VHDL Type Conversion

Last updated 1/7/25

VHDL Type Conversion

- 2 primary types of conversions
  - Vector conversions
    - Fully synthesizable – all about interpretation
    - Similar to a type cast in programming
    - std_logic_vector <-> signed
    - std_logic_vector <-> unsigned
    - signed <-> unsigned
  - Integer conversions
    - Not synthesizable
      - compile time calculations
      - simulations
    - Requires a function to make the conversion
    - signed <-> integer
    - unsigned <-> integer

VHDL Type Conversion

- Vector conversions
  - SLV to signed

```
my_4bit_signed <= signed(my_4bit_slv);
```

  - SLV to unsigned

```
my_4bit_unsigned <= unsigned(my_4bit_slv);
```

  - signed to SLV

```
my_6bit_slv <= std_logic_vector(my_6bit_signed);
```

  - unsigned to SLV

```
my_8bit_slv <= std_logic_vector(my_8bit_unsigned);
```

  - unsigned to signed

```
my_12bit_signed <= signed(my_12bit_unsigned);
```

  - signed to unsigned

```
my_16bit_unsigned <= unsigned(my_16bit_signed);
```

VHDL Type Conversion

- Integer conversions
  - Integers can only convert to/from signed/unsigned
    - SLV has no numerical interpretation
  - Vector to integer conversions
    - signed to int

```
my_comparison_int <= to_integer(my_12bit_signed);
```

    - unsigned to int

```
my_limit_int <= to_integer(my_16bit_unsigned);
```

  - Integer to vector conversions require the number of bits to use in the vector
    - integer to signed

```
my_8bit_signed_sim_input <= to_signed(22, 8);
```

    - integer to unsigned

```
my_4bit_unsigned_sim_input <= to_unsigned(11, 4);
```

VHDL Type Conversion