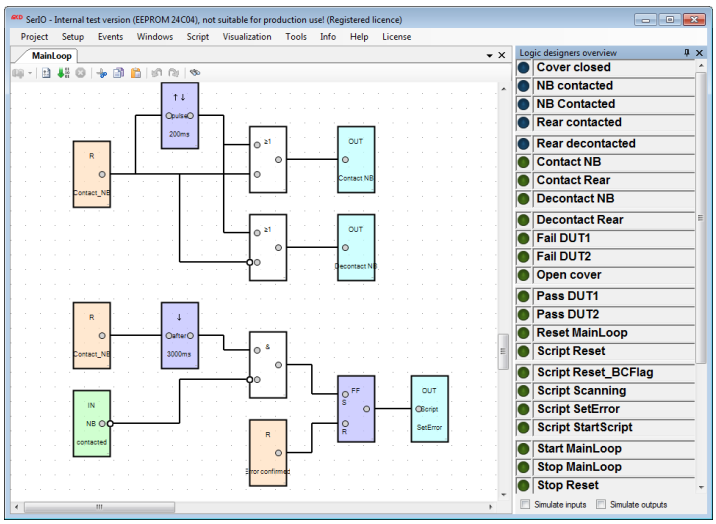
Bedienungsanleitung

Toolmonitor Logic Designer



GET IN touch WITH SENSITIVE TESTING

Softline

Modline

Conline

Boardline

Avidline

Pixline

Applikation

MCD Elektronik GmbH
Hoheneichstr. 52
75217 Birkenfeld
Telefon 0 72 31/78 405-0
Telefax 0 72 31/78 405-10
info@mcd-elektronik.de
www.mcd-elektronik.de
Sitz: Birkenfeld
Geschäftsführer: Bruno Hörter
Registergericht Mannheim
HRB 505692

Vorlagenversion: 5.4 / 2018-04-09
V1.7 2018-10-09 HB (TS)

Inhalt

1. ALLGEMEIN 4

2. ALLGEMEINES ÜBER DIE BENUTZEROBERFLÄCHE 6

2.1. GRAFISCHE BEDIENELEMENTE IM DESIGNMODUS 6

2.2. GRAFISCHE BEDIENELEMENTE IM AUSFÜHRUNGSMODUS 7

3. ERSTELLEN EINES EINFACHEN NETZWERKES 8

3.1. ANLEGEN EINES LOGIC DESIGNS 8

3.2. LAYOUT - KONFIGURATIONEN IM DESIGNMODUS 9

3.3. LOGIKELEMENTE HINZUFÜGEN UND BEARBEITEN 11

3.4. LOGIKELEMENTE VERBINDEN 13

3.5. LÖSCHEN VORHANDENER LOGIKELEMENTE 13

4. TESTEN / AUSFÜHREN / DEBUGGEN DER SCHALTUNG IM ARBEITSMODUS 13

5. LOGIKELEMENTE IM DETAIL 15

5.1. ALLGEMEINE EIGENSCHAFTEN 17

5.2. LOGIKELEMENT INPUT 17

5.3. LOGIKELEMENT AND 18

5.4. LOGIKELEMENT COMMENT 20

5.5. LOGIKELEMENT DELAY 20

5.6. LOGIKELEMENT EDGE / PULSE 21

5.7. LOGIKELEMENT DEBOUNCER 23

5.8. LOGIKELEMENT FLASH 23

5.9. LOGIKELEMENT FLIPFLOP 25

5.10. LOGIKELEMENT GLOBAL VARIABLE 26

5.11. LOGIKELEMENTE OR / XOR 29

5.12. LOGIKELEMENT OUTPUT 30

5.13. LOGIKELEMENT HIGH / LOW 35

5.14. LOGIKELEMENT SCRIPT 35

6. LOGIC DESIGNERS CONTROL 39

6.1. ÜBERSICHT 39

6.2. ÖFFNEN DES LOGIC DESIGNERS CONTROL 39

6.3. GRAFISCHE ELEMENTE 40

6.4. AKTUELLER STATUS 40

6.5. DAS AUSFÜHRUNGSINTERVALL UND DIE VERARBEITUNGSDAUER 41

6.6. DIREKTE STEUERUNG DER LOGIC DESIGNS: STARTEN / STOPPEN UND ÖFFNEN 41

6.7. DETAILANSICHT 42

1. Allgemein

Der Logic Designer ermöglicht die grafische Programmierung von Ein - und Ausgängen in MCD Toolmonitoren. Im Allgemeinen können logische Verknüpfungen zwischen Ein - / und Ausgängen implementiert werden. Hierbei kann jede Größe, die im MCD Toolmonitor über den „GetValue“ - Befehl abgefragt werden kann, als Eingang definiert, und jede Größe, die über den „SetValue“ - Befehl gestellt werden kann als Ausgang definiert werden. Diese Ein - und Ausgänge können dann mit den im Folgenden beschriebenen Logikbausteinen miteinander verknüpft werden. Stehen zu Testzwecken keine Ein - und Ausgänge zur Verfügung, so können diese auch simuliert werden. Des Weiteren besteht die Möglichkeit den Logic Designer mit Variablen, Funktionen und Events aus der Script Engine zu verknüpfen.

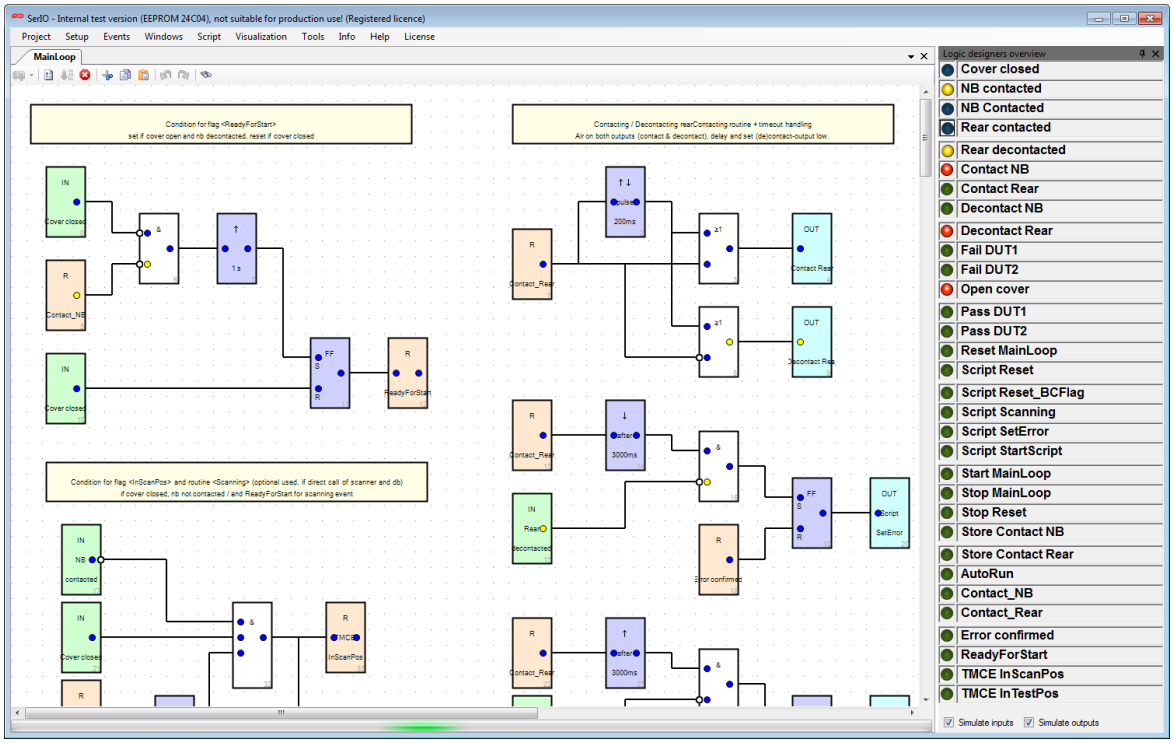


Abbildung 1: Beispielanwendung Logic Designer

Bestellnummer: # 150733

Im Logic Designer der MCD Toolmonitore steht eine Vielzahl an logischen Elementen und elektronischen Standardschaltungen (wie z.B. „FlipFlops“) zur Verfügung. Des Weiteren erlauben die Logikbausteine „Input“, „Output“, „Global Variable“ und „Script“ die nahtlose Integration der Logikschaltungen in die MCD Toolmonitore über die Schnittstelle „Virtual Interface“.

Die nachfolgende Tabelle gibt einen kurzen Überblick über die zur Verfügung stehenden Logikbausteine.

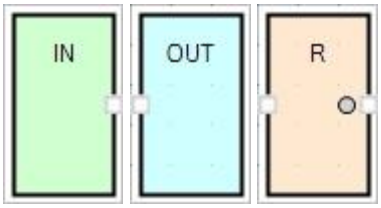
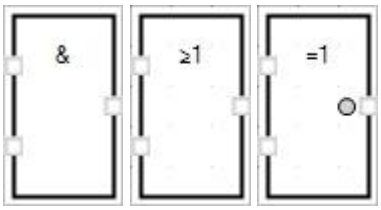
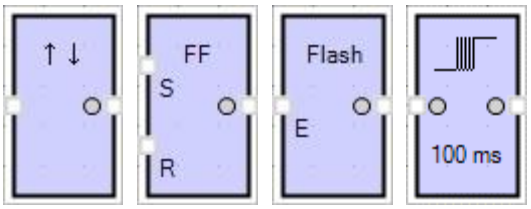
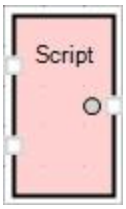
Input, **Output** und **Global Variable** *Input, Output* und *Global Variable* erlauben die Kommunikation über das *Virtual Interface* und dienen der Anbindung der logischen Schaltung an die zugreifbaren Größen des Toolmonitors.	
AND, **OR** und **XOR** Diese Gatter implementieren die einfachsten logischen Verknüpfungen von zwei (und mehr) Eingängen und einem Ausgang.	
Edge - / Pulse - Gatter zur Puls - / Flankenerkennung. **FlipFlop** - Gatter **Flash** - Baustein zum periodischen Ein - und Ausschalten eines Ausgangs. **Debouncer** - Baustein zum Entprellen von Eingängen.	
Script - Baustein zur direkten Verknüpfung der Logikschaltung mit einer Methode aus der Script Engine und zur freien Programmierung der Funktionalität des Scripts.	
Delay - Baustein zum Einbau von Ausführungsverzögerungen im Logikschaltplan. **High** und **Low** stellen dauerhaft ein High - / Low - Signal zur Verfügung.	
Comment - Baustein kann zur Kommentierung des Schaltplans verwendet werden.	

Abbildung 2: Überblick der zur Verfügung stehenden Logikbausteine

Die Funktionsweise und Verwendung der einzelnen Logikbausteine wird im Kapitel *Logikelemente im Detail* genauer erläutert.

2. Allgemeines über die Benutzeroberfläche

Grundlegend verfügt ein Logic Design über zwei Modi: den Designmodus (Entwurfsmodus) und den Ausführungs - / Arbeitsmodus. Das Erstellen / Bearbeiten eines Netzwerks / einer Schaltung erfolgt immer im Designmodus. Das Testen / Ausführen der Schaltung erfolgt im Arbeitsmodus. Im Arbeitsmodus sind alle Editierfunktionen deaktiviert.

Das Umschalten zwischen den beiden Modi erfolgt über die Kontextmenübefehle „Designmode“ und „Run“, oder die entsprechenden Symbolleistenschaltflächen und .

2.1. Grafische Bedienelemente im Designmodus

Im Designmodus unterteilt sich der Arbeitsbereich in:

- Layoutbereich (1.)
- Symbolleisten (2.)
- Eigenschaftenfenster (3.)

Über das Kontextmenü des Layoutfensters (4.) stehen erweiterte Editierfunktionen zur Verfügung.

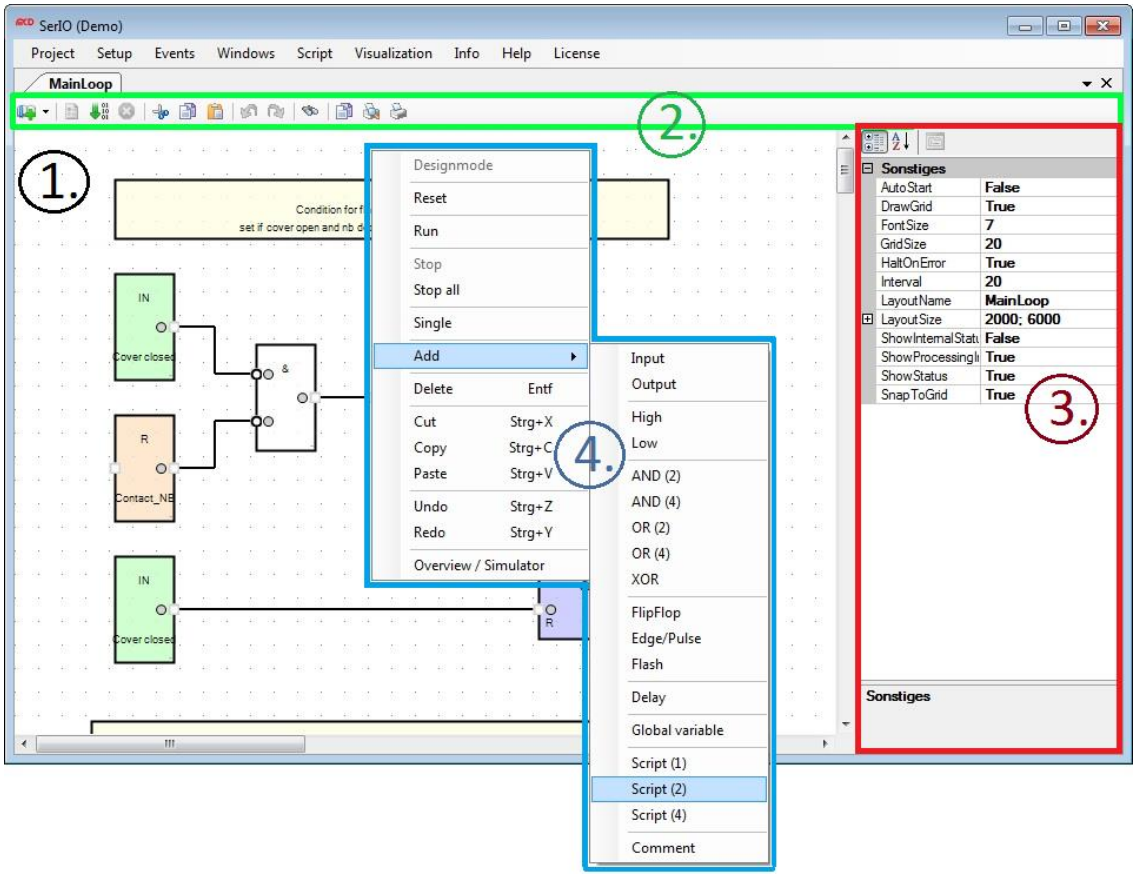


Abbildung 3: Übersicht der Bedienoberfläche im Designmodus

2.2. Grafische Bedienelemente im Ausführungsmodus

Im Ausführungsmodus unterteilt sich der Arbeitsbereich in:

- Layoutbereich (1.)
- Symbolleisten (2.)
- Zuschaltbares, optionales Fenster „Logic designers overview“ (3.)
- Fortschrittsbalken (4.)

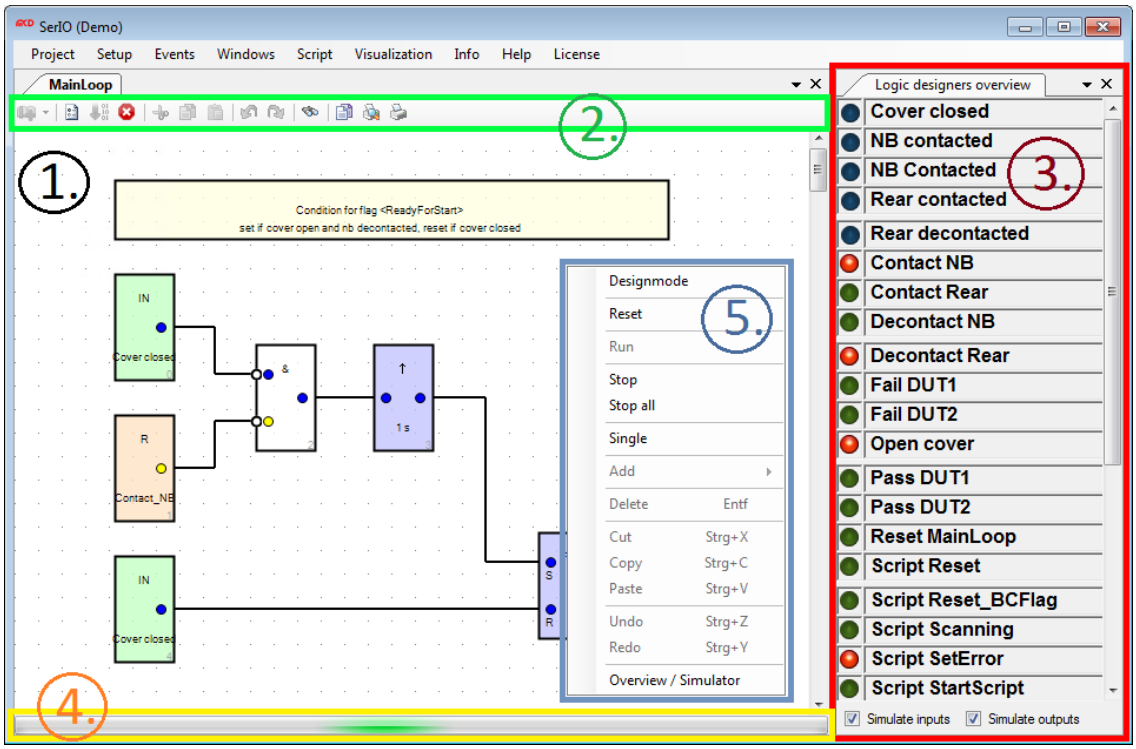


Abbildung 4: Übersicht der Bedienoberfläche im Ausführungsmodus

Die Ausführung des Programms erkennt man an dem animierten Fortschrittsbalken (4.). In dem optionalen, zuschaltbaren Fenster „Logic designers overview“ kann der aktuelle Zustand aller Ein - und Ausgänge visualisiert und überwacht werden (3.). Alle Editierfunktionen sind im Arbeitsmodus deaktiviert (5.).

3. Erstellen eines einfachen Netzwerkes

3.1. Anlegen eines Logic Designs

Um einen neuen Logic Designer anzulegen, muss, wie auf der unteren Abbildung dargestellt, in den Optionen des verwendeten Toolmonitors die Anzahl an Logic Designers verändert werden. Hier ist es auch möglich mit mehreren Designern zu arbeiten.

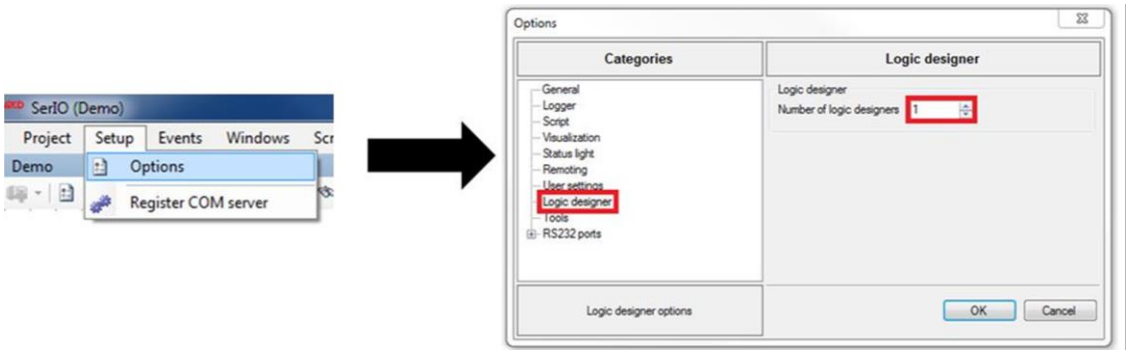


Abbildung 5: Logic Designer in den Toolmonitor - Options anlegen

Nachdem der Logic Designer in den Optionen angelegt wurde, erscheint nun ein neuer Menüeintrag unter „Script“ mit dem Namen „Logic Designer“. Über einen Linksklick kann der gerade angelegte Designer mit dem standardmäßigen Namen „Logic Designer1“ geöffnet werden.

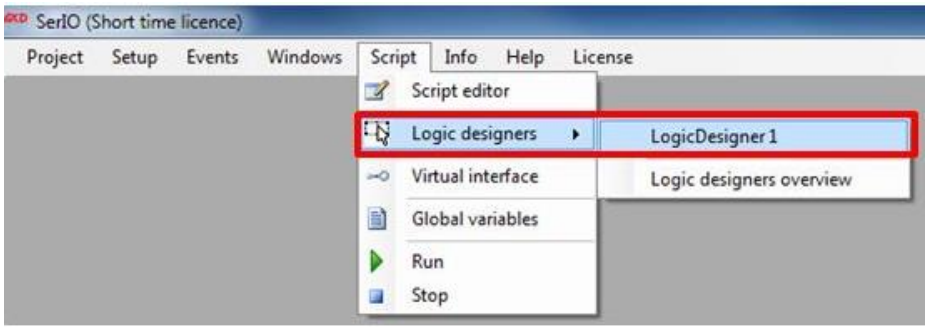


Abbildung 6: Logic Designer öffnen

Wird der angelegte Logic Designer das erste Mal geöffnet, erscheint die leere Benutzeroberfläche des Designers mit einem angedeuteten Raster.

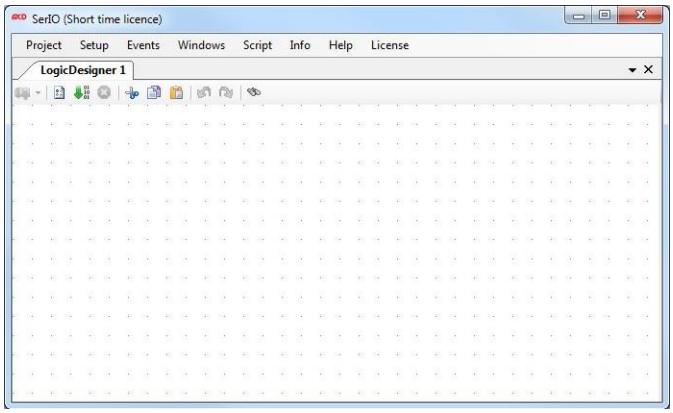


Abbildung 7: Leere Benutzeroberfläche

3.2. Layout - Konfigurationen im Designmodus

Um neue Bausteine in die Oberfläche einzufügen oder vorhandene Bausteine zu bearbeiten / löschen, muss immer in den sogenannten Designmodus, auch Entwurfsmodus, gewechselt werden. Dieser ist über einen Rechtsklick auf die weiße Oberfläche unter dem Kontextmenüeintrag „Designmode“ oder unter dem weißen Icon am oberen linken Bildrand erreichbar.

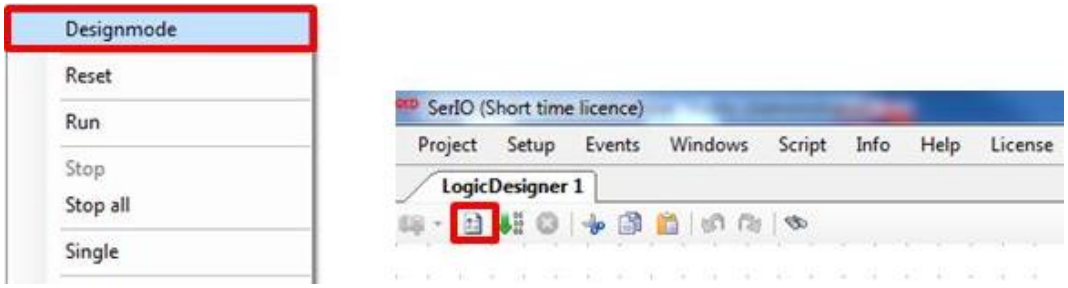


Abbildung 8: Aktivieren des Designmodus

Nachdem in den Designmode gewechselt wurde, erscheint am rechten Bildrand eine Tabelle mit Einstellmöglichkeiten zur Benutzeroberfläche.

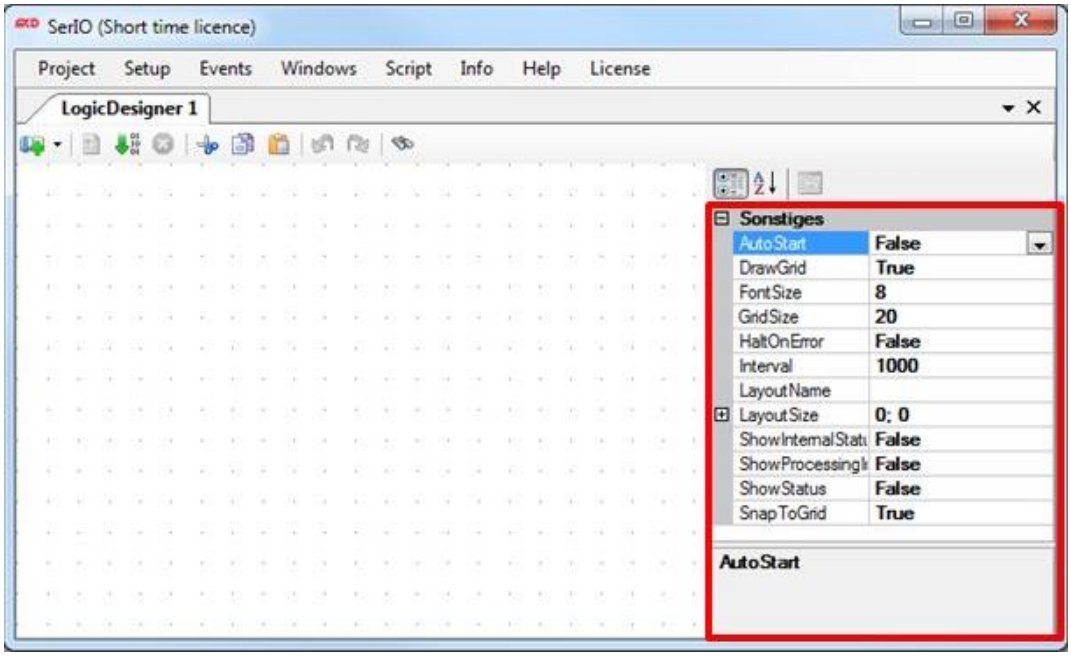


Abbildung 9: Einstellungen für die Benutzeroberfläche

Die nachfolgende Tabelle gibt einen kurzen Überblick über die zur Verfügung stehenden Einstellmöglichkeiten:

Eigenschaft / Einstellung	**Beschreibung**
Autostart	Definition, ob Inhalt des Logic Designers beim Laden des Toolmonitors / eines Presets automatisch gestartet wird
DrawGrid	Auswahl, ob Raster angezeigt werden soll
FontSize	Einstellung der Textgröße
GridSize	Veränderung der Größe des Rasters
HaltOnError	Einstellung, ob Ablauf bei Fehler / Exception anhalten soll
Interval	Einstellung der Zeit in Millisekunden, in der sich Zyklus wiederholen soll
LayoutName	Namensvergabe für Layout (verändert auch den Fenstertitel)
LayoutSize	Einstellung der Größe der Arbeitsfläche (wenn „0“ ausgewählt ist, wird gesamtes Fenster ausgefüllt)
ShowInternalStatus	Entscheidung, ob logischer Zustand der Ein - / Ausgänge oder Zustand auf der Leitung angezeigt wird
ShowProcessingIndex	Anzeige der Abarbeitungsreihenfolge einzelner Elemente
ShowStatus	Ein - und Ausschalten der Statusanzeige einzelner Elemente
SnapToGrid	Einstellung, ob sich Elemente an Raster anpassen sollen

Abbildung 10: Überblick, über die zur Verfügung stehenden Konfigurationsmöglichkeiten

Weitere Funktionen des Logic Designers stehen im Kontextmenü zur Verfügung. Um das Kontextmenü zu öffnen, muss mit der rechten Maustaste in eine freie Fläche des Designers geklickt werden.

Abbildung 11: Kontextmenü des Logic Designers

Die meisten Einträge des Kontextmenüs sind nur erreichbar, wenn sich der Designer im Designmodus befindet.

Die nachfolgende Tabelle gibt eine kurze Übersicht über die Einträge des Kontextmenüs. Einige davon sind auch über die Symbolleiste am oberen Bildrand erreichbar.

Kontextmenü	Symbolleiste	Funktion
Designmode		In den Designmode wechseln (nur im Ausführungsmodus)
Reset		Logikverknüpfungen zurücksetzen
Run		Logikschaltung starten
Stop		Logikschaltung stoppen (nur im Ausführungsmodus)
Stop all		Alle vorhandenen Formulare des Logic Designers stoppen
Single		Gesamten Zyklus nur einmal durchlaufen
Add		Neue Elemente hinzufügen
Delete		Elemente löschen
Cut		Elemente ausschneiden
Copy		Ausgewähltes Element in den Zwischenspeicher kopieren
Paste		Element aus dem Zwischenspeicher einfügen
Undo		Letzte Aktion rückgängig machen
Redo		Letzte Aktion wiederholen
Overview / Simulator		Fenster „Logic designers overview” öffnen
Copy image		Logic Design (oder einen Ausschnitt des Logic Designs) kopieren und als Bild in die Zwischenablage kopieren
Print preview		Dialogfenster „Print preview“ öffnen
Print		Logic Design an den Standarddrucker senden

Abbildung 12: Übersicht über die Einträge des Kontextmenüs

3.3. Logikelemente hinzufügen und bearbeiten

Das Hinzufügen neuer Elemente erfolgt entweder über den Kontextmenüeintrag „Add“, oder über den Button mit dem Buchsymbol am linken Rand der Symbolleiste. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

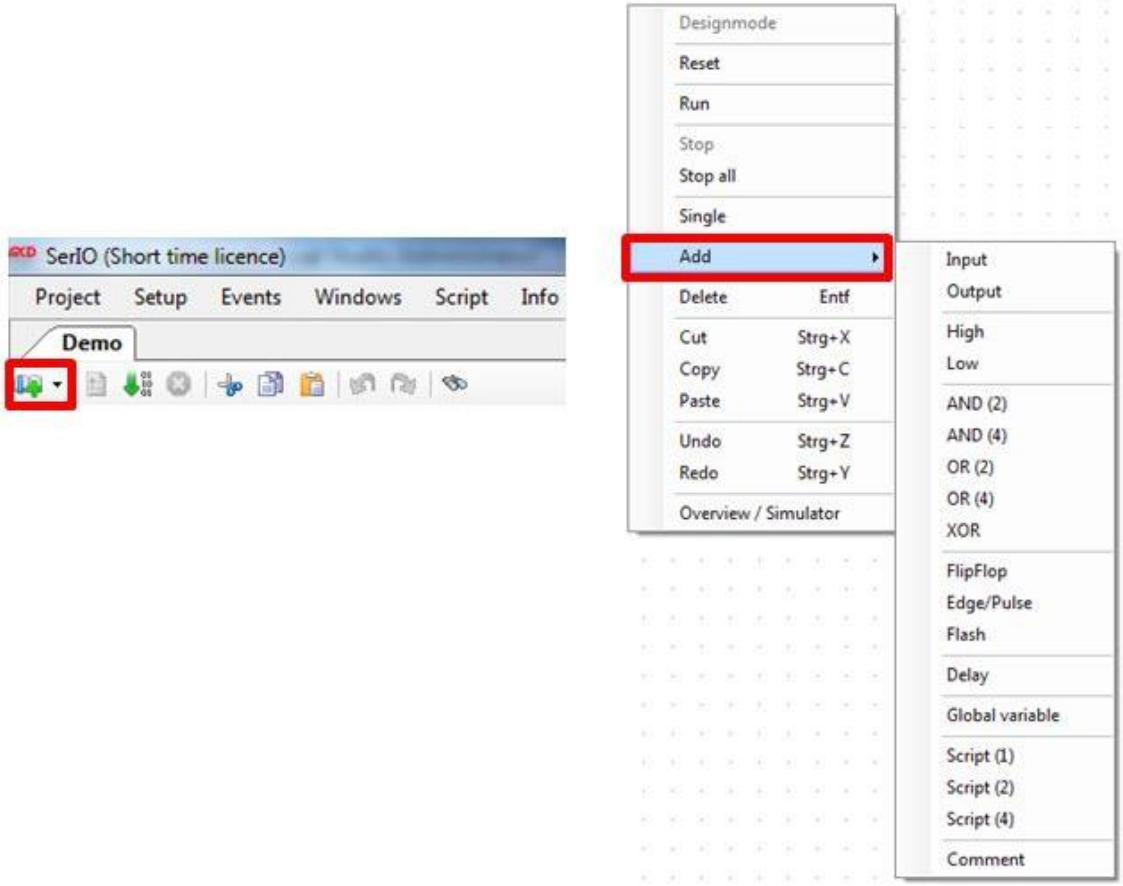


Abbildung 13: Logikelemente einfügen

Das Aussehen und das Verhalten der eingefügten Logikbausteine kann über das Eigenschaftenfenster angepasst werden. Im Eigenschaftenfenster werden die Eigenschaften der aktuell ausgewählten Elemente angezeigt. So kann über die Einstellung „FillColor“ die Farbe des Bausteins verändert werden.

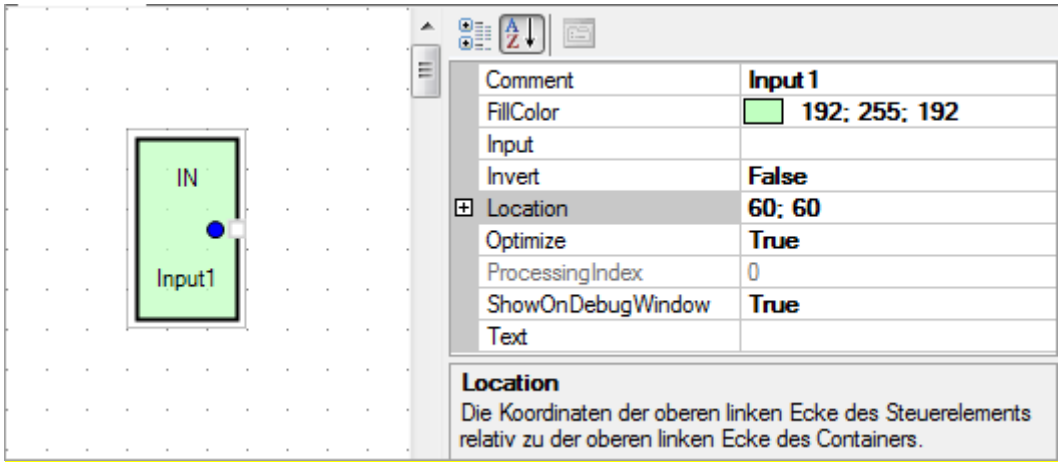


Abbildung 14: Eigenschaften des markierten Logikbausteines

Über die Eigenschaft „Location“ kann die Bausteinposition in X - / und Y - Richtung bestimmt werden. Der Baustein kann auch mit dem Mauszeiger mit gedrückter, linker Maustaste an die gewünschte Position auf der Oberfläche gezogen werden.

3.4. Logikelemente verbinden

Nachdem die Elemente eingefügt wurden, müssen diese noch miteinander verbunden werden. Hierzu klickt man auf den gewünschten, weiß markierten Ein - / Ausgang eines Elements und verbindet ihn durch einen weiteren Linksklick mit dem gewünschten Ein - / Ausgang des anderen Elements. Soll eine Verbindung zwischen zwei Punkten wieder gelöst werden, klickt man erneut auf einen der beiden Ein - / Ausgänge und anschließend auf eine leere, weiße Stelle in der Bedienoberfläche. Zu beachten ist hierbei, dass das Erstellen und Löschen von Verbindungen zwischen den Logikelementen nur im Designmodus möglich ist.

3.5. Löschen vorhandener Logikelemente

Um ganze Logikelemente zu entfernen, muss das entsprechende Element markiert werden und kann dann entweder über die Entfernen - Taste „Entf“, oder über den Kontextmenüeintrag „Delete“ gelöscht werden.

4. Testen / Ausführen / Debuggen der Schaltung im Arbeitsmodus

Nachdem die logische Schaltung im Designmodus erstellt wurde, kann diese nun mit dem Befehl „Run“ ausgeführt werden. Dieser kann über das Kontextmenü des Logic Designs und über die Symbolleiste gestartet werden. Dabei wechselt der Logic Designer automatisch in den Arbeitsmodus.

Der Arbeitsmodus bzw. die Ausführung der Schaltung wird durch den animierten Fortschrittsbalken im unteren Bereich des Logic Designs symbolisiert. Im Arbeitsmodus stehen folgende Funktionen zur Verfügung:

Kontextmenü	Symbolleiste	Funktion
Designmode		Ausführung der Logikschaltung anhalten und in Designmode wechseln
Reset		Logikverknüpfungen zurücksetzen
Stop		Logikschaltung stoppen
Stop all		Alle vorhandenen Formulare des Logic Designers stoppen
Single		Gesamten Zyklus nur einmal durchlaufen
Overview / Simulator		Fenster „Logic designers overview” öffnen

Abbildung 15: Funktionen des Arbeitsmodus

Im Arbeitsmodus sind alle Bearbeitungsfunktionen deaktiviert.

Im Arbeitsmodus gibt die „Logic designers overview“ einen Überblick über die Ein - und Ausgänge der Schaltung. Diese kann, wie oben beschrieben, über den Eintrag des Kontextmenüs „Overview / Simulator“ oder über das Fernglassymbol der Symbolleiste geöffnet werden. Sind die Eingänge oder Ausgänge nicht mit einem Signal verbunden, können diese zu Testzwecken auch simuliert werden. Hierzu müssen die beiden Checkboxen „Simulate inputs“ und / oder „Simulate outputs“ am unteren Rand des „Logic designers overview“ gesetzt werden.

Im unteren Beispiel wurden zwei Eingänge mit einer UND - Verknüpfung auf einen Ausgang geführt. Die Eingänge „Input1“ und „Input2“ werden nun in der „Logic designers overview“ angezeigt. „Input1“ ist „High“ (gelbes Licht), „Input2“ ist „Low“ (blaues Licht) und somit ist auch der Output1 „Low“ (grünes Licht). Die beiden Checkboxen für die Simulation der Ein - und Ausgänge sind aktiviert.

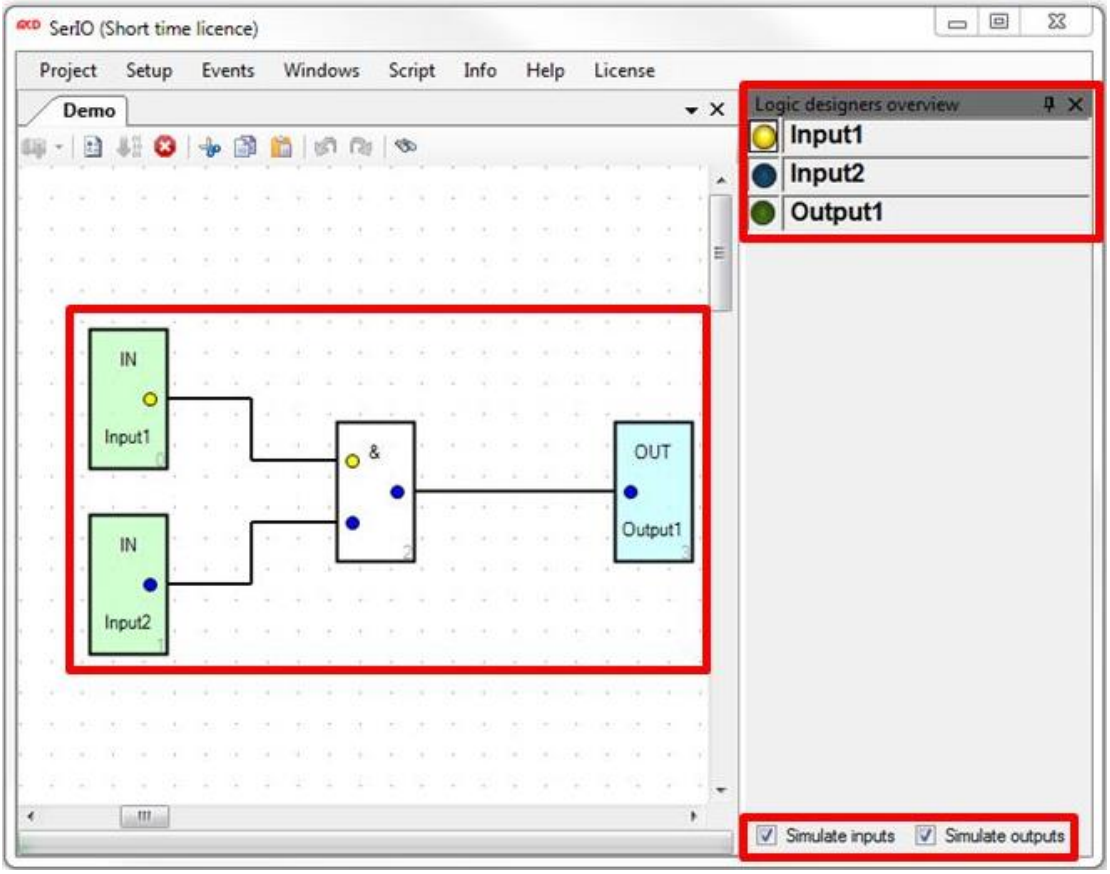
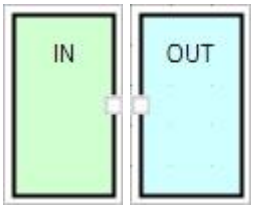
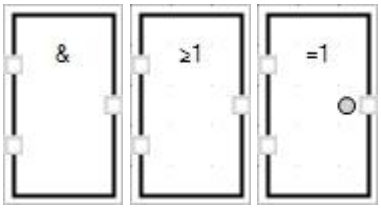
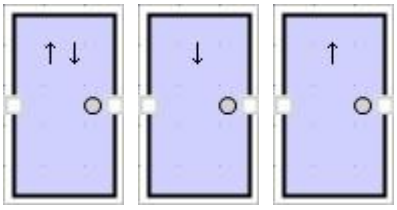
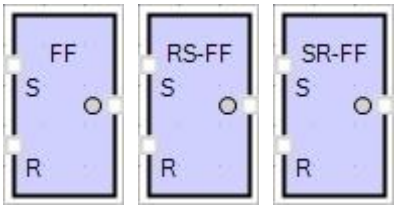
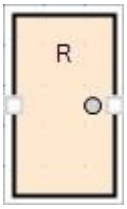


Abbildung 16: Beispiel des Logic Designer Overview

5. Logikelemente im Detail

Im Logic Designer der MCD Toolmonitore steht eine Vielzahl an logischen Elementen zur Verfügung, ein paar Spezial-Controls mit erweiterter Funktionalität und ein paar Controls zur Implementierung der Schnittstelle *Virtual Interface*.

Die nachfolgende Tabelle gibt einen Überblick über die zur Verfügung stehenden Logikbausteine:

Input und **Output** Input und Output, oder Ein - und Ausgang, erlauben die Kommunikation über das *Virtual Interface* und dienen der Anbindung der logischen Schaltung an die zugreifbaren Größen des Toolmonitors. Der Input kann mit einer beliebigen, über „Get“ - Methoden zugreifbaren, Größe des Toolmonitors verknüpft werden. Der Output kann mit einer beliebigen, über „Set“ - Methoden zugreifbaren, Größe verknüpft werden.	
AND, **OR** und **XOR** Diese Gatter implementieren die einfachsten logischen Verknüpfungen von zwei (und mehr) Eingängen und einem Ausgang.	
Edge - / Pulse - Gatter zur Puls - / Flankenerkennung.	
FlipFlop - Gatter mit und ohne Priorisierung von Set - / Reset - Eingängen.	
Global Variable Dieses Control erlaubt die Kommunikation über das *Virtual Interface*.	

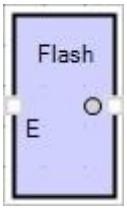
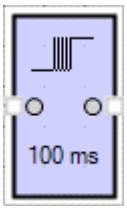
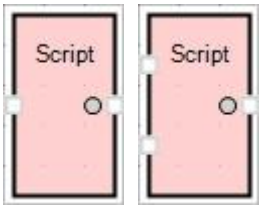
Flash - Baustein Dieses Control kann dazu verwendet werden einen Ausgang in einstellbaren Zeitintervallen ein - und auszuschalten.	
Debouncer - Baustein Dieser Baustein dient der Entprellung von Eingängen. Die Entprellzeit kann beliebig eingestellt werden.	
Delay - Baustein Mit Hilfe des Delay - Bausteins können im Logikschaltplan Ausführungsverzögerungen eingebaut werden.	
High und **Low** Diese Bausteine stellen dauerhaft ein High - / Low - Signal zur Verfügung und können z.B. bei Reset - oder Initialisierungsschaltungen dazu verwendet werden, um einen Eingang auf den definierten Zustand zu setzen.	
Script - Baustein Der Script - Baustein dient der direkten Verknüpfung der Logikschaltung mit einer Methode aus der Script Engine und ermöglicht dem Benutzer so Bausteine mit eigener Funktionalität zu erstellen.	
Comment Das Comment - Control dient nur der Kommentierung des Schaltplans und hat keine sonstige Funktionalität.	

Abbildung 17: Zur Verfügung stehenden Logikbausteine

5.1. Allgemeine Eigenschaften

Logik - Controls können über das Eigenschaftenfenster konfiguriert werden.

Alle Logik - Controls verfügen über einen Satz an allgemeinen Eigenschaften: „Location“, „FillColor“, „Comment“, „Text“ und „ProcessingIndex“.

- Location: Bestimmung der Position des Bausteins in X - / Y - Richtung. Alternativ kann der Baustein mit gedrückter, linker Maustaste auch an die gewünschte Position auf der Oberfläche gezogen werden.
- FillColor: Einstellung der Farbe des Elements
- Comment: Änderung des Textes im oberen Bereich des Bausteins
- Text: Änderung des Textes im mittleren Bereich des Bausteins
- ProcessingIndex: Ein - und Ausschaltung der Darstellung der Abarbeitungsreihenfolge, in der die Knoten nacheinander in jedem Durchlauf abgearbeitet werden.

Ansonsten unterscheiden sich die einstellbaren Eigenschaften je nach Art und Funktion des Logik - Controls. Auf die Control - spezifischen Eigenschaften wird in den folgenden Kapiteln eingegangen.

5.2. Logikelement INPUT

Die wichtigsten Bauelemente einer Logikverknüpfung sind der Eingang und der Ausgang, oder auch Input und Output. Diese Logikelemente müssen in jeder Schaltung vorhanden sein. Um einen Eingang einzufügen, klicken Sie im Kontextmenü des Logic Designs auf Untermenü „Add“, und dann auf den Menüeintrag „Input“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

Wird der neu eingefügte Eingang im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellmöglichkeiten für diesen Baustein.

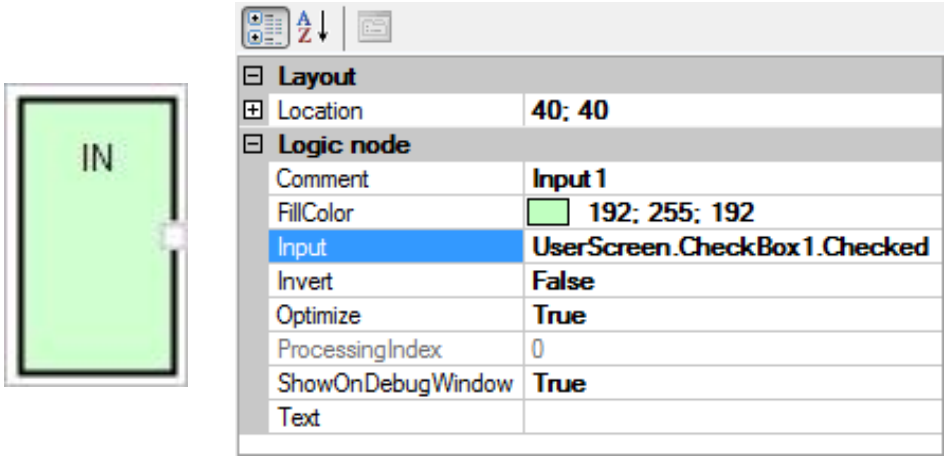


Abbildung 18: Input - Control und seine Einstellungen

Über die Eigenschaft „Input“ wird ausgewählt, welches Attribut der „GetValue“ - Funktion als Eingang verwendet wird.

Im unteren Beispiel wird der Status einer zuvor im „UserScreen“ angelegten „CheckBox1“ mit dem „Input1“ und der Status einer „CheckBox2“ mit „Input2“ verknüpft. Hierzu wird in der Einstellung „Input“ die Eigenschaft der „CheckBox“ über „UserScreen1.CheckBox1.Checked“ (rot markiert), bzw. „UserScreen1.CheckBox2.Checked“ (blau markiert) angegeben. Um zu testen wird nun über den Befehl „Run“ des Kontextmenüs die Logikverknüpfung gestartet. Aktiviert man nun die „CheckBox1“, geht der „Input1“ auf „High“. Aktiviert man die „CheckBox2“, geht der „Input2“ auf „High“. Die angelegten Checkboxen sind gelb markiert. Steht zu Testzwecken keine Quelle für einen

Eingang zur Verfügung, kann diese auch simuliert werden, in dem die Checkbox „Simulate inputs“ im „Logic Designers Overview“ aktiviert wird.

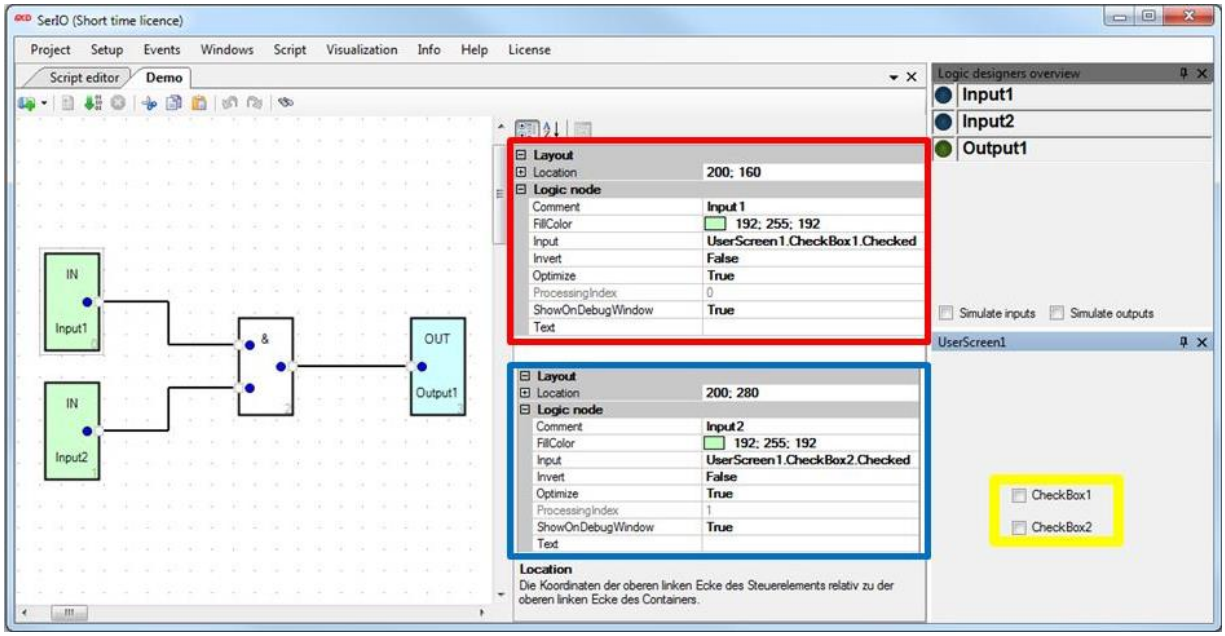


Abbildung 19: Beispiel für Verknüpfung von Eingängen mit Checkboxen

Die Eigenschaft „Invert“ ermöglicht es den Eingang zu invertieren. Wenn die Eigenschaft „Invert“ also auf „True“ gesetzt ist, ist der Eingang „High“, wenn seine Quelle „Low“ ist und umgekehrt.

Über „Optimize“ kann eingestellt werden, dass der Logic Designer nicht bei jedem neuen Knoten, den er abarbeitet, den Eingang abfragt, sondern sich den Status des Eingangs für einen Durchlauf merkt. Bei einer Mehrfachbenutzung von einem Eingang kann dies hinsichtlich der Performance der Logikverknüpfung große Vorteile haben.

Unter „ShowOnDebugWindow“ kann eingestellt werden, ob das Bauelement im „Logic Designers Overview“ angezeigt werden soll, oder nicht.

5.3. Logikelement AND

Ein AND - Gatter (UND - Verknüpfung) dient der logischen Verknüpfung mehrerer Eingänge mit einem Ausgang. Um eine UND - Verknüpfung einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „AND(2)“ für ein UND - Gatter mit zwei Eingängen, bzw. „AND(4)“ für ein UND - Gatter mit vier Eingängen. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

Wird das eingefügte UND - Gatter im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellungs - möglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein UND - Gatter mit vier Eingängen dargestellt. Bei einem UND - Gatter mit zwei Eingängen entfallen die beiden Einstellungen, um den Eingang drei und vier zu invertieren. Ansonsten verhalten sich die Bausteine identisch. Wird bei einem UND - Gatter mit vier Eingängen ein Eingang nicht belegt, so wird dieser beim Schalten des Ausgangs nicht berücksichtigt.

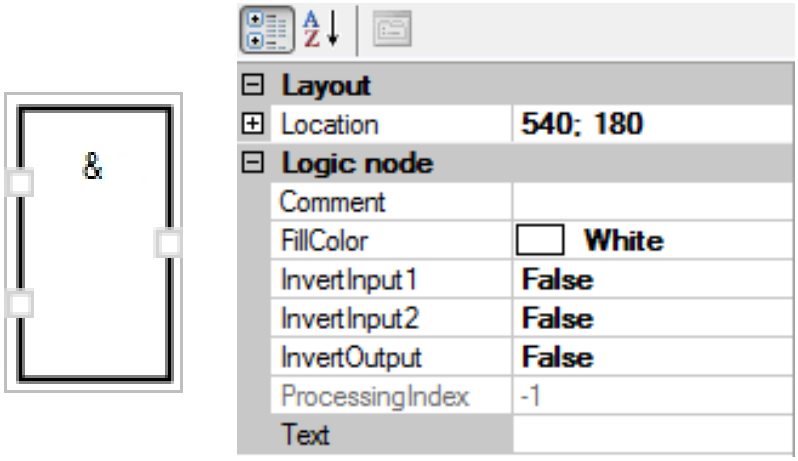


Abbildung 20: AND - Gatter und verfügbare Einstellungen

Über die Eigenschaften „InvertInput1/2/3/4“ lassen sich die jeweiligen Eingänge des Bausteins invertieren. Der oberste Eingang des Bausteins ist hierbei „Input1“ und der unterste entsprechend „Input2“.

Über die Eigenschaft „InvertOutput“ lässt sich der Ausgang des Gatters invertieren.

Im unteren Beispiel ist eine UND - Verknüpfung von zwei Eingängen zu sehen. Die Eingänge wurden „Input1“ und „Input2“ genannt, der Ausgang „Output1“. Entsprechend ihrer Benennung erscheinen die Ein - und Ausgänge dann auch in der „Logic Designers Overview“ in der rechten oberen Ecke.

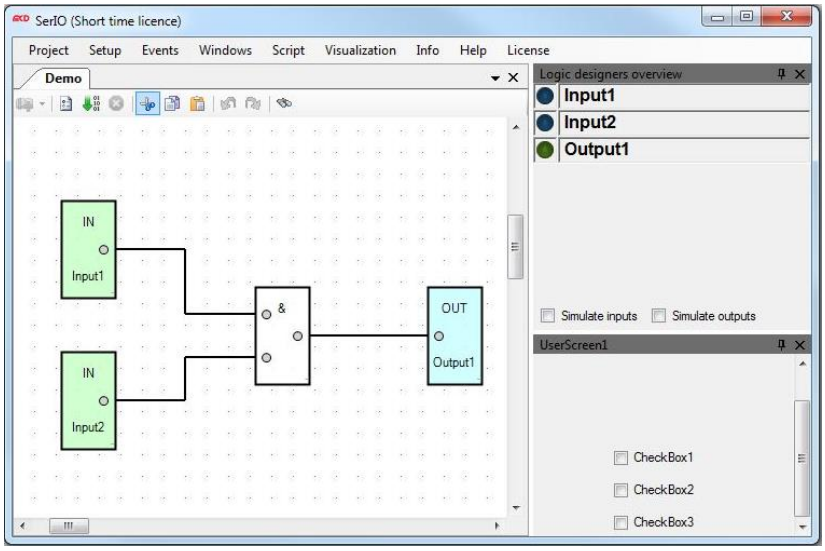


Abbildung 21: Beispiel einer UND - Verknüpfung von zwei Eingängen

5.4. Logikelement Comment

Das Comment - Baustein dient nur der Kommentierung des Schaltplans und hat keine sonstige Funktionalität.

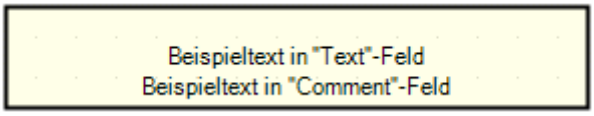


Abbildung 22: Beispiel eines Comment - Bausteins

Über den Comment - Baustein kann ein Kommentarfeld in die Oberfläche eingefügt werden. Um ein solches Kommentarfeld einzufügen, klicken Sie im Menü „Add“ auf den Menüeintrag „Comment“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

Über die Eigenschaft „Size“ kann, über die Breite (Width) und Höhe (Height) die Größe des Bausteins eingestellt werden. Die Angabe ist in Pixeln. Über die Eigenschaft „Comment“ kann der Text im oberen Bereich des Bausteins verändert werden. Über die Eigenschaft „Text“ kann der Text im mittleren Bereich des Bausteins verändert werden.

5.5. Logikelement Delay

Mit Hilfe des Delay - Bausteins können im Logikschaltplan Ausführungsverzögerungen eingebaut werden.

Über den Menüeintrag „Delay“ kann ein Verzögerungsbaustein eingefügt werden. Liegt an dessen Eingang ein „High“ an, so sperrt dieser die gesamte Logikverknüpfung für eine bestimmte, einstellbare Zeit. Um ein solches Delay einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Delay“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur im Designmodus, und nicht im Ausführungsmodus, verfügbar ist.

Wird das eingefügte Delay - Gatter im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellungs - möglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein Delay - Gatter dargestellt.

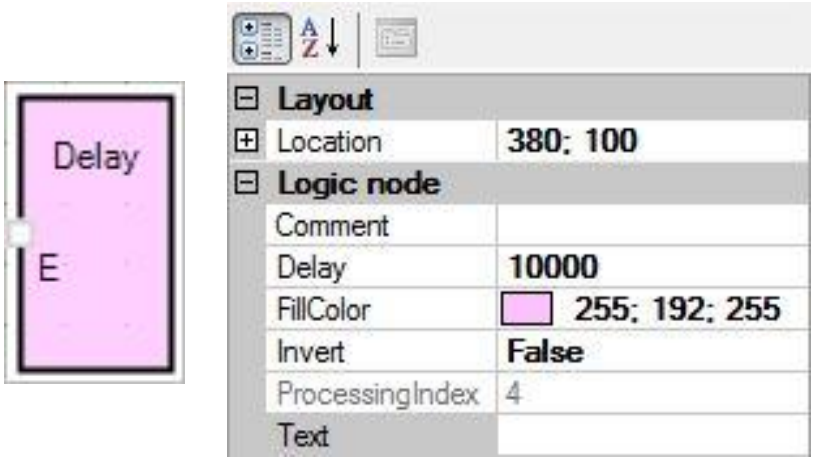


Abbildung 23: Delay - Baustein und Einstellungen des Delay - Gatters

Über die Eigenschaft „Delay“ wird die Zeit in Millisekunden eingestellt, für die die Logikverknüpfung gesperrt wird, wenn an dessen Eingang ein „High“ anliegt.

Ist die Eigenschaft „Invert“ auf „True“ gesetzt, so wird der am Eingang des Delay - Bausteins anliegende Zustand invertiert.

Im unteren Beispiel wurden die beiden Eingänge „Input1“ und „Input2“ mit UND verknüpft und auf einen Ausgang „Output“ geführt. Das angehängte Delay - Gatter sorgt für eine entsprechende Verzögerung.

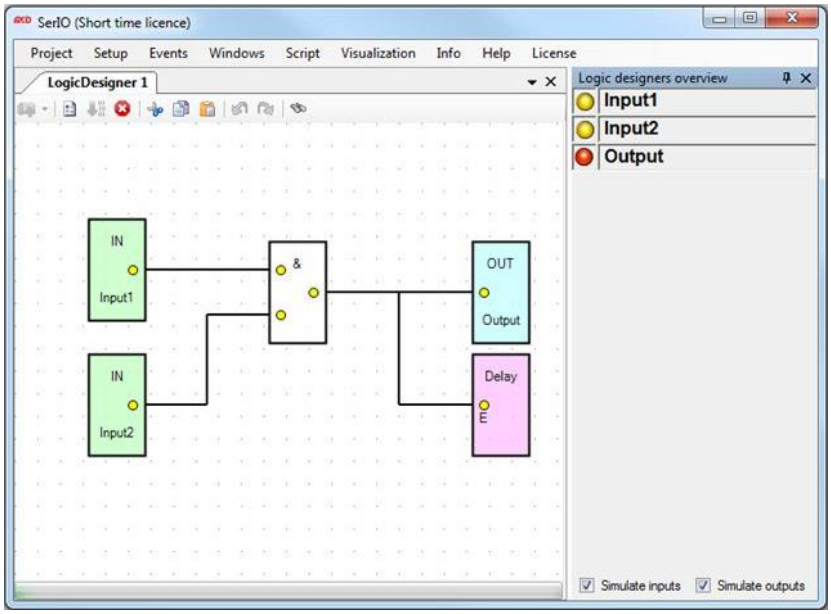


Abbildung 24: Beispiel einer Verknüpfung mit einem Delay - Baustein

5.6. Logikelement EDGE / PULSE

Das Edge - / Pulse - Gatter ist ein Baustein, der auf sich verändernde Flanken reagiert und in Reaktion darauf einen Puls ausgibt.

Um ein Edge - / Pulse - Element einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Edge / Pulse“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

Wird das eingefügte Edge - / Pulse - Gatter im Designmode selektiert, öffnet sich eine Übersicht über die Einstellmöglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein Edge - / Pulse - Gatter dargestellt.

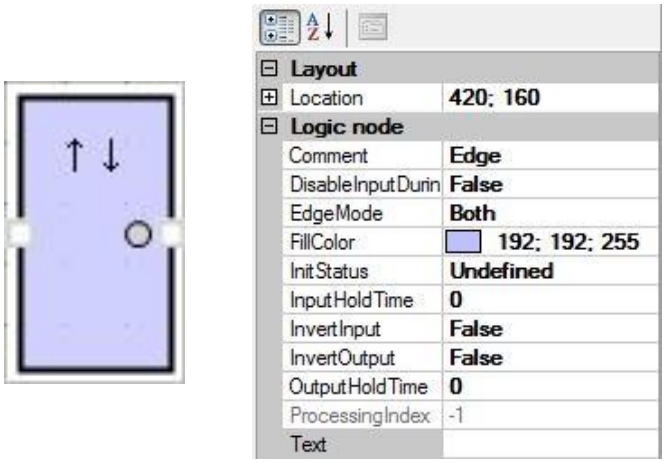


Abbildung 25: Edge - / Pulse - Baustein und Einstellungen des Edge - / Pulse - Gatters

Über die Eigenschaft „DisableInputDuringOutput“ kann der Eingang des Bausteins deaktiviert werden, solange der Ausgang auf „High“ ist.

Über „EdgeMode“ kann festgelegt werden, auf welche Flanken der Baustein reagieren soll. Ist „Raising“ eingestellt, reagiert der Baustein auf die steigende Flanke. Wird also eine steigende Flanke erkannt, wird am Ausgang des Edge - Bausteins ein Puls ausgegeben. Ist „Falling“ eingestellt, reagiert der Baustein auf fallende Flanken. Soll der Baustein sowohl auf fallende, als auch steigende Flanken triggern, muss der „EdgeMode“ auf „Both“ gestellt werden.

EdgeMode = Both EdgeMode = Falling EdgeMode = Raising

Abbildung 26: Edge - / Pulse - Gatter in unterschiedlichen Modi

Der „InitStatus“ legt fest, welchen Zustand der Ausgang des Bausteins beim Initialisieren der Logikverknüpfung hat. Zur Auswahl stehen hier „High“, „Low“ und „Undefined“.

Die „InputHoldTime“ legt fest, wie lange der Eingang des Bausteins „High“ sein muss, um einen Puls am Ausgang zu schalten. Die Zeit wird in Millisekunden angegeben. Steht die Zeit auf „0“, so wird der Puls geschalten, sobald die Flanke erkannt wird. Also spätestens nach einem Durchlauf der Verknüpfung. Über die Eigenschaft „InvertInput“ lässt sich der Eingang des Bausteins invertieren. Über die Eigenschaft „InvertOutput“ lässt sich der Ausgang des Bausteins invertieren. Die Eigenschaft „OutputHoldTime“ legt die Länge des ausgegebenen Pulses in Millisekunden fest. Steht diese Zeit auf „0“, dauert der Puls so lange wie ein Durchlauf der Logikverknüpfung.

Im unteren Beispiel wurden die beiden Eingänge „Input1“ und „Input2“ mit UND verknüpft und über ein Edge - / Pulse - Gatter auf einen Ausgang geführt. Sobald beide Eingänge „High“ sind, detektiert das Edge - / Pulse - Gatter eine steigende Flanke und gibt am Ausgang einen Puls aus. Wird einer der beiden Eingänge wieder „Low“, so liegt auch am Eingang des „Edge - / Pulse - Bausteins ein „Low“ an, der Baustein detektiert die fallende Flanke und gibt erneut einen Puls an seinem Ausgang aus. Die Einstellungen für den Baustein wurden so gewählt, wie oben dargestellt.

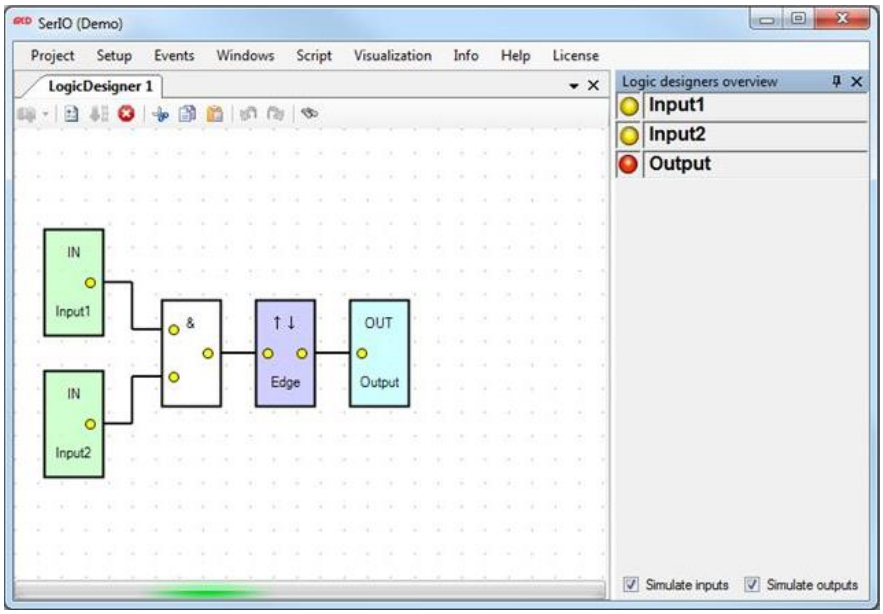


Abbildung 27: Beispiel einer Verknüpfung mit einem Edge- / Pulse-Baustein

5.7. Logikelement Debouncer

Der Debouncer - Baustein dient der Entprellung von Eingängen und kann zum sicheren Schalten von Tastern und Schaltern eingesetzt werden. Die Entprellzeit kann über die Einstellung „DebounceTime“ beliebig, in Millisekunden, eingestellt werden.

Um ein Debouncer - Element einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Debouncer“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

Wird das eingefügte Debouncer - Gatter im Designmode selektiert, öffnet sich eine Übersicht über die Einstellungsmöglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein Debouncer - Gatter dargestellt.

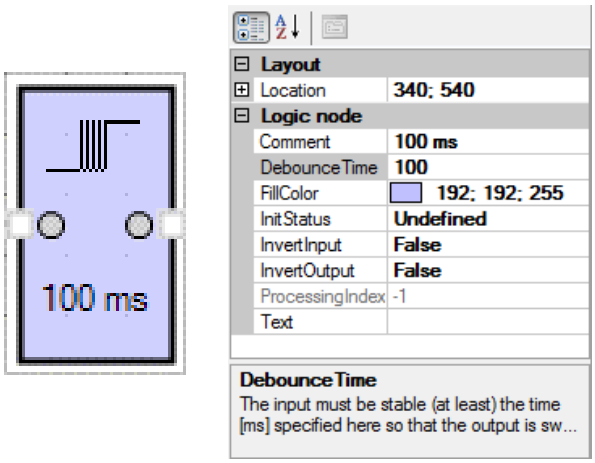


Abbildung 28: Debouncer - Baustein und Einstellungen des Debouncer - Gatters

Die Einstellung „DebounceTime“ legt fest, wie lange das Signal am Eingang des Bausteins unverändert anliegen muss, bevor es auf den Ausgang durchgeschaltet wird. Die Entprellzeit wird in Millisekunden angegeben.

Der „InitStatus“ legt fest, welchen Zustand der Ausgang des Bausteins beim Initialisieren der Logikverknüpfung hat. Zur Auswahl stehen hier „High“, „Low“ und „Undefined“. Über die Eigenschaft „InvertInput“ lässt sich der Eingang des Bausteins invertieren. Über die Eigenschaft „InvertOutput“ lässt sich der Ausgang des Bausteins invertieren.

5.8. Logikelement Flash

Über den Menüeintrag „Flash“ kann ein Baustein eingefügt werden, der, vorausgesetzt es liegt ein „High“ an dessen Eingang an, seinen Ausgang in einstellbaren Zeitintervallen ein - und ausschaltet. Kurz gesagt kann man den Ausgang dieses Bausteins „blinken“ lassen. Um ein Flash - Element einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Flash“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da das Menü „Add“ sonst nicht ausgewählt werden kann.

Wird das eingefügte Flash - Gatter im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellungs - möglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein Flash - Gatter dargestellt.

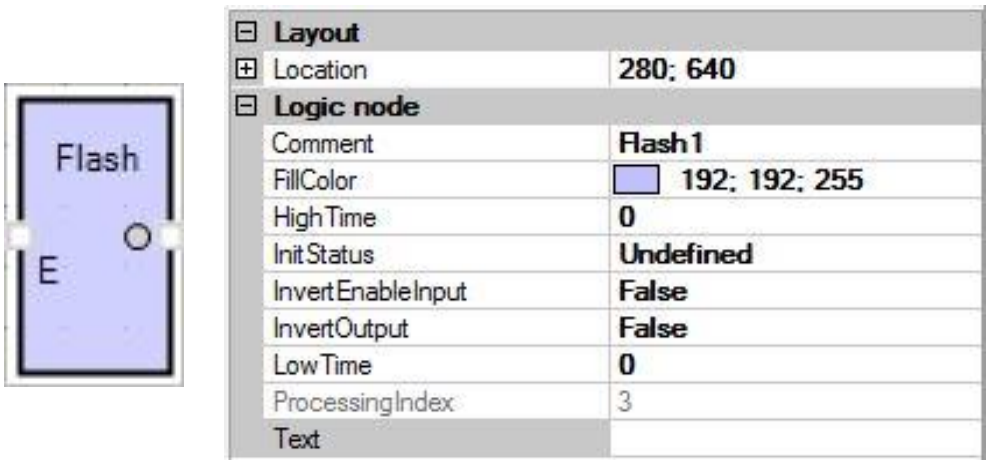


Abbildung 29: Flash - Baustein und Einstellungen des Flash - Gatters

Über die Eigenschaft „HighTime“ wird die Zeit in Millisekunden eingestellt, für wie lange der Flash - Baustein seinen Ausgang auf „High“ schaltet, wenn an seinem Eingang ein „High“ anliegt. Ist hier eine „0“ eingetragen, so hält der Baustein seinen Ausgang genau für einen Durchlauf der Logikverknüpfung auf „Low“, bevor er wieder auf „High“ umschaltet.

Die Eigenschaft „InitStatus“ legt fest, welchen Zustand der Ausgang des Bausteins beim Initialisieren der Logikverknüpfung hat. Zur Auswahl stehen hier „High“, „Low“ und „Undefined“. Über die Eigenschaft „InvertEnableInput“ kann der Eingang des Bausteins invertiert werden. Über die Eigenschaft „InvertOutput“ kann der Ausgang des Bausteins invertiert werden.

Die Eigenschaft „LowTime“ legt die Zeit in Millisekunden fest, für wie lange der Ausgang des Bausteins auf „Low“ liegt, wenn an seinem Eingang ein „High“ anliegt. Ist hier eine „0“ eingetragen, so hält der Baustein seinen Ausgang genau für einen Durchlauf der Logikverknüpfung auf „Low“, bevor er wieder auf „High“ umschaltet.

Im unteren Beispiel wurden die beiden Eingänge „Input1“ und „Input2“ mit UND verknüpft und über ein Flash - Gatter auf einen Ausgang geführt. Sobald „Input1“ und „Input2“ „High“ sind, liegt auch am Eingang des Flash - Bausteins ein „High“ an, der nun abwechselnd seinen Ausgang zwischen „High“ und „Low“ umschaltet.

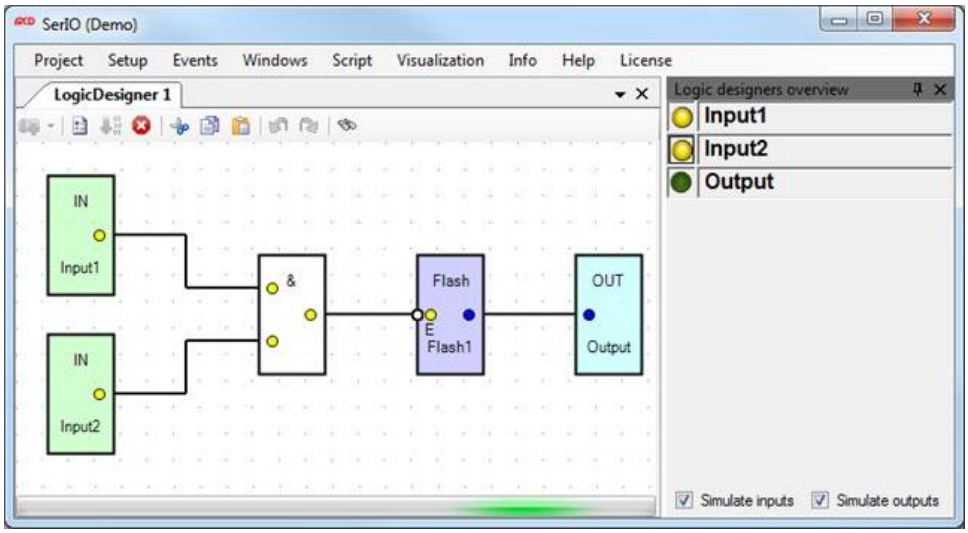


Abbildung 30: Beispiel einer Verknüpfung mit einem Flash - Baustein

5.9. Logikelement FlipFlop

Um einen FlipFlop - Baustein einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „FlipFlop“.

Es werden drei Flipflop - Typen unterstützt:
klassisches Flipflop (ohne Priorisierung der Eingänge), RS - und SR - FlipFlop.

Der FlipFlop - Typ kann in den Einstellungen des Bausteins verändert werden. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

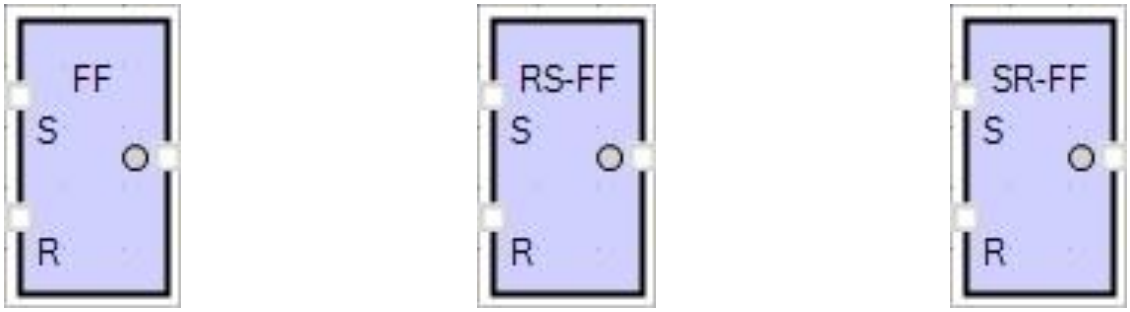
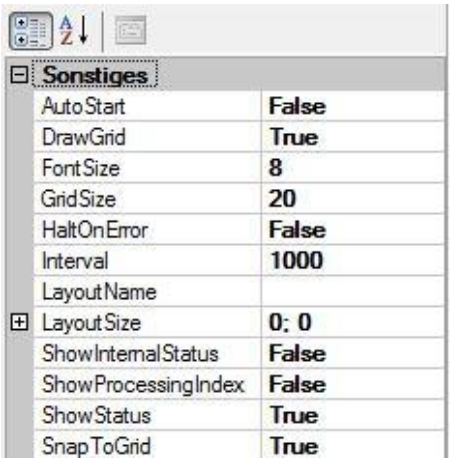


Abbildung 31: FlipFlop - Bausteine ohne Priorisierung, RS - FlipFlop mit priorisiertem Reset - Eingang und SR - FlipFlop mit priorisiertem Set - Eingang

Wird der eingefügte FlipFlop - Baustein im Designmode selektiert, öffnet sich die unten abgebildete Übersicht über die Einstellmöglichkeiten für diesen Baustein.



Sonstiges	
AutoStart	False
DrawGrid	True
FontSize	8
GridSize	20
HaltOnError	False
Interval	1000
LayoutName	
LayoutSize	0; 0
ShowInternalStatus	False
ShowProcessingIndex	False
ShowStatus	True
SnapToGrid	True

Abbildung 32: Einstellungen des FlipFlop - Bausteins

Die Eigenschaft „FlipFlopType“ entscheidet über den Typ des „FlipFlops“, wenn beide Eingänge, also „Set“ und „Reset“, „High“ sind. Ist in dieser Option „FF“ eingestellt, so „gewinnt“ immer der Eingang, an dem zuerst ein „High“ anliegt. Ist „RSFF“ eingestellt, so wird der Ausgang, wenn beide Eingänge (oder nur der „Reset“ - Eingang) „High“ sind resetet. Sind beide Eingänge „High“, „gewinnt“ also der „Reset“ - Eingang. Bei der Einstellung „SRFF“ „gewinnt“, sofern beide Eingänge „High“ sind, der „Set“ - Eingang gegen den „Reset“ - Eingang.

Der „InitStatus“ legt fest, welchen Zustand der Ausgang des Bausteins beim Initialisieren der Logikverknüpfung hat. Zur Auswahl stehen hier „High“, „Low“ und „Undefined“. Steht der Property - Eintrag „InvertOutput“ auf „True“, so wird der Ausgang des „FlipFlops“ invertiert. Steht der Property - Eintrag „InvertReset“ auf „True“, so wird der „Reset“ - Eingang des „FlipFlops“ invertiert. Steht der Property - Eintrag „InvertSet“ auf „True“, so wird der „Set“-Eingang des „FlipFlops“ invertiert.

Im unteren Beispiel wurden die beiden Eingänge „Input1“ und „Input2“ mit UND verknüpft und auf den „Set“-Eingang des „FlipFlops“ geführt. Der „Input3“ wurde direkt auf den „Reset“ - Eingang des „FlipFlops“ gelegt. Der

Ausgang des „FlipFlops“ führt wiederum auf den Ausgang. Da es sich um ein „RS - FlipFlop“ handelt und alle drei Inputs „High“ sind, ist der Ausgang des „FlipFlops“ in diesem Fall „Low“. Der „Reset“ - Eingang „gewinnt“ bei einem „RS - FlipFlop“ gegen den „Set“ - Eingang.

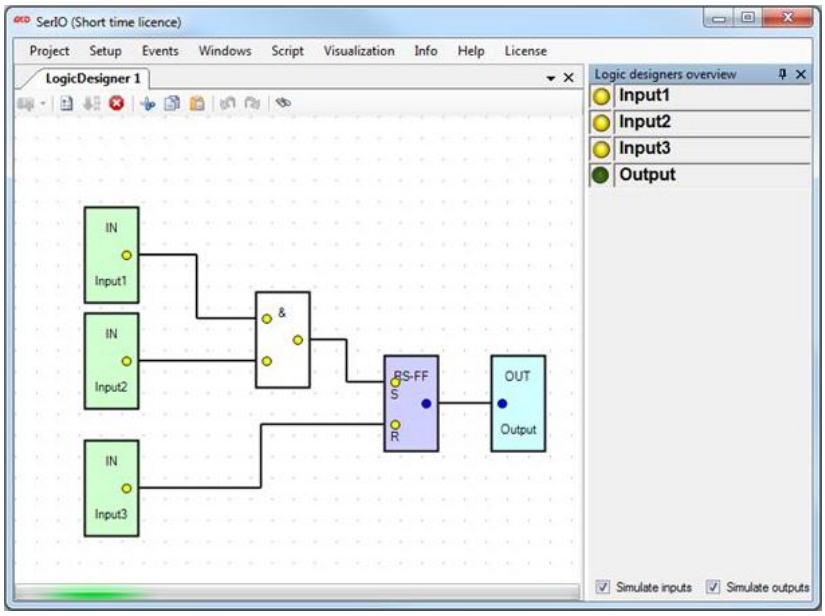


Abbildung 33: Beispiel einer Verknüpfung mit einem FlipFlop - Baustein

5.10. Logikelement Global Variable

Der Baustein „Global Variable“, oder auch Register - Baustein, ermöglicht es eine Logikverknüpfung mit einer globalen Variable zu verbinden. Die Möglichkeiten dieses Bausteins werden in den nachfolgenden Beispielen beschrieben. Um einen Register - Baustein einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Global Variable“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur zur Entwurfszeit verfügbar ist.

Wird das eingefügte Register - Gatter im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellungs - möglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein Register - Gatter dargestellt.

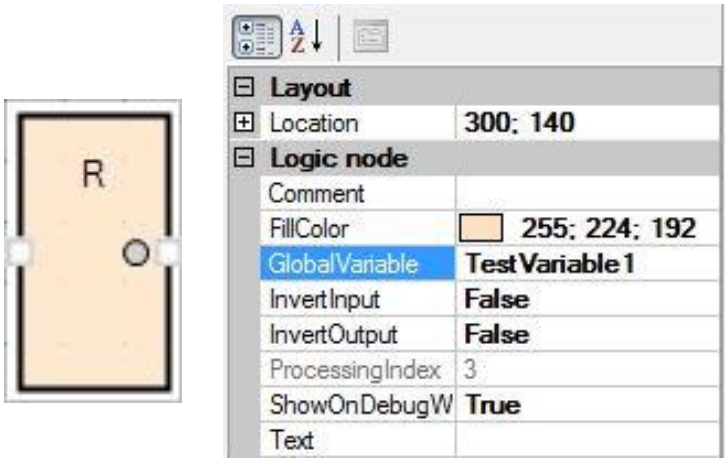


Abbildung 34: Global Variable – Baustein und Einstellungen des Register - Gatters

Über „GlobalVariable“ wird der Name der globalen Variablen angegeben, mit der der Baustein verknüpft sein soll. Für das Beispiel unten wurde der Name „TestVariable1“ vergeben. Über die Eigenschaft „InvertInput“ lässt sich der Eingang des Bausteins invertieren. Über die Eigenschaft „InvertOutput“ lässt sich der Ausgang des Bausteins invertieren. Unter „ShowOnDebugWindow“ wird eingestellt, ob die Variable im Debug - Fenster angezeigt wird, oder nicht.

Um im folgenden Beispiel die Funktion des Register - Bausteins zu erklären, wurde, wie im unteren Bild zu sehen ist, zuvor die Übersicht der globalen Variablen geöffnet. Hier können die aktuellen Werte der angelegten globalen Variablen überwacht werden.

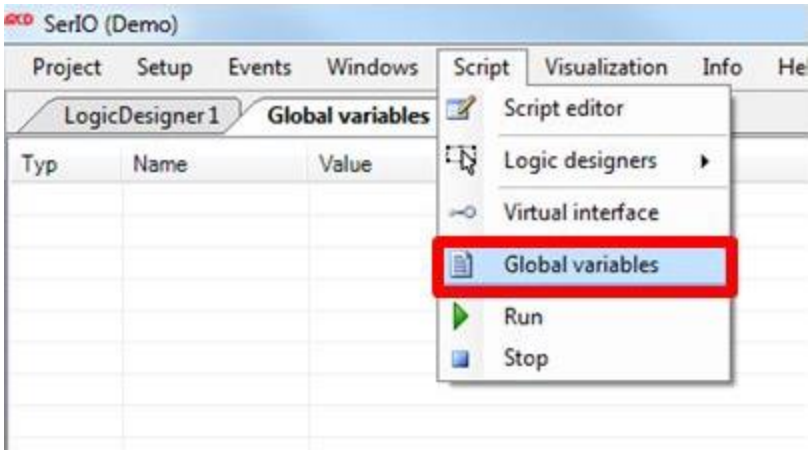


Abbildung 35: Global Variables - Fenster öffnen

Die beiden Eingänge „Input1“ und „Input2“ wurden nach einer UND - Verknüpfung auf einen Ausgang geführt. Der Wert, der in den Ausgang geführt wird, wird ebenfalls auf den Eingang des Register - Bausteins geführt. Dies ist die eine Möglichkeit, wie ein solcher Baustein verwendet werden kann. Der Wert der Variable kann dann beispielsweise im TestManager oder in einem Skript verwendet werden. Auf diese Art und Weise können zum Beispiel sogenannte „Shiftregister“ gebaut werden, also Register, die sich das Ergebnis, oder den Zustand, eines vorherigen Durchlaufs merken. Die folgenden beiden Bilder zeigen, wie sich der Wert der angelegten „TestVariable1“ (siehe oben) verhält, wenn der Eingang des Bausteins „Low“ (Abbildung 36) bzw. „High“ (Abbildung 37) ist.

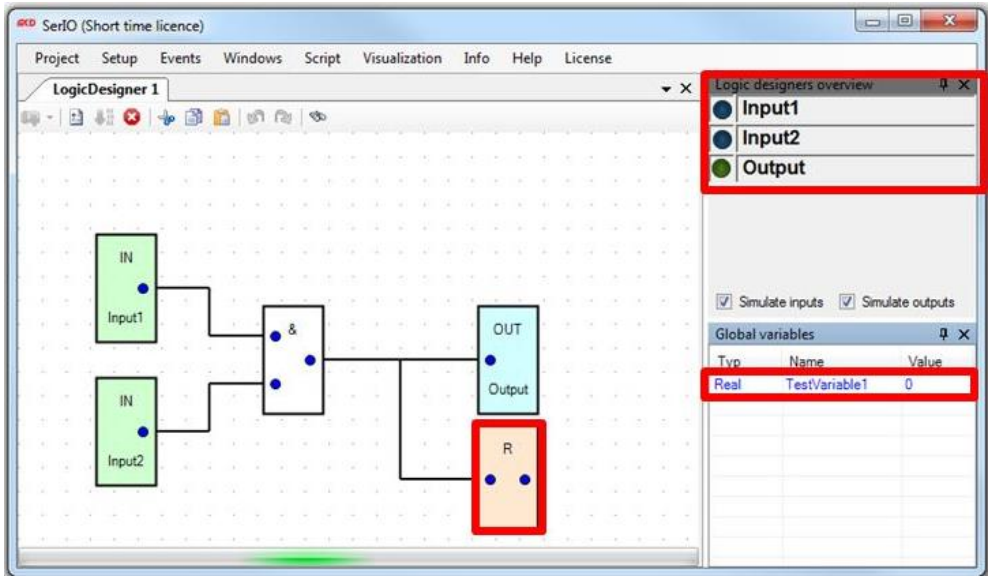


Abbildung 36: Eingang Register - Baustein ist Low, TestVariable1 ist 0

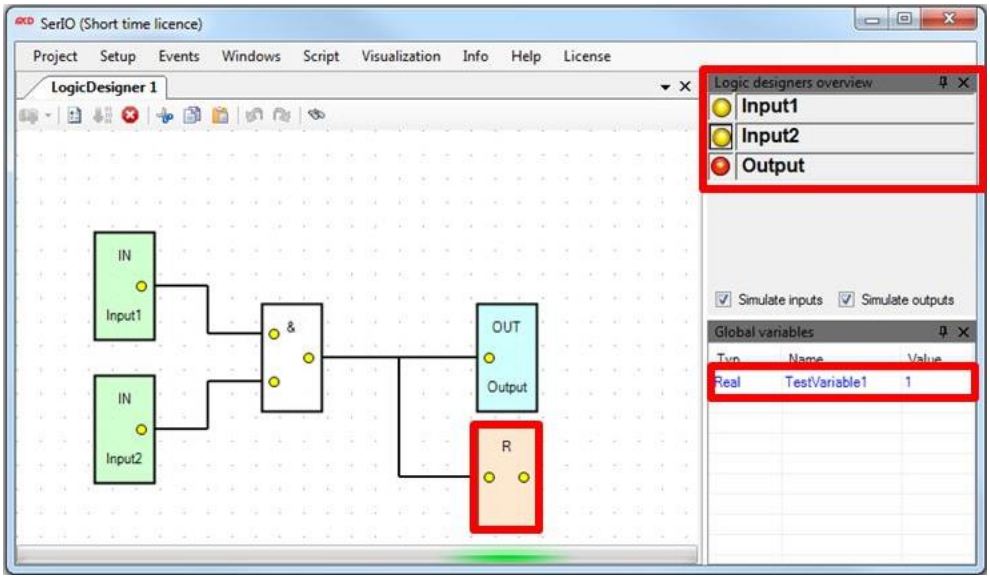


Abbildung 37: Eingang Register - Baustein ist High, TestVariable1 ist 1

Im zweiten Beispiel werden zwei Register - Bausteine eingefügt, die mit derselben Variable „TestVariable1“ verknüpft werden. Schaltet man den „Input1“ auf „High“, geht auch die „TestVariable1“ auf „1“ und somit wird der „Output“ auch „High“. Hier wird die „TestVariable1“ also einerseits über ein bestimmtes Signal verändert, andererseits wird sie benutzt um das Ausgangssignal zu setzen.

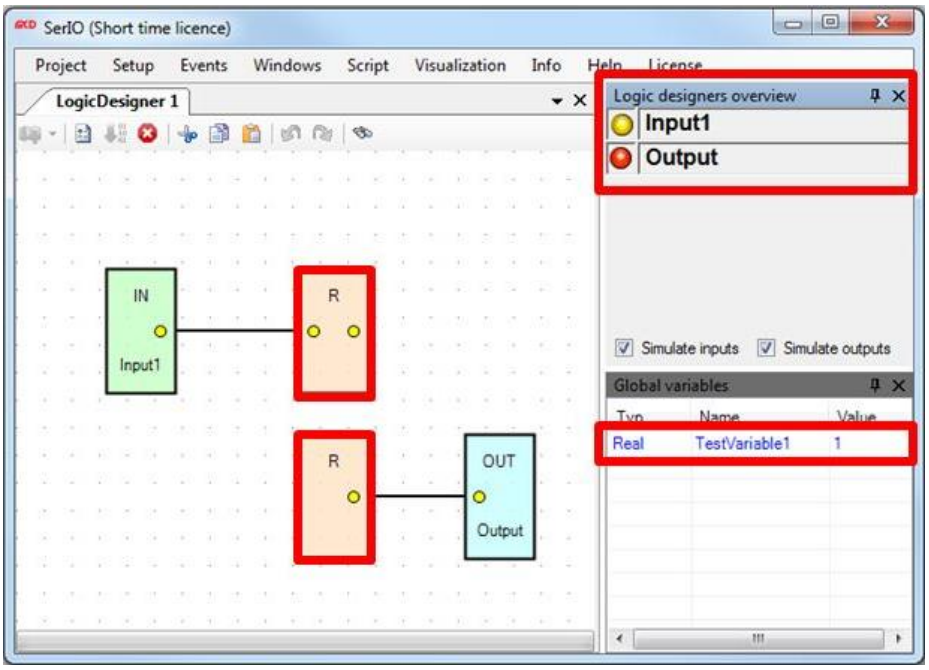


Abbildung 38: Beispiel einer Verknüpfung mit zwei Register - Bausteinen

Ändert man nun, wie hier im dritten Beispiel gezeigt, die Reihenfolge in der die Bausteine abgearbeitet werden, kann man sich, wie oben bereits erwähnt, eine Art „Shiftregister“ bauen. Um diesen Effekt zu verdeutlichen (das Umschalten würde sonst zu schnell gehen) wurde zwischen den unteren Bildern im Kontextmenü nicht der „Run“ - Befehl ausgewählt, sondern der „Single“ - Befehl. Wird jetzt über den „Single“ - Befehl ein einzelner Schritt abgearbeitet, verdeutlicht sich der zuvor beschriebene Effekt. Der Wert von „TestVariable1“ wird von „1“ auf „0“ geändert, während sich der Ausgang allerdings noch auf „High“ befindet. Der Wert eines vorherigen Zyklus kann sich durch dieses Verhalten also „gemerkt“ werden.

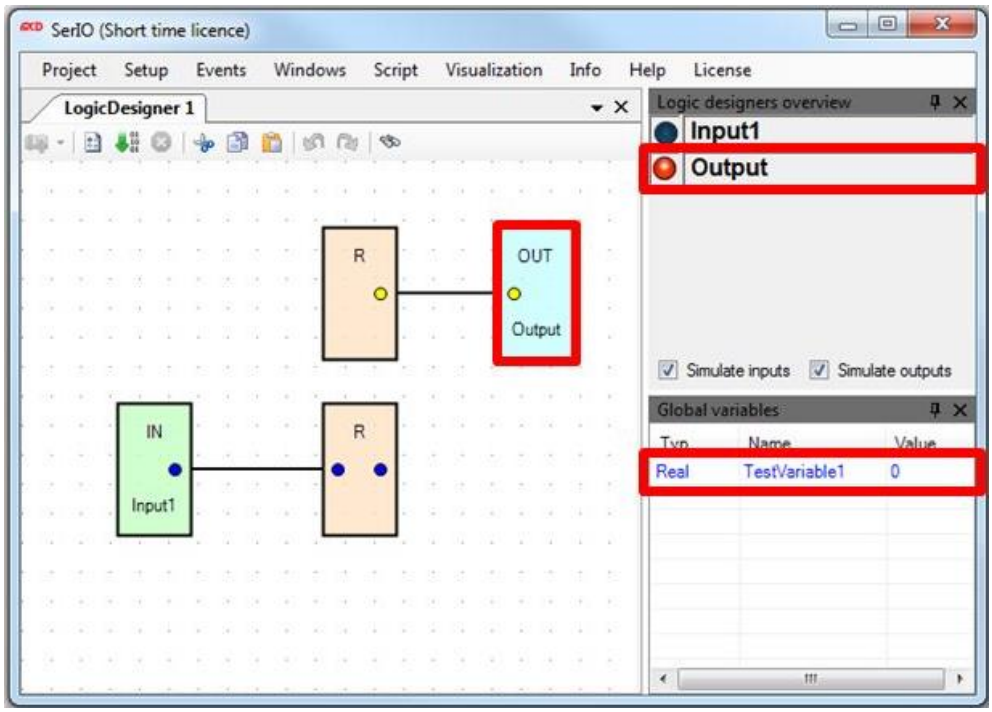


Abbildung 39: Beispiel eines Shiftregisters

5.11. Logikelemente OR / XOR

Ein OR - Gatter (ODER - Verknüpfung) dient der logischen ODER - Verknüpfung mehrerer Eingänge mit einem Ausgang. Ein XOR - Gatter (XOR - Verknüpfung, exklusives ODER) dient der logischen „Entweder - Oder“ - Verknüpfung zweier Eingänge mit einem Ausgang.

Um eine ODER - Verknüpfung einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „OR(2)“ für ein OR - Gatter mit zwei Eingängen, bzw. auf „OR(4)“ für ein OR - Gatter mit vier Eingängen. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur im Designmodus, und nicht im Ausführungsmodus, verfügbar ist.

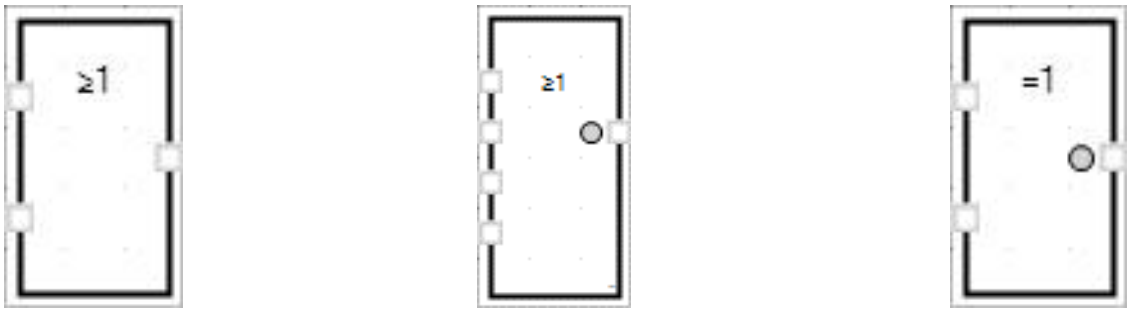


Abbildung 40: OR - Gatter mit zwei Eingängen, OR - Gatter mit vier Eingängen sowie XOR - Gatter

Über den Menüeintrag „XOR“ kann eine XOR - Verknüpfung eingefügt werden. Die Einstellmöglichkeiten sind identisch zur ODER - Verknüpfung mit zwei Eingängen.

Wird das eingefügte ODER - Gatter im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellungsmöglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für ein ODER - Gatter mit vier Eingängen dargestellt. Bei einem ODER - Gatter mit zwei Eingängen entfallen die beiden Einstellungen, um den Eingang drei und vier zu invertieren. Ansonsten verhalten sich die Bausteine identisch. Wird beispielsweise bei einem ODER - Gatter mit vier Eingängen ein Eingang nicht belegt, so wird dieser beim Schalten des Ausgangs nicht berücksichtigt.

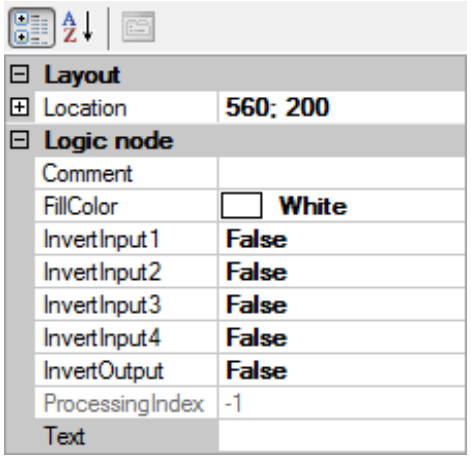


Abbildung 41: Einstellungen des ODER - Gatters

Über die Eigenschaften “InvertInput1/2/3/4“ lassen sich die jeweiligen Eingänge des Bausteins invertieren. Der oberste Eingang des Bausteins ist hierbei „Input1“ und der unterste entsprechend „Input2“. Über die Eigenschaft „InvertOutput“ lässt sich der Ausgang des Gatters invertieren.

Im unteren Beispiel ist eine ODER - Verknüpfung von zwei Eingängen zu sehen. Die Eingänge wurden „Input1“ und „Input2“ genannt, der Ausgang „Output1“. Entsprechend ihrer Benennung erscheinen die Ein - und Ausgänge dann auch in der „Logic Designers Overview“ in der rechten oberen Ecke.

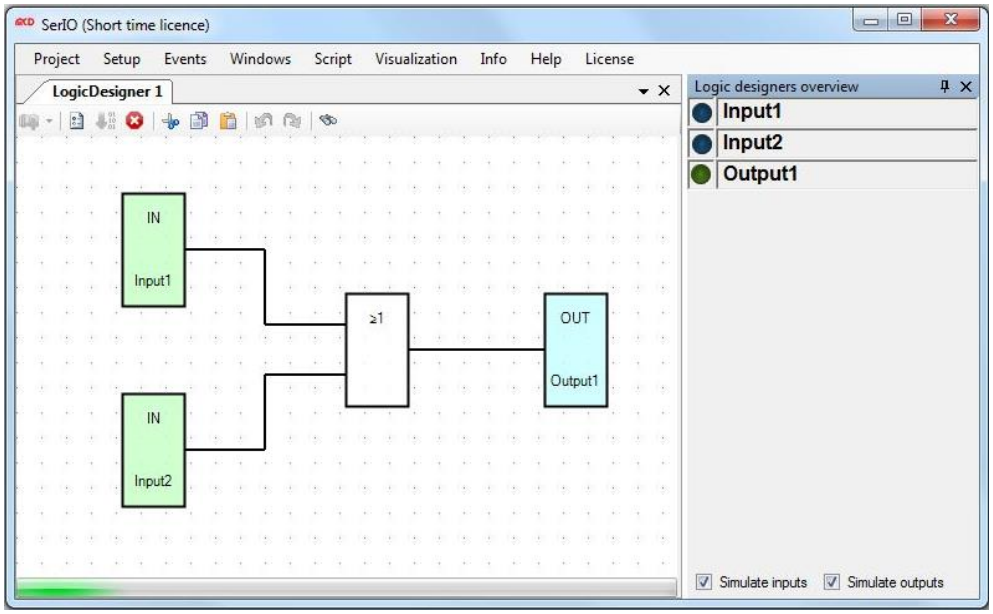


Abbildung 42: Beispiel einer ODER - Verknüpfung von zwei Eingängen

5.12. Logikelement OUTPUT

Die wichtigsten Bauelemente einer Logikverknüpfung sind der Eingang und Ausgang, oder auch „Input“ und „Output“. Diese Logikelemente müssen in jeder Schaltung vorhanden sein. Um einen Ausgang einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Output“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur im Designmodus verfügbar ist.

Wird der neu eingefügte Ausgang im Designmodus selektiert, öffnet sich eine Übersicht über die Einstell-möglichkeiten für diesen Baustein.

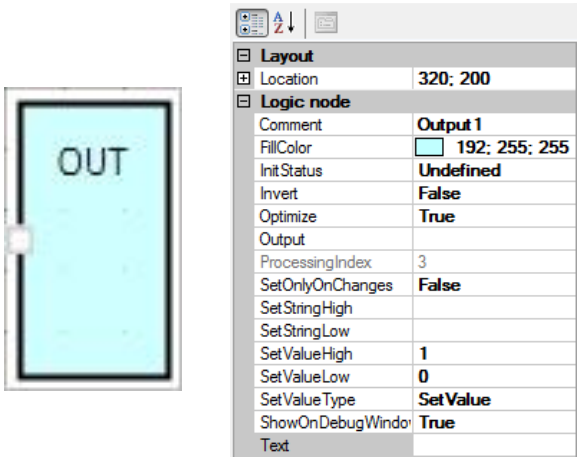


Abbildung 43: Output - Baustein und dessen Einstellungen

Über „InitStatus“ kann der Zustand des Ausgangs beim Starten der Logikverknüpfung festgelegt werden. Zur Auswahl stehen „High“, „Low“ und „Undefined“.

Die Eigenschaft „Invert“ ermöglicht es den Eingang zu invertieren. Wenn die Eigenschaft „Invert“ also auf „True“ gesetzt ist, ist der Eingang „High“, wenn seine Quelle „Low“ ist und umgekehrt.

Über die Eigenschaft „Optimize“ kann eingestellt werden, dass der Logic Designer nicht bei jedem neuen Knoten, den er abarbeitet, den Ausgang abfragt, sondern sich den Status des Ausgangs für einen Durchlauf merkt. Ist „Optimize“ auf „False“ gesetzt, wird der Ausgang sofort gesetzt, wenn er im Ablauf gestellt wird. Wird hier ein Ausgang mehrfach benutzt, wird er auch mehrfach während eines Durchlaufs gestellt. Ist „Optimize“ auf „True“ gesetzt, wird der Ausgang erst am Ende eines Durchlaufs gesetzt. Ändert sich der Zustand des Ausgangs theoretisch während eines Durchlaufs mehrfach, „gewinnt“ der letzte Zustand. Bei einer Mehrfachbenutzung von einem Ausgang kann dies, hinsichtlich der Performance der Logikverknüpfung, große Vorteile haben.

Über die Eigenschaft „Output“ wird angegeben, mit was der Ausgang über das *Virtuelle Interface* tatsächlich verknüpft wird. Allgemein führt ein solcher Ausgang einen „SetValue“ - Befehl aus, dementsprechend kann der Ausgang mit allen Attributen des „SetValue“ - Befehls verknüpft werden. Im unteren Beispiel wurde der Ausgang, analog zu den Eingängen, mit einer zuvor angelegten „CheckBox3“ (gelb markiert) verbunden. Für die eigentliche Verbindung wird dann in der Einstellung „Output“ die „Checked“ - Eigenschaft der „CheckBox3“ mit „UserScreen1.CheckBox3.Checked“ angegeben (siehe Abbildung).

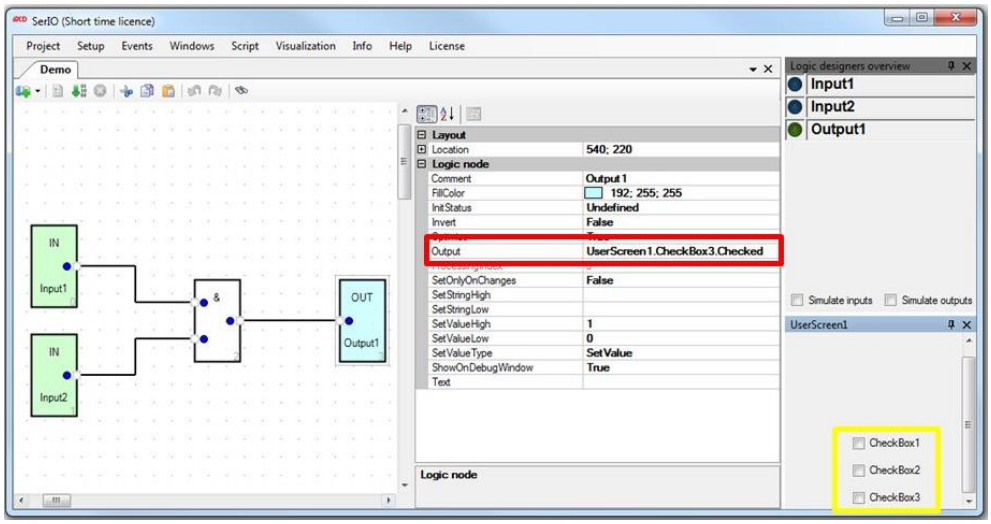


Abbildung 44: Beispiel einer Verknüpfung von Ausgang mit CheckBox

Die Eigenschaft „SetOnlyOnChanges“ dient ebenfalls der Performanceoptimierung. Ist diese auf „True“ gesetzt, wird der Ausgang nur neu gestellt, wenn sich diese im Vergleich zum letzten Durchgang auch verändert hat. Zu beachten ist hier, dass eine Zustandsänderung nicht detektiert wird, wenn sich der Ausgang durch einen anderen Grund, wie einen Befehl aus dem Logic Designer ändert. Ist „SetOnlyOnChanges“ auf „False“ gesetzt, wird der Ausgang bei jedem Durchlauf neu gesetzt, ob er sich geändert hat oder nicht.

Über die Einstellungen „SetValueHigh“ bzw. „SetValueLow“ kann festgelegt werden, welchen Wert der Ausgang zurückgeben soll, wenn er „High“, bzw. wenn er „Low“ ist. Im unteren Beispiel wurde zur Verdeutlichung das mit UND verknüpfte Signal von zwei Eingängen auf zwei Ausgänge („Output1“ und „Output2“) gelegt. Die beiden Ausgänge wurden mit zwei, zuvor im Designer angelegten, „NumericUpDown“ - Controls verknüpft, die deren Rückgabewert anzeigen. Im oberen Bild sind die Einstellungen zu den beiden Ausgängen zu sehen. Als Einstellung der Eigenschaft „Output“ wurde die „Value“ - Eigenschaft des jeweiligen „NumericUpDown“ - Controls angegeben. „Output1“ gibt „0“ zurück für „Low“ und „1“ für „High“. „Output2“ gibt „10“ zurück für „Low“ und „20“ für „High“. Hier kann mit beliebigen „Integer“ - Werten gearbeitet werden. Die unteren beiden Bilder zeigen die Reaktion der Controls, wenn die Ausgänge „Low“ sind (links) und wenn die Ausgänge „High“ sind (rechts). „Input1“ ist, wie die Beispiele zuvor, mit „CheckBox1“ verknüpft und „Input2“ mit „Checkbox2“.

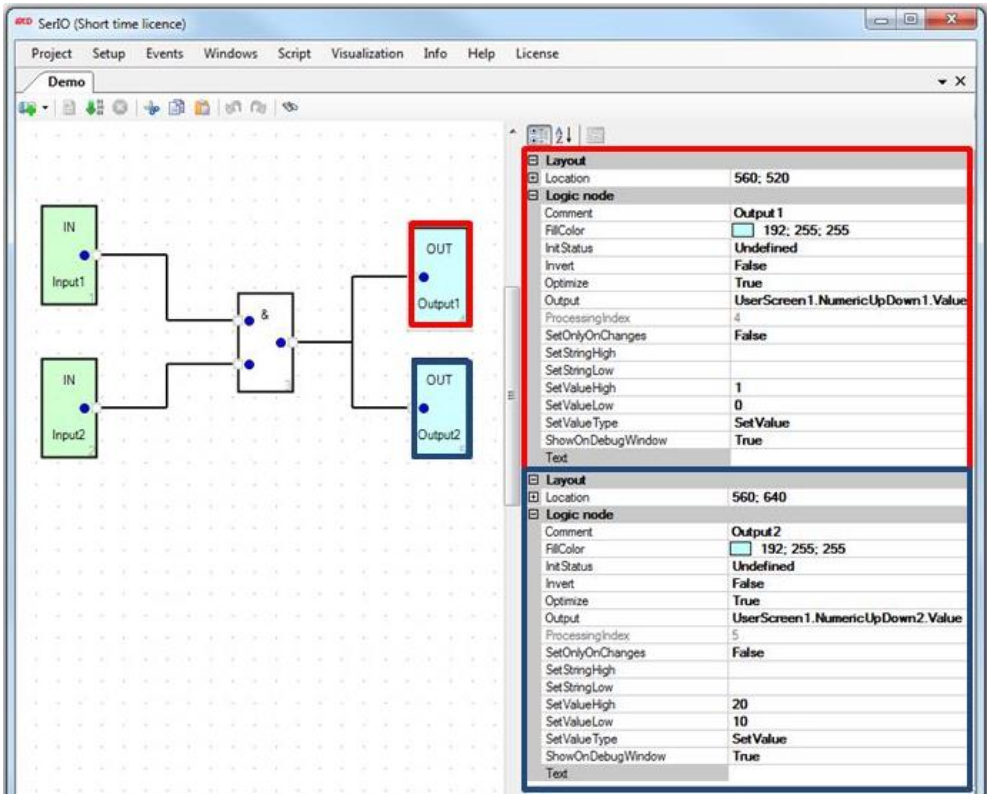


Abbildung 45: Einstellungen für Output1 (rot) und Output2 (blau)

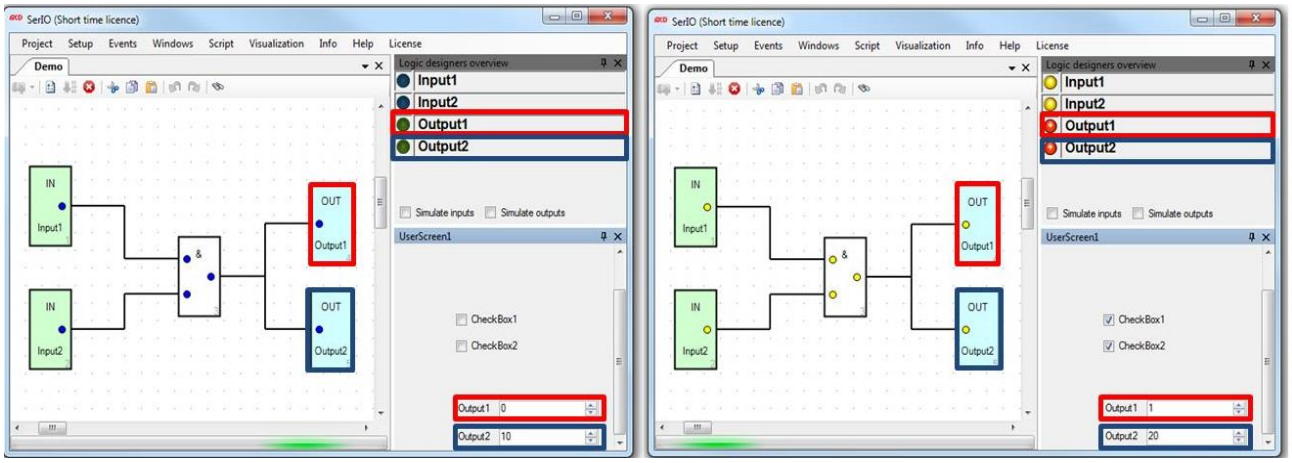


Abbildung 46: Ausgabe der Rückgabewerte von Output1 und Output2 auf NumericUpDown - Controls

Soll von einem Ausgang kein Zahlenwert, sondern ein „String“ zurückgegeben werden, so kann dieser in den Einstellungen „SetStringHigh“ und „SetStringLow“ festgelegt werden. Über die Eigenschaft „SetValueType“ kann dann ausgewählt werden, welcher Datentyp zurückgegeben werden soll. Im unteren Beispiel gibt ein Ausgang („Output1“) seinen Rückgabewert („String“) in einer „TextBox“ aus. Ist der Zustand des Outputs „High“, so zeigt auch die „TextBox“ das Wort „High“ (siehe Abbildung: hier im Low - Zustand).

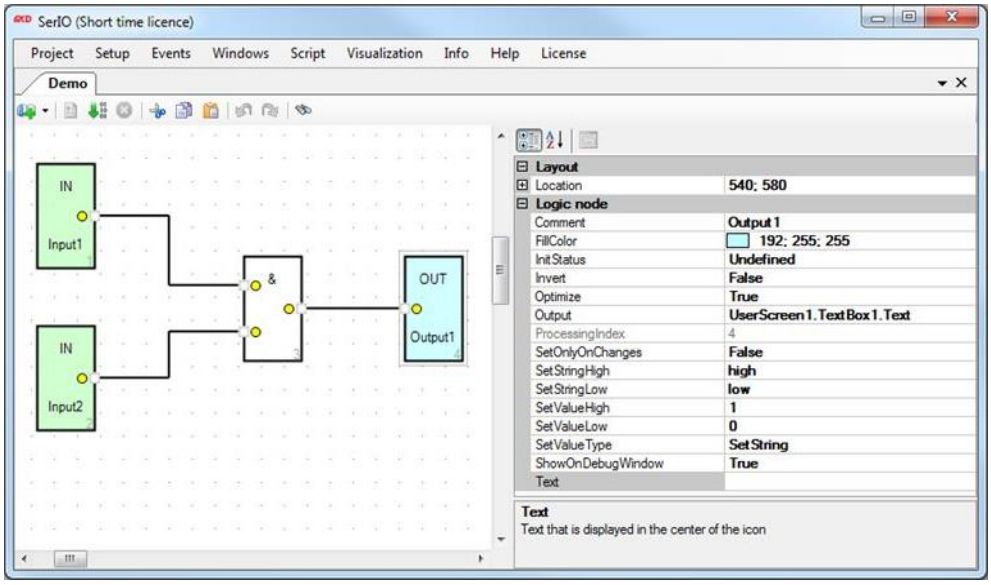


Abbildung 47: Einstellungen des Output1 für String - Rückgabewert und Ausgabe in TextBox1 (Low - Zustand)

Im oberen Bild sind die Einstellungen des Ausgangs zu sehen. Als „SetValueType“ wurde „Set String“ angegeben und als „Output“ die Text - Eigenschaft der „TextBox1“. Ist der Zustand des „Outputs“ „Low“, so zeigt auch die „TextBox“ das Wort „Low“ (siehe Abbildung: hier im Low - Zustand).

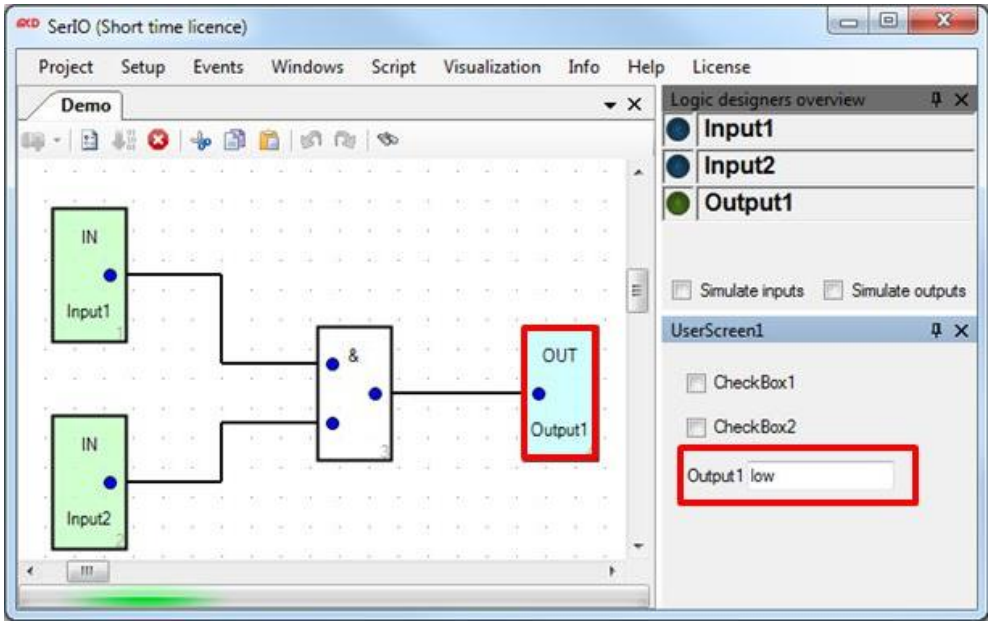


Abbildung 48: Ausgabe des String - Rückgabewertes von Output1 in TextBox1 (Low - Zustand)

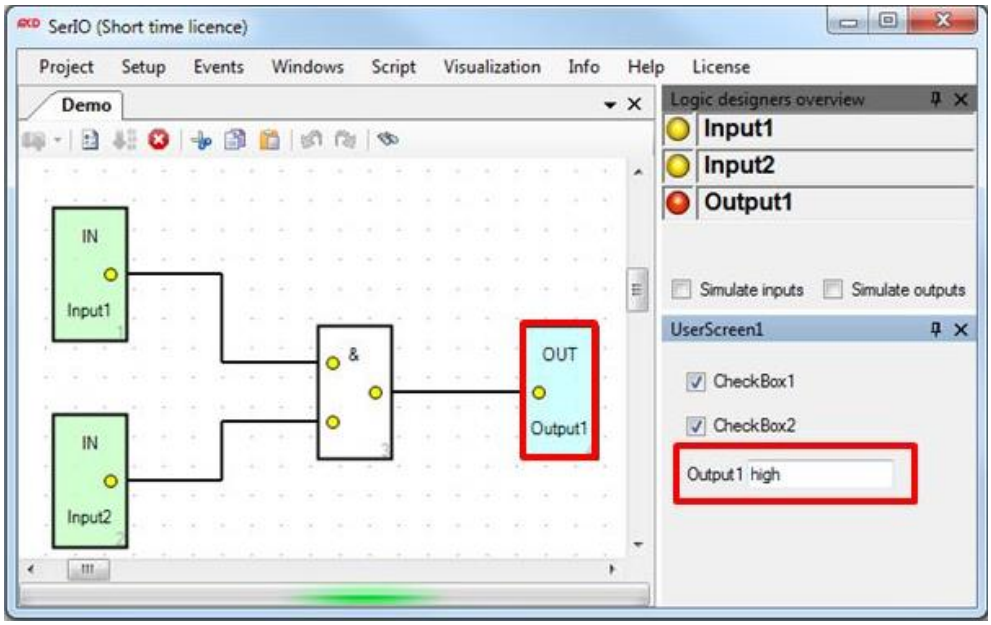


Abbildung 49: Ausgabe des String - Rückgabewertes von Output1 in TextBox1 (hier High - Zustand)

Unter „ShowOnDebugWindow“ kann eingestellt werden, ob das Bauelement in der „Logic Designers Overview“ angezeigt werden soll, oder nicht.

5.13. Logikelement HIGH / LOW

Die Bausteine „High“ und „Low“ stellen dauerhaft ein „High“ - / „Low“ - Signal zur Verfügung und können z.B. bei Reset - oder Initialisierungsschaltungen dazu verwendet werden, einen Eingang auf einen definierten Pegel zu setzen.

Abbildung 50: Logikelemente High und Low

5.14. Logikelement Script

Um zwischen verschiedenen Signalen Funktionalität herzustellen, die mit herkömmlichen Bausteinen zu kompliziert oder zu umständlich ist, steht ein sogenannter Script - Knoten zur Verfügung. Dieser Knoten steht mit einem, zwei oder vier Eingängen zur Verfügung. Der Script - Knoten ist eine Verknüpfung vom Logic Designer zu einer beliebig einstellbaren Script - Methode. In den folgenden Beispielen wird die benutzte Script - Methode mit „MyNode“ benannt.

Abbildung 51: Script - Bausteine mit einem und zwei Eingängen

Um einen solchen Script - Knoten einzufügen, klicken Sie im Kontextmenü des Logic Designs auf das Untermenü „Add“, und dann auf den Menüeintrag „Script“. Wichtig ist hierbei, dass sich der Logic Designer im Designmodus befindet, da die Bearbeitungsfunktion „Add“ nur im Designmodus verfügbar ist.

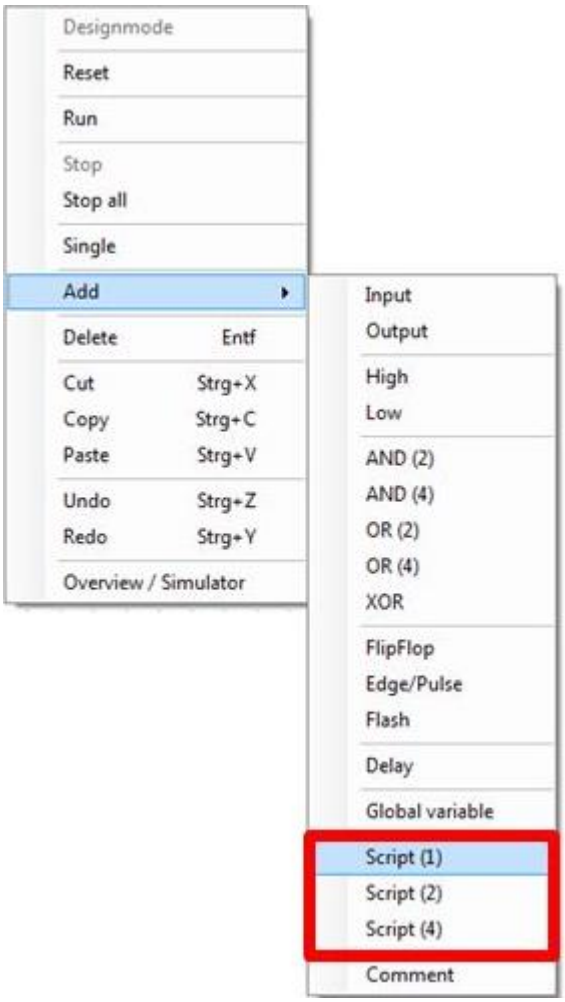
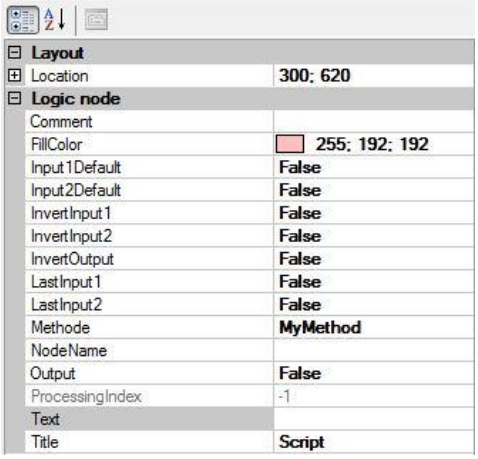


Abbildung 52: Script - Knoten einfügen

Wird der eingefügte Script - Knoten im Designmodus selektiert, öffnet sich eine Übersicht über die Einstellungsmöglichkeiten für diesen Baustein. Im unteren Bild sind die Einstellungsmöglichkeiten für einen Script - Knoten mit zwei Eingängen dargestellt.



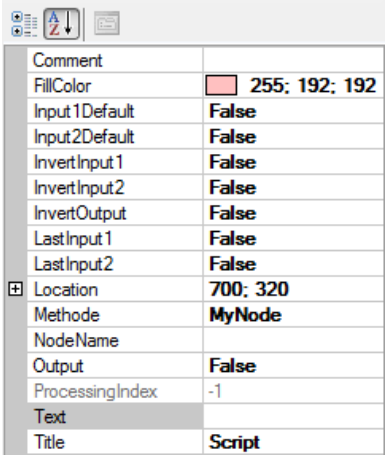
Layout	
Location	300; 620
Logic node	
Comment	
FillColor	255; 192; 192
Input1Default	False
Input2Default	False
InvertInput1	False
InvertInput2	False
InvertOutput	False
LastInput1	False
LastInput2	False
Methode	MyMethod
NodeName	
Output	False
ProcessingIndex	-1
Text	
Title	Script

Abbildung 53: Einstellungen des Script - Knotens

Über die Eigenschaft „Input1Default“ kann für den „Input1“ ein Default - Wert eingestellt werden, wenn dieser nicht verbunden ist. Über die Eigenschaft „Input2Default“ kann für den „Input2“ ein Default - Wert eingestellt werden, wenn dieser nicht verbunden ist. Über die Eigenschaft „InvertInput1“ lässt sich der „Input1“ des Gatters invertieren. Über die Eigenschaft „InvertInput2“ lässt sich der „Input2“ des Gatters invertieren. Über „InvertOutput“ kann der Ausgang des Bausteins invertiert werden. Über „LastInput1“ kann der Status von „Input1“ beim letzten Durchlauf eingestellt werden. Über „LastInput2“ kann der Status von „Input2“ beim letzten Durchlauf eingestellt werden.

Unter „Methode“ wird die Script - Methode eingestellt, mit der der Script - Knoten verknüpft sein soll. Unter „NodeName“ kann dem Script - Knoten ein Name vergeben werden. Unter Output kann der aktuelle Ausgangszustand eingestellt werden. Über die Eigenschaft „Titel“ kann dem Script - Knoten ein Titel vergeben werden, der dann im oberen Teil des Bausteins angezeigt wird.

Wie oben bereits erwähnt, folgt nun ein kurzes Beispiel, um die Funktion des Script - Knotens besser darzustellen. Der Script - Knoten wurde, wie oben, über die Einstellung „Methode“ mit der Beispielmethode „MyNode“ verknüpft.



Comment	
FillColor	255; 192; 192
Input1Default	False
Input2Default	False
InvertInput1	False
InvertInput2	False
InvertOutput	False
LastInput1	False
LastInput2	False
Location	700; 320
Methode	MyNode
NodeName	
Output	False
ProcessingIndex	-1
Text	
Title	Script

Abbildung 54: Verknüpfung des Script - Knotens mit MyNode - Methode

Da die Methode „MyNode“ aktuell aber noch nicht existiert, würde der „Compiler“ beim Klick auf den Befehl „Run“ eine Fehlermeldung anzeigen, da er die Funktion nicht finden kann. Sobald der Methodenname vergeben wurde, wird diese Methode aber durch einen Doppelklick auf den rosafarbenen Script - Knoten automatisch im Script Editor erstellt (im Bild unten rot markiert). Als Standardeinstellung wird hier zu Beginn eine logische UND - Verknüpfung zwischen den beiden Einstellungen eingefügt (im Bild unten blau markiert).

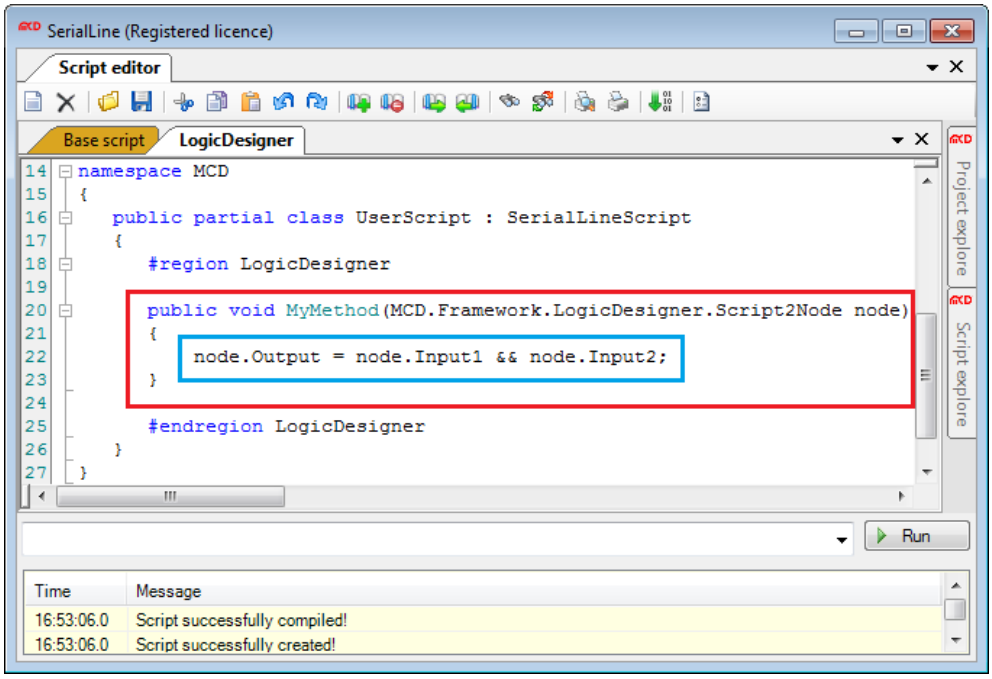


Abbildung 55: MyNode - Methode im Script Editor

Nach einem Klick auf den Befehl „Run“ sind die beiden Eingänge „Input1“ und „Input2“ nun über ein logisches UND verknüpft. Statt der UND - Verknüpfung können hier beliebige Logikverknüpfungen implementiert werden.

6. Logic Designers Control

6.1. Übersicht

Das „Logic Designers Control“ ist ein Werkzeug der MCD Toolmonitore zur Anzeige, Steuerung und Analyse der im Projekt vorhandenen Logic Designs. Die Features des „Logic Designers Controls“ sind:

- Übersichtliche / tabellarische Darstellung aller Logic Designs mit jeweils dem aktuellen Status und Verarbeitungszeiten
- Starten / Stoppen der Logic Designs von der zentralen Stelle
- Detailansicht mit der Auflistung der relevanten „Logic - Nodes“ mit jeweiligen Verarbeitungszeiten
- Schnelles Öffnen der Logic Designs per Doppelklick auf den Namen

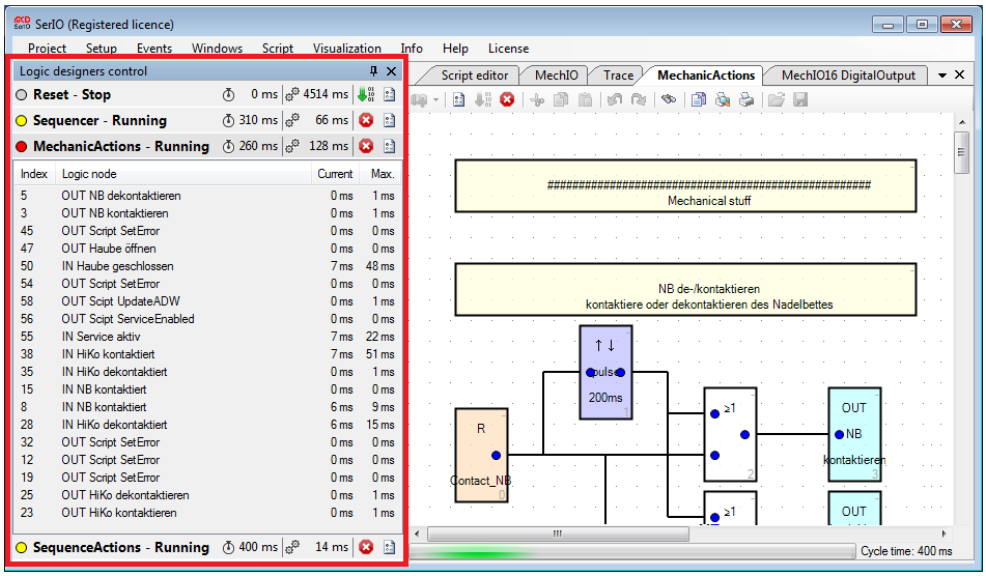


Abbildung 56: Links angedocktes Logic Designers Control (rot markiert)

6.2. Öffnen des Logic Designers Control

Das „Logic Designers Control“ kann über das Menü **Script → Logic designers → Logic designers control** aufgerufen werden.

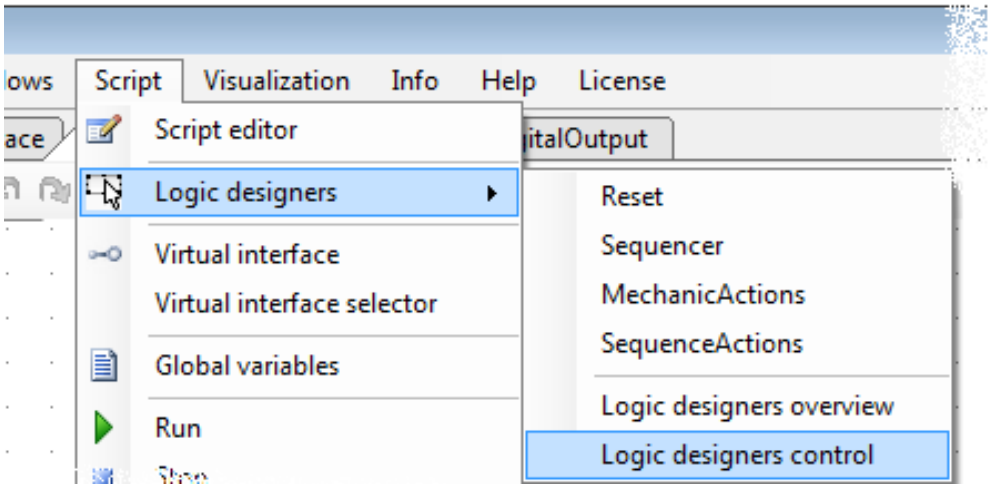


Abbildung 57: Logic Designers Control öffnen

6.3. Grafische Elemente

Das „Logic Designers Control“ enthält alle in der Anwendung definierten Logic Designs. Diese sind zeilenweise untereinander dargestellt. Ein Logic Design - Eintrag enthält:

- Den Namen und den aktuellen Status des Logic Designs (1.)
- Das Ausführungsintervall („Last Cycle Interval“) und die Verarbeitungszeit („Processing Cycle Duration“) (2.)
- Die Buttons zur Steuerung der Logic Designs (3.)

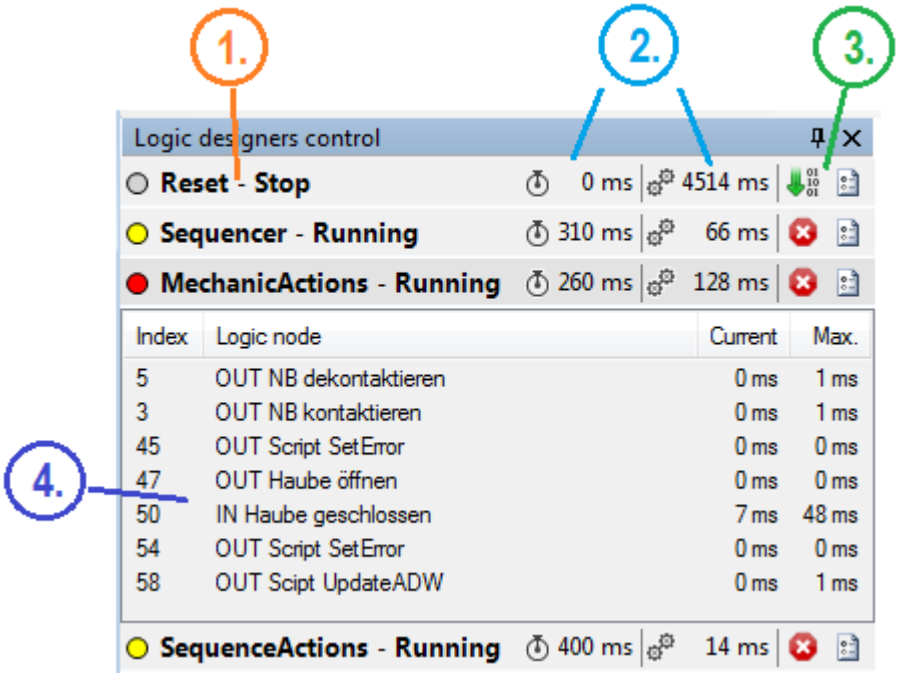


Abbildung 58: Darstellung eines Logic Design - Eintrags

Beim Anklicken eines Logic Designs mit der linken Maustaste wechselt die Darstellung des Eintrags in den Detailmodus (4.) und die im Logic Design enthaltenen „Logic - Nodes“ werden eingeblendet.

6.4. Aktueller Status

Der aktuelle Status wird durch das farbige Symbol links in der Zeile visualisiert.

Symbol / Farbe	Status	Beschreibung
Grau	Stop	Das Logic Design ist gestoppt und wird nicht verarbeitet.
Gelb	Running, Idle	Das Logic Design wird ausgeführt und wartet auf die Verarbeitung.
Rot	Running, Processing	Das Logic Design wird ausgeführt und wird momentan verarbeitet.

Abbildung 59: Status - Visualisierung

6.5. Das Ausführungsintervall und die Verarbeitungsdauer

Das Ausführungsintervall, auch „Last Cycle Interval“, und die Verarbeitungsdauer, auch „Processing Cycle Duration“, eines Logic Designs sind wichtige Kenngrößen zur Analyse der Performance der Logic Designs.

Das Ausführungsintervall ist die absolute Zeit in Millisekunden, die zwischen der letzten und der aktuellen Ausführung verstrichen ist. Diese entspricht im Optimalfall der in der Eigenschaft „Intervall“ eingestellten Zeit und mindestens der tatsächlichen Verarbeitungsdauer aller aktiven Logic Designs. Das Ausführungsintervall ist mit dem Stoppuhr - Symbol ( ) gekennzeichnet.

Abbildung 60: Darstellung des Ausführungsintervalls

Die Verarbeitungsdauer**,** gekennzeichnet mit dem Zahnrad - Symbol ( ), ist die tatsächliche Dauer der Verarbeitung des Logic Designs. Diese setzt sich aus der Verarbeitungszeit aller „Logic - Nodes“ des logischen Netzwerks zusammen. Die Verarbeitungsdauer der einzelnen „Logic - Nodes“ kann in der Detailansicht angezeigt / analysiert werden.

6.6. Direkte Steuerung der Logic Designs: Starten / Stoppen und Öffnen

Zur direkten Steuerung der Logic Designs aus dem „Logic Designers Control“ stehen die Buttons „Run“ ( ), „Stop“ ( ) und „Designmode“ ( ) zur Verfügung.

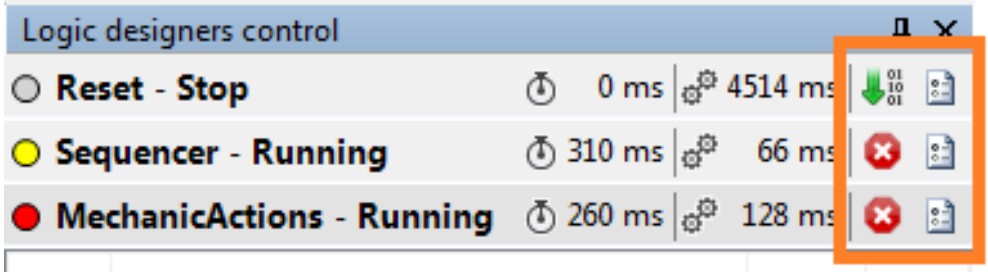


Abbildung 61: Buttons zur Steuerung der Logic Designs

Ein weiteres nützliches Feature des „Logic Designers Controls“ ist die Möglichkeit, schnell zwischen den Logic Designs zu navigieren: Per Doppelklick mit der linken Maustaste wird das Logic Design geöffnet und fokussiert. Dabei wird der aktuelle Ausführungsmodus beibehalten.

Button / Funktion	Funktion
Run ( )	Die Logikschaltung wird gestartet
Stop ( )	Die Logikschaltung wird gestoppt (nur im Ausführungsmodus)
Designmodus ( )	Öffnet das Logic Design zum Bearbeiten (nur im Ausführungsmodus)
Doppelklick auf den Namen	Öffnet das Logic Design und setzt den Fokus darauf

Abbildung 62: Zusammenfassung der Steuerungsfunktionen

6.7. Detailansicht

Klickt man im „Logic Designers Control“ auf einen Eintrag, so wird die Liste seiner Logikelemente dargestellt. Die Liste enthält alle für die Performance - Analyse relevanten Logikelemente: „Inputs“, „Outputs“, „Register“ und „Scripts“. Für diese Logikelemente werden folgende Daten ausgegeben: „Processing Index“ (Spalte Index), der Name des Logikelements, die aktuelle Verarbeitungszeit und die maximale Verarbeitungszeit (gemessen seit dem letzten Start des Logic Designs).

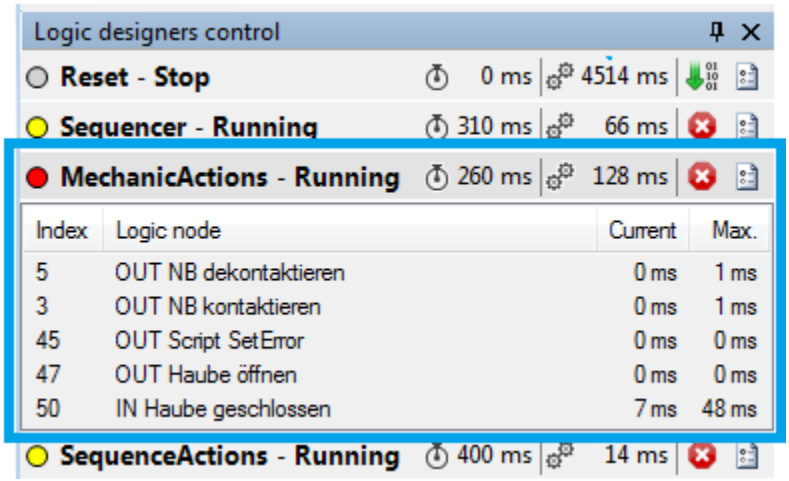


Abbildung 63: Liste der Logikelemente

Spaltenname	Beschreibung
Index	Processing Index: Verarbeitungsreihenfolge des Logikelements innerhalb des Logiknetzwerkes
Logic node	Name / Beschriftung des Logikelements
Current	Letzte erfasste Verarbeitungsdauer des Logikelements in Millisekunden
Max.	Maximale Verarbeitungsdauer dieses Logikelements seit dem letzten Start des Logic Designs

Abbildung 64: Zusammenfassung der Datenausgabe