

# Infos zur Verwendung von RAM und ROM

## Student Group

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

## Table of Contents

|  |   |
|--|---|
| <b>Infos zur Verwendung von RAM und ROM</b> .....      | 2 |
| <b><i>Nutzung von RAM/ROM in Simulide</i></b> .....    | 2 |
| <b><i>Beispiel in Simulide</i></b> .....               | 2 |
| Konvertierung von *.wav / *.mp3 in *.dat / *.wav ..... | 2 |
| Dateien .....  | 3 |

# Infos zur Verwendung von RAM und ROM

## Nutzung von RAM/ROM in Simulide

- Bei Verwendung der Komponente RAM/ROM Memory in Simulide können die Daten über das Kontextmenu (Rechtsklick aus die Komponente) geladen und abgespeichert werden.
- Falls Daten aus einer Datei geladen werden sollen, so sind folgende Varianten möglich:
  - Variante 1: nicht löscher
    1. Bei den Eigenschaften der Komponente ist nicht löscher ausgewählt
    2. Daten können dann **vor oder nach dem Start** per Kontextmenu geladen werden
  - Variante 2: löscher
    1. Bei den Eigenschaften der Komponente ist nicht löscher **nicht** ausgewählt
    2. Daten können dann nur **nach dem Start** per Kontextmenu geladen werden
  - Andernfalls wird der Speicher beim Start des Simulation wieder mit 0 gefüllt
- Für das Auslesen der Daten muss der Eingang /WE auf  $\$5\text{rm}\sim V\$$  und /OE auf  $\$0\text{rm}\sim V\$$  und gelegt werden.  
Dann sind folgende Varianten möglich
  - Variante 1: asynchron
    1. Bei den Eigenschaften der Komponente ist asynchron ausgewählt
    2. Die Daten liegen an D0...D7 direkt an
  - Variante 2: synchron
    1. Bei den Eigenschaften der Komponente ist asynchron **nicht** ausgewählt
    2. Die Daten liegen an D0...D7 erst nach einer fallenden Flanke am Eingang /CS

## Beispiel in Simulide

### Konvertierung von \*.wav / \*.mp3 in \*.dat / \*.wav

Um die Daten zu konvertieren, kann das Freeware Tool [Audacity](#) genutzt werden. In diesme sind folgende Schritte notwendig:

1. im Footer von Audacity die Audioposition auf samples stellen (falls dies nicht schon automatisch eingestellt ist)



2. Datei (wav / mp3) laden.
  1. Hier kann z.B. auf freie, selbst-erzeugte Songs von [Suno](#) oder Loops aus [FreeSound.org](#)

zurückgegriffen werden

2. Audacity hat mit [OpenVINO](#) auch ein eigenes AI Tool zum lokalen Erzeugen von Musik. Dies ist aber nur bei hinreichend schneller Hardware empfohlen (Ein Laptop von 2021 reicht nicht).
3. Den gewünschten Bereich markieren und eine separate Spur daraus herstellen (Auswahl mit Maus, <Strg>+X, <Strg>+V, Rest löschen über: markieren und <Entf>)
4. Die Daten über Spuren » Mix » Stereo zu Mono heruntermischen auf nur eine Mono-Spur zusammenlegen
5. Es empfiehlt sich nun die Daten durch ein Tiefpass zu filtern, z.B. mit 4000Hz und : Effekt » Low-Pass-Filter. . » Frequenz: 4000,0 und Rolloff: 48 dB/Octave. Damit kann die Menge an Daten und die Wiedergabefrequenz im Anschluss reduziert werden.
6. Abspeichern der Daten:
  1. Der Einfachste Weg ist das Abspeichern eines Audiofiles:
    1. Datei » Audio exportieren
    2. Format: WAV, Kanäle: Mono, Abtaste: 8000 Hz, Encodierung: Unsigned 8-bit PCM
  2. Es gibt aber auch einen umständlichen Weg über den Sample-Export, der im Anschluss noch konvertiert werden muss:
    1. Markieren aller Daten im Track: <Strg>+A
    2. Werkzeuge » Sample-Datenexport
    3. Ausgabe beschränken auf die ersten: 65535 (oder Samplewert des letzten Samples), Messskala: linear, Datei exportieren nach: Pfad + Name eingeben, Index: Aus, Headerinfo: Aus
    4. Konvertieren der Daten: Die Daten liegen als float \$[-1.00, ... , +1.00]\$ vor und müssen auf \$[0, ... , 255]\$ konvertiert werden. Dazu kann die folgende Excel-Datei genutzt werden: [conversiontool.xlsx](#)
7. 8-bit \*.wav Dateien und \*.data Dateien mit Komma-separierten Werten können beide direkt in Simulide in ein RAM Baustein laden:
  1. Rechtsklick auf RAM-Baustein
  2. Daten laden

## Dateien

- Simu-Datei: [audio\\_ram\\_example.sim1](#)
- Beispiel-Dateien für Audio: [ram\\_rom\\_data.zip](#)

Die Simulide Datei ermöglicht:

- ohne Mikrocontroller
- die Daten aus einem adressierbaren RAM/ROM Byte für Byte auszulesen
- aus diesen 8-Bit Daten Analogwerte zu generieren
- diese Analogwerte auf einem Scope und Lautsprecher auszugeben

Ich empfehle, zur Performancesteigerung folgendes:

- Schaltungsoptionen optimieren: Einstellungen (⚙) » Schaltung
  - Farbanimation deaktivieren
  - Canvas aktualisieren nur mit 10 Bilder pro Sek.
- Nicht notwendige Ausgaben entfernen
  - 7-Segmentanzeige für Adresse und Ausgabe löschen
  - Oszi löschen

From:

<https://wiki.mexle.org/> - **MEXLE Wiki**

Permanent link:

[https://wiki.mexle.org/microcontrollertechnik/infos\\_zur\\_verwendung\\_von\\_ram\\_und\\_rom](https://wiki.mexle.org/microcontrollertechnik/infos_zur_verwendung_von_ram_und_rom)

Last update: **2024/04/17 02:00**

