

**Hinweis:**

Dieses Skript wurde vor der Präsenzvorlesung veröffentlicht. Das Beispiel in der Vorlesung kann daher leicht abweichen.

## 1 Automaten

### 1.1 Allgemeines

Neben der bereits bekannten Darstellung eines Algorithmus als Ablaufdiagramm ist für eine große Klasse von Anwendungen die Modellierung des Ablaufs als endlicher Automat verbreitet.

Bei der Modellierung als Automat geht man davon aus, dass ein bestimmter Ablauf in einzelne Zustände zerlegt wird, zwischen denen gewechselt werden kann (oder in denen bei Bedarf auch verblieben werden kann). Dieser Wechsel der Zustände bestimmt das Verhalten. Bei einem typischen Berechnungsalgorithmus ist man dagegen daran interessiert, zu bestimmten Eingaben ein Ergebnis zu erhalten, die dazu nötigen Zwischenschritte (Zustände) interessieren meist ebenso wenig wie die Reihenfolge, in der sie durchlaufen werden.

### 1.2 Außenbetrachtung

Ein Automat hat, als Blackbox betrachtet, eine Menge von Eingangssignalen, eine Menge von Ausgangssignalen und möglicherweise einen von außen zugeführten Takt (Abbildung 1).

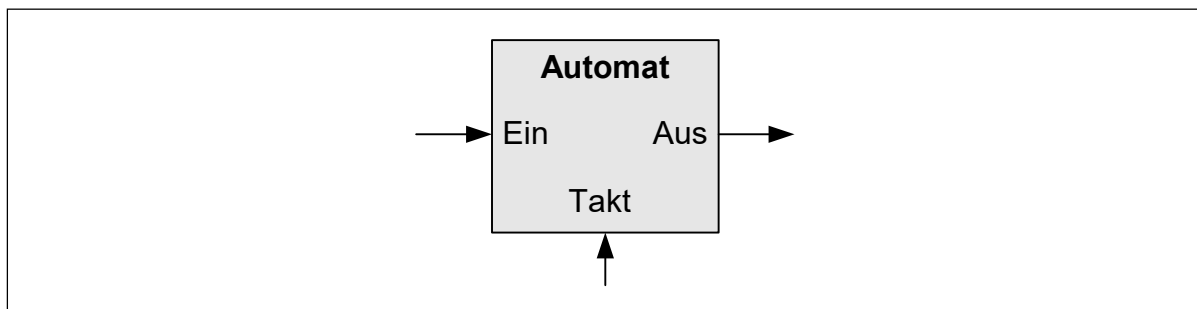


Abbildung 1: Automat als Black-Box

Die Eingangssignale liefern dem Automaten Informationen. Das sind meist Sensoren (z.B. Druck, Temperatur etc.) oder Tasten zur Bedienung eines Geräts. Mit Hilfe dieser Signale wird der **interne Ablauf** gesteuert. Mit Hilfe der Ausgangssignale kann der Automat seine Umwelt beeinflussen (Aktoren wie z.B. Motoren) oder Informationen für den Bediener ausgeben (Display). Der Takt bestimmt das **Zeitraaster**, in dem der Automat auf die Eingangssignale reagiert oder in dem er Ausgaben vornimmt.

Es gibt verschiedene Typen von Automaten, von denen der sog. **Moore-Automat** der einfachste ist. Er wird sehr gerne in der Hardwareentwicklung verwendet, kann aber auch gut als Grundlage für eine Softwareentwicklung dienen.

Der im Praktikum verwendete  $\mu\text{C}$  (wie auch alle anderen  $\mu\text{C}/\mu\text{P}$ ) ist mit an Sicherheit grenzender Wahrscheinlichkeit als eine Menge von zusammenarbeitenden Moore-Automaten entworfen worden.

Der Moore-Automat zeichnet sich dadurch aus, dass zwischen zwei Takten keine Reaktion auf die Eingabesignale erfolgen kann und ebenso wenig eine Änderung der Ausgabesignale

möglich ist. Ohne einen Takt ist der Moore-Automat handlungsunfähig, jede Änderung eines Ausgangssignals erfordert mindestens einen Zustandswechsel.

### 1.3 Zustände

Der Entwurf eines Automaten erfolgt Top-Down und häufig mit dem Hilfsmittel „Zustandsübergangsdiagramm“. Das ist ein Diagramm, in dem alle benötigten Zustände für den gewünschten Ablauf eingetragen werden. Zudem wird dort eingetragen, welche Zustandswechsel in Abhängigkeit von den Eingangssignalen erfolgen können. Bei sehr wenigen Ausgangssignalen kann man dort auch angeben, was der Automat in einem Zustand ausgeben soll. Zunächst betrachtet man den Normalablauf, erst später verfeinert man dann den Ablauf. Als Beispiel soll eine sehr einfache Kaffeemaschine dienen.

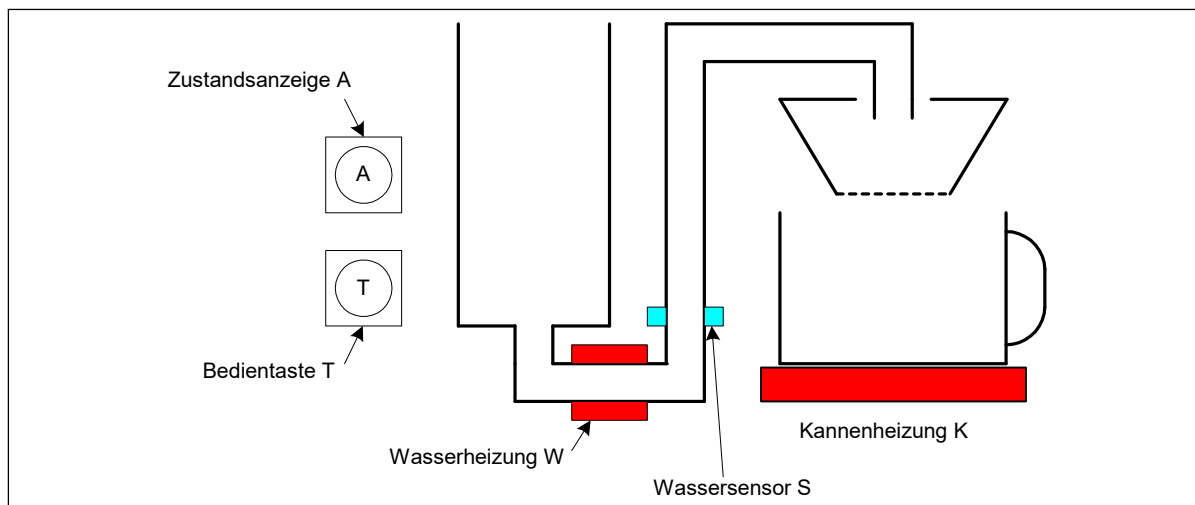


Abbildung 2: Kaffeemaschine

In Abbildung 2 sind die zur Verfügung stehenden Eingangssignale sowie die anzusteuernenden Ausgangssignale angegeben.

Der Normalablauf ist folgender: Die meiste Zeit wird die Maschine mit leerem Wasserbehälter herumstehen. In diesem Zustand soll die Anzeige A ausgeschaltet sein, um anzuzeigen, dass die Maschine nicht bereit ist. Der Zustand kann nur verlassen werden, wenn Wasser eingefüllt wird – dazu kann der Wassersensor S abgefragt werden. Der Zustand soll hier „Ruhe ohne Wasser“ (ROW) heißen. Wenn Wasser vorhanden ist, dann wechselt die Maschine in den Zustand „Ruhe mit Wasser“ (RMW). In diesem Zustand geschieht noch nichts weiter, außer dass die Anzeige jetzt „Bereit“ anzeigt. Aus diesem Zustand kann durch einen Druck auf die Bedientaste T die Durchlaufphase (DP) gestartet werden. Auch dies ist ein Zustand. Dabei sind die Wasserheizung W und die Kannenheizung K eingeschaltet, die Anzeige zeigt „Kaffee wird zubereitet“ an. Erst wenn der Wassersensor meldet, dass kein Wasser mehr vorhanden ist, wird der Zustand verlassen und in den Zustand „Wärmephase“ (WP) gewechselt. In diesem Zustand zeigt die Anzeige „Kaffee fertig“ an und es ist nur noch die Kannenheizung eingeschaltet. Aus diesem Zustand wird in den Zustand ROW gewechselt, wenn die Taste T erneut betätigt wird. Aus diesem Normalablauf lassen sich bereits mindestens vier Zustände herauslesen, die für die erste Lösung benötigt werden (ROW, RMW, DP und WP). Ein Zustand wird in einem Zustandsübergangsdiagramm (engl. state chart) i.d.R. durch einen Kreis dargestellt.

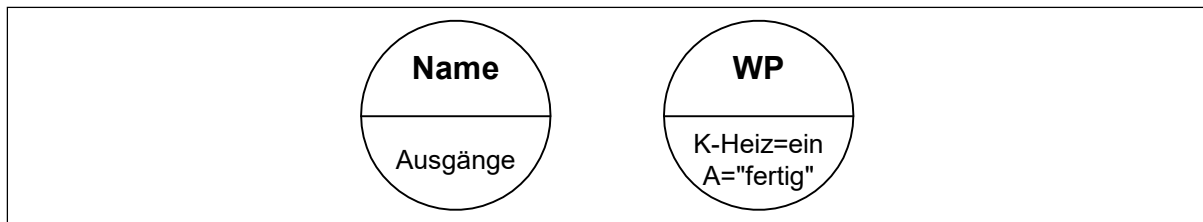


Abbildung 3: Zustände

In Abbildung 3 links ist der allgemeine Fall dargestellt. Zu jedem Zustand gehört ein **eindeutiger** Name. Zustände innerhalb desselben Diagramms mit demselben Namen sind auch identisch. Wenn es der Platz zulässt, dann ist es beim Moore-Automaten möglich, die Werte für die Ausgänge ebenfalls im Zustand unterzubringen (meist unter dem Namen). Zu beachten ist dabei aber, dass **alle** Ausgänge, deren Wert von Bedeutung (also nicht egal) ist, dort eingetragen werden müssen. Der Zustand rechts aus dem Beispiel für die Kaffeemaschine ist daher **nicht vollständig** beschrieben. Der Grund ist, dass in diesem Zustand die Wasserheizung abgestellt sein soll, das müsste also ebenfalls explizit vermerkt sein.

## 1.4 Übergänge

Der Automat kann nur unter genau definierten Bedingungen von einem Zustand in einen anderen wechseln. Dies wird durch einseitig gerichtete Pfeile dargestellt, die vom *aktuellen Zustand* (AZ) zum jeweiligen *Folgezustand* (FZ) zeigen (Abbildung 4).



Abbildung 4: Übergänge

Die Bedingung wird an den Pfeil geschrieben, sie kann sich nur auf Eingangssignale beziehen. Für den ersten Entwurf kann man das noch in freier Schreibweise (rechts in Abbildung 4) machen, für die Realisierung benötigt man dann eine formellere Schreibweise.

Zu einem AZ kann es viele FZ geben. Sehr häufig muss der Automat für mehr als einen Takt in demselben Zustand bleiben, in dem Fall gibt es einen Übergang, in dem der FZ der AZ ist – also einen Pfeil, der im Kreis zurück führt. Diesen Übergang kann man nicht weglassen, wenn er nötig ist.

Um die Bedingungen formaler aufschreiben zu können und auch, um später Prüfungen auf formale Fehler durchführen zu können, ist es nötig, die Bedeutung der Eingangssignale zu können. In der Digitalelektronik (auf die die  $\mu$ C-Technik aufbaut) gibt es in der Regel nur Signale mit den Werten 0 und 1.

Für das Beispiel sollen jetzt die Eingangssignale nach Tabelle 1 definiert sein.

Signal	Wert	Bedeutung
T	0	Taste nicht betätigt
	1	Taste betätigt
S	0	Kein Wasser vorhanden
	1	Wasser vorhanden

Tabelle 1: Definition der Eingangssignale

Eine formale Darstellung erfolgt jetzt mit Hilfe von logischen Verknüpfungen der Einzelsignale. Für die Bedingung „Wasser ist da“ wäre zu schreiben  $S=1$ . Für die Bedingung „Taste T betätigt“ oder Wasser vorhanden“ aus dem Normalablauf wäre zu schreiben  $(T=1)$  oder  $(S=1)$ . Als Bedingungen werden üblicherweise nur *und* und *oder* benutzt, wobei das oder

dem *inklusiv oder* entspricht. Die Bedingung ( $T=1$ ) oder ( $S=1$ ) ist also auch dann erfüllt, wenn sowohl die Taste gedrückt wird als auch Wasser vorhanden ist.

Als Sonderfall kann es vorkommen, dass ein Übergang *immer* erfolgt, also ganz unabhängig von den Eingängen. Dieser Fall wird mit  $1$  an dem Übergang gekennzeichnet. In dem Fall kann es natürlich für den aktuellen Zustand auch nur einen Folgezustand geben, in den mit dem nächsten Takt immer gewechselt wird.

Zur weiteren Abkürzung ist die folgende Konvention üblich:

$T=1$  wird mit  $T$  abgekürzt,  $T=0$  mit  $\bar{T}$  (gesprochen: T quer oder T nicht). Eine Verknüpfung mit *und* wird durch eine Multiplikation gekennzeichnet, eine Verknüpfung mit *oder* durch eine Addition. Wenn keine Verwechslungsgefahr besteht, kann man das Multiplikationszeichen weglassen.

Achtung: Die Regel Punkt vor Strich gilt hier **nicht**.

## 1.5 Zustandsübergangsdiagramm

Die Zustände sowie die Übergänge werden jetzt in ein gemeinsames Diagramm gezeichnet.

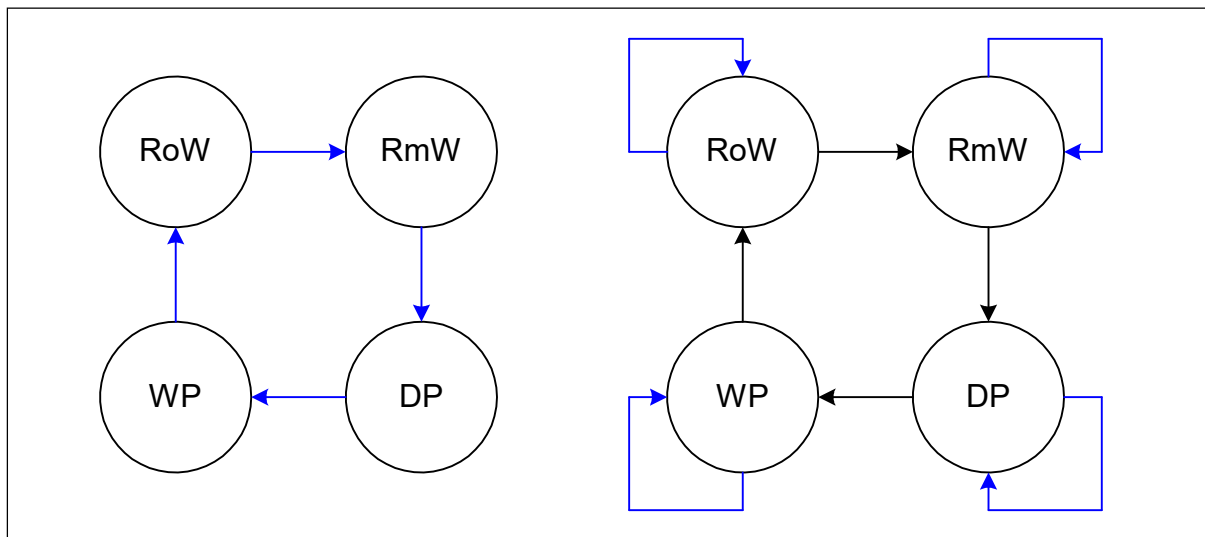


Abbildung 5: Zustandsübergangsdiagramm., erster Entwurf

Auf der linken Seite ist der vorgesehene Normalablauf zu sehen, rechts ist er um die Übergänge ergänzt, bei denen der jeweils aktuelle Zustand beibehalten wird. Die rechte Variante ist diejenige, die der Bediener sieht, die linke ist für den Entwickler nötig. Auch das rechte Diagramm ist noch nicht vollständig, weil weder die Ausgaben in jedem Zustand angegeben sind noch die Bedingungen für die Übergänge.

### 1.5.1 Ausgabetabelle

Eine übersichtliche Darstellung der Ausgaben für jeden Zustand kann in Form einer Ausgabetabelle erfolgen. Dabei wird einfach für jeden Zustand angegeben, welche Werte die Ausgänge annehmen sollen. Dazu ist es wieder nötig, die Ausgaben in Form von Signalen mit den Werten 0 und 1 zu kennen.

Signal	Wert	Bedeutung
K	0	Kannenheizung aus
	1	Kannenheizung an
W	0	Wasserheizung aus
	1	Wasserheizung an
A1, A0	0, 0	Anzeige aus

	1, 0	Anzeige „Bereit“ (grünes Licht)
	0, 1	Anzeige „Kaffee wird zubereitet“ (rotes Licht)
	1, 1	Anzeige „Kaffee ist fertig“ (oranges Licht)

Tabelle 2: Definition der Ausgangssignale

Hier fallen zwei Einträge auf. Zunächst besteht das bisherige Einzelsignal A jetzt aus zwei Signalen A1 und A0, die in einer bestimmten Kombination die gewünschte Anzeige ergeben. Im Beispiel würde man A1 mit einer grünen LED und A0 mit einer roten LED verbinden. Leuchten beide LED, dann ergibt sich die Mischfarbe Orange.

Mit den Definitionen aus Tabelle 2 kann jetzt die folgende Ausgabetabelle aufgestellt werden:

Zustand	K	W	A1	A0
ROW	0	0	0	0
RMW	0	0	1	0
DP	-	1	0	1
WP	1	0	1	1

Tabelle 3: Ausgabetabelle

In der Spalte für K steht bei dem Zustand DP ein -. Das bedeutet „es ist egal“. Man kann also später bei der Realisierung des Automaten hier 0 oder 1 ausgeben, ohne dass das die Funktion des Automaten beeinträchtigt. Allerdings ist es Energieverschwendung, die Kannenheizung schon in der Durchlaufphase einzuschalten. Man wird als wohl K=0 wählen.

### 1.5.2 Zustandsübergangstabelle

In diesem Beispiel sollen die noch fehlenden Bedingungen für die Übergänge ebenfalls in Form einer Tabelle angegeben werden. Dazu haben die Übergänge eine Referenznummer erhalten (Abbildung 6). Ignorieren Sie die roten Übergänge zunächst noch.

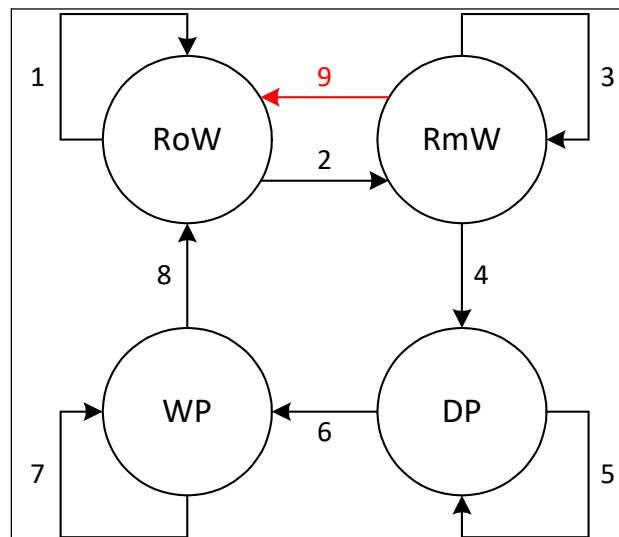


Abbildung 6: Erweitertes Zustandsübergangsdiagramm

Übergang	Langschreibweise	Formale Darstellung
1	Kein Wasser	$\bar{S}$
2	Wasser vorhanden	$S$
3	Taste nicht betätigt	$\bar{T}$
4	Taste betätigt	$T$
5	Noch immer Wasser da	$S$

6	Kein Wasser mehr da	$\bar{S}$
7	Taste nicht betätigt	$\bar{T}$
8	Taste betätigt	$T$

Tabelle 4: Zustandsübergangstabelle

An dieser Stelle kann man den Ablauf schon einmal in Gedanken durchgehen, und sich überlegen, welche abnormalen Situationen vorkommen könnten und wie dann darauf zu reagieren wäre. Eine Möglichkeit (das sind nicht alle) wäre: Der Bediener schüttet das Wasser wieder aus dem Vorratsbehälter, bevor er die Maschine startet (z. B. weil er zu viel eingefüllt hat) oder die Wasserheizung ist defekt und demzufolge wird die Durchlaufphase nicht beendet. Die entsprechende Reaktion ist in Abbildung 6 rot nachgetragen, in Tabelle 5 ist der Übergang eingetragen.

Übergang	Langschreibweise	Formale Darstellung
9	Kein Wasser	$\bar{S}$

Tabelle 5: Zusätzlicher Übergang

## 2 Formale Prüfungen

Wenn das Zustandsübergangsdiagramm in der ersten Version aufgestellt ist, dann sollte man in jedem Fall zwei formale Prüfungen durchführen. Dies entspricht in etwa dem Kompilieren eines C-Programms: Solange noch Warnungen auftreten, sollte man sich das Programm daraufhin ansehen. Für ein Zustandsübergangsdiagramm gibt es zwei derartige Prüfungen, die Prüfung auf Widerspruchsfreiheit und die Prüfung auf Vollständigkeit. Die Prüfungen werden für **jeden** Zustand durchgeführt.

Ein Widerspruch tritt auf, wenn für dieselbe Bedingung für einen aktuellen Zustand zwei verschiedene Folgezustände vorgesehen sind. In der Praxis heißt das, dass der Automat dann nicht weiß, was er tun soll – er könnte zufällig einen FZ auswählen und das könnte der falsche sein.

Ein Zustand ist unvollständig, wenn für eine mögliche Bedingung kein Übergang definiert ist. Auch hier hat der Automat in der Praxis eine unerwünschte Wahlfreiheit. Beide Prüfungen lassen sich wie folgt zusammenfassen:

**Für jeden Zustand muss gelten: Zu jeder Bedingung gibt es genau einen Folgezustand.**

Die Prüfung kann man sehr leicht systematisch mit Hilfe einer Tabelle durchführen (allerdings muss man diese Überprüfungstabelle für jeden Zustand getrennt aufstellen). Zunächst schreibt man sich alle Signale auf, die an Übergängen auftreten, die von dem Zustand wegführen. Alle anderen Signale kann man sich sparen, ebenso lässt man alle Übergänge unberücksichtigt, die **nur** zu diesem Zustand hinführen.

Dann spielt man alle Kombinationen der Signale durch und schreibt die vorgesehenen Folgezustände auf. In jeder Zeile muss dann **genau ein** FZ stehen (Zustände gleichen Namens sind identisch). Als Beispiel seien RMW und ROW betrachtet.

Signale	FZ
S	
0	ROW
1	RMW

Tabelle 6: Prüftabelle für ROW

Der Zustand ist vollständig und widerspruchsfrei.

Signale		FZ
S	T	
0	0	ROW (wegen 9), RMW (wegen 3)
0	1	ROW (wegen 9), DP (wegen 4),
1	0	RMW (wegen 3)
1	1	DP (wegen 4)

**Tabelle 7: Prüftabelle RMW**

Dieser Zustand ist nicht widerspruchsfrei. Wenn die Taste betätigt wird und zugleich kein Wasser mehr vorhanden ist, dann ist der FZ nicht eindeutig, Ebenso gibt es ein Problem, wenn kein Wasser mehr vorhanden ist, aber die Taste nicht betätigt wird.

Hier muss man sich überlegen, wie der Konflikt in der Praxis aufgelöst werden soll und die Bedingungen entsprechend anpassen.

Offenbar ist es in jedem Fall richtig, ohne Wasser in den Zustand ROW zu wechseln. Dazu passt man die Bedingungen entsprechend an (Tabelle 8).

Übergang	Langschreibweise	Formale Darstellung
3	Taste nicht betätigt <b>und</b> Wasser da	$\bar{T}S$
4	Taste betätigt <b>und</b> Wasser da	$TS$
9	Kein Wasser	$\bar{S}$

**Tabelle 8: Korrigierte Bedingungen für Zustand RMW**