

main
 1 Branch
 0 Tags

Go to file

Go to file

Code

This branch is [1 commit ahead of](#) [naavaneetha/SYNCHRONOUS-UP-COUNTER:main](#) .

ganeshprabhu2005 Update README.md
 2 minutes ago

db	EXP11	3 months ago
incremental_db	EXP11	3 months ago
output_files	EXP11	3 months ago
simulation	EXP11	3 months ago
LICENSE	Initial commit	3 months ago
README.md	Update README.md	2 minutes ago
UPCOUNTER.qpf	EXP11	3 months ago
UPCOUNTER.qsf	EXP11	3 months ago
UPCOUNTER.qws	EXP11	3 months ago
UPCOUNTER.v	EXP11	3 months ago
UPCOUNTER.v.bak	EXP11	3 months ago
Waveform.vwf	EXP11	3 months ago

README
 License

SYNCHRONOUS-UP-COUNTER

AIM:

To implement 4 bit synchronous up counter and validate functionality.

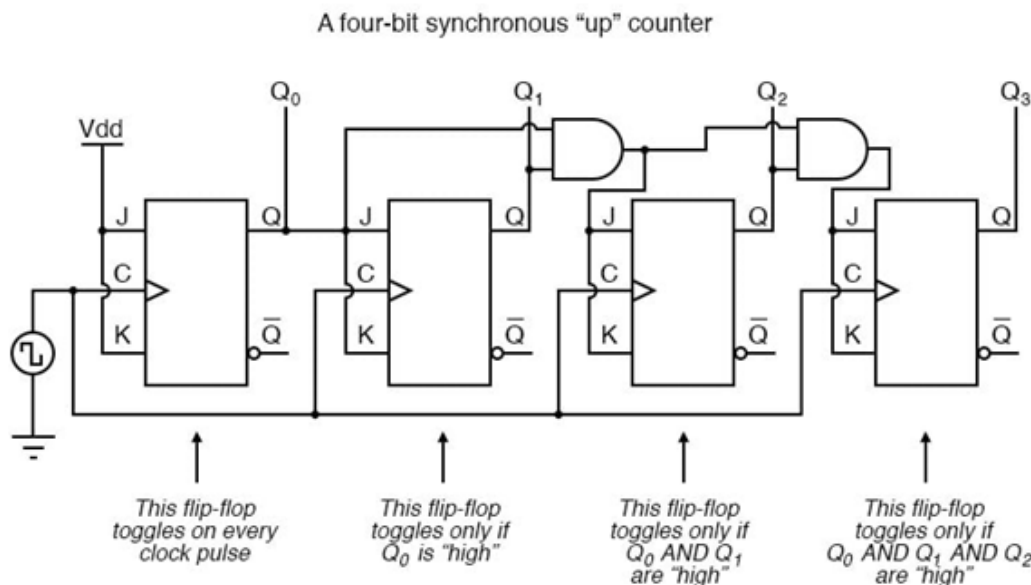
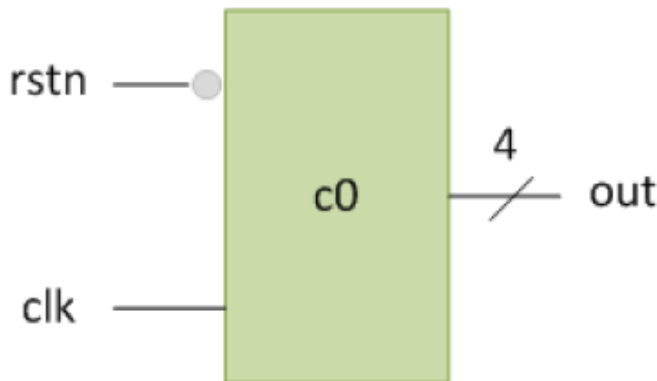
SOFTWARE REQUIRED:

Quartus prime

THEORY

4 bit synchronous UP Counter

If we enable each J-K flip-flop to toggle based on whether or not all preceding flip-flop outputs (Q) are "high," we can obtain the same counting sequence as the asynchronous circuit without the ripple effect, since each flip-flop in this circuit will be clocked at exactly the same time:



Each flip-flop in this circuit will be clocked at exactly the same time. The result is a four-bit synchronous "up" counter. Each of the higher-order flip-flops are made ready to toggle (both J and K inputs "high") if the Q outputs of all previous flip-flops are "high." Otherwise, the J and K inputs for that flip-flop will both be "low," placing it into the "latch" mode where it will maintain its present output state at the next clock pulse. Since the first (LSB) flip-flop needs to toggle at every clock pulse, its J and K inputs are connected to Vcc or Vdd, where they will be "high" all the time. The next flip-flop need only "recognize" that the first flip-flop's Q output is high to be made ready to toggle, so no AND gate is needed. However, the remaining flip-flops should be made ready to toggle only when all lower-order output bits are "high," thus the need for AND gates.

Procedure

1. Initialize the shift register to a known state (e.g., all zeros).
2. Input a bit serially into the shift register.

CLK	CLR	LD	EN	Q3	Q2	Q1	Q0
0	0	0	0	X	X	X	X
0	0	0	1	X	X	X	X
0	0	1	0	X	X	X	X
0	0	1	1	X	X	X	X
0	1	X	X	0	0	0	0
1	X	X	X	Q3	Q2	Q1	Q0

RESULTS

Hence a 4 bit synchronous up counter is implemented correctly

Releases

No releases published

Packages

No packages published

Languages

● VHDL 47.6% ● Stata 20.5% ● HTML 20.4% ● Verilog 9.7% ● Standard ML 1.8%