6 Introduction to Computer Architecture (swm11)

The SystemVerilog code below is for a simple input/output processor that is focused on handling state changes on input and output pins.

```
typedef enum { CONST, JMP, BNZ, PUSH,
               SET, GET, INC, DEC } opcode;
typedef enum { RX, RY } regname;
typedef logic [31:0] word;
typedef logic [3:0]  immediate;
typedef logic [3:0]  pins;
typedef struct {opcode op;
                regname r;
                immediate i; } instruction;
module pio_proc
  (
   input  logic   clk,
   input  logic   rst,
   input  pins    inpins,
   output pins    outpins,
   output word    outw
   );
instruction imem[0:15] =
  '{'{op:CONST, r:RY, i:4'd0},  //  0
    '{op:GET,   r:RX, i:4'd3},  //  1
    '{op:DEC,   r:RX, i:4'd0},  //  2
    '{op:BNZ,   r:RX, i:4'd1},  //  3
    '{op:GET,   r:RX, i:4'd3},  //  4
    '{op:BNZ,   r:RX, i:4'd4},  //  5
    '{op:INC,   r:RY, i:4'd0},  //  6
    '{op:GET,   r:RX, i:4'd3},  //  7
    '{op:DEC,   r:RX, i:4'd0},  //  8
    '{op:BNZ,   r:RX, i:4'd6},  //  9
    '{op:INC,   r:RY, i:4'd0},  // 10
    '{op:GET,   r:RX, i:4'd3},  // 11
    '{op:GET,   r:RX, i:4'd3},  // 12
    '{op:BNZ,   r:RX, i:4'd10}, // 13
    '{op:PUSH,  r:RY, i:4'd0},  // 14
    '{op:JMP,   r:RX, i:4'd0}}; // 15
```

```
instruction ir;
immediate pc;
word rf [regname];
always_comb ir = imem[pc];
always_ff @(posedge clk or posedge rst)
  if(rst)
    pc <= 0;
  else begin
    case(ir.op)
      CONST:rf[ir.r] <= {28'b0, ir.i};
      JMP:  pc <= ir.i;
      BNZ:  pc <= (rf[ir.r]!=0) ? ir.i : pc+1;
      PUSH: outw <= rf[ir.r];
      SET:  outpins[ir.i[1:0]] <= rf[ir.r][0];
      GET:  rf[ir.r] <= {31'b0,inpins[ir.i[1:0]]};
      INC:  rf[ir.r] <= rf[ir.r]+1;
      DEC:  rf[ir.r] <= rf[ir.r]-1;
    endcase // case (ir.op)
    if((ir.op!=JMP) && (ir.op!=BNZ))
      pc <= pc+4'd1;
  end
endmodule // pio_proc
```

(*a*) How many clock cycles does it take to fetch, decode and execute an instruction? Explain your answer. [2 marks]

(*b*) What are the eight instructions and what function do they each perform? Give a short English explanation of each. [6 marks]

(*c*) The imem memory is initialised with a program. Produce commented pseudo code that represents the program. [6 marks]

(*d*) In the SystemVerilog module, what registers are not reset and will they cause the processor to malfunction? [3 marks]

(*e*) What is the meaning of the outw output? Is the instruction timing of the program important? [3 marks]