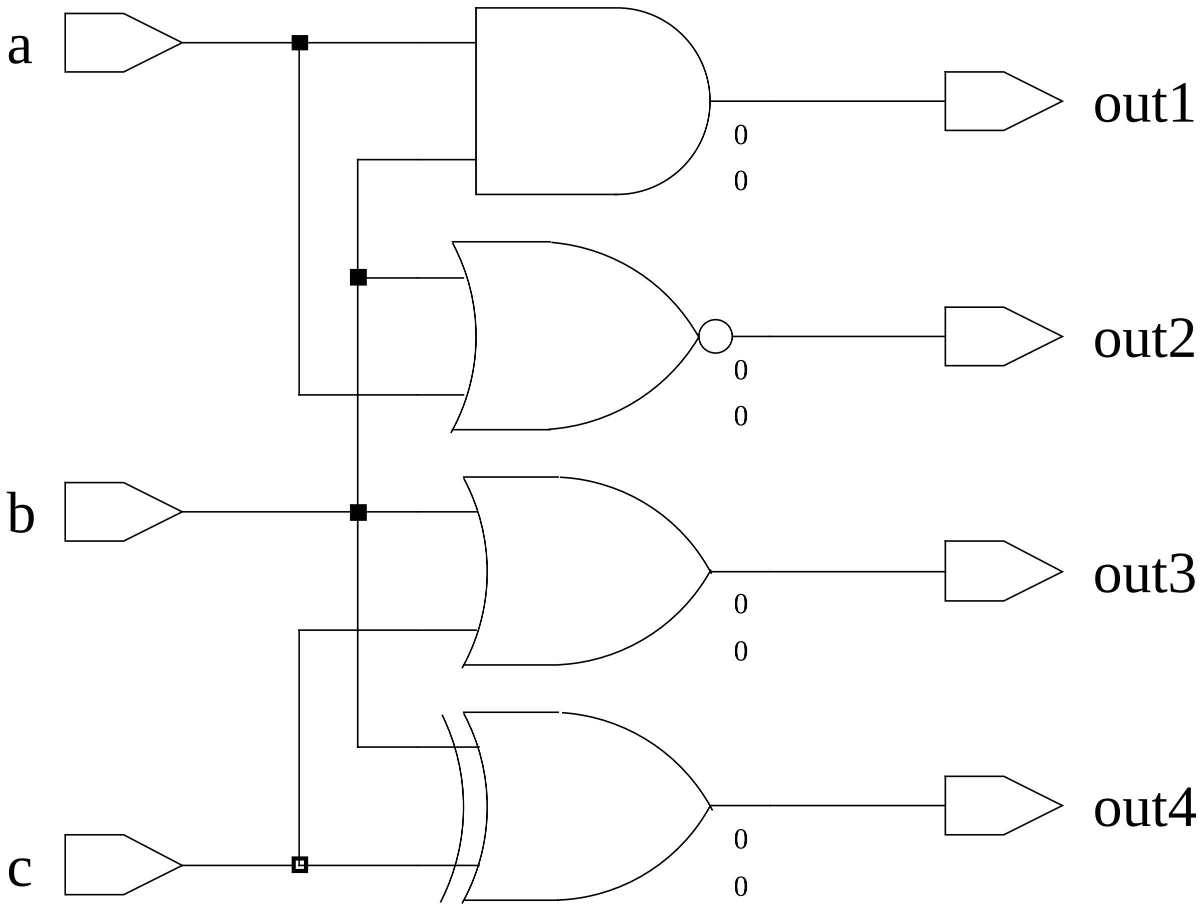
Concurrency

To model reality, VHDL processes certain statements concurrently.

Example:



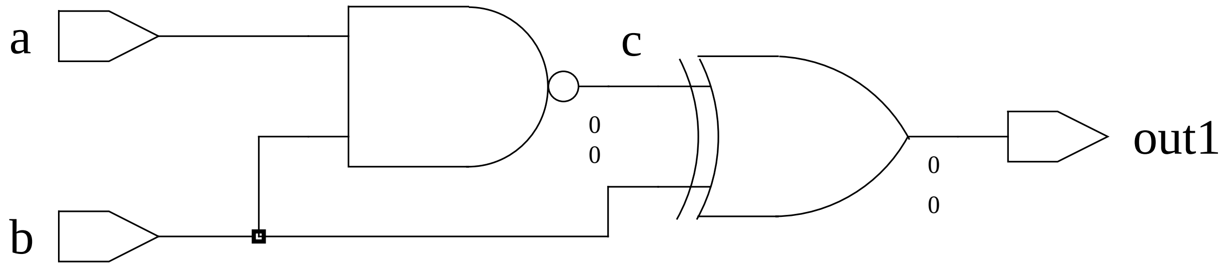
```
ARCHITETURE example of concurrent IS
BEGIN
  out1 <= a AND b;
  out2 <= a NOR b;
  out3 <= b OR c;
  out4 <= b XOR c;
END example;
```

Statement Activation

Signals connect concurrent statements.

Concurrent statements activate or “fire” when there is an event on a signal “entering” the statement.

Example:



```
ARCHITECTURE example OF concurrent IS
  SIGNAL c  :  std_logic;
  BEGIN
    c    <= a NAND b; --nand gate
    out1 <= c XOR b;  --xor gate
END example;
```

The NAND statement is activated by a change on either the a or b inputs.

The XOR statement is activated by a change on either the b input or signal c.

Note that additional signals (those not defined in the PORT clause) are defined in the architecture’s declarative area.

Concurrency Again

VHDL is inherently a concurrent language.

All VHDL processes execute concurrently.

Basic granularity of concurrency is the *process*.

Concurrent signal assignments as actually one-line processes.

```
   c    <= a NAND b;  --”one line process”
   out1 <= c XOR b;   --”one line process”
```

VHDL statements execute sequentially *within a process*.

```
ARCHITECTURE example OF concurrency IS
  BEGIN
    hmmm: PROCESS (a,b,c)
    BEGIN
     c    <= a NAND b;   --”do sequentially”
     out1 <= c XOR b;    --”do sequentially”
  END PROCESS hmmm;
```

How much time did it take to do the stuff in the process statement?

Concurrency

The body of the ARCHITECTURE area is composed of one or more concurrent statements. The concurrent statements we will use are:

- **Process - the basic unit of concurrency**
- **Assertion - a reporting mechanism**
- **Signal Assignment - communication between processes**
- **Component Instantiations - creating instances**
- **Generate Statements - creating structures**

Only concurrent statements may be in the body of the architecture area.

```
ARCHITECTURE showoff OF concurrency_stmts IS
BEGIN
------concurrent club members only----------
--BLOCK
--PROCESS
--ASSERT
--a <= NOT b;
--PROCEDURE
--U1:nand1 PORT MAP(x,y,z);  --instantiation
--GENERATE
------concurrent club members only----------
END showoff;
```