DESIGN AND IMPLEMENTATION OF A DIGITAL CLOCK TIMER USING VERILOG

By:-
KOTHAPALLI MAHITH VATHSAV

MAY 5TH , 2025

KOTHAPALLI MAHITH VATHSAV

Objective:-

The objective of this project is to design and implement a digital clock timer using Verilog HDL that counts seconds, minutes, and hours up to 23:59:59 based on a clock pulse input. The design aims to simulate real-time clock functionality using synchronous counters and rollover logic.

Theory:-

1) Digital Clock Basics

- A digital clock works by counting seconds, minutes, and hours using counters. A real-time clock counts from 00:00:00 to 23:59:59 and resets thereafter.

2) Verilog Counters

- In this project, synchronous counters are used, where counting happens on every positive clock edge. Three 6-bit counters represent:
- ss (seconds)
- mm (minutes)
- hh (hours)

3) Rollover Logic

- Seconds increment from 0 to 59. After 59, it resets to 0 and increments minutes.
- Minutes increment from 0 to 59. After 59, it resets to 0 and increments hours.
- Hours increment from 0 to 23. After 23, it resets to 0.

4) Hardware Description Language (HDL)

Verilog is used for describing digital systems. Here, the clock timer is implemented with always blocks triggered on the rising edge of the clock, using if-else logic for rollover conditions.

5) Hardware Description Language (HDL)

- Verilog is used for describing digital systems. Here, the clock timer is implemented with always blocks triggered on the rising edge of the clock, using if-else logic for rollover conditions.

6) Methodology / Design Steps

1. Define Inputs and Outputs
    - Inputs: Clock (clk), Reset (rst)
    - Outputs: 6-bit outputs for seconds (ss), minutes (mm), and hours (hh)
2. Write Counter Logic in Verilog
    - Use synchronous logic (always @(posedge clk)).
    - Implement nested rollover conditions for sec ➔ min ➔ hour.

Design part :-

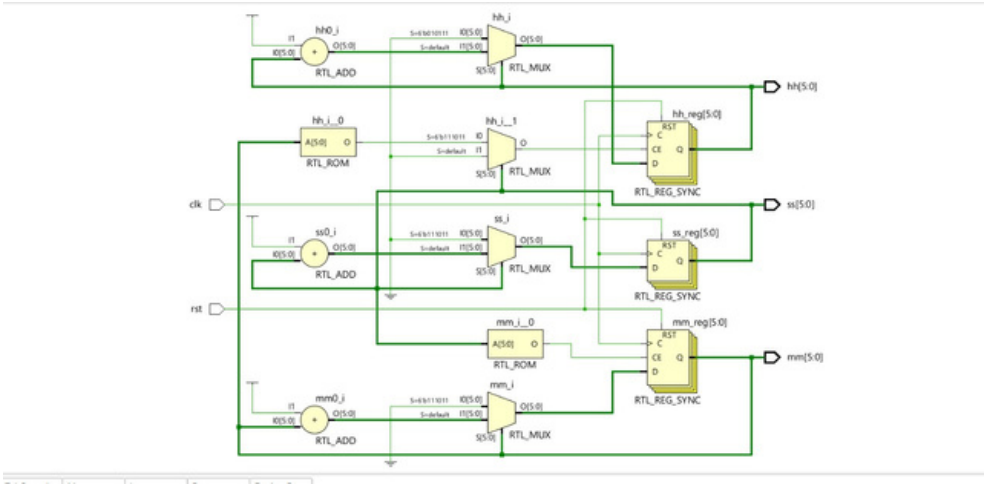
```
`timescale 1ns / 1ps
// Create Date: 05/03/2025 09:49:26 PM
// Designer Name: KOTHAPALLI MAHITH VATHSAV
// Module Name: digclktim
// Project Name: Digital clock timer
module digclktim(
input clk,
input rst,
output reg[5:0]ss,
output reg[5:0]mm,
output reg[5:0]hh
);
always@(posedge clk) begin
if(rst==1)begin
ss<= 6'b000000;
mm<= 6'b000000;
hh<= 6'b000000;
end
else  begin
if(ss==6'b111011)begin //once the ss reaches to 59 it will reset to 0 seconds
ss<= 6'b000000;
if(mm==6'b111011)begin //once the minutes reaches to 59 min it will reset to 0 min
mm<= 6'b000000;
if(hh==6'b010111)begin //once the hour hand reaches 23 hrs it will also reset to 0 hr
hh<= 6'b000000;
end
else begin // if not it will increment by 1 hr from 0 hr
hh<= hh+6'b000001;
end
end
else begin
```

```
else begin
mm<=mm+6'b000001; // if not it will increment by 1 mm from 0 mm
end
end
else begin
ss<= ss+6'b000001;  //if not it will increment by 1 second from 0 seconds
end
end
end



endmodule

```

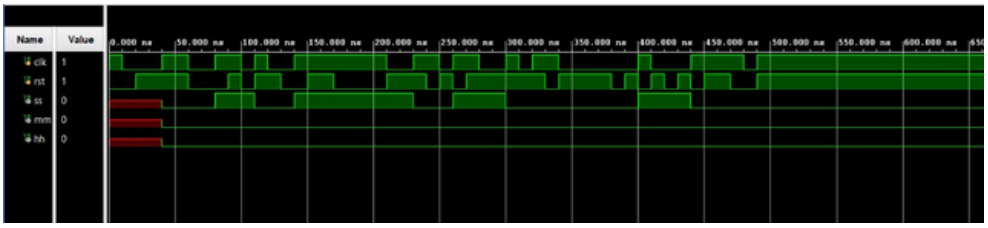
Testbench part :-

```
// Create Date: 05/04/2025 01:01:56 AM
// Designer Name: KOTHAPALLI MAHITH VATHSAV
// Module Name: digcltmtb
// Project Name: Digital clock timer
module digcltmtb;
reg clk;
reg rst;
wire ss;
wire mm;
wire hh;
digclktim dut(
.clk(clk),
.rst(rst),
.ss(ss),
.mm(mm),
.hh(hh)
);
initial begin
$monitor("clk=%b,rst=%b,ss=%b,mm=%b,hh=%b",clk,rst,ss,mm,hh);
end
initial begin
repeat(50)begin
clk= $urandom_range(0,1);
rst=$urandom_range(0,1);
#10;
end
end


endmodule
```

Eloborated design part :-



Output Waveform part :-



Thank you!

Want to discuss on this :-

vathsavkothapalli12092004@gmail.com

Follow this page for more VLSI content :-

KOTHAPALLI MAHITH VATHSAV

KOTHAPALLI MAHITH VATHSAV