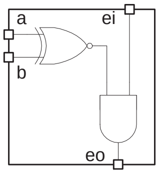
EE 3755 Homework 2 Due: 22 February 2002

The solutions to the problems below are in the form of schematic diagrams. None of the problems include technology mapping. In each diagram be sure to label the module ports as they are labeled in the diagram to the right.



Problem 1: Show the output, in the form of a schematic diagram showing gates, of the synthesizer inference step (before mapping and before optimization) for the module below (the solution to Homework 1, Problem 1). *Hint: This is really easy.*

```
module decode_2_to_4(x0,x1,x2,x3,a);
   input [1:0] a;
   output      x0, x1, x2, x3;

   not n0(a0n,a[0]);
   not n1(a1n,a[1]);

   and a0(x0,a1n,a0n);
   and a1(x1,a1n,a[0]);
   and a2(x2,a[1],a0n);
   and a3(x3,a[1],a[0]);

endmodule
```

Problem 2: Show the output of the synthesizer inference step for the module below (based on the solution to Homework 1, Problem 2). Show the adder as a module, not as individual gates.

```
module atoi_implicit(i,s);
   input [7:0] s;
   output [3:0] i;

   assign  i[3:0] = s[3:0] + ( s[6] ? 4'd9 : 4'd0 );

endmodule
```

Problem 3: For the module above, show how the synthesizer might optimize the inferred hardware (but not the adder). *Hint: using a pencil, draw four multiplexors, one for each bit.*

Problem 4: For the module below, show the output of the inference step and also show output after optimization. (Ignore technology mapping.) In the unoptimized version show the comparison operator as a module, if the comparison operator is present in the optimized version show it using gates. The optimized version should use one multiplexor, and the number of multiplexor inputs should be a power of 2. Do not make assumptions about how the module will be used.

```
module m1(x,s,a,b,c);
   input [1:0] s;
   input [31:0] a, b, c;
   output [31:0] x;

   assign x =
          s == 1 ? a :
          s == 2 ? b : c;

endmodule
```

Problem 5: For the module below (which is different in an important way than the one above), show the output of the inference step. (Ignore technology mapping.) In the unoptimized version show the comparison operator in the diagram. Do not make assumptions about how the module will be used.

```
module m2(x,s,v1,v2,a,b,c);
   input [1:0] s, v1, v2;
   input [31:0] a, b, c;
   output [31:0] x;

   assign x =
          s == v1 ? a :
          s == v2 ? b : c;

endmodule
```

Problem 6: For the module above, suppose the synthesis program optimizer has replaced the two multiplexors with a single multiplexor. Show a schematic of such a circuit. If the comparison operator is present show it as a module, not as individual gates. *Hint: Extra logic is needed at the multiplexor control input.*