

MOTIVATION

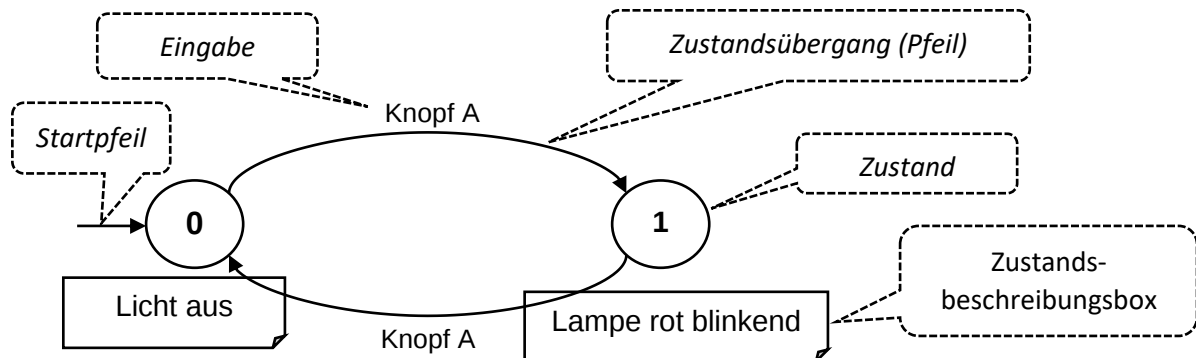
Der Calliope besitzt eine helle LED Leuchte in der Mitte. Diese soll als **Fahrradlampe** dienen. Ein Druck auf Knopf A schaltet die LED ein, nochmal auf A gedrückt schaltet die LED wieder aus.

A) Versuche, die Aufgabenstellung mit dem Calliope zu programmieren.

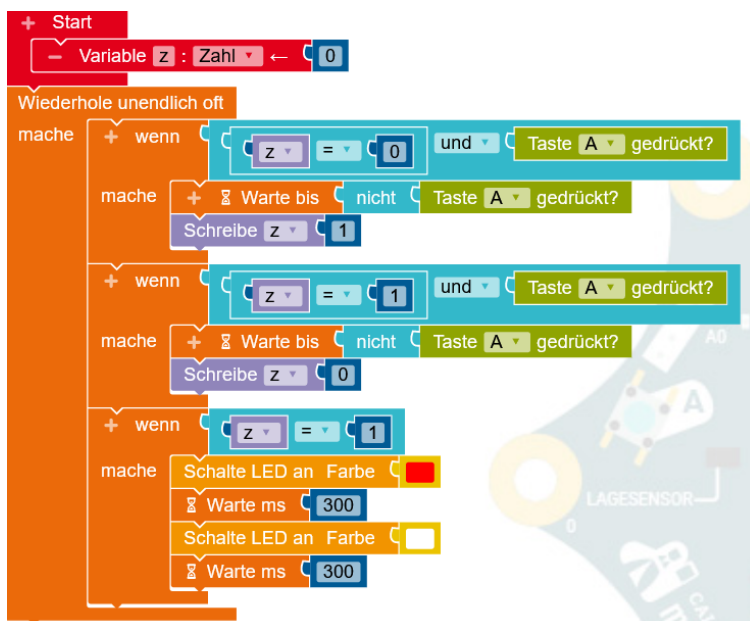
B) Benenne die Probleme, die sich bei der Umsetzungen ergeben.

ÜBERSICHT

Ein Zustandsmodell besteht aus **Zuständen**, **Zustandsübergängen (Pfeile)**, **Eingaben** und **Zustandsbeschreibungsboxen**. Ein **Startpfeil** gibt an, in welchem Zustand das Modell startet.



Das **Implementieren** des **Zustandsmodells** mit dem **Calliope** birgt in den Codeblöcken einige Besonderheiten. Das soll hier am Beispiel einer Warnlampe, die beim Druck auf Knopf A rot blinkt und bei nochmaligem Druck stoppt, aufgezeigt werden.



(1) Der Übergang von Zustand 0 zu Zustand 1 erfolgt, wenn **(z=0) und (A gedrückt)**

(2) Da A ein Taster ist und das Programm beim Druck auf A "durchrutscht", sollte die Eingabe erst erfolgen, wenn A losgelassen wird: **"Warte bis | nicht | Taste A gedrückt"**.

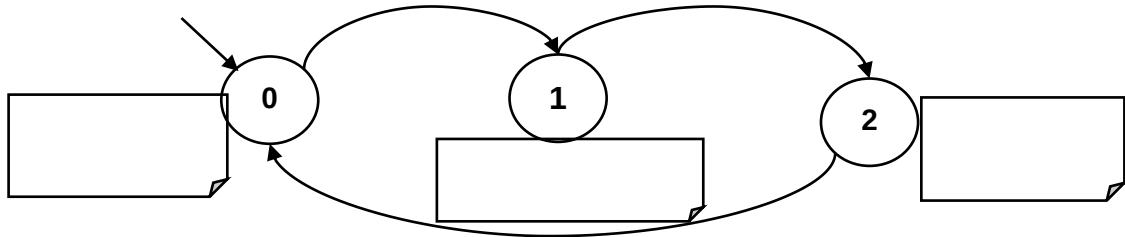
(3) Das Blinken wird mithilfe von **"Warte ms | 300"** realisiert. Dadurch blockiert das Programm etwas. Allerdings ist das eine einfache günstige Lösung.

ANWENDEN

Aufgabe 1

Eine **Fahrradlampe** hat beim Leuchten 2 Helligkeitsstufen (hell / mittelhell / aus). Beim Einschalten ist sie hell und kann über Knopfdruck (Knopf A) auf mittelhell und aus geschaltet werden. Ist sie aus, schaltet Knopf A die Lampe wieder an (hell).

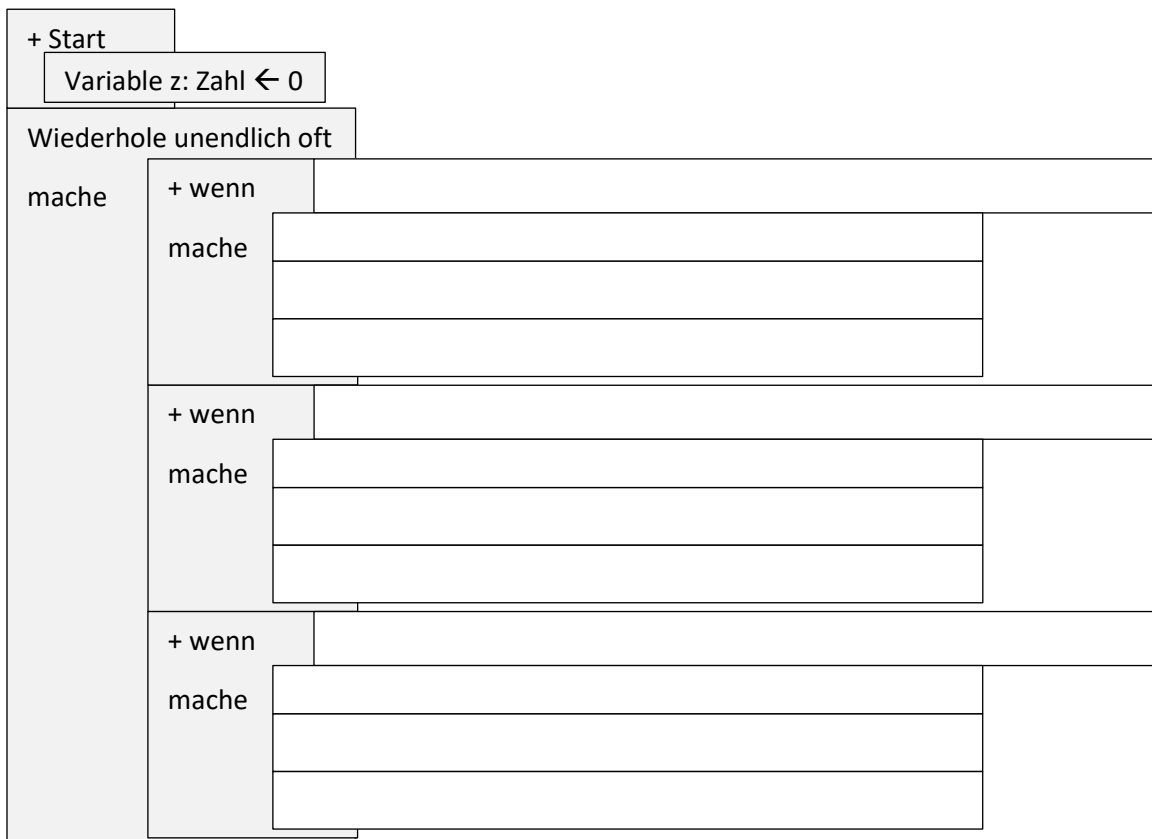
a) Ergänze das Zustandsmodell in der Vorgabe.



b) Ergänze die Programmvorlage mithilfe der vorgegebenen Befehle:

<i>(z = 0) und</i>	<i>Taste A gedrückt?</i>	<i>⌚ Warte bis nicht Taste A gedrückt</i>
<i>(z = 1) und</i>	<i>Taste A gedrückt?</i>	<i>⌚ Warte bis nicht Taste A gedrückt</i>
<i>(z = 2) und</i>	<i>Taste A gedrückt?</i>	<i>⌚ Warte bis nicht Taste A gedrückt</i>

Schreibe z	0	<i>schalte LED an: Farbe weiß</i>
Schreibe z	1	<i>schalte LED an: Farbe grau</i>
Schreibe z	2	<i>schalte LED aus</i>



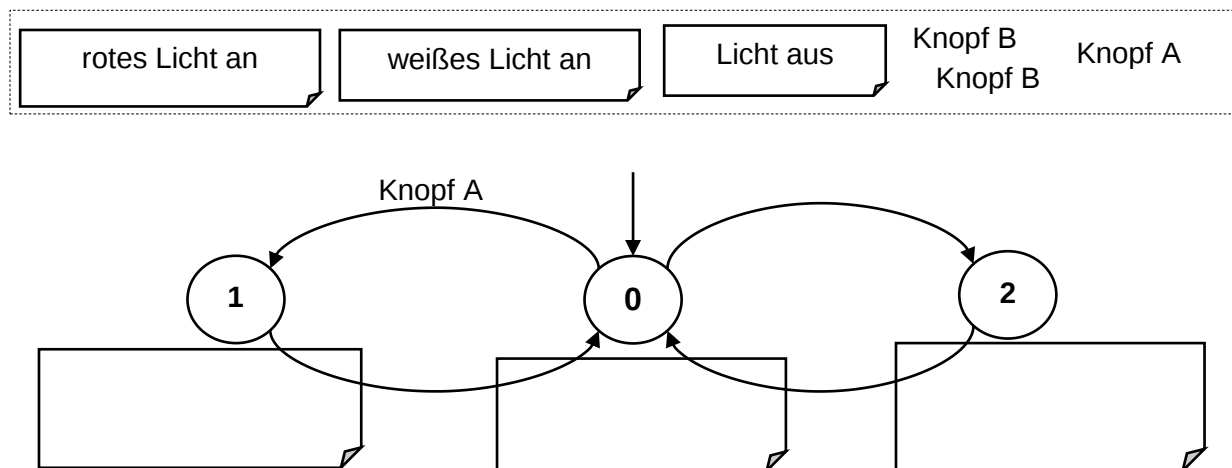
Aufgabe 2

Eine **Pannnenlampe für das Auto** soll zwei Farben bekommen. Weißes Licht, um bei einem Defekt im Dunkeln eine Reparatur durchführen zu können, rotes Licht, um im Dunkeln andere Autofahrer vor dem Pannnenfahrzeug warnen zu können.
Mit *Knopf A* wird weißes Licht an und ausgeschaltet, mit *Knopf B* rotes Licht.

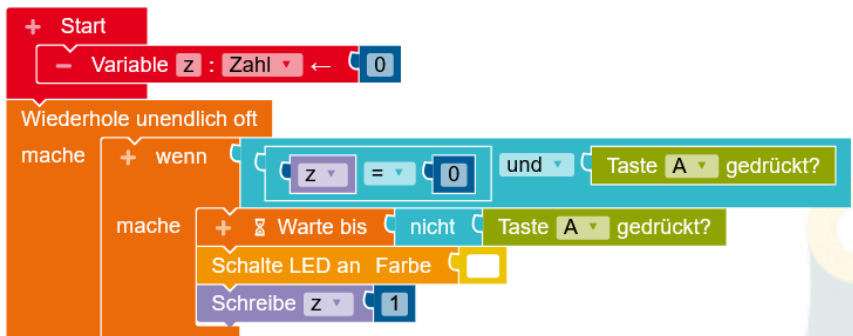
a) Ergänze das Zustandsmodell, indem du die in der Vorgabe abgebildeten

- Ein-/Ausgaben an die Zustandsübergänge (Pfeile) und
- die Zustandsbeschreibungen in die Beschreibungsboxen

schreibst.



b) Implementiere mit dem Calliope. Der Anfang des Programms ist schon vorgegeben.



Aufgabe 3

Ein Fahrrad hat eine **3-Gang-Schaltung**. Ein Druck auf Knopf A schaltet einen Gang hoch, Knopf B schaltet einen Gang runter. Von Gang 3 aus kann nicht weiter hoch geschaltet werden, von Gang 1 aus nicht weiter runter. Der Calliope soll die Schaltung simulieren und zur Anzeige des aktuellen Gangs dienen. Es werden also die Zahlen "1", "2" oder "3" angezeigt.

a) Ergänze das vorgegebene Zustandsmodell.

b) Setze das Modell mit dem Calliope um.

	Automaten
	Zustandsmodellierung mit dem Calliope

Aufgabe 4

Ein **Fahrradrücklicht** kann aus sein, rot leuchten oder rot blinken. Die 3 Zustände sollen mit der Taste A durchgeschaltet werden können. Modelliere und implementiere.

→ Hilfe zum "Blinken" erhältst du in der ÜBERSICHT (Seite 1).

Aufgabe 5

Ein **Fahrrad** soll mit einem **Blinker (Fahrtrichtungsanzeiger)** ausgestattet werden.

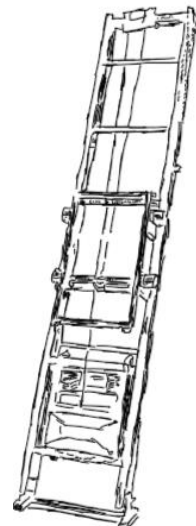
Handkonsole am Lenker | Anzeigekonsole nach hinten am Sattel



Die *Handkonsole* hat zwei Tasten L und R. Das sind beim Calliope die Tasten A und B. Bei Betätigen des L-Tasters (Taste A) leuchtet eine Pfeil in der *Anzeigekonsole* nach links, nochmaliger Druck schaltet das Blinken wieder aus. Ebenso beim Taster R (Taste B). Hier soll ein Pfeil nach rechts leuchten, der über R wieder ausgeschaltet wird. Zusätzlich soll von links nach rechts und umgekehrt umgeschaltet werden. D.h. leuchtet der linke Pfeil und Taste B wird gedrückt, erscheint sofort der rechte Pfeil (und umgekehrt). Entwirf das Zustandsmodell und implementiere.

Aufgabe 6

Mit dem **Dachdeckeraufzug** kann ein Dachdecker Ziegel und Baumaterial auf das Dach eines Hauses fahren lassen. Die Taste A fährt den Aufzug (wenn er unten ist) nach oben. Das dauert 2 Sekunden. Die Taste B fährt den Aufzug (wenn er oben ist) wieder runter (Dauer: auch 2 Sekunden). Die Fahrtrichtung (hoch/runter) soll während dieser 2 Sekunden angezeigt werden. Erstelle das Zustandsmodell und implementiere.



→ Hinweis: Benutze 4 Zustände.

Aufgabe 7

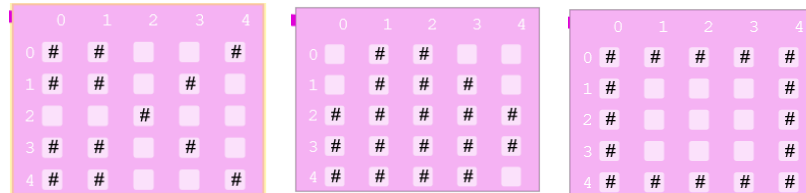
Ein **Cola-Automat** am Bahnhof kann gekühlte Cola-Flaschen ausgeben. Pro Flasche kostet jedes Getränk 2 Euro. Der Automat nimmt 1 und 2 Euro Stücke an. Es muss passend gezahlt werden, bei Überzahlung wird die letzte Münze wieder ausgeworfen. Die Getränkeflasche wird beim Erreichen der 2 Euro automatisch ausgegeben.

- Benenne die Eingaben und Zustände.
- Modelliere den Automaten mit einem Zustandsgraph.
- Erstelle ein Simulationsprogramm mit dem Calliope.

Aufgabe 8

"Schere, Stein, Papier" soll in einer ganz einfachen Version implementiert werden. Zunächst wählt der Spieler über die Taste A "Schere", "Stein" oder "Papier" aus, indem ein Tastendruck die Wahlmöglichkeiten durchschaltet: "Schere" → "Stein" → "Papier" → "Schere" → "Stein" → "Papier" etc.

Als Symbole:



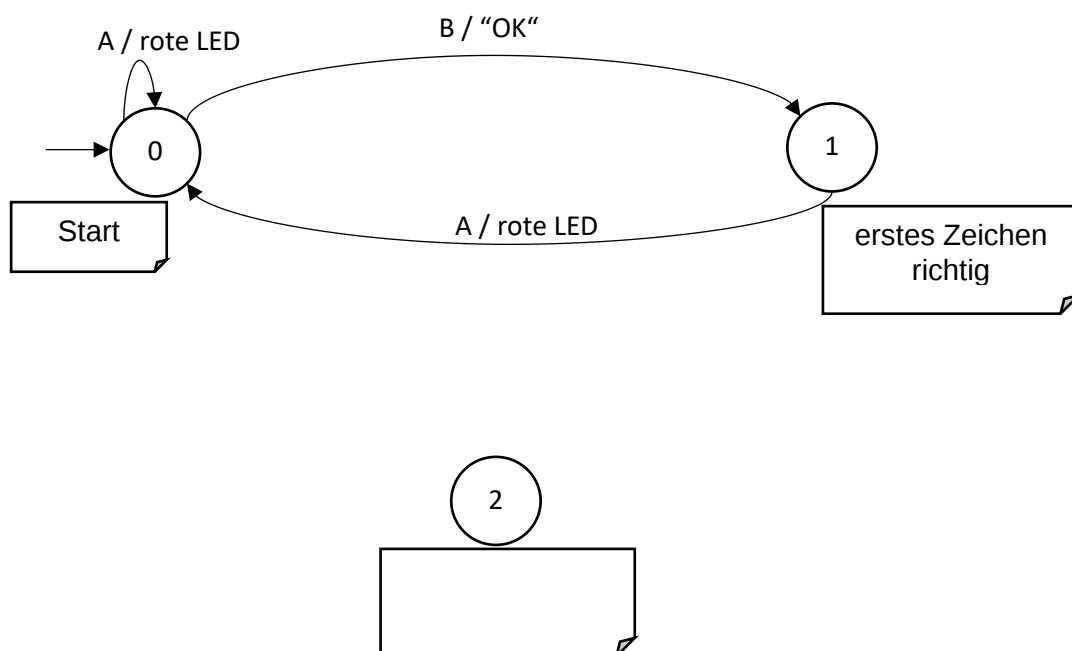
Dann, wenn auf Knopf B gedrückt wird, werden 2 Sekunden gewartet, und der Computer wählt eines der Symbole zufällig. Eine automatisierte Auswertung, wer gewinnt, findet nicht statt (Könnte aber theoretisch implementiert werden).

Aufgabe 9

PIN-Eingabe: Die Tasten A und B werden zur Eingabe einer Zeichenfolge (PIN) benutzt, mit der zum Beispiel der Zugang einer Tür gesichert werden kann. Die Zeichenfolge besteht aus drei Zeichen, **BBA**. Bei der Eingabe der Tastenfolge BBA leuchtet die LED für 2 Sekunden grün, bei einer fehlerhaften Eingabe für 2 Sekunden rot. Wird eine richtige Taste gedrückt, erscheint „OK“ auf dem Display.



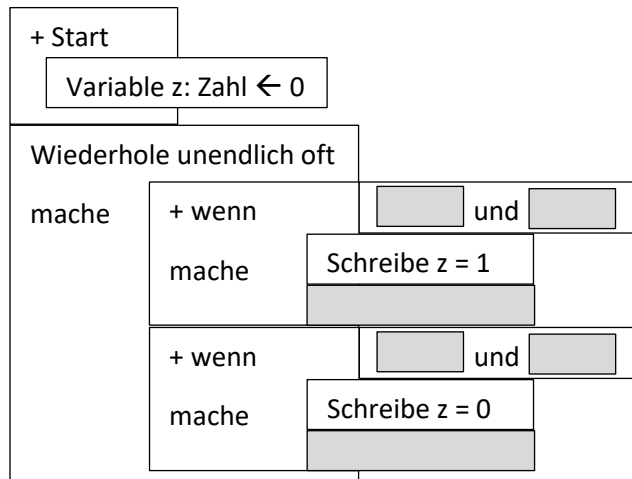
Ein Teil des Zustandsmodells ist schon vorgegeben. Ergänze das Modell:



Aufgabe 10

Schrank-Beleuchtung: Der Calliope liegt im Schrank. Geht die Tür auf (etwas Licht), leuchtet der Calliope, ist die Tür zu (ganz dunkel) geht der Calliope aus.

Ergänze das abgebildete Programm, indem du die Zahlen der fehlenden Codeschnipsel in die grauen Lücken schreibst und anschließend das Zustandsmodell zu dem Programm zeichnest.



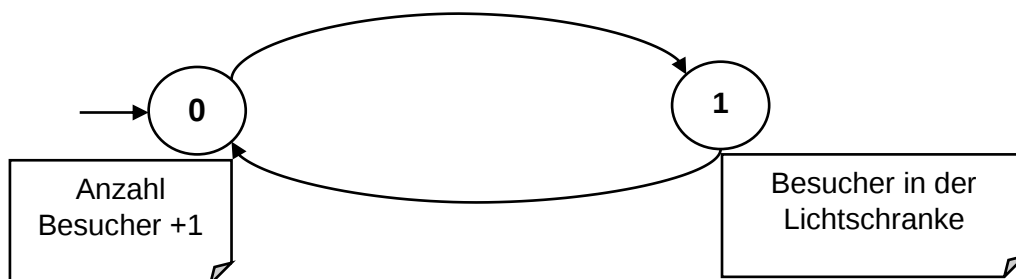
- 1.) $z = 1$
- 2.) $z = 0$
- 3.) gib „Wert“ % Lichtsensor > 30
- 4.) gib Wert ° Temperatursensor > 30
- 5.) gib „aufrecht“ Lage
- 6.) gib „Wert“ % Lichtsensor < 30
- 7.) schalte LED an: Farbe weiß
- 8.) schalte LED aus.
- 9.) Lösche Bildschirm

Aufgabe 11

Eine **Lichtschranke** zählt die Besucher eines Supermarkts. Bei jedem 10-ten Besucher wird eine Melodie gespielt.

Sensor: Lichtsensor *Zusatzmaterial:* Taschenlampe

a) Ergänze.



b) Implementiere.