

به نام خدا

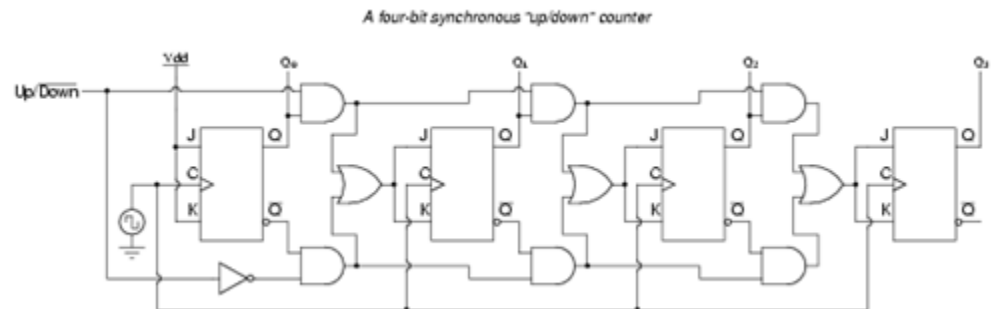
پیش گزارش آزمایش دوم

علیرضا سلیمیان 400105036

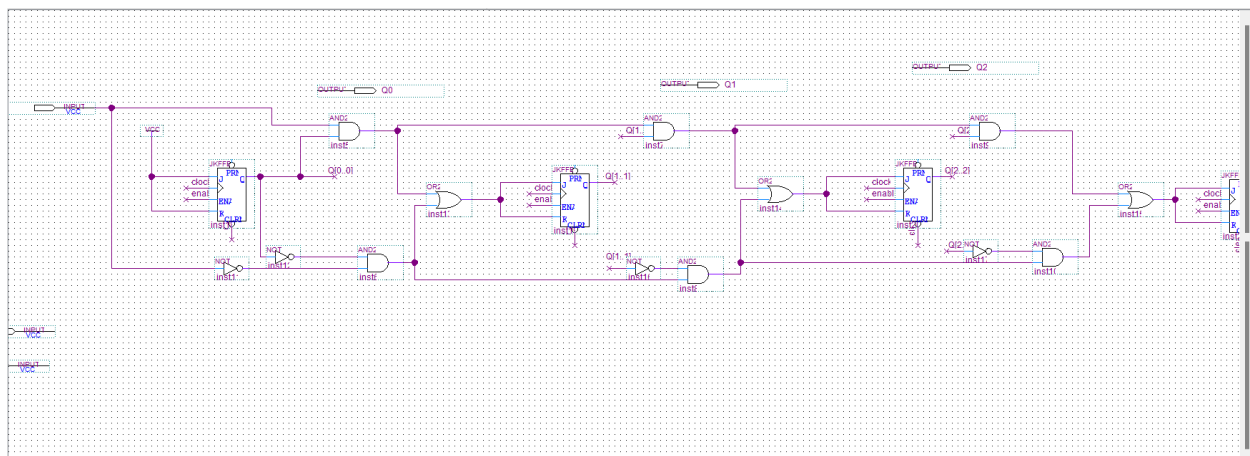
محمد پیام تائبی: 400104867

امیر حسین علمدار: 400105144

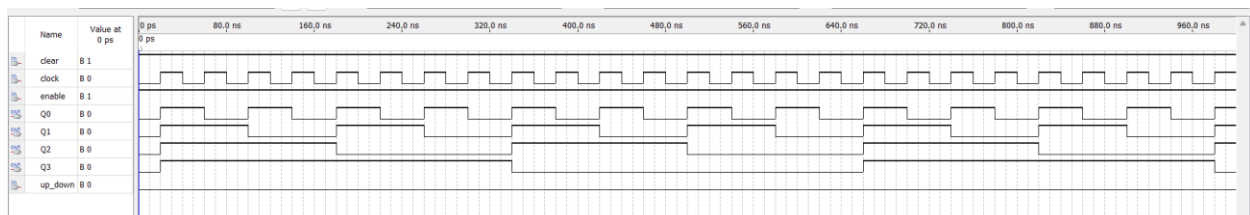
در این آزمایش قرار است مداری ترتیبی طراحی کنیم که در آن باید اجازه ورود به اتفاقی داده شود. ابتدا به توضیح چگونگی ساخت یک شمارنده میپردازیم. برای ساخت یک شمارنده که رو به بالا و پایین شمارش انجام دهد نیاز به تعدادی jk-ff داریم. در زیر شکل کلی یک شمارنده آورده شده است.



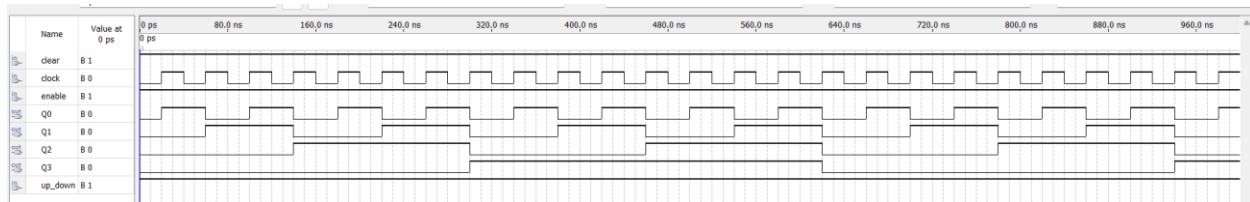
در صورت سوال آورده شده است که این شمارنده باید دارای قابلیت clear و یک سیگنال enable باشد. که برای اضافه کردن آنها کافی است که k را طوری انتخاب کنیم که این قابلیت هارا داشته باشد. در ادامه شکلی که برای ساخت این مدار در کوآرتوس کشیده شده است آورده شده است.



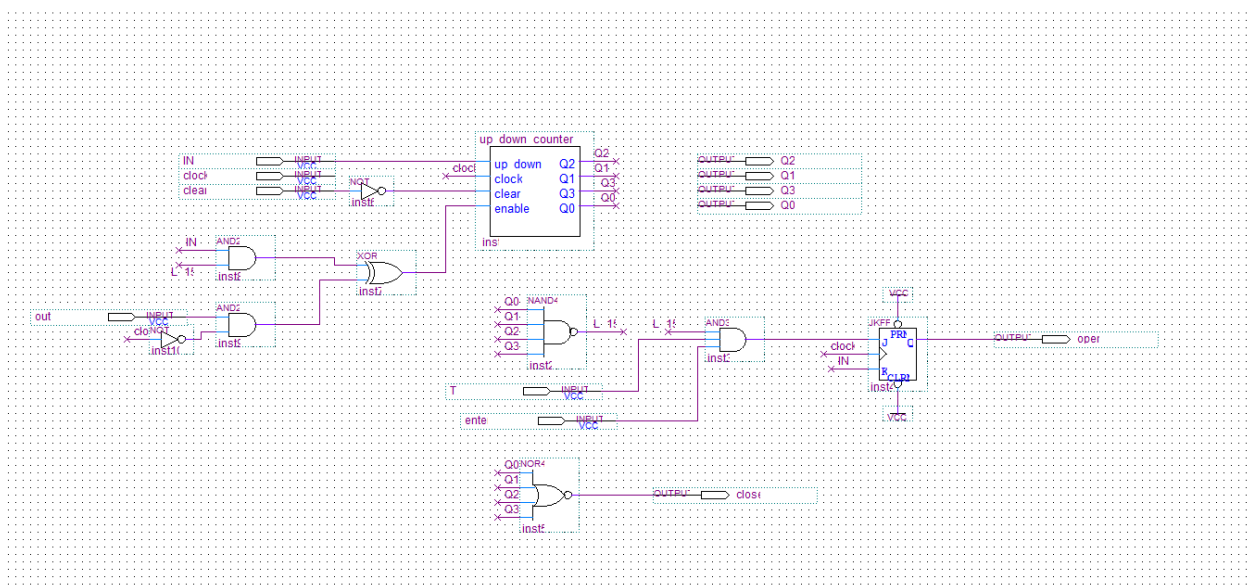
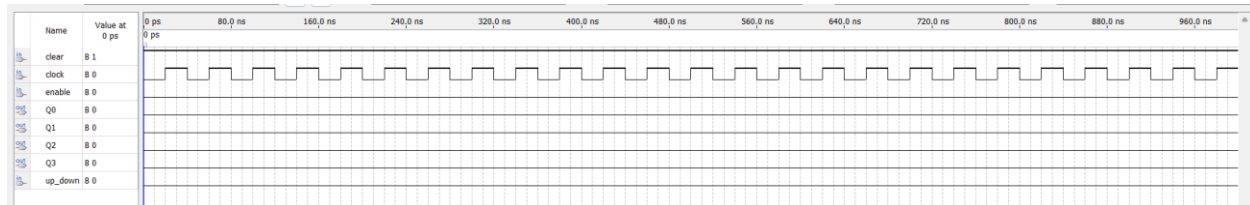
که البته به علت بزرگ شدن شکل، تمام آن در تصویر قرار نگرفت. در ادامه نیز تعدادی تست برای این مدار آورده شده است. شکل زیر برای شمارش رو به پایین آورده شده است.



شمارش رو به بالا:



تاثیر enable:



شکل بالا ساختار کلی مدار آورده شده است و در ادامه سیگنال های آن توضیح داده خواهد شد.

سیگنال **open**: در ابتدا که کاربر می خواهد وارد شود، باید دکمه **ent** را بزند و اگر تعداد افراد داخل اتاق کمتر از ۱۵ بود، این سیگنال یک می شود و تا زمانی که سیگنال **IN** وارد نشده باشد یک باقی می ماند. برای ساخت این سیگنال از یک JK FF استفاده کردیم به این صورت که اگر در زمان مجاز بودیم و تعداد افراد کمتر از ۱۵ بود و دکمه **ent** وارد شده بود، یک خروجی بدهد و بلافاصله بعد از اینکه سیگنال **IN** وارد شد سیگنال ما صفر می شود (البته باید منتظر کلاک باشیم).

Count enable: برای ساخت این سیگنال باید این را در نظر گرفت که اگر یکی از دو سیگنال **IN** و **OUT** فعال بود و همزمان یک مقدار نداشتند باید شمارش انجام شود، برای مثال اگر سیگنال **IN** و **OUT** هر دو صفر باشند، که شمارشی انجام نمی شود و اگر هر دو یک باشند باز هم شمارشی انجام نمی شود چرا که در این حالت یکی وارد شده و دیگری خارج شده است و تعداد افراد کل تغییری نمی کند، تنها در صورتی

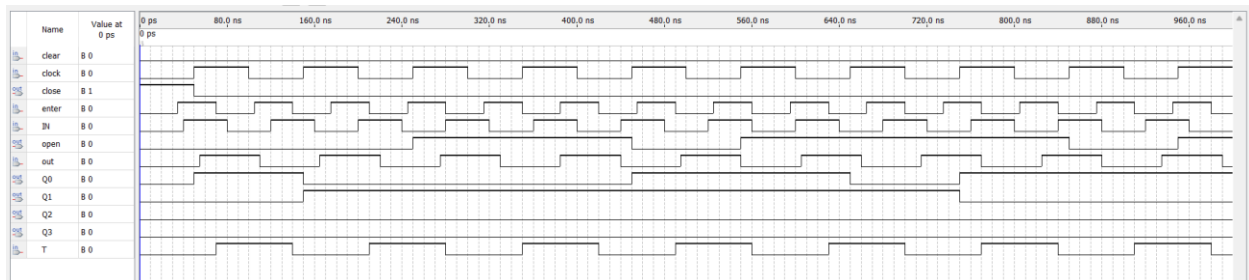
تغییر انجام می‌شود که فقط یکی از این دو سیگنال فعال باشند. در این قسمت، من شرط اینکه تعداد افراد کمتر از ۱۵ باشد و در خروجی بسته نباشد را نیز در نظر گرفته‌ام.

برای چک کردن اینکه تعداد افراد کمتر از ۱۵ هست به بیت‌های خروجی نیاز داریم، چون ۴ بیت در اختیار داریم، پس اگر تمام بیت‌ها یک نباشد (عدد ۱۵ نباشد) میتوان فرد جدیدی وارد کرد برای این قسمت از nand استفاده کردیم.

برای صفر نشدن همه بیت‌ها نیز از یک nor استفاده کردیم.

سیگنالی نیز به عنوان کلاک به دستگاه داده شده است.

شکل زیر تست این مدار را نشان میدهد:



برای به دست آوردن فرکانس کاری این مدار، از خود کوآرتوس استفاده کردیم:

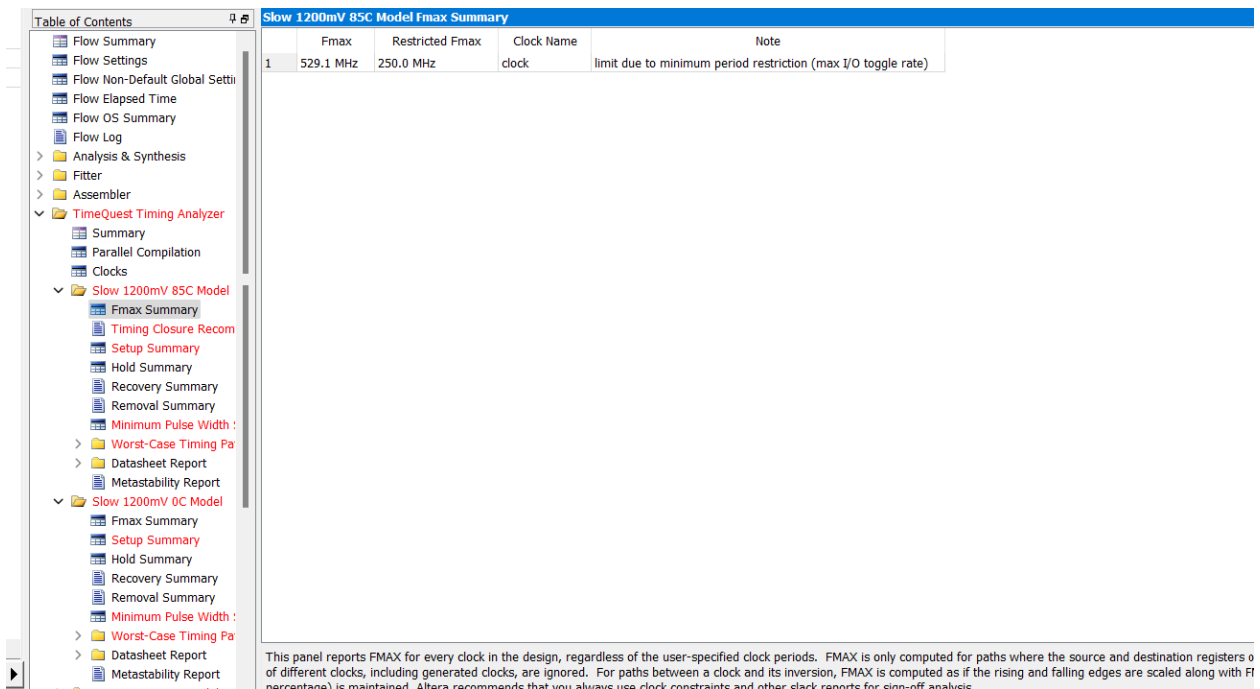


Table of Contents

Flow Summary

Flow Settings

Flow Non-Default Global Settli

Flow Elapsed Time

Flow OS Summary

Flow Log

> Analysis & Synthesis

> Fitter

> Assembler

> TimeQuest Timing Analyzer

Summary

Parallel Compilation

Clocks

> Slow 1200mV 85C Model

Fmax Summary

Timing Closure Recom

Setup Summary

Hold Summary

Recovery Summary

Removal Summary

Minimum Pulse Width :

> Worst-Case Timing Pa

> Datasheet Report

Metastability Report

> Slow 1200mV 0C Model

Fmax Summary

Setup Summary

Hold Summary

Recovery Summary

Removal Summary

Minimum Pulse Width :

> Worst-Case Timing Pa

> Datasheet Report

Metastability Report

Slow 1200mV 0C Model fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	592.07 MHz	250.0 MHz	clock	limit due to minimum period restriction (max I/O toggle rate)

This panel reports FMAX for every clock in the design, regardless of the user-specified clock periods. FMAX is only computed for paths where the source and destination registers or ports are driven by the same clock. Paths of different clocks, including generated clocks, are ignored. For paths between a clock and its inversion, FMAX is computed as if the rising and falling edges are scaled along with FMAX, such that the duty cycle (in terms of a percentage) is maintained. Altera recommends that you always use clock constraints and other slack reports for sign-off analysis.