



```

ENTITY numarator IS
    GENERIC( delay: TIME := 10 ns);
    PORT(clock, reset, load: IN BIT;
        minus2: IN BIT;
        intrare_paralela: IN INTEGER;
        iesire: OUT INTEGER);
END numarator;

ARCHITECTURE behave OF numarator IS
    --declaratii de semnale, componente, etc
    BEGIN

        PROCESS( clock, reset)
            VARIABLE temp: INTEGER;
            BEGIN

                IF reset='0' THEN
                    temp := 0;
                ELSIF clock='1' AND clock'EVENT and clock'LAST_VALUE='0' THEN
                    IF load='1' THEN
                        temp := intrare_paralela;
                    ELSE
                        IF minus2='0' THEN
                            temp := temp-1;
                        ELSIF minus2='1' THEN
                            temp := temp-2;
                        ELSE
                            temp := temp;
                        END IF;
                    END IF;
                END IF;

                iesire <= temp AFTER delay;
            END PROCESS;

        END ARCHITECTURE behave;

```

```

ENTITY clk_gen IS
    GENERIC(t_high: TIME:=30 ns; t_period: TIME:=50 ns; t_reset: TIME:=10 ns);
    PORT(clock: OUT BIT:='1'; reset : OUT BIT);
END clk_gen;

ARCHITECTURE behave OF clk_gen IS
    BEGIN
        reset<='0', '1' AFTER t_reset;

        PROCESS
            BEGIN
                IF NOW > 10 us THEN
                    WAIT;
                END IF;
            END PROCESS;

```

```
        clock<='1', '0' AFTER t_high;
        WAIT FOR t_period;
    END PROCESS;
END ARCHITECTURE;
```

```
--      TESTBENCH
```

```
ENTITY test IS
```

```
END test;
```

```
ARCHITECTURE struct OF test IS
```

```
    COMPONENT numarator IS
        GENERIC( delay: TIME := 10 ns);
        PORT(clock, reset, load: IN BIT;
            minus2: IN BIT;
            intrare_paralela: IN INTEGER;
            iesire: OUT INTEGER);
    END COMPONENT;
```

```
    COMPONENT clk_gen IS
        GENERIC(t_high: TIME:=30 ns; t_period: TIME:=50 ns; t_reset: TIME:=10 ns);
        PORT(clock: OUT BIT:='1'; reset : OUT BIT);
    END COMPONENT;
```

```
    SIGNAL clock_s, reset_s, load_s : BIT;
    SIGNAL minus2_s : BIT;
```

```
    SIGNAL intrare_paralela_s: INTEGER := -5;
```

```
    SIGNAL iesire_s : INTEGER;
BEGIN
```

```
et1: numarator GENERIC MAP(delay => 5ns)
    PORT MAP ( clock => clock_s, reset => reset_s, load => load_s, minus2 => minus2_s, intrare_paralela =>
intrare_paralela_s, iesire => iesire_s);
```

```
et2: clk_gen GENERIC MAP (t_high => 40ns, t_period => 100ns, t_reset => 30ns) PORT MAP (clock => clock_s,
reset => reset_s );
```

```
load_s <='0', '1' after 240 ns, '0' AFTER 440 ns;
```

```
minus2_s <='0', '1' after 5 us;
```

```
intrare_paralela_s <= 100 AFTER 150 ns;
```

```
END struct;
```
