

# Entwicklung von komplexeren Falstad-Beispielen

## Student Group

| First Name | Surname | Matrikel Nr. |
|------------|---------|--------------|
|            |         |              |
|            |         |              |
|            |         |              |

## Table of Contents

|                                                      |   |
|------------------------------------------------------|---|
| Entwicklung von komplexeren Falstad-Beispielen ..... | 2 |
| <i>Details</i> .....                                 | 2 |
| <i>Aufgaben</i> .....                                | 2 |

# Entwicklung von komplexeren Falstad-Beispielen

## Details

- [Falstad](#) (bzw. [CircuitJs](#)) wird in verschiedenen meiner Fächer für anschauliche Erklärungen genutzt (z.B. in [Schaltungstechnik](#) oder [Digitaltechnik](#)).
- Seit Kurzem steht für Falstad eine [JavaScript Schnittstelle](#) zur Verfügung.
- Diese Schnittstelle ermöglicht eine Anbindung von anderen HTML-Komponenten außerhalb mit der Simulation.

## Aufgaben

Entwicklung von JavaScript Code für Anbindung von Ein-/Ausgabe auf einer Webseite an CircuitJs

1. Verbinden von Zustandsmaschinen in JS und Verbinden mit der Schaltung
2. automatisiertes Erstellen von Schaltungen aus Funktionstabelle / KV-Diagramm / Zustandsmaschine
3. Standard Ein-/Ausgabe Feld für Binärwerte (für diverse Digitalschaltungen)
4. Diverse Verbesserungen für Elektrotechnik- und Elektronik-Simulationen (z.B. Eingabe über Phasoren)
5. Scripts zum Erstellen von elektrischen Schaltungen (z.B. für Übungen)

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

Permanent link:  
[https://wiki.mexle.te.hs-heilbronn.de/studentische\\_arbeiten/entwicklung\\_von\\_komplexeren\\_falstad-beispielen?rev=1646691056](https://wiki.mexle.te.hs-heilbronn.de/studentische_arbeiten/entwicklung_von_komplexeren_falstad-beispielen?rev=1646691056)

Last update: **2022/03/07 23:10**

