ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG
INSTITUT FÜR INFORMATIK

Dr. Jürgen Ruf
Dipl.-Inf. Ilia Polian

Freiburg, 7. Mai 2002

2. Übungsblatt zur Vorlesung

Rechnerarchitektur

Aufgabe 1
Multiplizieren Sie die 8-bit Zahlen 01000110_2 und 11100111_2 nach der Methode von Booth.

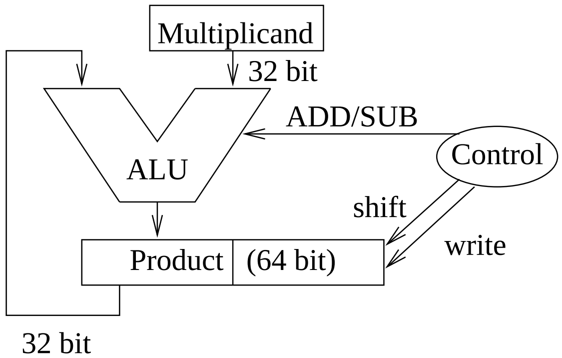
Aufgabe 2

a) Addieren Sie die Zahlen $0.110_2 (= \frac{1}{2} + \frac{1}{4} + \frac{0}{8})$ und 0.011_2 als Floating point-Zahlen (IEEE 754 Standard, 32 Bit).

b) Verdoppeln Sie die Zahl 0.11011_2 (Multiplikation mit 2) im IEEE Standard.

c) Gibt es mehr normalisierte oder mehr nicht-normalisierte Floating point-Zahlen in diesem Standard?

Aufgabe 3
Geben Sie Boolesche Gleichungen für die Steuer-Signale der ALU sowie der Steuer-Signale 'shift right' und 'write' des 64-Bit-Registers einer Hardware-Realisierung des Methode Booth an. Sie dürfen davon ausgehen, dass die beiden Bits y_i und y_{i+1} gegeben sind.



Aufgabe 4
Eine Maschine arbeitet mit 32-bit Wörtern. Es existiert ein LOAD-Befehl `LOAD Reg Addr`, welcher die Daten aus der Adresse `Addr` des Speichers in das Register `Reg` schreibt. Es sind nur Adressen erlaubt, die durch 32 teilbar sind. Ein zusätzlicher Befehl `LOADA` (load arbitrary) ist zu entwerfen, der Daten von einer beliebigen Stelle des Speichers auslesen kann (d. h. die o. g. Einschränkung auf durch 32 teilbare Adressen gilt nicht). Bitte geben Sie ein „Mikroprogramm“ (eine Folge von anderen Befehlen) an, das `LOADA` implementiert. Sie können für den Zugriff auf den Speicher `LOAD` verwenden und auf Registern Operationen mit beliebigen Befehlen durchführen (arithmetische Befehle, Verzweigungen, `MOV`, ...). Bitte geben Sie die ggf. benötigten zusätzlichen Ressourcen (wie Hilfsregister) an.

Aufgabe 5

Ein Rechner wird entworfen, welcher sieben Befehle `A, B, C, D, E, F` und `G` unterstützen soll. Es ist bekannt, daß die Befehle folgende Häufigkeiten haben werden:

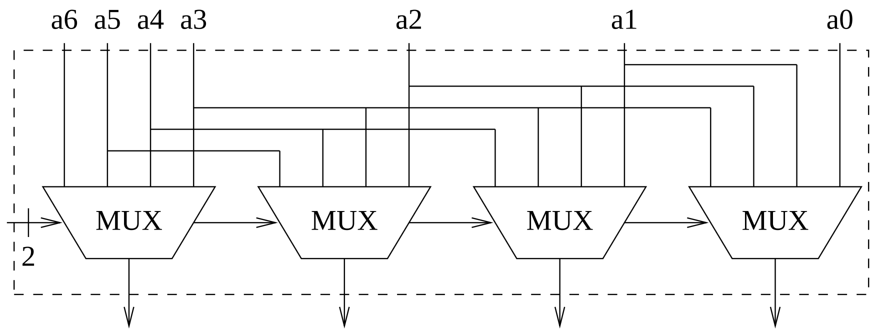
`A`	70%	`C`	5%	`E`	3%	`G`	1%
`B`	15%	`D`	4%	`F`	2%		

Es wird diskutiert, ob die Opcodes (Kodierungen) für die Befehle gleichlang sein sollen.

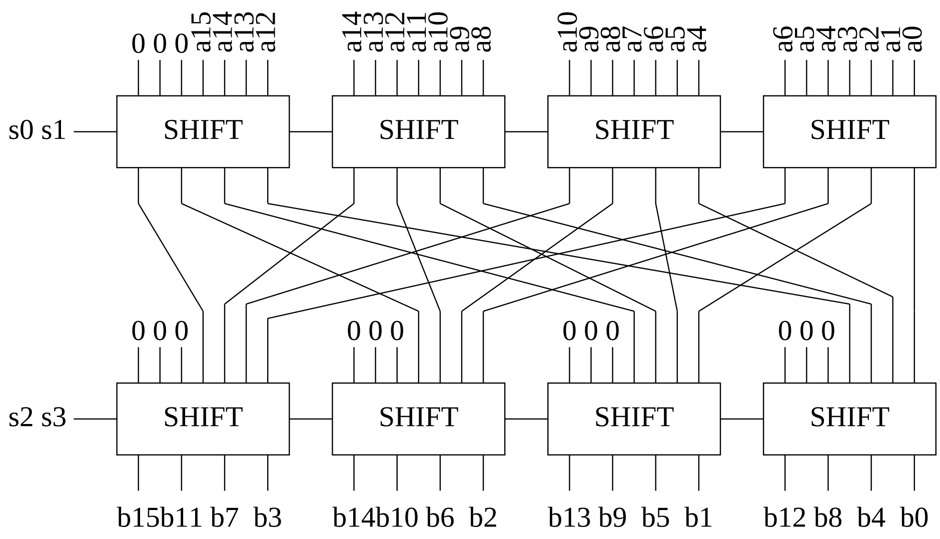
a) Die Kodierungen von allen Befehlen müssen gleich lang sein (die Länge muß keine Zweierpotenz sein). Geben Sie eine mögliche Kodierung für alle Befehle an. Wie lang ist die Kodierung eines Programms aus 100 Befehlen mit Häufigkeiten gemäß obiger Tabelle?

b) Die Kodierungen von verschiedenen Befehlen dürfen unterschiedliche Länge haben. Allerdings darf keine Kodierung mit der Kodierung eines *anderen* Befehls anfangen. Ist der Befehl `C` zum Beispiel als `11` kodiert, darf kein anderer Befehl als `110` oder `11101` kodiert sein (warum macht diese Beschränkung Sinn?). Bestimmen Sie eine Kodierung für die Befehle, so daß die Länge des Programms aus 100 Befehlen mit Häufigkeiten gemäß obiger Tabelle möglichst klein ist und geben Sie diese Länge an.

Hinweis: Hier ist der (16, 4) - Barrelshifter aus der Aufgabe 4 des letzten Übungsblattes abgebildet:



(a) SHIFT(2,4)



(b) Barrel-Shifter