
0.1	24.04.2018	Alexander Beyer	Erstellung: Allgemeine und einige wichtige Testfälle, deren Funktionalität hohe Priorität haben
0.2	30.04.2018	Alexander Beyer	Überarbeitung und Erweiterung um wichtige Testfälle: SANE_T_01_02 SANE_T_02_01 SANE_T_03_01 SANE_T_04_01

## SANE\_T\_00\_01

**Titel:** Öffnen der Anwendung – Lässt sich die Anwendung auf jedem spezifizierten Endgerät öffnen?

**Beschreibung:** Es soll getestet werden, ob sich die Anwendung auf jedem spezifizierten Endgerät öffnen lässt. Dazu soll das Öffnen der Anwendung mit mehreren Endgeräten getestet werden. Die in Frage kommenden Endgeräte werden zu den Äquivalenzklassen Smartphone (Android und Windows), Tablet-PC, Laptop und PC zusammengefasst, wobei angenommen wird, dass Geräte einer Äquivalenzklasse ähnliche Verhaltensweisen während des Tests aufzeigen.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Beobachte, ob die Seite geöffnet wird
3. Beobachte, ob die Seite ordentlich skaliert
4. Beobachte, ob die Anwendung erwartungsgemäß auf Interaktion reagiert.

**Erwartetes Ergebnis:** Die Anwendung soll geladen werden, sich an die jeweilige Bildschirmgröße des Endgerätes anpassen und einheitlich auf Interaktionen reagieren.

---

## SANE\_T\_00\_02

**Titel:** Navigation – Die Navigation zeigt die vom Nutzer ausgewählten Views an

**Beschreibung:** Beim Klicken der einzelnen Unterpunkte bzw. Namen der einzelnen Views im Navigationsmenü soll beobachtet werden, ob die gewünschte View richtig angezeigt wird bzw. die gewünschten Views richtig angezeigt werden.

**Vorgehen:**

1. Internetseite in Chrome laden
2. View-Namen im Navigationsmenü klicken bzw. berühren
3. Beachte: Im Dual-View-Mode sollten zwei Views ausgewählt werden

**Erwartetes Ergebnis:** Die ausgewählten Views werden angezeigt.

## SANE\_T\_01\_01

**Titel:** View: Wertetabelle – Änderung der Tabellenstruktur

**Beschreibung:** Der Nutzer kann der Wertetabelle neue Ein- bzw. Ausgangsparameter hinzufügen.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Wertetabelle-View in der Navigation auswählen
3. Klick/Tap auf die Schaltflächen "+" bzw. "-", um die Anzahl der Ein- bzw. Ausgangsparameter zu erhöhen bzw. zu verringern  
Beachte: Es gibt eine minimale und maximale Anzahl an Ein- und Ausgangsparametern

**Erwartetes Verhalten:** Die Zeilen der Wertetabelle verdoppeln/halbieren sich beim Betätigen der "+"/"-Schaltfläche über den Eingangsparametern. Die "+"/"-Schaltflächen über den Ausgangsparametern fügen der Wertetabelle eine Spalte für einen weiteren Ausgangsparameter hinzu bzw. entfernen diese.

---

## SANE\_T\_01\_02

**Titel:** View: Wertetabelle - Klick oder Tap auf die einzelnen Tabelleneinträge

**Beschreibung:** Der Nutzer kann durch Klicken/Berühren der einzelnen Tabelleneinträge, diese zwischen klar definierten Werten rotieren lassen.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Wertetabelle-View in der Navigation auswählen
3. Klick/Tap auf die Einträge der Ausgabefunktionen
4. Klick/Tap auf die Einträge des h\*-Ausdrucks

**Erwartetes Verhalten:** Die Werte der Ausgabefunktionen sollten zwischen 0 und 1 rotieren, die Werte des h\*-Ausdrucks zwischen "leer" und \*.

## SANE\_T\_02\_01

**Titel:** View: Karnaugh-Veitch-Diagramm - Klick oder Tap auf die einzelnen Tabelleneinträge

**Beschreibung:** Ein Klick oder Tap auf ein Bit schaltet dieses zwischen 0, 1 und \* um.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Karnaugh-Veitch-Diagramm-View in der Navigation auswählen
3. Klick/Tap auf ein Bit

**Erwartetes Verhalten:** Das Bit sollte zwischen 0, 1 und \* umschalten.

---

## SANE\_T\_03\_01

**Titel:** View: Boolesche Ausdrucksalgebra - richtige Interpretation der eingegebenen Ausdrücke

**Beschreibung:** Es soll überprüft werden, ob der Parser eingegebene boolesche Ausdrücke bzw. Terme korrekt interpretiert. Dieser Test soll mit Hilfe der Internetseite "https://pegjs.org" durchgeführt werden. Zum Testen soll ein Datensatz verwendet werden, von dem angenommen werden darf, dass, wenn alle Daten einmal eingegeben wurden und alles richtig gedeutet wurde, auch alle anderen möglichen Kombinationen korrekt interpretiert werden (Äquivalenzklassen).

**Vorgehen:**

1. Quelltext vom Parser auf "https://pegjs.org" einfügen
2. Eingeben aller Ausdrücke aus dem Datensatz

**Erwartetes Ergebnis:** Eingegebene Schachtelungen und Zeichen werden vom Parser korrekt gedeutet. In der nachfolgenden Liste wird definiert, wie der Parser bestimmte Zeichen interpretieren soll.



Zu prüfende Zeichen:

- "&" symbolisiert "and" (logisches Und)
- "+" symbolisiert "or" (logisches Or)
- "/" symbolisiert "not" (logisches Nicht)
- "~" symbolisiert "equiv" (Äquivalenz)
- "/~" symbolisiert "antival" (Antivalenz)
- ">" symbolisiert "imply" (Implikation)

---

## SANE\_T\_04\_01

**Titel:** View: Quine-McCluskey-Algorithmus - richtiges Minimieren mit Hilfe des QMC-Algorithmus

**Beschreibung:** Die zentrale boolesche Funktion soll vom QMC-Algorithmus vollständig und korrekt minimiert werden.

Beachte: Für diesen Test ist ein Datensatz an booleschen Ausdrücken nötig, deren Verhalten gegenüber der Minimierung durch den QMC-Algorithmus bekannt ist. Außerdem wird ein fortgeschrittenes Wissen des Testers über den QMC-Algorithmus vorausgesetzt.

**Vorgehen:**

1. Internetseite im Browser (vorzugsweise Chrome) laden
2. Quine-McCluskey-Algorithmus-View in der Navigation auswählen
3. Eine Ausgangsfunktion aus dem oben beschriebenen Datensatz auswählen (muss vorher eingegeben bzw. importiert worden sein)

**Erwartetes Verhalten:** Alle Kürzungstabellen, die Liste der Primimplikanten und der für die endgültigen minimierten Terme verwendeten Implikanten sowie die Minimierung stimmen mit den Minimierungen aus dem Datensatz überein.

---

## SANE\_T\_11\_01

**Titel:** Settings – Darstellung und Reaktionsfähigkeit der Settings-Optionen

**Beschreibung:** Es soll geprüft werden, ob die implementierten Settings-Optionen angezeigt werden und ob die Unterpunkte auf Interaktionen reagieren. Dieser Test kann iterativ wiederholt werden, sobald die Settings um eine neue Funktion erweitert wurden.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Ansteuern des Settings-Icons
3. Beobachte, ob die Unterpunkte korrekt dargestellt werden
4. Klicke auf die einzelnen, bereits hinzugefügten Funktionen

**Erwartetes Verhalten:** Die Unterpunkte der Settings werden wie gewünscht dargestellt und reagieren auf Klicks.

---

## SANE\_T\_11\_02

**Titel:** Erscheinungsbild - Änderung des Erscheinungsbildes der Anwendung über die Settings

**Beschreibung:** Der Nutzer sollte in den Settings erfolgreich zwischen einem Tag- und einem Nachtmodus umschalten können.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Ansteuern des Settings-Icons
3. Klick/Tap auf den Tag/Nacht-Switch

**Erwartetes Verhalten:** Die Anwendung sollte aus dem Tag- in den Nachtmodus oder umgekehrt wechseln, je nachdem in welchem Modus sie vorher war.

---

## SANE\_T\_18\_01

**Titel:** Spracheinstellungen

**Beschreibung:** Der Nutzer sollte in den Settings erfolgreich zwischen den Sprachen Englisch und Deutsch umschalten können.

**Vorgehen:**

1. Internetseite in Chrome laden
2. Ansteuern des Settings-Icons
3. Klicken bzw. Berühren des DE-/EN-Switches

**Erwartetes Verhalten:** Die Sprache der Anwendung sollte zwischen Deutsch und Englisch wechseln.