

Aufgabe: Speichern und Rechnen Spielekonsole mit 8GB RAM

a)

Eine Binary Cell kann genau eine 0 oder 1 speichern, also 1 Bit.
Für einen Arbeitsspeicher mit 8 GB RAM gilt:

$$8 \text{ GB} = 8 * 10^3 \text{ MB} = 8 * 10^6 \text{ KB} = 8 * 10^9 \text{ Byte} = 6,4 * 10^{10} \text{ Bit}$$

Antwort: Man braucht $6,4 * 10^{10}$ Binary Cells, um einen Arbeitsspeicher von 8 GB zu realisieren.
Da jede Binary Cell genau ein RS-Flip Flop enthält, bräuhete man ebenso viele RS-Flip Flops

b)

Ein Demultiplexer

c)

Anzahl der Zeilen:

$$\frac{6,4 * 10^{10} \text{ Bit}}{64 \text{ Bit}} = 10^9$$

Anzahl der benötigten Leitungen n:

$$2^n = 10^9$$

$$\log_2(10^9) \approx 29.897$$

Antwort: Man bräuhete mindestens 30 Leitungen

Zum Rechenwerk

a) Addition von 2 64-Bit Zahlen

Man benötigt 1 Halbaddierer für die erste Addition und 63 Volladdierer für alle weiteren. Jeder Volladdierer beinhaltet 2 Halbaddierer. Somit benötigt man insgesamt **127 Halbaddierer**.

b) Subtraktion in der ALU

Belegung der Steuerleitungen:

z1: beliebig

z2: beliebig

m=1, d.h. Übertragsbit (c_{neu}) wird durchgeschaltet

u=0, v=1

$c_{alt}=1$

- Wenn u=0 und v=1 wird z2 bevor es in den Volladdierer geleitet wird immer negiert und zu $z2' = \text{NOT}(z2)$
- Eine Subtraktion kann bei Binärzahlen gemacht werden, indem man z1 und das Zweierkomplement von z2 addiert
- Das Zweierkomplement einer Binärzahl wird gebildet, indem man diese invertiert und 1 addiert. In diesem Fall wird das Zweierkomplement von z2 gebildet indem man zu $z2'$ das carrybit $c_{alt}=1$ addiert
- Übrig bleibt nur noch die Addition von z1 und dem Zweierkomplement von z2, womit die Subtraktion geleistet ist.
- Da m=1 wird die Subtraktion ganz normal ausgeführt und c_{neu} durchgeschaltet

Aufgabe: Optimierung einer Schaltung

a)

a	b	c	d	y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

b)

$$y = (a \wedge c) \vee (\neg a \wedge b \wedge c) \vee (\neg a \wedge c \wedge \neg d) \vee \neg(c \vee d)$$

c)

