CS2310: Foundations of Computer System Design Lab
July-November 2021
Shania Mitra

AIM:

To build a simple 16-bit HACK Computer using a bottom-up approach assuming that only NOT gate, 2-input AND, OR and NAND gates are available, using the Nand2Tetris hardware emulator.

APPROACH:

1. In the first step we build the following fundamental encoders and decoders:
    a. 4-to-2 bit encoder
    b. 4-to-2 bit priority encoder
    c. 2-to-4 and 3-to-8 bit decoder
2. Next, we build the following:
    a. 2-bit 4-to-1 multiplexer
    b. 2-bit 4-to-1 priority multiplexer
    c. 2-bit 1-to-8 demultiplexer
    d. 8-bit comparator
    e. 8-bit parity generator and parity checker
3. Using the above-mentioned combinational circuits, we build the following adders, taking care of overflow detection logic,
    a. 16-bit ripple carry adder
    b. 4-bit CLA based 16-bit adder without using higher-level propagate and generate terms
    c. 4-bit CLA based 16-bit adder using higher-level propagate and generate terms
4. In the next step, we designed an 8-bit Wallace tree multiplier circuit using carry save addition (CSA) based 3-to-2 reducers followed by CLA with higher-level propagate and generatem terms.
    a. Unsigned integer multiplier
    b. Signed integer multiplier
5. Using the adders, multipliers and the previously built circuits, we build an 8-bit ALU that can perform the following:
    a. Unsigned addition
    b. Unsigned subtraction with X > Y
    c. Signed addition
    d. Signed subtraction
    e. Unsigned multiplication
    f. Signed multiplication
6. Additionally, a 16-bit floating point multiplier was built to handle the following format of floating point numbers:
    a. Sign bit
    b. Biased Exponent: 8 bits
    c. Mantissa Fraction: 7 bits

7. In the next step, we implement a HACK ALU that generates the outputs Zr and Ng for each of the 18 possible combinations of control signals, i.e., Zx, Nx, Zy, Ny, f, No.
8. Finally, we build a HACK Computer based on the HACK CPU. The computer consists of a 64x16-bit memory. Locations 0 to 15 in memory are reserved. Locations 16 to 63 can be used to store data and instructions.

RESULTS AND CONCLUSIONS:

To test the HACK Computer, the following set of instructions were executed on the Computer:

1. d = a + b - c (All operands are unsigned)

2. if (a > b) then c = a - b else c = b - a (All operands are unsigned)

3. int i = 1
   int sum = 0
   while (i < 100)
   { sum = sum + i;
   i = i + 1;
   }

The outputs were consistent with the expected output.
