CMPUT 229 - Quiz # 4 - Fall 2010

Name: Solution

Question 1 (100 points): A traditional multiplication algorithm tests each bit of the multiplier and performs the addition accordingly. The figure below contains the Wikipedia description of the Booth algorithm (this is the simpler version that does not work with the largest negative number).

> Booth's algorithm involves repeatedly adding one of two predetermined values *A* and *S* to a product *P*, then performing a rightward arithmetic shift on *P*. Let **m** and **r** be the multiplicand and multiplier, respectively; and let *x* and *y* represent the number of bits in **m** and **r**.
>
> 1. Determine the values of *A* and *S*, and the initial value of *P*. All of these numbers should have a length equal to (*x* + *y* + 1).
>   1. A: Fill the most significant (leftmost) bits with the value of **m**. Fill the remaining (*y* + 1) bits with zeros.
>   2. S: Fill the most significant bits with the value of (−**m**) in two's complement notation. Fill the remaining (*y* + 1) bits with zeros.
>   3. P: Fill the most significant *x* bits with zeros. To the right of this, append the value of **r**. Fill the least significant (rightmost) bit with a zero.
> 2. Determine the two least significant (rightmost) bits of *P*.
>   1. If they are 01, find the value of *P* + *A*. Ignore any overflow.
>   2. If they are 10, find the value of *P* + *S*. Ignore any overflow.
>   3. If they are 00, do nothing. Use *P* directly in the next step.
>   4. If they are 11, do nothing. Use *P* directly in the next step.
> 3. Arithmetically shift the value obtained in the 2nd step by a single place to the right. Let *P* now equal this new value.
> 4. Repeat steps 2 and 3 until they have been done *y* times.
> 5. Drop the least significant (rightmost) bit from *P*. This is the product of **m** and **r**.

1. (**50 points**) For each of the pair of values for **m** and **r** below, indicate the number of shift and addition operations that would be performed by a traditional multiplication algorithm and by the Booth algorithm.

	m	**r**	Traditional		Booth	
			Shifts	Additions	Shifts	Additions
(a)	0x05	0x7E	8	6	8	2
(b)	0x33	0x55	8	4	8	8

2. (**50 points**) Assume that a shift operation costs 1 time unit and that an addition costs 10 times units. For each of the two operations listed above, which of the two algorithms is faster and by how much?

(a) time(trad) = 8 + 60 = 68;
time(Booth) = 8 + 20 = 28
⇒ Booth is 68/28 = 2.4 times faster than traditional.

(b) time(trad) = 8 + 40 = 48;
time(Booth) = 8 + 80 = 88
⇒ Traditional is 88/48 = 1.8 times faster than traditional.