LSU EE 4755 **Homework 5** Solution **Due: 20 November 2019**

Problem 1: Solve 2018 Final Exam Problem 3, in which the inferred hardware for a `misc` module is to be found (a) and the state of the event queue over time simulating `misc` (b) is to be found.

See the final exam solution at `https://www.ece.lsu.edu/koppel/v/2018/fe_sol.pdf`.

Problem 2 on next page.

Problem 2: Appearing below is a solution to Homework 4 Problem 1. Show the hardware that will be inferred for this module after some optimization. Show the `pop` module as a box.

- Clearly show all input and output ports.
- Please don't get parameters and ports confused.

Solution appears below. The solution uses enable signals on the registers, but it would also be correct to use an extra mux instead. Because optimization is applied the `wv-wk` term is shown as a constant, not as a subtraction unit. The `>>=1` is shown as a bit renumbering instead of has a shift unit.

```
module best_match
  #( int wv = 32, int wk = 10,  int wvb = $clog2(wv),      int wkb = $clog2(wk+1) )
   ( output logic [wvb:1] pos,  output logic [wkb:1] err,  output logic ready,
     input uwire [wv-1:0] val,  input uwire [wk-1:0] k,    input uwire start, clk );

   logic [wvb-1:0] curr_pos;
   logic [wv-1:0] sh_val;
   uwire [wkb-1:0] e;
   pop #(wk,wkb) p( e, k ^ sh_val[wk-1:0] );


   always_ff @( posedge clk )

     if ( start == 1 ) begin

        ready = 0;
        curr_pos = 0;
        sh_val = val;
        err = ~0;

     end else if ( !ready ) begin

        if ( e < err ) begin
          err = e;
          pos = curr_pos;
        end

        ready = curr_pos == wv-wk;
        curr_pos++;
        sh_val >>= 1;

     end

endmodule
```