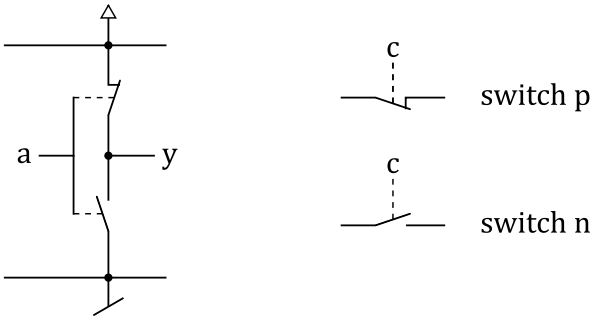
Logische Zustände

Übungen Digitales Design

1 | LST - Logikgatter, welche nur einen Zustand liefern

1.1 Schalter-Schaltungen

Bestimmen Sie die Beziehung zwischen Ein- und Ausgang der Schaltung der folgenden Abbildung.

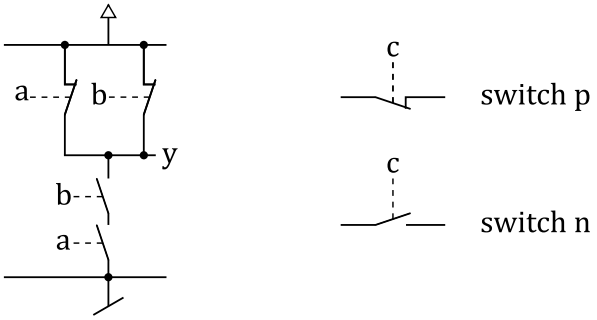


Ein n-Typ-Schalter ist zu, wenn der Steuersignal auf `'1'` ist. Ein p-Typ-Schalter ist offen, wenn der Steuersignal auf `'1'` ist.

lst/one-state-01-01

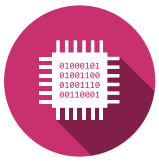
1.2 Schalter-Schaltungen

Bestimmen Sie die Beziehung zwischen Ein- und Ausgang der Schaltung der folgenden Abbildung.



Ein n-Typ-Schalter ist zu, wenn der Steuersignal auf `'1'` ist. Ein p-Typ-Schalter ist offen, wenn der Steuersignal auf `'1'` ist.

lst/one-state-01-02



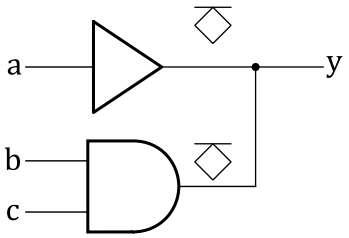
1.3 Schalter-Schaltungen

Nur mit Hilfe von Schaltern, schlagen Sie ein Schema eines XOR-Gatters mit 2 Eingängen vor.

lst/one-state-01-03

1.4 Open-Source Schaltung

Ergänzen Sie die Schaltung der nebenstehenden Abbildung.
Geben Sie die Wahrheitstabelle des Ausgangs **y** als Funktion der 3 Eingängen



lst/one-state-02-01

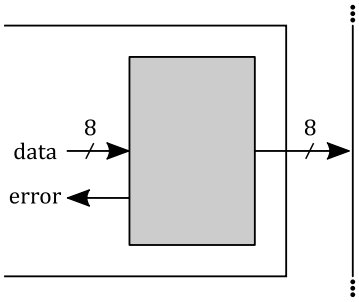
1.5 Alarmschaltung

Schlagen Sie das Schema einer Brand-Alarmschaltung in einem Gebäude vor.
Das Gebäude enthält 16 Rauchsensore, welche in verschiedenen Räumen gestellt sind. Diese werden durch einen 3-Draht-Kabel zusammengeknüpft: 2 für die Speisung und 1 für die Informationsübermittlung. Die Aktivierung eines einzelnen Sensores soll eine Sirene heulen lassen.
Ein Sensor liefert eine `'1'` bei Rauchaufspürung. Die Sirene heult, sobald ihr Steuersignal auf `'1'` schaltet.

lst/one-state-02-02

1.6 Zusammenstossaufspürung

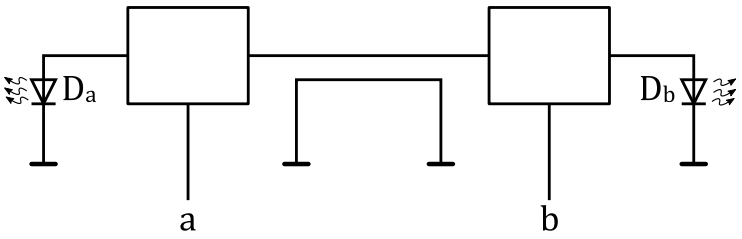
Zeichnen Sie das Schema eines Bus-Interfaces, wo mehrere Bausteine fähig sind, auf einem gemeinsamen Bus einen 8-Bit Wert zu schreiben und dessen Wert kontrollieren zu können.
Ist der Wert auf der Linie verschieden als den vorgesehenen, so liefert das Bus-Interface dem Baustein ein Irrtumanzeigesignal, um ihm den Übertragungsproblem wissen zu lassen.
Schlagen Sie eine Methode vor, um den Daten auf dem Bus eine Priorität geben zu können.



lst/one-state-02-03

1.7 Informationsübermittlung auf einem einzigen Draht

Geben Sie das innere Schema der Blöcke der Schaltung der folgenden Abbildung.



a	b	D_a	D_b
0	0	0	0
0	1	1	0
1	0	0	1
1	1	0	0

lst/one-state-02-04

2 | LST - Logikgatter mit Ausgängen hoher-Impedanz

2.1 Serieverknüpfung von Peripheriebausteinen

Schlagen Sie ein Schema zur Verknüpfung von Bausteinen zu einem gemeinsamen Datenbus vor. Das System enthält 3 Komponente, welche je einen 8-Bit Wert liefern. Um einen davon zu wählen erstellt die Systemsteuerung eine auf 2 Bit codierte Adresse. Die Funktionsweise ist die folgende:

- ist die Adresse gleich 0, so übermittelt kein Baustein einen Wert,
- ist die Adresse gleich 1, so übermittelt der Baustein 1 seinen Wert,
- ist die Adresse gleich 2, so übermittelt der Baustein 2 seinen Wert,
- ist die Adresse gleich 3, so übermittelt der Baustein 3 seinen Wert.

lst/hiz-01

2.2 Erstellung einer Funktion Tristate Schaltungen

Implementieren Sie einen 2 zu 1 Multiplexer unter Verwendung von Puffern mit hochohmigem Ausgang (Tri-State).

lst/hiz-02

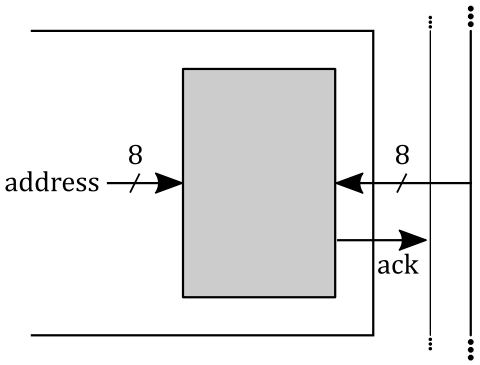
2.3 Erstellung einer Funktion mit Hilfe von Tristate Schaltungen

Schlagen Sie das Schema einer offenen-Kollektor-Inverters abhand von Tristate-Invertern vor.

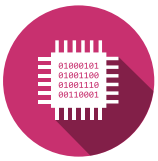
lst/hiz-03

2.4 Zusammenstossaufspürung

Schlagen Sie das Schema eines Bus-Interfaces vor, wo die Bausteine ihre Anwesenheit bei Erscheinung ihrer Adresse angeben.
Wenn die Adresse auf dem externen Bus gleich die des Bausteines ist, so gibt das Bus-Interface ihre Anwesenheit auf der **`ack`** (acknowledge) Linie an, welche allen Bausteinen gemeinsam ist.
Schlagen Sie eine Technik vor, um diesem System während des Betriebes identische Komponente eines nach dem anderen zuzufügen, ohne diese manuell konfigurieren zu müssen.

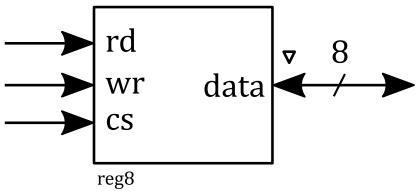


lst/hiz-04

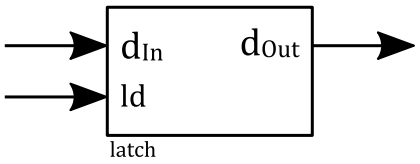


2.5 Register mit bidirektionellem Datenbus

Geben Sie das innere Schema eines 8-Bit Registers mit bidirektionalem Datenbus.
Das Register ist mit einem bidirektionalem Datenbus verknüpft. Für einen Schreibzugriff wird ein 8-Bit Wert auf dem Datenbus gestellt und die Kontrollsignale **`cs`** (chip select) und **`wr`** (write) werden aktiviert. Für einen Lesezugriff werden die Kontrollsignale **`cs`** und **`rd`** (read) aktiviert und das Register stellt sein gespeicherter Wert auf dem Datenbus.



Der Register ist aus 8 Speicherelemente (latch) wie derjenige der nebenstehenden Abbildung zusammengestellt. Wenn der Eingang **`ld`** (load) aktiv ist, so wird der Wert d_{In} (data in) im Element gespeichert, und dieser bringt ihn direkt auf seinem Ausgang d_{Out} (data out). Das Speicherelement behält den Wert unverändert solange **`ld`** nicht aktiv bleibt.
Mit Hilfe des erzeugten Komponentes erstellen Sie das Schema eines 8-Bit Lese-Schreib-Speichers.



lst/hiz-05