General VHDL Format

entity <name> **is port** {
<define signals to outside>
}

architecture <name> **of** <name> **is**
<define global types>
begin

<proc1> : **process** (sensitivity list)
<define local vars>
begin
<statements>
end process

end architecture

Entity for Det-101

```
library ieee;
use ieee.std_logic_1164.all;

--Define the “black box” for the object

entity det_101 is port {
  signal clk, s, reset:  in std_logic;
  signal q:              out std_logic
};
end det_101;
```

Architecture for Det-101

architecture version1 **of** det_101 **is**

   type state_type **is** (s0, s1, s2, s3);
   signal cur_state, next_state**:** state_type;

begin

   <state_logic process>

   <state_register process>

end version1;

Process State_Logic

```
state_logic: process (cur_state, s)
begin
  case cur_state is
    when s0 =>
      q <= '0';
      if s = '1' then
        next_state <= s1;
      else
        next_state <= s0;
      end if;
    when s1 =>
      q <= '0';
      if s = '1' then
        next_state <= s1;
      else
        next_state <= s2;
      end if;
```

Process State_Logic (cont)

```
when s2 =>
  q <= '0';
  if s = '1' then
    next_state <= s3;
  else
    next_state <= s0;
  end if;

when s3 =>
  q <= '1';
  if s = '1' then
    next_state <= s0;
  else
    next_state <= s2;
  end if;
end case;
end process state_logic;
```

Process State_Register

```
state_register:process (clk, reset)
begin
  if reset = '1' then
    cur_state <= s0;
  elsif rising_edge(clk) then
    cur_state <= next_state;
  end if;

end process state_register;
```