ganeshprabhu2005 / T-FLIPFLOP-POSEDGE

Code Pull requests Actions Projects Wiki Security Insights Settings

GPL-3.0 license

0 stars 86 forks 0 watching 1 Branch 0 Tags Activity

Public repository · Forked from naavaneetha/T-FLIPFLOP-POSEDGE

main 1 Branch 0 Tags Go to file t Go to file Add file Code

This branch is 1 commit ahead of naavaneetha/T-FLIPFLOP-POSEDGE:main . Contribute Sync fork

ganeshprabhu2005 Update README.md now

db	EXP9	3 months ago
incremental_db	EXP9	3 months ago
output_files	EXP9	3 months ago
simulation	EXP9	3 months ago
LICENSE	Initial commit	3 months ago
README.md	Update README.md	now
TFLIPFLOPPOSEDGE.qpf	EXP9	3 months ago
TFLIPFLOPPOSEDGE.qsf	EXP9	3 months ago
TFLIPFLOPPOSEDGE.qws	EXP9	3 months ago
TFLIPFLOPPOSEDGE.v	EXP9	3 months ago
TFLIPFLOPPOSEDGE.v.bak	EXP9	3 months ago
Waveform.vwf	EXP9	3 months ago

README License

T-FLIPFLOP-POSEDGE

AIM:

To implement T flipflop using verilog and validating their functionality using their functional tables

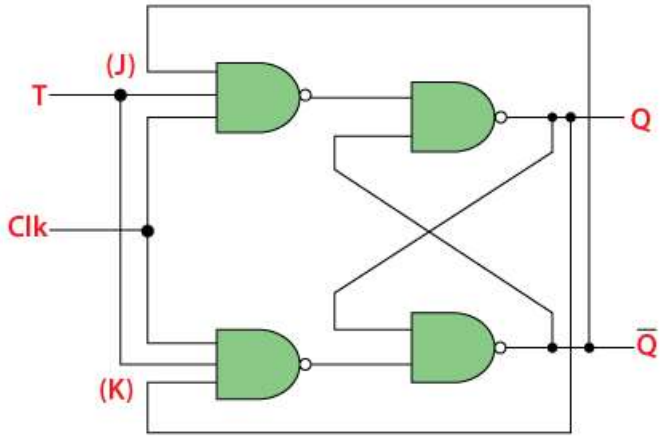
SOFTWARE REQUIRED:

Quartus prime

THEORY

T Flip-Flop

T flip-flop is the simplified version of JK flip-flop. It is obtained by connecting the same input 'T' to both inputs of JK flip-flop. It operates with only positive clock transitions or negative clock transitions. The circuit diagram of T flip-flop is shown in the following figure.



This circuit has single input T and two outputs Qtt & Qtt'. The operation of T flip-flop is same as that of JK flip-flop. Here, we considered the inputs of JK flip-flop as J = T and K = T in order to utilize the modified JK flip-flop for 2 combinations of inputs. So, we eliminated the other two combinations of J & K, for which those two values are complement to each other in T flip-flop. The following table shows the state table of T flip-flop.

Here, Qtt & Qt+1t+1 are present state & next state respectively. So, T flip-flop can be used for one of these two functions such as Hold, & Complement of present state based on the input conditions, when positive transition of clock signal is applied. The following table shows the characteristic table of T flip-flop. Inputs Present State Next State

Inputs	Present State	Next State
T	Q t	Q $t+1$
0	0	0
0	1	1
1	0	1
1	1	0

From the above characteristic table, we can directly write the next state equation as Q(t+1)=T'Q(t)+TQ(t)' ⇒Q(t+1)=T⊕Q(t)

Procedure

Step 1: Open Quartus II in your laptop.

Step 2: Write code to implement SR flipflop using verilog and validating their functionality using their functional tables.

Step 3: Run compilation to check for errors.

Step 4: Open waveform output and load input values.

Step 5: Run simulation to get the output.

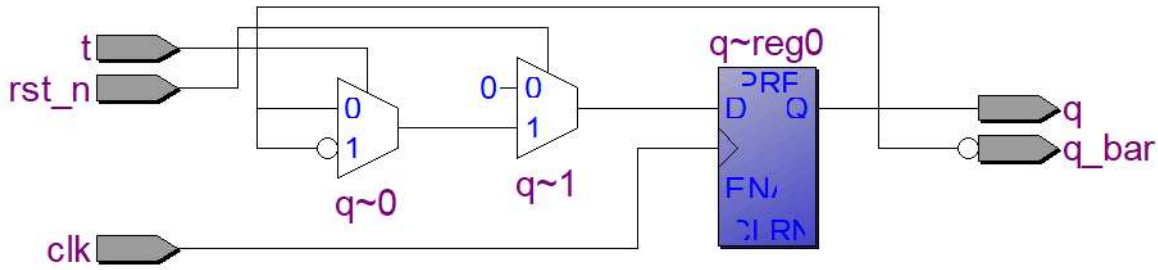
Step 6: Open in RTL viewers to get RTL diagram output.

PROGRAM

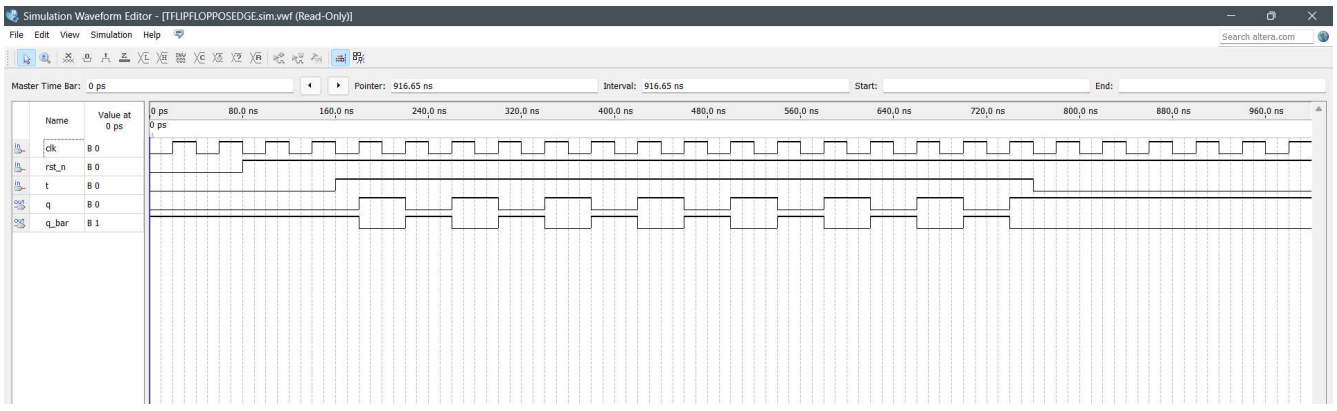
/* Program for flipflops and verify its truth table in quartus using Verilog programming. Developed by: GANESH PRABHU J RegisterNumber: 212223220023

```
module TFLIPFLOPPOSEDGE( input clk, rst_n, input t,
output reg q,
output q_bar
);
always@(posedge clk)
begin
if(!rst_n)
q<=0;
else
if(t)
q<=~q;
else
q<=q;
end
assign q_bar = ~q;
endmodule
```

RTL LOGIC FOR FLIPFLOPS



TIMING DIGRAMS FOR FLIP FLOPS



RESULTS Hence, T flipflop using verilog and validating their functionality using their functional tables is implemented.

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

VHDL 48.8% Stata 19.2% HTML 17.1% Verilog 13.4% Standard ML 1.5%

Suggested workflows

Based on your tech stack

SLSA Generic generator Configure

Generate SLSA3 provenance for your existing release workflows

Jekyll using Docker image Configure

Package a Jekyll site using the jekyll/builder Docker image.

More workflows Dismiss suggestions