

در ادامه توضیحاتی در مورد ساختار کد و طراحی خواهیم دید.

یک bus به اسم **dataIN** داریم که بعد از دیدن سیگنال **start** ورودی های 5 بیتی را در سه کلاک پیاپی برای برنامه به ارمغان می آورد. ورودی اول 5 بیت پر ارزش مقسوم است که در شیفت رجیستر **A** ذخیره می شود. ورودی دوم 5 بیت کم ارزش مقسوم است که در شیفت رجیستر **Q** ذخیره می شود. ورودی سوم مقسوم علیه است که در رجیستر **D** ذخیره می شود.

یک bus به اسم **dataOUT** داریم که بعد از اتمام عملیات تقسیم ، بعد از فعال کردن سیگنال **done** دو خروجی از برنامه خارج میکند. که خروجی اول خارج قسمت خواهد بود و خروجی دوم باقی مانده را نشان می دهد.

یک سیگنال **OV** هم داریم که هر وقت مقدار مقسوم علیه از 5 بیت پر ارزش مقسوم کوچکتر یا مساوی آن باشد در یک کلاک مقدار **1** به خودش میگیرد.

یک سیگنال به اسم **DivByZero** هم داریم که زمانی که مقسوم علیه صفر باشد مقدار **1** را در یک کلاک به خودش میگیرد.

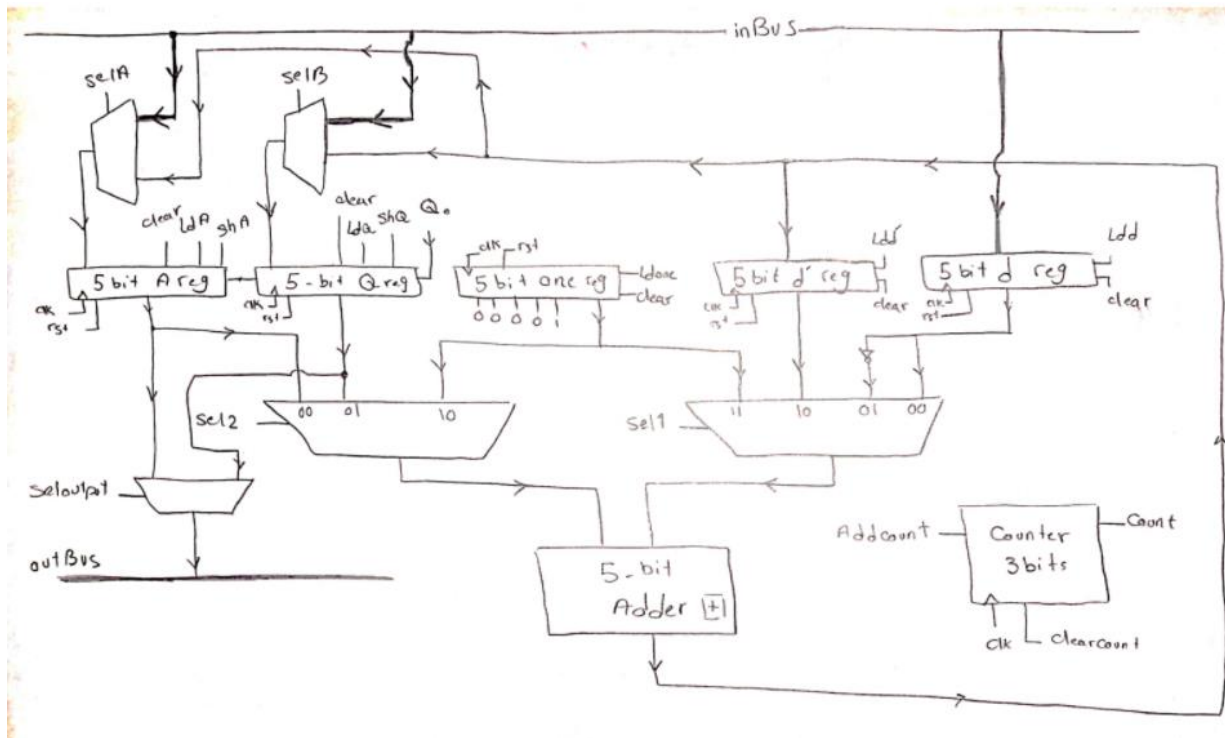
اگر هر کدام از سیگنال های بالا **1** شوند ، برنامه از تقسیم خودداری میکند و تا رسیدن ورودی های بعدی ، صبر میکند.

ما از الگوریتم با بازیابی و **performing** استفاده کرده ایم.

در طرح این پروژه ورودی ها مثبت فرض شده بنابراین بیت پر ارزش **A** و **D** هم صفر فرض شده است.

در ادامه تصویری از مسیر داده و کنترلر خواهیم دید.

در شکل زیر مسیر داده (data path) را مشاهده میکنیم.



در تصویر زیر مسیر کنترلر (controller) را مشاهده میکنیم.

