University of the Saarland
Department 6.2 - Informatik
Prof. Dr. W.J. Paul

Computer Architecture II - WS 05/06

(due: Monday, 7.11.2005)

Excercise 1: (Two's Complement Numbers) **(10 points)**

In the lecture we defined the representation of two's complement numbers as follows:

$$a, b \in \{0,1\}^n$$

Def:

$$[a] \;=\; -a_{n-1} \cdot 2^{n-1} + \langle a[n-2:0] \rangle$$
$$[a] \in \{-2^{n-1}, \ldots, 2^{n-1}-1\} = T_n$$

To be proven:

$$[a_{n-1}\; a] + [b_{n-1}\; b] \in T_{n+1}$$

Excercise 2: (Booth Recoding: Sign Bits) **(10 points)**

To avoid summing of negative numbers C_{2j} we consider positive numbers E_{2j} instead:

$$\begin{array}{ll} E_{2j} = C_{2j} + 3 \cdot 2^{n+1} & \quad j > 0 \\ E_0 \;= C_0 + 4 \cdot 2^{n+1} & \end{array}$$

The corresponding binary representations are:

$$\begin{array}{ll} e_{2j} = bin_{n+3}(E_{2j}) & \quad j > 0 \\ e_0 \;= bin_{n+4}(E_0) & \end{array}$$

In the lecture we argued that we can compute the binary representations of E_{2j} and E_0 by:

$$\begin{array}{ll} \langle e_{2j} \rangle = \langle 1\, \overline{s_{2j}}\, (d_{2j} \oplus s_{2j}) \rangle + s_{2j} & \quad j > 0 \\ \langle e_0 \rangle \;= \langle \overline{s_0}\, s_0\, s_0\, (d_0 \oplus s_0) \rangle + s_0 & \end{array}$$

To be proven: Correctness of the computation algorithm for $j > 0$ and $j = 0$.

Excercise 3: (Booth Recoding: The Last Sign Bit) **(5 points)**

In the construction of the partial products for multiplication (using Booth Recoding) we incorporated the sign bits s_{2j-2} into the g_{2j} instead of adding them to g_{2j-2} directly:

$$\begin{array}{ll} g_{2j} = (e_{2j}\, 0\, s_{2j-2}) & \quad j > 0 \\ g_0 \;= (e_0\, 0\, 0) & \end{array}$$

To be proven: $s_{2m'-2} = 0$, i.e. we dont need to perform an additional summation in order to add the last sign bit.

University of the Saarland
Department 6.2 - Informatik
Prof. Dr. W.J. Paul

Computer Architecture II - WS 05/06

(due: Monday, 7.11.2005)

Excercise 4: (Weight of Nodes in the Addition Tree) **(10 points)**

We define the weight $w(v)$ of a node v in the addition tree of multipliers by:

- For leaves:

$$w(v) = \begin{cases} c & \text{for 3/2 adders} \\ c+1 & \text{for 4/2 adders} \end{cases}$$

- For an inner node v with left father x and right father y:

$$\begin{aligned} L(v) &= w(x) \\ R(v) &= w(y) \\ w(v) &= L(v) + R(v) \end{aligned}$$

To be proven by induction on the level l in the addition tree: For two nodes v and v' on the same level l, such that the node v is left of v', we have:

$$w(v) \leq w(v')$$