ELEC-311
Project 3
Adder-Subtractor

Charles Pittman

October 29, 2013

1 Objective

- Design a combinational logic circuit that adds, or subtracts, two 4-bit 2's complement numbers
- Describe the circuit using VHDL

2 Discussion

The circuit to be constructed was a 2-bit comparator. A 2-bit comparator takes two 2-bit numbers, $P = P_1P_0$ and $Q = Q_1Q_0$, and produces an output, $GT = 1 \iff P > Q$. The truth table describing this function is shown in Table **??**. This can be displayed more concisely as a summation of min-terms: $\sum_m(P_1, P_0, Q_1, Q_0) = (4, 8, 9, 12, 13, 14)$.

Using a Karnaugh map (Fig **??**), a minimal form of the function was found: $GT = P_0\overline{Q_1Q_0} + P_1P_0\overline{Q_0} + P_1\overline{Q_1}$. This function was then translated into the circuit shown in Fig **??**.

The NAND gate is considered to be a functionally complete set because any logic function can be created with only NAND gates. This is demonstrated graphically in Figure **??**. Using these conversions, the circuit was again translated into one using only NAND gates (Fig **??**), and truth table verified (not shown).

3 Results

X	Y	X	Y	X + Y	X - Y
3	1	0011	0001	0100	0010
1	5	0001	0101	0110	1100
-2	3	1110	0011	0001	1011
-2	-4	1110	1100	1010	0010
4	6	0100	0110	01010	1110

Table 1: Test vectors and expected values.

4 VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
use work.project3_gates.all;

entity circuit is
  port (X    : in  std_logic_vector(3 downto 0);
        Y    : in  std_logic_vector(3 downto 0);
```

```
        Z    : out std_logic_vector(3 downto 0);
        SEL  : in  std_logic;
        Cout : out std_logic);
end circuit;

architecture Structural of circuit is

  signal C : std_logic_vector(2 downto 0);
  signal S : std_logic_vector(3 downto 0);

begin

  x3 : ExclusiveOR port map(SEL, Y(3), S(3));
  x2 : ExclusiveOR port map(SEL, Y(2), S(2));
  x1 : ExclusiveOR port map(SEL, Y(1), S(1));
  x0 : ExclusiveOR port map(SEL, Y(0), S(0));

  fa3 : FullAdder port map(X(3), S(3), C(2), Cout, Z(3));
  fa2 : FullAdder port map(X(2), S(2), C(1), C(2), Z(2));
  fa1 : FullAdder port map(X(1), S(1), C(0), C(1), Z(1));
  fa0 : FullAdder port map(X(0), S(0), SEL, C(0), Z(0));

end Structural;
```

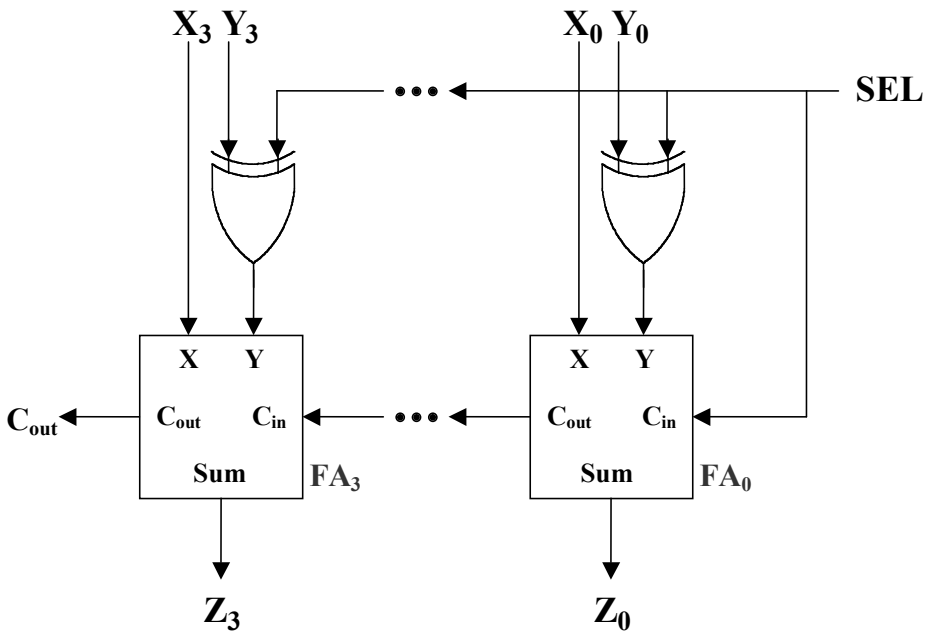

Figure 1: 4-bit adder/subtractor.