



Aufgabe 1: (Zyklische Shifter) (3 Punkte)

In der Vorlesung haben wir einen zyklischen Linksshifter mit variabler Shiftdistanz $b[k - 1 : 0]$ entworfen. Dazu haben wir k -viele zyklische Linksshifter konstanter Shiftdistanz benutzt. Diese konstanten Shifter haben einen Select-Eingang s der auswählt, ob sie shiften oder nicht. Das Select-Signal des i -ten¹ konstanten Shifters (dieser shiftet um 2^i Stellen) ist mit dem Bit $b[i]$ verbunden. Der Ausgang des i -ten konstanten Shifters heisst r^i .

Beweisen Sie folgendes Lemma aus der Vorlesung: $\forall i : r^i = \text{cls}(a, \langle b[i : 0] \rangle)$.

Aufgabe 2: (Halbdecoder) (3 + 3 Punkte)

In der Vorlesung haben wir die Funktion eines Halbdecoders definiert.

1. Geben Sie eine rekursive Konstruktion für einen n -Bit Halbdecoder an.
2. Beweisen Sie die Korrektheit ihrer Konstruktion.

Aufgabe 3: (Decoder) (3 + 3 Punkte)

Ein n -Decoder ist ein Schaltkreis mit Eingängen $x[n - 1 : 0]$ und Ausgängen $Y[2^n - 1 : 0]$, so dass für alle i gilt:

$$Y_i = 1 \iff \langle x \rangle = i.$$

1. Geben Sie eine rekursive Konstruktion eines n -Decoders an. Ihre Konstruktion sollte ein Delay haben, das logarithmisch in n ist.
2. Beweisen Sie die Korrektheit ihrer Konstruktion.

Aufgabe 4: (Einsparen eines Inkrementers) (4 + 3 Punkte)

Im Schaltkreis *dist*, der die Shiftdistanz im Shifter-Environment bestimmt, wird ein fünf Bit Inkrementer benutzt. Dieser Inkrementer wird bei uzyklischen Rechtsshifts benutzt.

Der Inkrementer kann aus dem *dist*-Schaltkreis entfernt werden. Dadurch erhält man natürlich in gewissen Fällen eine falsche Shiftdistanz. Dieser Fehler kann aber durch einen zusätzlichen Multiplexer in den Datenpfaden des Shifter-Environments korrigiert werden.

1. Welche Änderungen im Shifter-Environment sind notwendig, um korrekte Shifts ohne den Inkrementer zu erreichen?
2. Beweisen Sie, dass Ihr modifiziertes Shifter-Environment das gleiche Verhalten hat wie die ursprüngliche Konstruktion.

¹ $i \in \{0, \dots, k - 1\}$

Aufgabe 5: (Korrekt ausgerichtete Speicherzugriffe)**(4 Punkte)**

Beweisen Sie, dass Speicherzugriffe im DLX-Prozessor aligned sind. Dazu müssen Sie folgendes zeigen:

$$M(r(x, d)) = \text{bytes}_{br(x,d)}\text{word}(M, x)$$

Aufgabe 6: (Arithmetische Einheit)**(4 Punkte + 6 Bonuspunkte)**

In der Vorlesung haben wir Conditional-Sum-Addierer und Carry-Look-Ahead-Addierer eingeführt. Diese haben wir anschließend zu arithmetischen Einheiten erweitert (einschließlich *overflow* und *negative* Signale). Sei $D_{AU}(n)$ das maximale Delay der n -bit AU und sei $D_{AU}(s[1:0]; n)$ das Delay der beiden untersten Bits der Summe.

1. Beweisen Sie, dass für die Arithmetische Einheit mit dem Conditional-Sum-Addierer und beliebige $n \geq 2$ das Delay der zwei untersten Summenbits folgendermaßen abgeschätzt werden kann:

$$D_{AU}(s[1:0]; n) \leq D_{AU}(2).$$

2. Beweisen Sie das Gleiche für die arithmetische Einheit mit dem Carry-Lookahead-Addierer.