

Aufgaben 3.2

Bau einer 4-Bit ALU

In Logisim ist eine 4-Bit ALU (Arithmetic and Logic Unit) zu bauen, die nachfolgende Berechnungen – via die Selektoreingänge S auswählbar – ausführen kann:

- A or B, bitweise (für S = 0), stets mit Overflow Flag OF=0
- A and B, bitweise (für S = 1), stets mit Overflow Flag OF=0
- not A, bitweise (für S = 2), stets mit Overflow Flag OF=0
- not B, bitweise (für S = 3), stets mit Overflow Flag OF=0
- A plus B (A, B: 4-bit unsigned integer) (für S = 4), mit Overflow Flag OF
- A minus B (A, B: 4-bit unsigned integer) (für S = 5), mit Underflow Flag OF
- 2 mal A (für S = 6), mit Overflow Flag OF
- 5 mal A (für S = 7), mit Overflow Flag OF

A ist der erste 4-Bit Operand, B der zweite 4-Bit Operand. R ist das 4-Bit Resultat. OF repräsentiert das 1-Bit Flag Overflow für die Addition/Multiplikation und Underflow für die Subtraktion. S ist der Operations-Selektor. Die 7-Segment-Anzeigen zeigen Operanden und Resultat in hexadezimal an.

Hinweise zur Abgabe

Sie erhalten die Datei *teamname 4-Bit ALU.circ*, die zwingend zu verwenden ist. Insbesondere ist für die Abgabe zu beachten:

- Als Hauptschaltung muss das Schaltungsblatt *4-Bit ALU* gewählt sein. Dort ist ein "Teamname" Ihrer Wahl einzutragen und sind die Mailadressen aller Team-Mitglieder anzugeben.
- Nehmen Sie eine Umbenennung von *teamname 4-Bit ALU.circ* vor. "Teamname" ersetzen Sie mit dem gewählten Namen.
- In der Interface-Ansicht (Symbol mit Maus-Tip-Anzeige "*Aussehen der angezeigten Schaltung als Teilschaltung bearbeiten*") des Schaltungsblatts *4-Bit ALU* ist "Teamname" im Symbol mit Ihrem gewählten Namen zu ersetzen. Ansonsten sind hier keine weiteren Änderungen zugelassen.
- Im Schaltungsblatt *Interface* und in weiteren Schaltungsblättern für die Unterschaltungen realisieren Sie ihre 4-Bit ALU.

Weitere Hinweise

- Die Aufgabe wird (i.d.R.) in Vierergruppen realisiert. Drei Personen realisieren die 8 Unterschaltungen, die vierte Person ist für die Integration der Unterschaltungen in eine ALU verantwortlich.
- Die Schaltung soll auch getestet werden.
- Die Subsaltungen können unabhängig realisiert werden und müssen untereinander nicht optimiert sein.
- Subsaltungen können als *Logisim Libraries* eingesetzt werden (vgl. Help). Es ist dann darauf zu achten, dass *alle* Files in entsprechenden Ordnern abgegeben werden (Ganzes als *.zip, *.rar o.ä. abgeben).
- Die im Unterricht abgegebene ALU (Skript) darf nicht verwendet werden. Auch Schaltungen aus den Bibliotheken *Arithmetik* und *Speicher* von Logisim sind hier nicht zugelassen, alle andern Schaltungen hingegen schon.

Schliesslich

- Bezeichnen Sie Anschlüsse und Bausteine.
- Stellen Sie Ihre Schaltungen übersichtlich dar. I.A. gilt: Je weniger Leitungsüberkreuzungen umso übersichtlicher.

→

Testing

Logisim bietet die Möglichkeit, die Korrektheit einer Schaltung mit Black Box Tests zu prüfen. Für diese Aufgabe wurde durch den Dozenten eine Testbench-Schaltung erstellt. Mittels eines Zählers werden alle möglichen Eingangsmuster erzeugt, welche an eine Referenzschaltung (Master) und Ihre Schaltung angelegt werden. Darauf werden die Resultate verglichen. Nach korrekt durchgelaufenem Vergleich ist die LED im Zustand grün und der Zählerstand steht auf 7FF. Andernfalls leuchtet die rote LED: Ihre Schaltung weist einen oder mehrere Fehler auf.

Test Configuration

Testing by comparison:

Insert Test Circuit via Logisim Library within red corners.

Ctrl-R to reset everything. Then Ctrl-K to start counter.

TestPassed shows green: Circuit OK, TestFailed shows red: not OK.

Test output into file via cmd-tool:

java -jar "logisim-generic-2.7.1.jar" testbench4BitALU.circ -tty table > testbench4BitALUResults.log

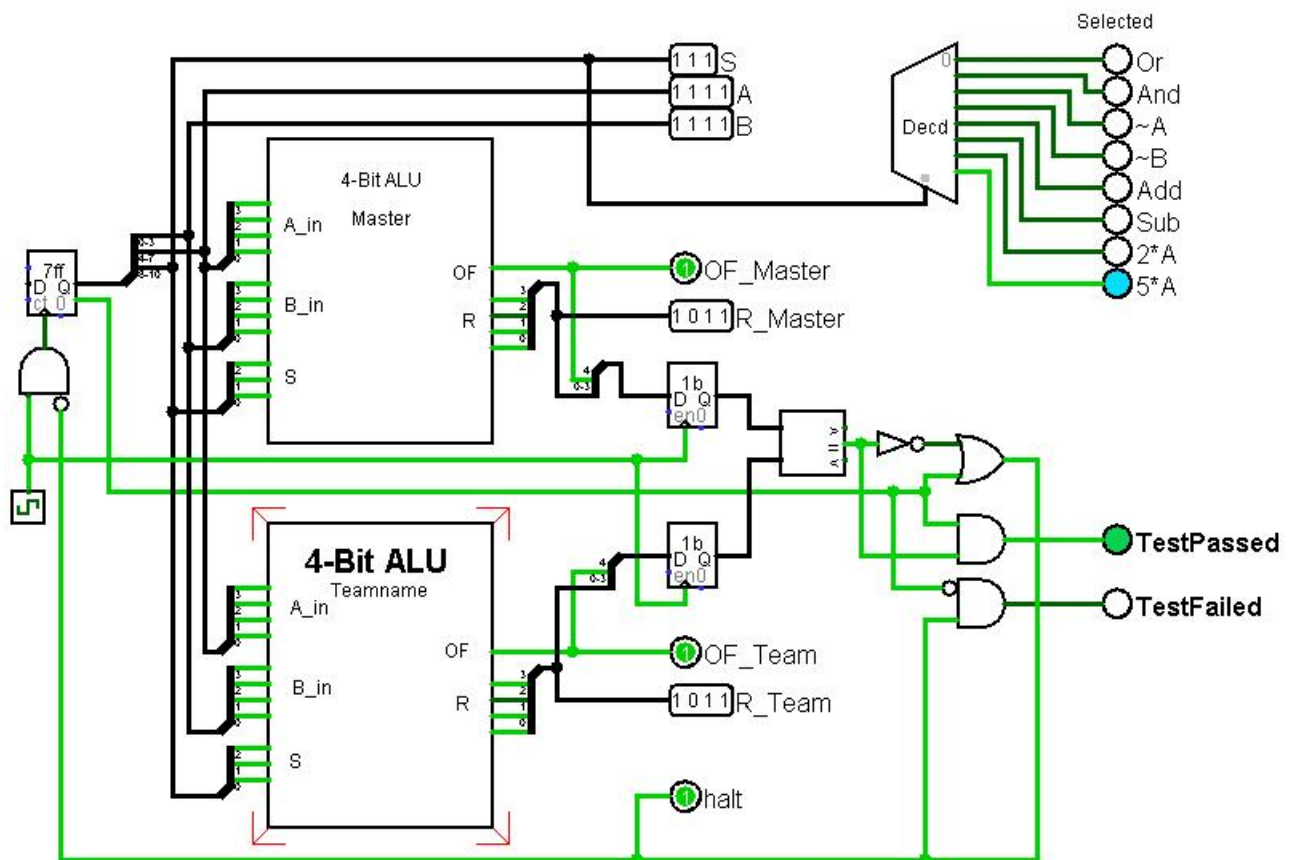


Fig. 1: Erfolgreich abgeschlossener Test

Obigen Test können Sie mittels der Datei *testbench4BitALU.circ* selbst vornehmen. Achten Sie darauf, dass sich im gleichen Verzeichnis wie die Testbench-Datei auch die Datei *chpALU.jar* befindet. Ihre Lösung in *teamname 4-Bit ALU.circ* bringen Sie in der Testbench-Datei via das Menü *Projekt>Bibliothek laden>Logisim Bibliothek...* in die Testbench und setzen die Blackbox Ihrer Lösung - links unter den üblichen Bibliotheken auffindbar - in die roten Ecken ein.

