

# HARDWARE-PRAKTIKUM

Versuch L-3

Synchrone Schaltwerke

Fachbereich Informatik

Universität Kaiserslautern

## Versuch L-3

In einer pfälzischen Universitätsstadt soll auf dem Gelände eines bedeutenden Lehr- und Forschungsbetriebes (LFB) ein Neubau für den expandierenden Fachbereich 'Allgemeinbildung' errichtet werden. Im Zeichen der Energieeinsparung und gemäß dem Motto 'Trimm Dich' sollen viele Treppen, aber nur ein Aufzug für Lasten und ganz Bequeme gebaut werden.

Wie nun die Erfahrung in den anderen Gebäuden des LFB gezeigt hat, gilt auch hier das Gesetz von Murphy, das besagt, daß ein Aufzug immer dann nicht kommt, wenn man ihn am nötigsten braucht. Ein findiger Kopf kam nun auf die Idee, die Aufzugsteuerung von den Betroffenen selbst entwerfen zu lassen, um so jede Kritik bei einem Ausfall im Keim zu ersticken.

Vom Fachbereich 'Motorentchnik' wurde die Antriebseinheit entworfen, die nur die beiden digitalen Eingänge AUF und AB hat. Eine logische 1 am Eingang AUF bzw. AB bewirkt eine Aufwärts- bzw. Abwärtsbewegung des Aufzugs. Eine 0 an beiden Eingängen bewirkt ein Anhalten. Eine 1 an beiden Eingängen zerstört die Antriebseinheit, ebenso ein AUF-Befehl im obersten bzw. ein AB-Befehl im untersten Stockwerk.

Vom Fachbereich 'Kabinenbau' wurde die Aufzugskabine mit der automatischen Türsteuerung entworfen. Es werden 3 Stockwerkssignale E1, E2 und E3 ausgegeben, die dann 1 werden, wenn die Kabine die Stockwerke 1, 2 oder 3 erreicht. Die Türsteuerung wird durch ein Signal TÜR angestoßen. Wird dieses Signal log. 1, so öffnet sich die Tür, ein Tür-geschlossen-Signal TG wird 0. Nach einiger Zeit schließt sich die Tür automatisch, das Signal TG wird 1. Die Türsteuerung arbeitet unabhängig von der Antriebseinheit, d.h. der Aufzug kann auch dann losfahren, wenn die Tür noch offen ist. Aufgrund eines vorauszusehenden Planungsfehlers wurde eine Schaltung vergessen, die verhindert, daß Personen eingeklemmt werden. Dies soll jedoch nicht stören, da die Studenten des LFB als sehr anpassungsfähig gelten.

Der Fachbereich 'Impulswesen' entwarf die Tastensteuerung für die Anforderungstasten außen und die Wahlstasten in der Kabine. Es werden 3 Anforderungssignale ANF1, ANF2 und ANF3 generiert, die dann 1 werden, wenn in einem Stockwerk die Anforderungstaste oder im Aufzug das entsprechende Stockwerk angewählt wurde. Gleichzeitig leuchten die entsprechenden Lampen in den Tasten auf. Die Signale ANF1..ANF3 werden automatisch dann zurückgesetzt, wenn die Tür in dem entsprechenden Stockwerk geöffnet wird. Es steht damit die folgende Schnittstelle nach Bild 1 für das Hauptsteuerwerk zur Verfügung.

### Annahmen

- Die Öffnungszeit der Tür ist größer als die Taktperiode.
- Ein Fahrtsignal (AUF oder AB) an die Motorsteuerung läßt den Aufzug auch dann losfahren, wenn die Tür noch offen ist, d.h. das zu entwerfende Steuerwerk muß diesen Fehlerfall abfangen.
- Liegt keine Anforderung vor, soll die Tür geschlossen sein.
- Das Steuerwerk werde mit einem langsamen Takt betrieben (ca. 1Hz). Es kann daher angenommen werden, daß nach einem Fahrbefehl das entsprechende Stockwerksignal (E1..E3) vor der nächsten Taktflanke bereits zurückgesetzt ist (siehe Bild 2).
- Der Aufzug befinde sich im Grundzustand in Ruhe.

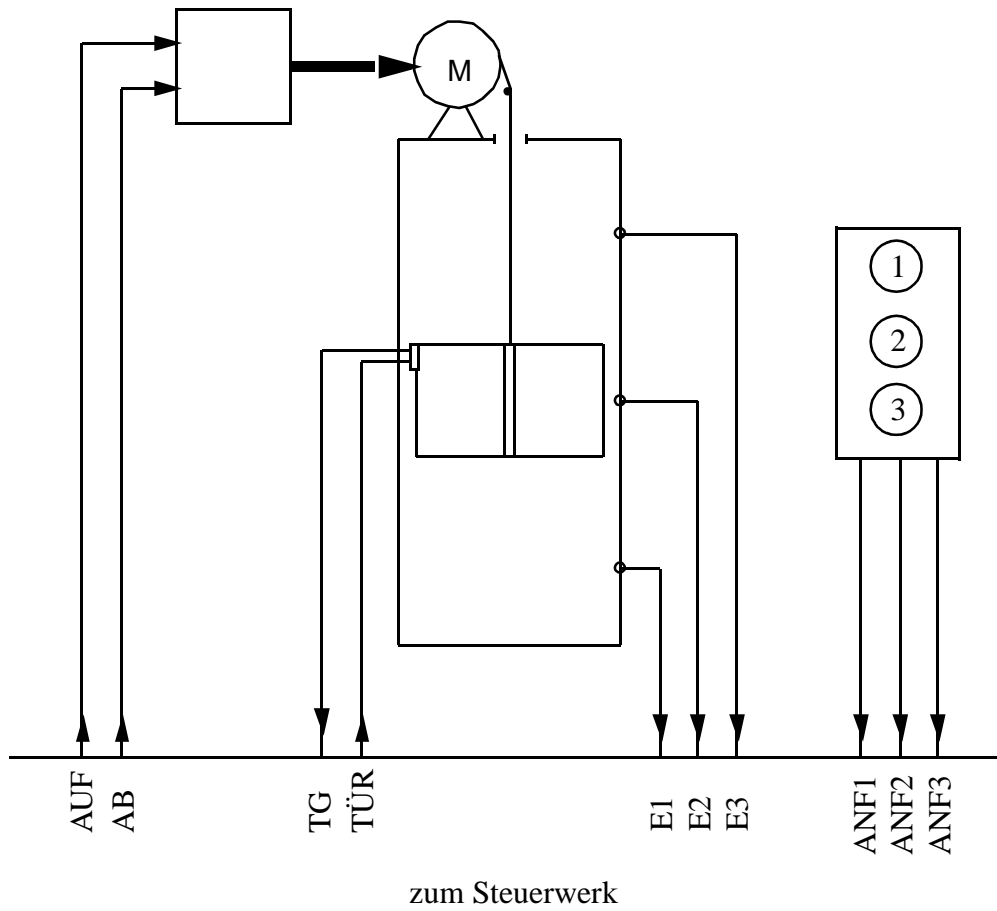


Abbildung 1: Schnittstelle Aufzug

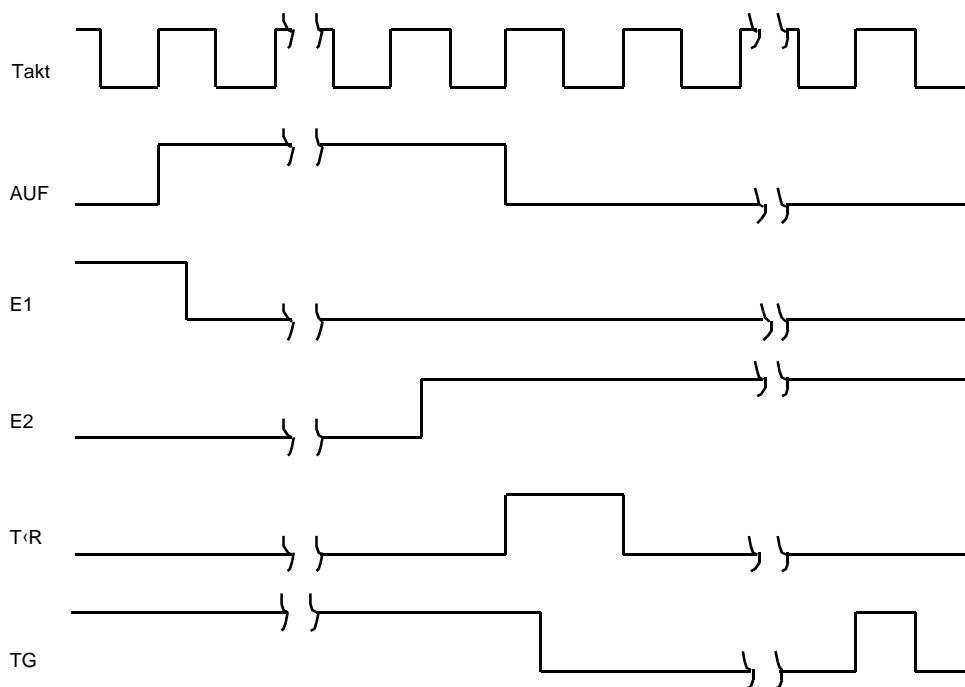


Abbildung 2: Beispiel für einen Ablauf

## Aufgaben

1. Entwerfen sie ein Schaltnetz, das aus den Signalen E1..E3, ANF1..ANF3 und TG ein Ausgangssignal  $Y = ( Y[2], Y[1], Y[0] )$  generiert mit folgender Bedeutung:

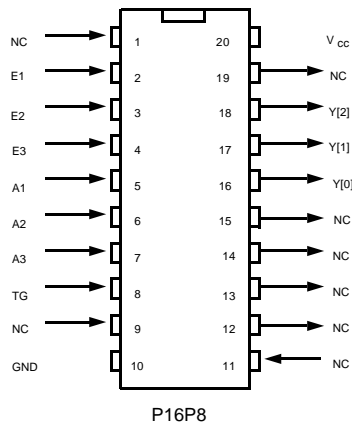
| Y[2] | Y[1] | Y[0] | Symbol         | Bedeutung  |
|------|------|------|----------------|--|
| 0    | 0    | 0    | T_offen        | Tür offen  |
| 0    | 1    | 0    | T_zu_halt_k_A  | Tür geschlossen, Aufzug hält, keine Anforderung                |
| 0    | 0    | 1    | Zwischenstock  | Aufzug befindet sich zwischen zwei Stockwerken                 |
| 0    | 1    | 1    | A_StW_wo_Aufz  | Anforderung aus dem Stockwerk, in dem sich der Aufzug befindet |
| 1    | 1    | 1    | Anf_oben_unten | Anforderung von oben und unten                                 |
| 1    | 1    | 0    | Anf_unten      | Anforderung aus Stockwerken unterhalb des Aufzugs              |
| 1    | 0    | 1    | Anf_oben       | Anforderung aus Stockwerken oberhalb des Aufzugs               |

Die Prioritäten sind von oben nach unten fallend, d.h. 'Tür offen' hat höchste Priorität.

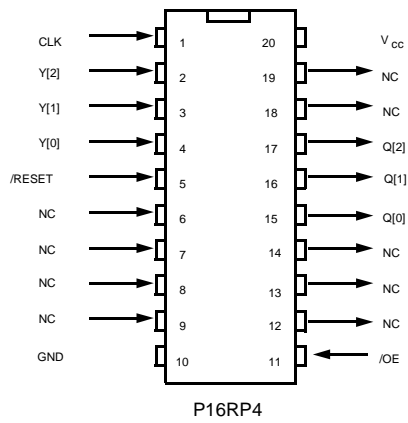
- 1.1 Entwerfen Sie ein PLPL-Programm zur Beschreibung des Schaltnetzes. Zur Realisierung der Schaltung verwenden Sie ein PAL vom Typ P16P8. Die Anschlußbelegung ist durch Tabelle 2 im Anhang vorgegeben.
- 1.2 Bauen Sie das Schaltnetz auf und testen Sie es.
2. Entwerfen Sie einen Speicher-Automaten mit einem PAL des Typs P16RP4 zur Steuerung von Motor und Tür des Aufzugs, der nur das Signal Y als Eingangsvariable kennt. Dadurch wird das Schaltwerk 'stockwerksneutral', d.h. es arbeitet unabhängig von der Anzahl der Stockwerke.
- 2.1 Zeichnen Sie den Zustandsgraphen. Dabei soll bei gleichzeitiger Anforderung von oben und unten das obere Stockwerk stets zuerst bedient werden. Daß Personen in unteren Stockwerken durch diese Strategie u.U. zu sehr langem Warten verurteilt sind soll hier nicht interessieren.
- 2.2 Stellen Sie die Schaltwerkstafel auf und kodieren Sie die Zustände.
- 2.3 Entwerfen Sie ein PLPL-Programm zur Realisierung des Schaltwerkes. Vergessen Sie die Rücksetzmöglichkeit nicht! Die Anschlußbelegung des PALs entnehmen Sie Tabelle 3 des Anhangs.
- 2.4 Bauen Sie das gesamte Steuerwerk auf und testen Sie es.
3. Modifizieren Sie das Schaltwerk so, daß der Aufzug bei gleichzeitiger Anforderung von oben und unten in der bisherigen Richtung weiterfährt.

Anschlußbelegung des Protoboards

|    | Belegung         |
|----|------------------|
| A0 | AUF              |
| A1 | AB               |
| A2 | TÜR              |
| A3 |                  |
| A4 | TAKT             |
| A5 | E3               |
| A6 | E2               |
| A7 | E1               |
| B0 | TG               |
| B1 | ANF3             |
| B2 | ANF2             |
| B3 | ANF1             |
| B4 |                  |
| B5 |                  |
| B6 |                  |
| B7 |                  |
| C0 |                  |
| C1 |                  |
| C2 |                  |
| C3 |                  |
| C4 |                  |
| C5 |                  |
| C6 |                  |
| C7 |                  |
| D0 |                  |
| D1 |                  |
| D2 |                  |
| D3 |                  |
| D4 |                  |
| D5 |                  |
| D6 |                  |
| D7 |                  |
| E0 |                  |
| E1 |                  |
| E2 |                  |
| E3 |                  |
| E4 |                  |
| E5 |                  |
| E6 |                  |
| E7 | Reset (Act. Low) |
| F0 |                  |
| F1 |                  |
| F2 |                  |
| F3 |                  |
| F4 |                  |
| F5 |                  |
| F6 |                  |
| F7 |                  |
| G0 |                  |
| G1 |                  |



Schaltnetz Aufzugsteuerung



Schaltwerk Aufzugsteuerung

## Fragen L-3

1. Grundlagen-Fragen siehe L-1, L-2
2. Wie können mit einem GAL/PAL sowohl Schaltnetze als auch Schaltwerke realisiert werden? Ist beides innerhalb desselben Bausteins möglich?
3. Was ist der Unterschied zwischen einem asynchronen und synchronen Schaltwerk bezügl. der Taktung? Können beide per PAL/GAL realisiert werden?
4. Wie entwirft man ein Schaltwerk, in welchen Schritten?
5. Welche Darstellungsmöglichkeiten für das Verhalten eines Schaltwerks kennen Sie? Wie äußert sich die gewählte logische Grundstruktur in diesen Darstellungen?
6. Welche Grundstruktur realisiert Ihr Aufzugswerk?
7. Sie sollten beim Testat in der Lage sein, ein Werk mit bis zu 8 Zuständen zu entwerfen!
8. Was ist der Unterschied zwischen Zustand und Zustandscodierung? Nach welchen Kriterien wird letztere vorgenommen?
9. Welcher PLPL-Sprachkonstrukt unterstützt besonders den Schaltwerksentwurf?
10. Können Hazards oder Races in PAL/GAL-Schaltwerken auftreten?