

## ورودی ها و خروجی های تقسیم کننده:

4 ورودی داریم و 3 خروجی، ورودی 32 بیتی A را بر ورودی 16 بیتی B تقسیم می‌کنیم، یک ورودی start به عنوان سیگنال کنترلی داریم. بعد از مقدار دادن به اعداد آنرا 1 می‌کنیم و سپس 0 و عمل تقسیم شروع می‌شود (با 1 شدن start، fsm به initial state می‌رود و اگر 0 نکنیم کد همواره به state اول می‌رود عملیات تقسیم شروع نمی‌شود.) و آخرین ورودی clock است.

خروجی‌ها یکی Done است که با 1 شدن آن یعنی عمل تقسیم تمام شده و دیگر خروجی‌ها عبارتند از outQ که خارج قسمت و baghimande که مقدار باقی مانده تقسیم است.

## الگوریتم تقسیم:

قبل از شروع شدن عملیات تقسیم مقدار قدر مطلق A و B را در temp a و temp b قرار می‌دهیم تا برای تقسیم اعداد منفی به مشکل نخوریم و تمامی processها روی این دو انجام می‌شود.

هر بار temp a، 16 بیت اول باقی مانده و 16 بیت دیگر را خارج قسمت و در sum ذخیره می‌کنیم، زیرا اگر 16 بیت بزرگتر مقسوم بیشتر از مقسوم علیه باشد برای نشان دادن خارج قسمت به 17 بیت نیاز داریم پس فرض شده کوچکتر است بنابراین در مرحله اول یک شیف به چپ داریم.

اگر آخرین بیت sum برابر 1 شد یعنی منفی است و آن عددی که در باقی مانده است از B کوچکتر است پس باید یک شیف به چپ می‌دهیم و 0 وارد بیت کوچکتر خارج قسمت می‌کنیم و یک شیف به چپ کلی هم می‌دهیم و به sc(sequence counter) یک واحد اضافه می‌کنیم.

اگر آخرین بیت sum برابر 0 شد یعنی مقدار در نظر گرفته شده برای a یا همان sum از b بزرگتر شده پس باید b را منهای sum کنیم و به sc(sequence counter) یک واحد اضافه می‌کنیم و مقدار sum را باید در temp a لود کنیم و شیف چپ دهیم.

برای اتمام عملیات تقسیم مقدار sc باید به تعداد بیت‌های B برسد تا Done 1 شود و جواب نهایی بدست آید.

برای تقسیم اعداد منفی مقدار اصلی اعداد A و B را در temp AA و temp BB ذخیره و برای بدست آوردن مقدار و علامت باقی مانده و خارج قسمت استفاده می‌کنیم.

## اعضا:

امیر مهدی زاده

محمد عساری

پایا فغانی

علی تابش پور