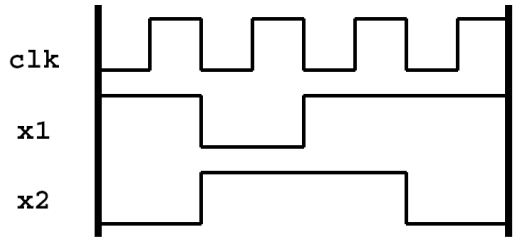
Assignment 1

Due June 5, 13:59

Important: Late submissions will NOT be accepted. Please submit a hardcopy of your solutions in the ECE 466 **drop-box** (ELW, second floor) and your SystemC code via the ECE 466 **CourseSpaces** webpage, following the submission guidelines posted on the course website.

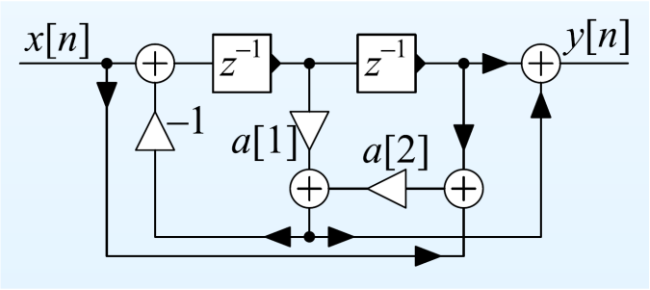
1. *[10 points]* Consider the SystemC code and the input waveforms of **clk**, **x1**, and **x2** shown below. Draw and briefly explain the corresponding output waveforms of **y1**, **y2**, and **y3**.




```
SC_MODULE (example) {

        sc_in <sc_logic> x1, x2;
        sc_out <sc_logic> y1, y2, y3;
        sc_in_clk clock;

        sc_signal <sc_logic> s;                 // signal "s"

        void example_process1() {
          sc_logic v;                           // variable "v"
          v = ~s.read();                        // ~ means NOT
          y1.write(x1.read() ^ v);              // ^ means XOR
        }

        void example_process2() {
          sc_logic u;                           // variable "u"
          u = ~x2.read();                       // ~ means NOT
          y2.write(s.read() & u);               // & means AND
        }

        void example_process3() {
          while(1) {
             s.write(x1.read() ^ x2.read());    // ^ means XOR
             y3.write(~s.read());               // ~ means NOT
             wait();
          }
        }
```

```
        SC_CTOR (example) {

              SC_METHOD(example_process1); sensitive << x1 << s;
              SC_METHOD(example_process2); sensitive << x2 << s;
              SC_CTHREAD(example_process3, clock.pos());

              s.write(SC_LOGIC_0);              // s initially 0
              y1.initialize(SC_LOGIC_0);        // y1 initially 0
              y2.initialize(SC_LOGIC_0);        // y2 initially 0
              y3.initialize(SC_LOGIC_0);        // y3 initially 0

        }

};
```

2. *[20 points]* Consider the digital filter shown below, where $\mathbf{z^{-1}}$ means a delay of 1 clock cycle. In addition to the **clock** and **reset** inputs, the filter has one `float`-type input **x** and one `float`-type output **y**. Assume that the (multiplication) coefficients have the constant values $\mathbf{a_1 = 0.5}$ and $\mathbf{a_2 = 0.25}$, with $\mathbf{a_0 = -1}$.
First, create individual SystemC modules for the `float`-type adder, multiplier, and clocked $\mathbf{z^{-1}}$ register with reset. Then, create instances of these modules as needed and connect them together to obtain a top-level structural description of our filter. Next, introduce a clocked stimulus generator and a clocked result monitor: your testbench must start with a reset, then send a 1-cycle unit pulse to **x** and observe the pulse response on **y**. Finally, write `sc_main()` with waveform tracing of **reset**, **clock**, **x**, and **y** over 20 data samples. Your code submission must be self-contained, i.e., ready to be compiled/executed on the lab workstations.
Submit an electronic copy of your solution via the ECE 466 **CourseSpaces** webpage (following the submission guidelines posted on the course website), and also submit a hardcopy in the ECE 466 **drop-box**.



3. *[10 points]* Given the same digital filter from the previous question, your task is to create its behavioral description using a single **SC_MODULE** with two processes: **SC_METHOD** (sensitive to **x** and the outputs of $\mathbf{z^{-1}}$ registers as needed, producing **y** and the inputs for $\mathbf{z^{-1}}$ registers as needed) and **SC_CTHREAD** (updating the outputs of $\mathbf{z^{-1}}$ registers at positive clock edges). This coding approach is similar to the way we describe Mealy-type FSMs (in our filter, **y** is directly affected by **x**).
Submit both an electronic copy (**CourseSpaces**) and a hardcopy (**drop-box**) of your complete solution, with the same stimulus generator, result monitor, and `sc_main()` as in **Question 2**.