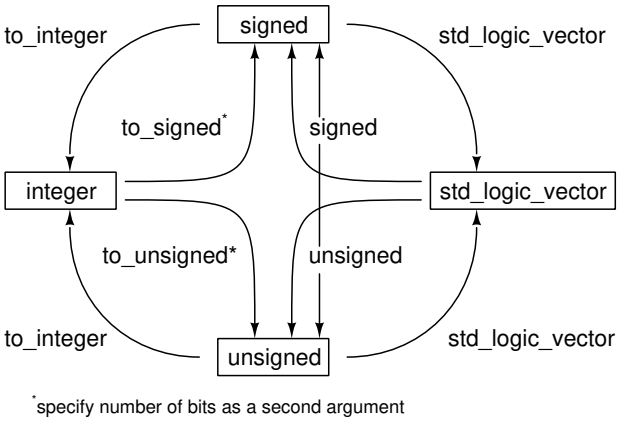
ELEX 2117 : Digital Techniques 2
2020 Winter Term

VHDL for Synthesis

entity, architecture

```
-- a comment

library ieee ;
use ieee.std_logic_1164.all ;
use ieee.numeric_std.all ;

entity entity_name is
  port (
     -- inputs and outputs, e.g.:
     a, b, c : in std_logic ;
     clock : in std_logic ;
     y : out std_logic
     ) ; -- no ';' after last port
end entity_name ;

architecture rtl of entity_name is
  -- internal signals, e.g.:
  signal u2, u2_next : unsigned(1 downto 0) ;
begin
  -- concurrent statements; order has no effect
end rtl  ;
```

type conversions

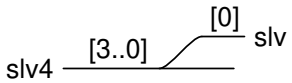


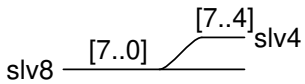
literals

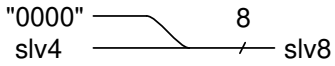
```
sl <= '0' ; -- std_logic
slv4 <= "0101" ; -- binary
slv8 <= x"0a" ; -- hex
u2 <= to_unsigned(1,2) ; -- decimal
```

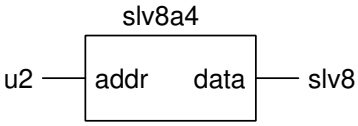
signals

```
signal sl : std_logic ;
signal slv4 : std_logic_vector (3 downto 0) ;
signal slv8 : std_logic_vector (7 downto 0) ;

-- initialized array (lookup table)

type slv8_type is array (natural range <>) of
  std_logic_vector(7 downto 0) ;
signal slv8a4 : slv8_type(0 to 3)
    := ( x"7E", x"30", x"6D", x"79" ) ;
```









concurrent assignments

```
-- slicing and concatenation
  sl <= slv4(0) ;
  slv4 <= slv8(7 downto 4) ;
  slv8 <= "0000" & slv4 ;

-- array element
  slv8 <= slv8a4(to_integer(u2)) ;

-- continous assignment
  y <= a and b ;

-- selected assignment
  y <= a when sl = '1' else b ;

-- tri-state output
  y <= a when sl = '1' else 'Z' ;

-- nested selected assignment
  y <= a when u2 = "00" else
       b when slv4 = "0001" else
       c ; -- always include this

--  selected assignment
  with slv4 select y <=
       a when "0001",
       b when "0010",
       c when others ; -- always!

-- flip-flop or register
  u2 <= u2_next when rising_edge(clock) ;

-- entity instantiation
  instance_name: entity work.entity_name2
    port map ( a, b, y ) ;
```