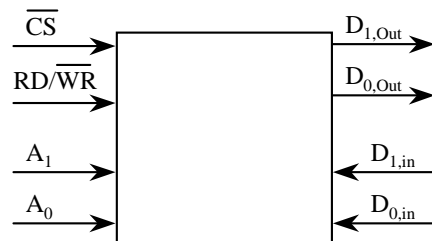


# 4. Übung

## Vorlesung Rechnerstrukturen WS 99/2000

---

1. Entwerfen Sie einen Speicherbaustein mit folgendem Anschlußschema:



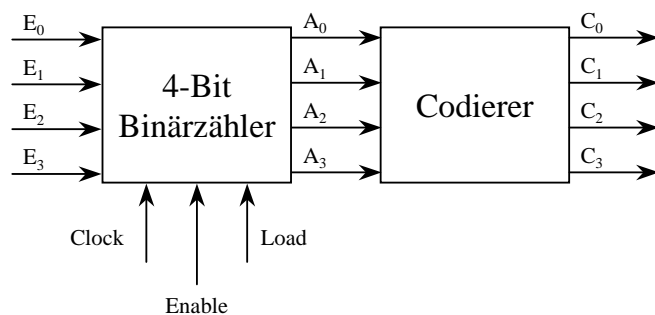
Die Anschlüsse verhalten sich wie aus der Vorlesung bekannt. Der Einfachheit halber gebe es getrennte, unidirektionale Datenleitungen für Ein- und Ausgabe und nicht wie bei handelsüblichen Bausteinen eine gemeinsame, bidirektionale Datenleitung. Beachten Sie, daß kein explizites Clocksignal bei der Konstruktion des Bausteins verwendet werden soll.

1.a. Überlegen Sie zunächst, wie Sie das Clocksignal für die Flip-Flops aus den im Diagramm dargestellten Eingangsleitungen erzeugen können. Welche Taktflanken der Signale sind in Ihrem Design ausschlaggebend, d.h. zu welchen Zeitpunkten werden die Daten von  $D_{in}$  zum Speichern in den Baustein übernommen bzw. wann werden die Daten an  $D_{out}$  zur Verfügung gestellt? Zeichnen Sie ein Timingdiagramm, in dem Sie einen Lese- und einen Schreibvorgang zeitlich darstellen. Markieren Sie dabei die eben besprochenen Datenübernahmezeitpunkte. Skizzieren Sie ferner in Hinblick auf benötigte Setup- und Hold-Zeiten, wann welche Signale zu einer korrekten Funktion des Bausteins stabil bleiben müssen. 5 P

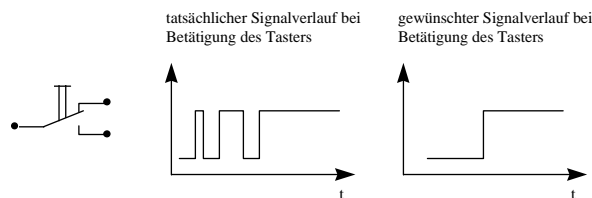
1.b. Entwerfen Sie den Speicherbaustein. Verwenden Sie in Ihrer Realisierung JK-Master/ Slave-Flip-Flops. 8 P

2. Beim Zählen im Binärcode (wie z.B. bei dem Binärzähler aus Aufgabenblatt 3) wird bei einigen Zustandsübergängen der Logikpegel mehrerer Ausgangssignale gleichzeitig geändert. Beispielsweise ändern beim Übergang von 0111 auf 1000 *alle* Ausgänge ihren Pegel. Es erweist sich —beispielsweise zur Vermeidung statischer Hazards— häufig als günstig, wenn sich bei jedem Zustandsübergang *genau ein* Bit der Ausgangssignale ändert.

Entwerfen Sie einen Codierer nach unten skizzierten Modell, der an die Ausgänge  $A_3..A_0$  eines 4-Bit Binärzählers angeschlossen werden kann. Die codierte Ausgabe  $C_3..C_0$  des Schaltnetzes soll —wie oben beschrieben— bei jedem Clock-Impuls am Binärzähler in *genau einem* Bit wechseln.



- 2.a. Schlagen Sie einen geeigneten Code vor, indem Sie eine Wahrheitstabelle angeben, die den aufeinanderfolgenden Zuständen  $A_3..A_0$  des Binärzählers jeweils eine codierte Zahl  $C_3..C_0$  zuordnet. 4 P
- 2.b. Entwerfen Sie ein Schaltnetz für die von Ihnen vorgeschlagene Codierung. 3 P
3. Das in der Vorlesung diskutierte Timingdiagramm für das R/S-Latch weist einen inkonsistenten Zustand aus, bei dem für kurze Zeit beide Ausgangssignale  $Q$  und  $\bar{Q}$  log. 0 sind. Die Existenz dieses Zustandes soll in der Übung praktisch durch Aufbauen der Schaltung auf einem sogenannten Protoboard überprüft werden. Bereiten Sie sich auf die praktische Übung wie folgt vor:
- 3.a. Skizzieren Sie eine Logik, die an die Ausgänge des R/S-Latches angeschlossen wird und eindeutig die Existenz des o.g. inkonsistenten Zustandes nachweist. 3 P
- 3.b. Zur Ansteuerung der Schaltung können beispielsweise Taster verwendet werden. Hierbei ergibt sich ein Problem, das auf den mechanischen Aufbau solcher Taster zurückzuführen ist und als *prellen* bezeichnet wird. Bei der Betätigung des Tasters neigen die Kontakte dazu, mehrmals zu öffnen bzw. zu schließen, bis der Zustand stabil bleibt (siehe mittleres Bild der Skizze). Somit ist das Signal, das ein solcher Taster liefert, oft nicht unmittelbar zur Ansteuerung von Digitalschaltungen geeignet. Schlagen Sie eine Schaltung zur Entprellung eines Tasters vor, so daß ein sauberes Signal gemäß dem rechten Bild der Skizze am Ausgang zur Verfügung steht. Gegeben ist hierbei ein Taster mit einem Wechselkontakt. 6 P



Ausgegeben: 26.11.99

Abgabe: bis spätestens Donnerstag 2.12.99, 12:00 Uhr in V 118 (Sekretariat)