Lab session #09

November 28, 2024

Michelangelo Barocci, michelangelo.barocci@polito.it

Politecnico di Torino,

PhD Student @ EDA Group - DAUIN

Dpt. of Computer and Control engineering (DAUIN)

Square root approximation circuit

```
read(a);
read(b);
t1 = abs(a);
t2 = abs(b);
x = max(t1, t2);
y = min(t1, t2);
t3 = x*0.125;
t4 = y*0.5;
t5 = x - t3;
t6 = t4 + t5;
t7 = max(t6, x)
res = t7;
```

- Design a circuit for approximating the result of a square root such that *res* ≅ $\sqrt{(a^2+b^2)}$, following the formula suggested in the lab guideline.
- Use the schedule in the picture on the right, which has the purpose of limiting some resources (e.g the registers)
- Input signals are 8-bit signed *a* and *b* and a *start* signal
- Output should be 9-bit *res* and a signal *ready*
- Resort to HLSM and FSM-D design

- Design a testbench to validate your design on relevant cases

Sequential multiplier

				a_3	a_2	a_1	a_0	multiplicand
$\times$				b_3	b_2	b_1	b_0	multiplier
				a_3b_0	a_2b_0	a_1b_0	a_0b_0	
			a_3b_1	a_2b_1	a_1b_1	a_0b_1		
		a_3b_2	a_2b_2	a_1b_2	a_0b_2			
$+$	a_3b_3	a_2b_3	a_1b_3	a_0b_3				
y_7	y_6	y_5	y_4	y_3	y_2	y_1	y_0	product

- Design a circuit for calculating the product of two 8-bit input signals A_{in} and B_{in} by using the Add-and-Shift method.
- Remember to use registers that store the input values A_{in} and B_{in} into two signals *A* and *B* so to avoid unexpected behaviour during the multiplication
- "Intelligently" shift the partial products *P* as described in the lab guideline.
- Output signals should be a 16-bit signal *res* with the result of the multiplication, and a *ready* signal that indicates the readiness of the output
- Design the circuit using FSM-D and HLSM approach.

- Design a testbench to validate your design on relevant cases