ELEC-311
Project 4
Synchronous Design

Charles Pittman

December 5, 2013

1 Objective

- Design and test a synchronous sequential circuit which implements a 2-bit binary counter.
- Describe the circuit using VHDL.

2 Discussion

	Inputs		PS		NS		Outputs	
mt	E	U	Q1	Q0	Q1+	Q0+	Z1	Z0
0	0	0	0	0	1	1	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	1	1	0
3	0	0	1	1	1	0	1	1
4	0	1	0	0	0	1	0	0
5	0	1	0	1	1	0	0	1
6	0	1	1	0	1	1	1	0
7	0	1	1	1	0	0	1	1
8	1	0	0	0	0	0	0	0
9	1	0	0	1	0	1	0	1
10	1	0	1	0	1	0	1	0
11	1	0	1	1	1	1	1	1
12	1	1	0	0	0	0	0	0
13	1	1	0	1	0	1	0	1
14	1	1	1	0	1	0	1	0
15	1	1	1	1	1	1	1	1

Table 1: Transition Table

$D_1 = Q_1^+ = \sum_m(0, 3, 5, 6, 10, 11, 14, 15)$
$D_0 = Q_0^+ = \sum_m(0, 2, 4, 6, 9, 11, 13, 15)$
$Q_1^+ = E'U'Q_1'Q_0' + U'Q_1Q_0 + E'UQ_1'Q_0 + UQ_1Q_0' + EQ_1$
$Q_0^+ = E'Q_0' + EQ_0$

3 VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;

library UNISIM;
use UNISIM.VComponents.all;

use work.project4_pkg.all;
```

```
entity counter is
  port (EL  : in  std_logic;  -- Active-low enable
        UP  : in  std_logic;  -- Active-high control (count up/down)
        CLK : in  std_logic;  -- Rising-edge trigger clock
        CLR : in  std_logic;  -- Active-high asynchronous clear
        Z   : out std_logic_vector (1 downto 0));  -- 2-bit number
end counter;

architecture Dataflow of counter is
  signal d        : std_logic_vector(1 downto 0);
  signal q        : std_logic_vector(1 downto 0);
  signal slow_clk : std_logic;

begin
  -- q next-state equations
  d(1) <= (not EL and not UP and not q(1) and not q(0)) or
          (not UP and q(1) and q(0)) or
          (not EL and UP and not q(1) and q(0)) or
          (UP and q(1) and not q(0)) or
          (EL and q(1));
  d(0) <= (not EL and not q(0)) or (EL and q(1));

  -- Stretch the clock input for slow_clk
  CDIV : CLK_DIV port map (CLK, slow_clk);

  -- D flip-flops to do the actual switching
  FF1 : FDC port map(q(1), slow_clk, CLR, d(1));
  FF0 : FDC port map(q(0), slow_clk, CLR, d(0));

  Z <= q;  -- Give Z the present state (for output)
end Dataflow;
```