XML als Speicherformat

Um dieses Kapitel nachzuvollziehen und zu verstehen, wie der XML Parser in diesem Programm funktioniert, werden simple XML Grundlagen benötigt. Diese kann man sich unter anderem mit zusätzlichen Informationen auf der Tutorial Seite von w3schools aneignen (https://www.w3schools.com/xml/).

XML Parser

Bei dem benützten XML Parser handelt es sich um eine selbstgeschriebenen Klasse, die Dateien ausliest und mithilfe von Stringalgorithmik analysiert. Dabei wird vor allem Wert auf die Attributwerte der XML Elemente gelegt. Der Parser ist dabei präventiv gestaltet, das heißt bei auftretenden Fehlern im XML Format bricht er in der Regel den Auslesevorgang für das einzelne Element oder aber für die gesamte XML Datei ab. Dies wurde eingebaut um Folgefehler und unvorhersehbares Verhalten des Schaltplaneditors durch falsche Änderungen an Dateien durch den Benutzer zu vermeiden. Als Zusatzfunktion enthält der Parser die Möglichkeit Änderungen an Dateien mittels Hash-Codes zu verfolgen. Anzumerken ist, dass es zum derzeitigen Stand nicht möglich ist einzelne Elemente oder Zeilen auszuklammern, der Parser wird sie trotzdem auslesen.

Speichern von Schaltplänen

Das Speichern von Schaltplänen und auswählen der Zieldatei erfolgt über den typischen betriebssystemabhängigen File Browser. Wird anstatt einer neuen Datei eine bereits existierende ausgewählt wird diese nach einer Bestätigung komplett überschrieben. Eine XML Schaltplandatei besteht aus der XML Deklaration und dem Root Element <Circuit>. Das Root Element enthält sämtliche Schaltplankomponenten und Verbindungen zwischen diesen. Die Speicherung von Werten der Komponenten erfolgt dabei als Attribut der XML Komponenten im typischen XML Stil, property=“value“. Ein Schaltplan kann dabei eine beliebig große Anzahl an Komponenten beinhalten, da jede Komponente in sich geschlossen ist und lediglich Verbindungen die IDs anderer Komponenten benützen. (Bild von circuit xml oder aber Ladevorgang in BlitzEdit)

Manipulation von Schaltplandateien

Eine gespeicherte Komponente hat den folgenden typischen Aufbau:

<component id="12" x="750" y="650" rot="0" type="Kondensator" hash="-1455387425">

<child comp="12" conn="10;11;" hash="1418336891"/>

<connector id="10" x="750" y="825" rot="0" relX="0" relY="100" hash="-1756379682">

<connection conn1="10" conn2="6" hash="975596792"/>

</connector>

<connector id="11" x="750" y="550" rot="180" relX="0" relY="-100" hash="-812529944">

<connection conn1="11" conn2="2" hash="-831857675"/>

</connector>

</component>

Jeder der Werte kann durch den Benutzer mit einem simplen Texteditor geändert werden. Jedoch ist zu beachten, dass keine Änderungen an den IDs durchgeführt werden sollte.

Das component Element enthält sämtliche Information über den Typ der Komponente und der Position auf dem Schaltplan. Die Zuordnung von Verbindungsstücken (connectors) mit dem Komponent erfolgt über das child Element. Verbindungsstücke enthalten die ursprüngliche relative Position und die aktuelle Position auf dem Schaltplan. Verbindungen werden über das connection Element erstellt. Dabei wird die ID von zwei zu verbindenden Verbindungsstücke angegeben.

Werden manuelle Änderungen durchgeführt und in BlitzEdit geladen, wird dieses anhand der hash Attributen der Elemente erkannt und bei fehlerhaften Änderungen das geänderte Element ausgegeben. Diese Überprüfung kann deaktiviert werden durch entfernen des circuithash Elementes am Anfang vom Root Element.

Laden von Schaltplänen

Das Laden erfolgt, wie bei dem Speichern, mittels dem File Browser. Sollte eine Änderung der Schaltplandatei durch den Benutzer durchgeführt worden sein, muss dieser den Ladevorgang bestätigen, da nicht feststellbar ist, ob die Datei noch kompatibel mit dem Parser ist. Akzeptiert der Benutzer versucht der Parser die Datei zu lesen und analysieren. Tritt ein Fehler auf, wie zB. das fehlende einbinden einer Komponente in die aktuellen Bibliotheken, wird eine Meldung als Pop-up angezeigt und der Ladevorgang unterbrochen. (Bild mit Warnung)

Es können auch Änderungen an der Schaltplandatei durchgeführt werden während diese in BlitzEdit geöffnet ist. Der geänderte Schaltplan kann über die Reload (Shortcut: F5) Funktion einfach erneut geladen werden. (Bild mit Warnung)

Komponenten im XML Format

Die Deklaration einer Komponente in XML hat das folgende Format:

<?xml version="1.0" encoding="utf-8"?>

<!-- Bauteil Deklaration: Spannungsquelle -->

<component type="Spannungsquelle">

<svg path="img/Spannungsquelle.svg"/>

<connector x="0" y="100" rot="0"/>

<connector x="0" y="-100" rot="180"/>

<property name="Spannungsquelle" type="decimal" unit="volt"/>

</component>

Eine Komponente besteht aus einer beliebigen Anzahl von Verbindungsstücken, einem SVG Bild und einer beliebigen Anzahl an Eigenschaften, wie z.B. einem Feld mit der Einheit Volt für eine Spannungsquelle. Die Pfadangabe zu einem SVG Bild erfolgt dabei relativ, sodass auch ganze Ordner einfach als Bibliotheken zwischen verschiedenen Benutzern und Computern ausgetauscht werden können. Verbindungsstücke bestehen aus Positions Attributen, die die relative Position zu der Komponente beschreiben, und ihre relative Rotation. Eigenschaften bestehen aus einem Namen, einem Datentyp (string, integral, decimal) und einer Einheit (ohm, farrad, volt, ampere, henry, watt, coulomb, hertz, windungen).

Modifikation von Komponenten

Um das Aussehen einer Komponente zu verändern muss das SVG Bild geändert werden. Dies kann man auch mittels einem einfachen Texteditor durchführen. Der SVG Renderer in BlitzEdit unterstützt bisher nur rechteckige Elemente und Rahmen. Um Verbindungsstücke oder Eigenschaften hinzuzufügen, können einfach vorhandene kopiert und abgeändert werden oder neu geschrieben werden. Dabei ist jedoch auf die Syntax und Tippfehler zu achten.

Erstellen von Komponenten

Um eine neue Komponente zu erstellen kann die bereits existierende Template Komponente im test Ordner benützt werden. Dazu muss diese an den gewünschten Zielort kopiert werden und der Name der Datei und der Komponente geändert werden. Schließlich muss lediglich der Pfad zu dem SVG Bild angegeben werden und ,wenn benötigt, Verbindungsstücke und Eigenschaften hinzugefügt werden. Wurde die Komponte erfolgreich erstellt kann diese über die Import Funktion in BlitzEdit hinzugefügt und benützt werden. (Bild mit Import?)