FPGA

Lab Report # 6a

Hassan Ahmed
Hamza Shabir
Neha Chaudhary

6.1 Objectives

The objectives of this lab are:

- To design a block diagram using a clock divider and separate counter instances for seconds, minutes, and hours.
- To implement a behavioral model of a digital watch and its stimulus.
- To generate a synthesis model and test the second-digits (8-bit numbers) on LEDs of the Spartan 3E Starter Board.

6.2 Background

ModelSim is a multi-language HDL simulation environment by Mentor Graphics, for simulation of hardware description languages such as VHDL, Verilog and SystemC, and includes a built-in C debugger. ModelSim can be used independently, or in conjunction with Intel Quartus Prime, Xilinx ISE or Xilinx Vivado. The Spartan-3E family builds on the success of the earlier Spartan-3 family by increasing the amount of logic per I/O, significantly reducing the cost per logic cell. New features improve system performance and reduce the cost of configuration. These Spartan-3E FPGA enhancements, combined with advanced 90 nm process technology, deliver more functionality and bandwidth per dollar than was previously possible, setting new standards in the programmable logic industry. Because of their exceptionally low cost, Spartan-3E FPGAs are ideally suited to a wide range of consumer electronics applications, including broadband access, home networking, display/projection, and digital television equipment.

A digital clock is a type of clock that displays the time digitally (i.e. in numerals or LEDs etc.) Digital clocks are often associated with electronic drives, but the "digital" description refers only to the display, not to the drive mechanism. Both analog and digital clocks can be driven either mechanically or electronically, but "clockwork" mechanisms with digital displays are rare.

Because digital clocks can be very small and inexpensive devices that enhance the popularity of product designs, they are often incorporated into all kinds of devices such as cars, radios, televisions, microwave ovens, standard ovens, computers and cell phones.

Diagram

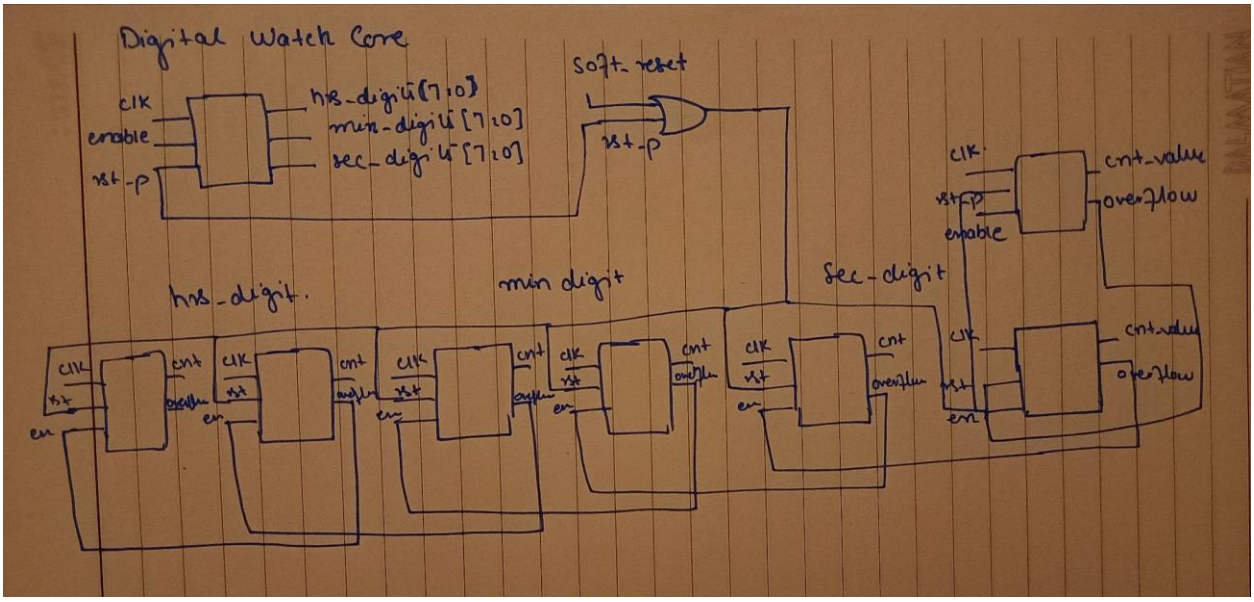

Figure 1: Block Diagram of Digital Watch

6.3 Equipment

- Computer/Laptop with Xilinx Design Suite 14.2 installed.
- ModelSim

6.4 Procedure

1. Designing of block diagram using clock divider and counter instances and mentioning the intermediate connections.
2. Implementation of behavioral abstractions of digital watch
3. Design a test bench in Verilog to check the outputs of digital watch.
4. Opening this project in ISE, synthesize the module and then do the implementation check.
5. Finally generate the BIT file to be uploaded on the FPGA.
6. Observe the output on LEDs.

6.5 Code and Simulations

As we are going to use just LEDs of the Sparten 3E the code is only for the seconds part of the digital watch.

Code:

```
module counter (clk, reset, enable, count_val, over_flow);
parameter Bits=4;
parameter Max=9;
input clk, reset, enable;
output[Bits-1:0] count_val;
output over_flow;
reg[Bits-1:0] count_val;
reg over_flow;

always@(posedge clk)
if(reset) begin
count_val<=0;
over_flow<=0;
end
else if(enable)
if(count_val==Max-1) begin
count_val<=0;
over_flow<=1;
end
else begin

count_val<=count_val+1;
over_flow<=0;
end
endmodule

module digital_watch(clk, reset, enable, seconds);
input clk, reset, enable;
output[7:0] seconds;
wire[3:0] seconds_least, seconds_most;
wire w1,w2;
wire[7:0] seconds;
//For sparten 3E we will use values 32 and 50_000_000.
counter #(2,2) C0(.clk(clk), .reset(reset), .enable(enable), .over_flow(w1), .count_val());
counter #(4,10) C1(.clk(clk), .reset(reset), .enable(w1), .count_val(seconds_least[3:0]),
.over_flow(w2));
counter #(4,6) C2(.clk(clk), .reset(reset), .enable(w2), .count_val(seconds_most[3:0]));
assign seconds={seconds_most,seconds_least};
endmodule
```

Test bench

```
module test_watch();
reg clk, reset, enable;
wire[7:0] seconds;
digital_watch D1(.clk(clk), .reset(reset), .enable(enable), .seconds(seconds));

initial begin
clk=1'b0;
reset=1'b0;
enable=1'b0;
#3 reset=1'b1;
#7 reset=1'b0;
#10 enable=1'b1;
#1000 $stop;
end
always #5 clk=~clk;
endmodule
```

Simulation Results:

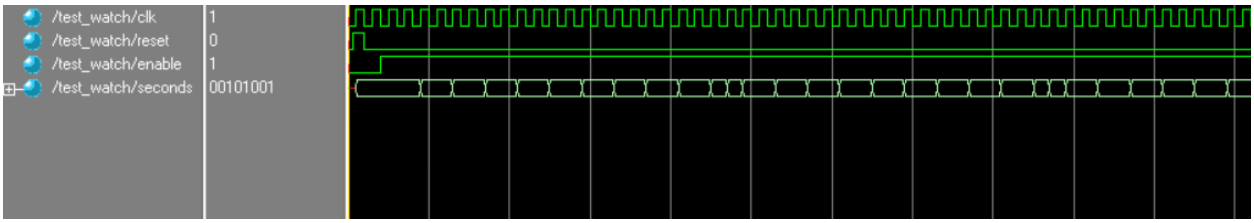

Figure 2: Simulation results of seconds part of digital clock

User Constraint File:

```
NET "clk" LOC = "C9" | IOSTANDARD = LVCMOS33 ;
#NET "clk" PERIOD = 20.0ns HIGH 50%;
NET "reset" LOC = "H13" | IOSTANDARD = LVTTL | PULLDOWN ;
NET "enable" LOC = "L13" | IOSTANDARD = LVTTL | PULLUP ;
NET "seconds<0>" LOC = "F12" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE =
8 ;
```

```
NET "seconds<1>" LOC = "E12" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
NET "seconds<2>" LOC = "E11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
NET "seconds<3>" LOC = "F11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
NET "seconds<4>" LOC = "C11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
NET "seconds<5>" LOC = "D11" | IOSTANDARD = LVTTL | SLEW = SLOW | DRIVE = 8 ;
```

6.6 Discussion:

Using the 50 MHz clock, we hat implement the digital watch for which the clock needs to be divided such that each clock cycle shows 1 sec, we divided the 50 MHz clock into 1Hz clock using counter and if condition, on each positive edge of this 1 Hz clock theres an increment of 1 second. After 59 seconds the seconds overflow increments a minute. Similarly after 59 minutes the minutes overflow increments hours. After 24 hours the watch will reset.

Only the seconds cycle is checked on the embedded digital LEDs of Sparten-3E FPGA board. UCF file is written where the specific pins are mapped to the pin codes for each switch or LED to make a connection; the cit file being generated is burned on the board and the results are verified.

The FPGA itself operates on 50MHz and is fetched to the watch module to avoid any conflicts.