HAUTE ÉCOLE
D'INGÉNIERIE
ET DE GESTION
DU CANTON
DE VAUD

Prof. Yann Thoma

Formal verification
Verification of a counter
CERN training - July 2025

Context

We are interested in verifying the correctness of a counter having the following interface :

```
entity counter is
generic (
    SIZE  : integer := 8;
    ERRNO : integer := 0);
port(
    rst_i      : in  std_logic;
    clk_i      : in  std_logic;
    d_i        : in  std_logic_vector(SIZE-1 downto 0);
    en_i       : in  std_logic;
    up_ndown_i : in  std_logic;
    load_i     : in  std_logic;
    value_o    : out std_logic_vector(SIZE-1 downto 0)
);
end counter ;
```

This counter has an enable for incrementing (and decrementing) and the possibility to load a value. Please note that the load has precedence over the enable, and that the `up_ndown_i` selects the operation (increment or decrement).
Propose a set of assertions that allows you to use formal verification to prove the correct behavior of this counter.
A second generic parameter `ERRNO` allows to inject errors in the design. Its behavior is the following :

1. When 0 the result is valid ;
2. When in the $[1, 4]$ interval, the result is invalid.

This generic parameter allows to test your assertions by trying various `ERRNO` values. Just modify the `counter.sby` file to test different values.