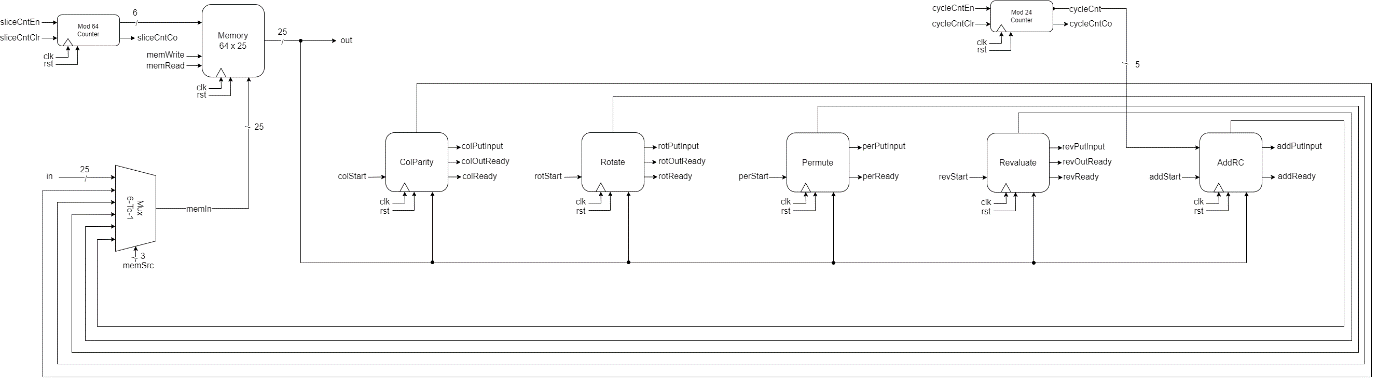
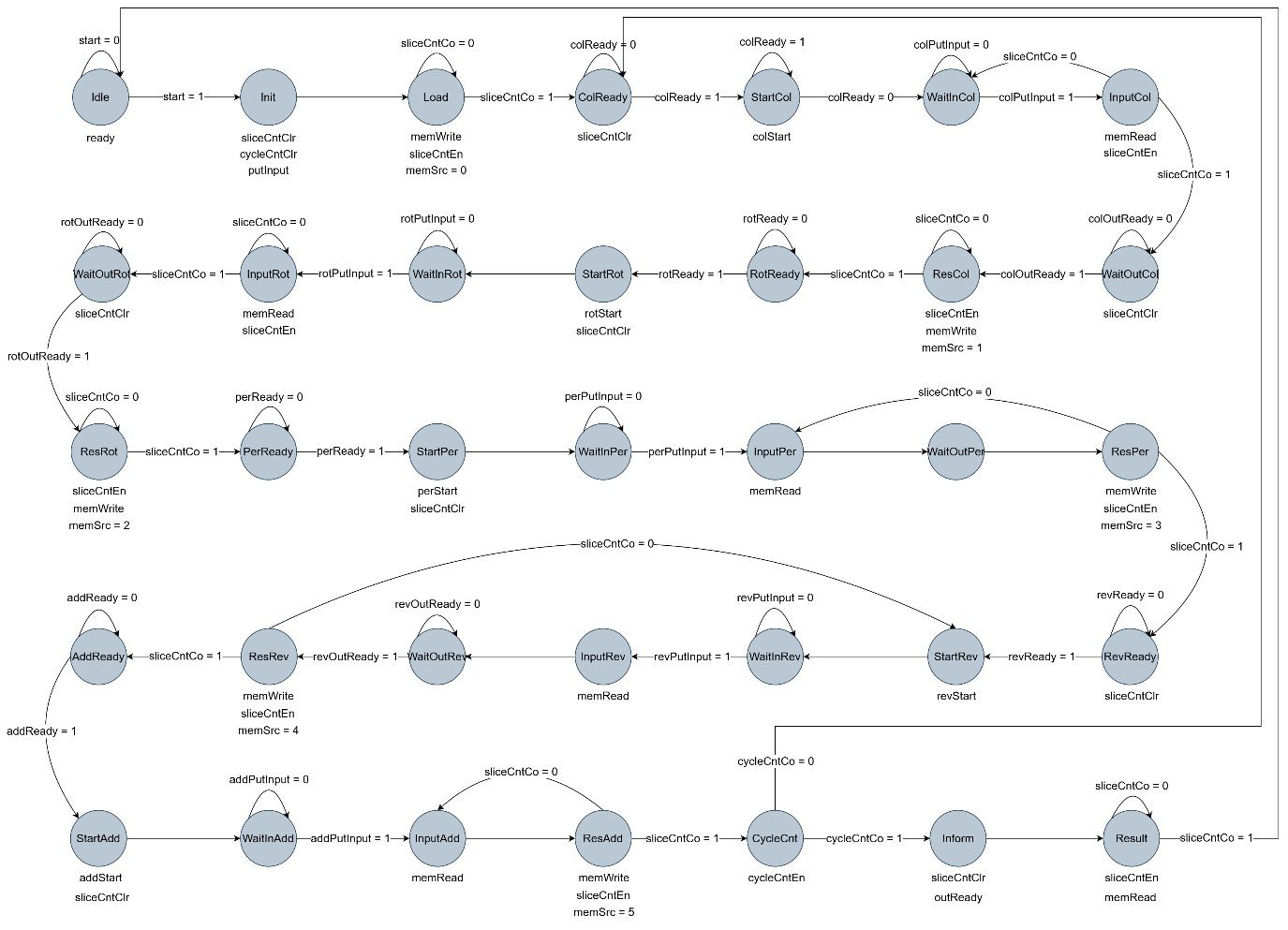
# ماژول Encoder

این ماژول، ابتدا ماتریس ورودی را طی 64 کلاک دریافت کرده و در مموری خود ذخیره می‌کند. سپس 5 تابع را روی مموری به ترتیب اعمال کرده و این عمل را 24 بار تکرار می‌کند. در نهایت طی 64 کلاک مقادیر مموری خود را خروجی می‌دهد.

## مسیر داده



## واحد کنترل



# ماژول ColParity

این ماژول در پروژه میانترم پیاده‌سازی شده و از همان در اینجا بدون تغییر استفاده شده است.

این تابع بر روی هر بیت ماتریس، به صورت زیر تعریف می‌شود:

برای پیاده‌سازی این تابع که یعنی هر بیت ماتریس را برابر xor دو ستون سمت چپ، و ستون سمت راست در لایه قبل می‌کند، از دو ماژول داخلی ColumnParityCalculator و ParityCalculator استفاده می‌شود.

ماژول ColumnParityCalculator، یک اسلایس را گرفته و parity پنج ستون آن را محاسبه می‌کند. برای این کار هر بار با استفاده از چهار adder اندیس عناصر ستون به دست آمده، 5 اندیس از اسلایس انتخاب و وارد xor می‌شوند. نتیجه در یک shift register ذخیره شده و پس از محاسبه هر 5 ستون، خروجی داده می‌شود.

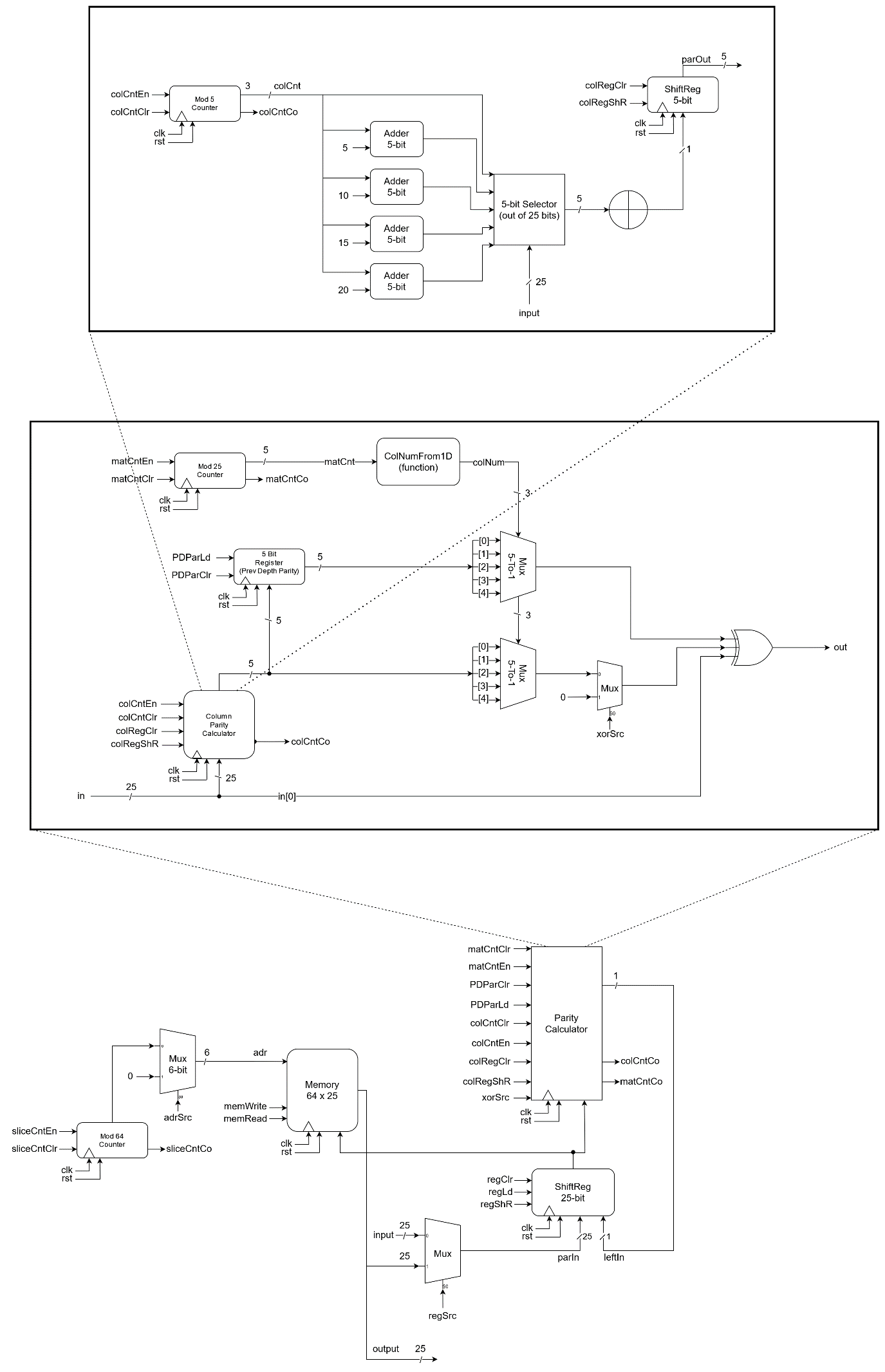
ماژول ParityCalculator ابتدا با استفاده از ماژول قبلی، parity ستون‌های اسلایس کنونی را محاسبه کرده و parity ستون‌های اسلایس قبلی را در یک رجیستر ذخیره نگه می‌دارد. سپس مقدار جدید بیت اول ورودی را طبق رابطه (که xor بیت اول، CurrColParity[i-1] و PrevColParity[i+1] است) خروجی می‌دهد.

ماژول اصلی، شامل یک مموری است که ابتدا طی 64 کلاک ورودی را دریافت کرده، و سپس هر اسلایس را وارد یک shift register می‌کند. این shift register هر بار وارد ParityCalculator شده و پس از محاسبه شدن مقدار جدید بیت اول آن، یک بار شیفت می‌خورد و مقدار جدید را وارد خود می‌کند.

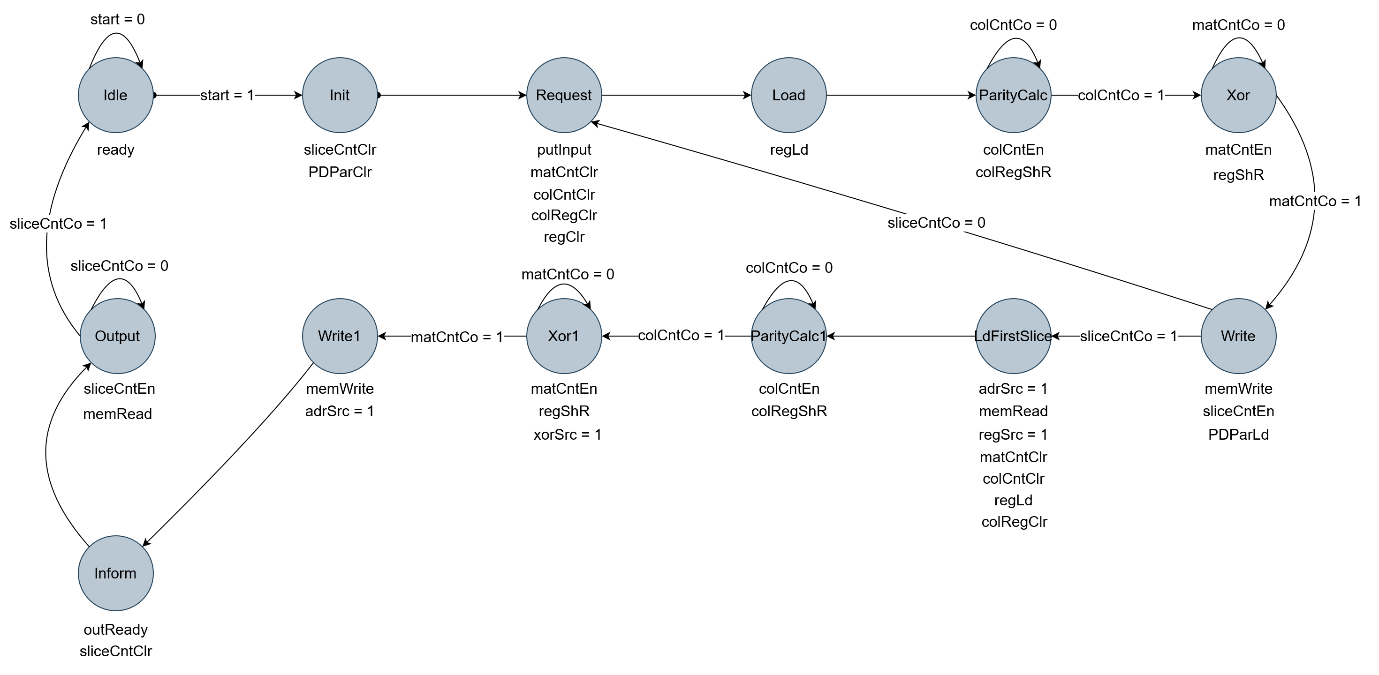
این کار 25 بار انجام شده و هر بار مقدار جدید یک بیت وارد shift register می‌شود و از آنجا که شیفت می‌خورد، ParityCalculator که فقط مقدار جدید بیت اول را می‌دهد، حاصل تمام بیت‌ها را تولید می‌کند.

در نهایت مقدار نهایی shift register وارد مموری شده و پس از پایان محاسبات، مموری خروجی داده می‌شود.

## مسیر داده



## واحد کنترل



# ماژول Rotate

این ماژول هر lane ماتریس را مقدار خاصی شیفت می‌دهد. مقادیر hard code نشده اند و محاسبه می‌شوند.

برای این کار دو ماژول MartixMult و ShiftSize داخل Rotate قرار دارند.

MatrixMult، عدد t را گرفته و ضرب ماتریسی ذکر شده در صورت سوال را برای به دست آوردن x و y انجام می‌دهد. سپس با استفاده از تابع Index2DTo1D، اندیس lane ای که باید شیفت بخورد را می‌گیرد.

ShiftSize، مداری combinational بوده که با گرفتن t، حاصل را حساب می‌کند که مقدار شیفت کردن است.

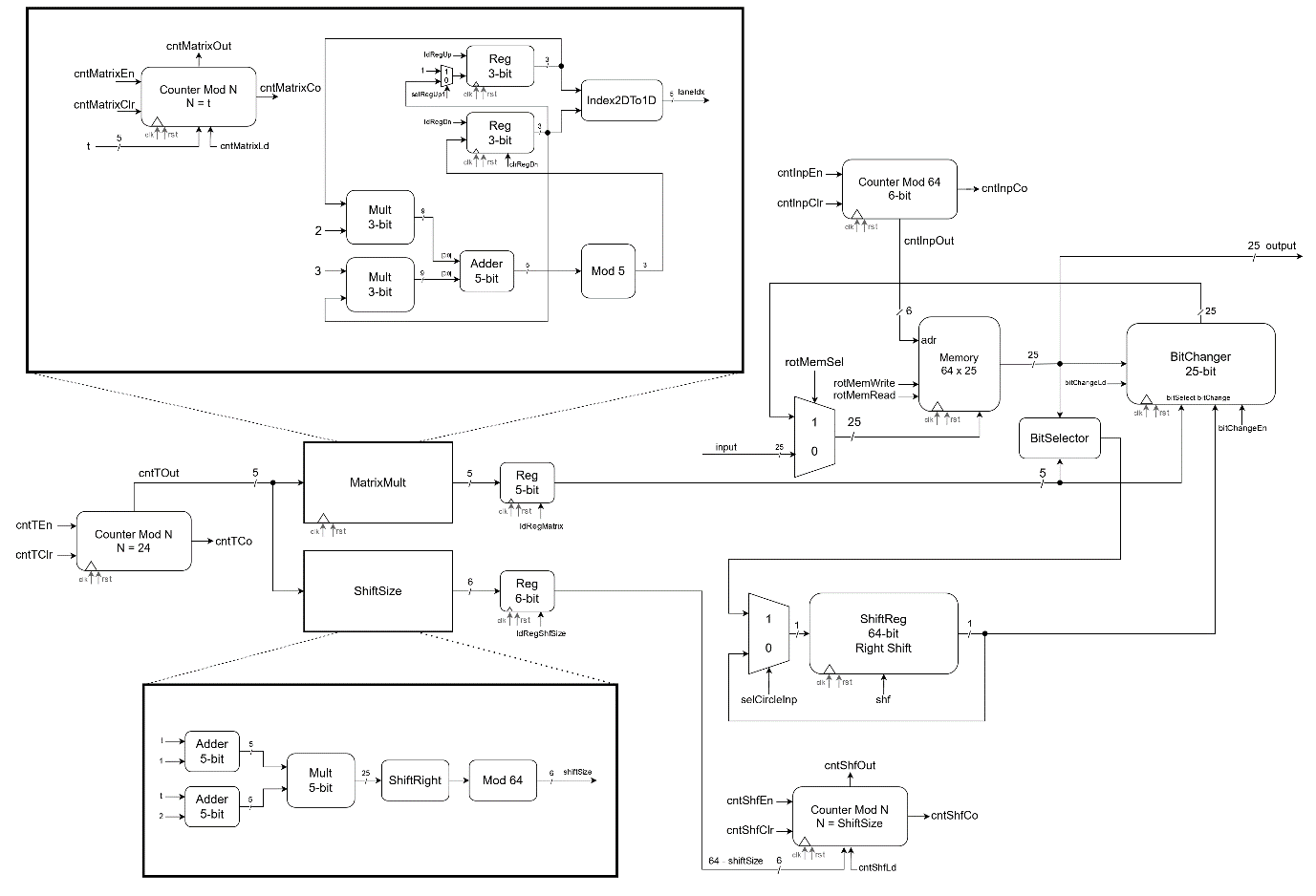
این ماژول در ابتدا طی 64 کلاک ورودی را در مموری داخل خود ذخیره می‌کند.

حال اندیس lane ای که باید شیفت شود و مقدار شیفت به دست آورده می‌شود. این مقادیر وابسته به t اند که با شمارنده mod 24 به دست می‌آید.

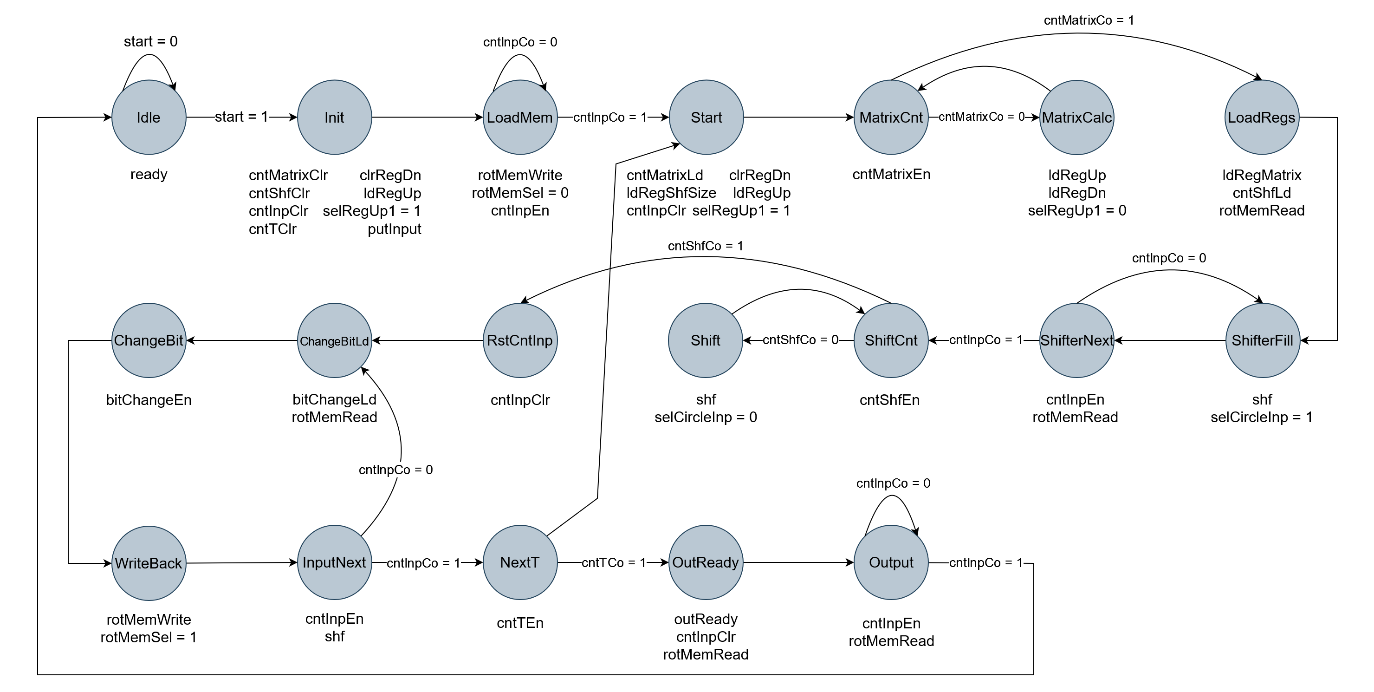
سپس lane مورد نظر را طی 64 کلاک در یک shift register وارد می‌کند و به اندازه حساب شده ShiftSize آن را شیفت می‌دهد. پس از شیفت شدن، این مقادیر به مموری ماژول بر می‌گردند.

پس از تکرار این پروسه به ازای هر t، خروجی طی 64 کلاک از مموری داخلی به خارج فرستاده می‌شود.

## مسیر داده



## واحد کنترل



# ماژول Permute

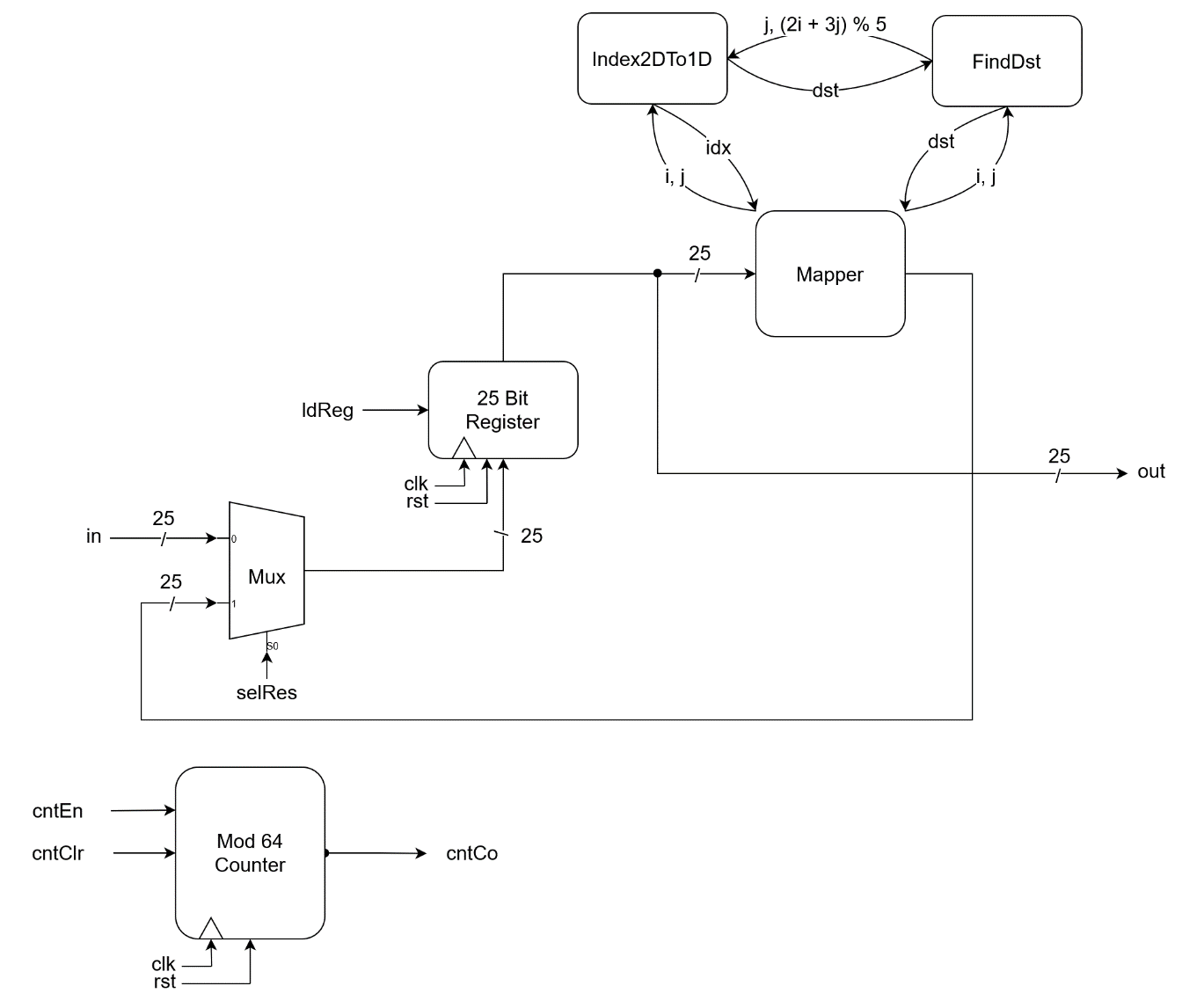
این ماژول در پروژه اول پیاده‌سازی شده و از همان در اینجا بدون تغییر استفاده شده است.

این تابع بر روی هر اسلایس ماتریس، نگاشت زیر را انجام می‌دهد:

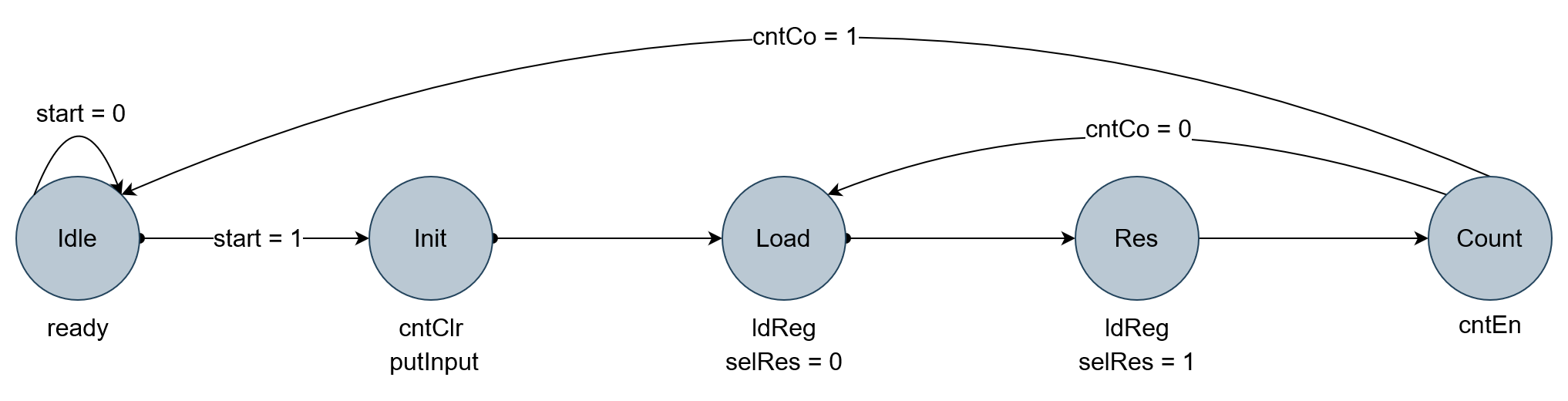
از یک رجیستر برای ورودی و خروجی اسلایس پس از انجام نگاشت استفاده شده است.

ماژول داخلی Mapper، کار نگاشت را انجام داده که از دو تابع Index2DTo1D برای تبدیل خانه ماتریس به اندیس خطی، و FindDst برای یافتن مقصد هر خانه پس از نگاشت استفاده می‌کند.

## مسیر داده



## واحد کنترل



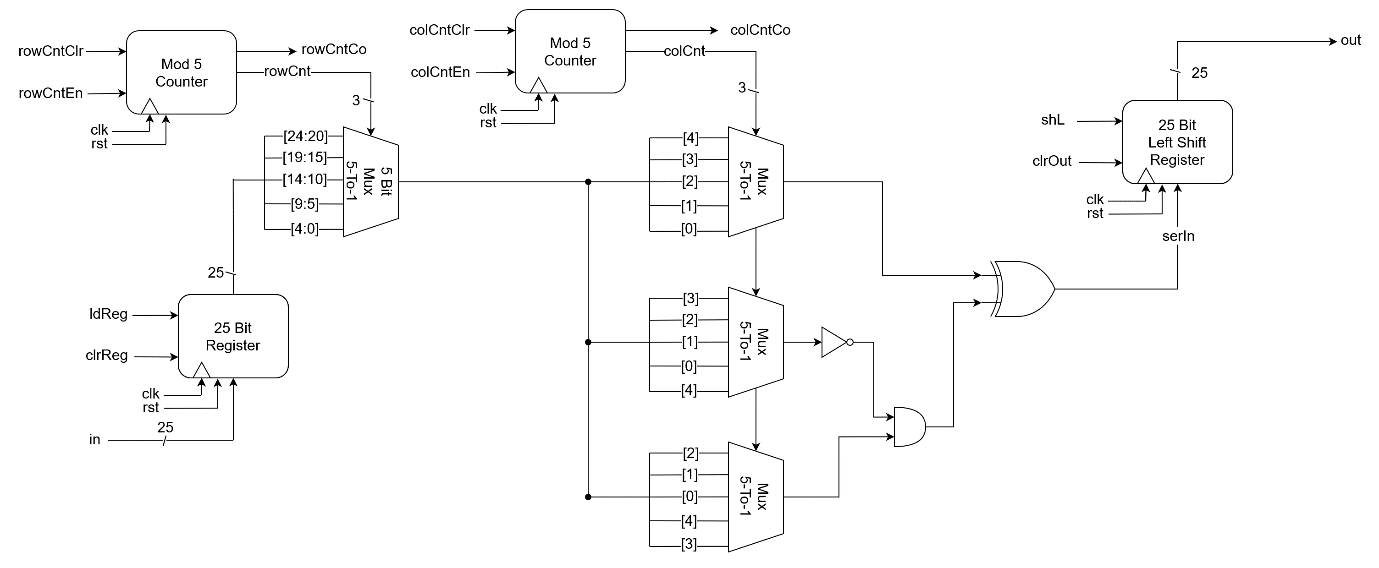
# ماژول Revaluate

این ماژول هر بیت ماتریس را به صورت زیر تغییر می‌دهد:

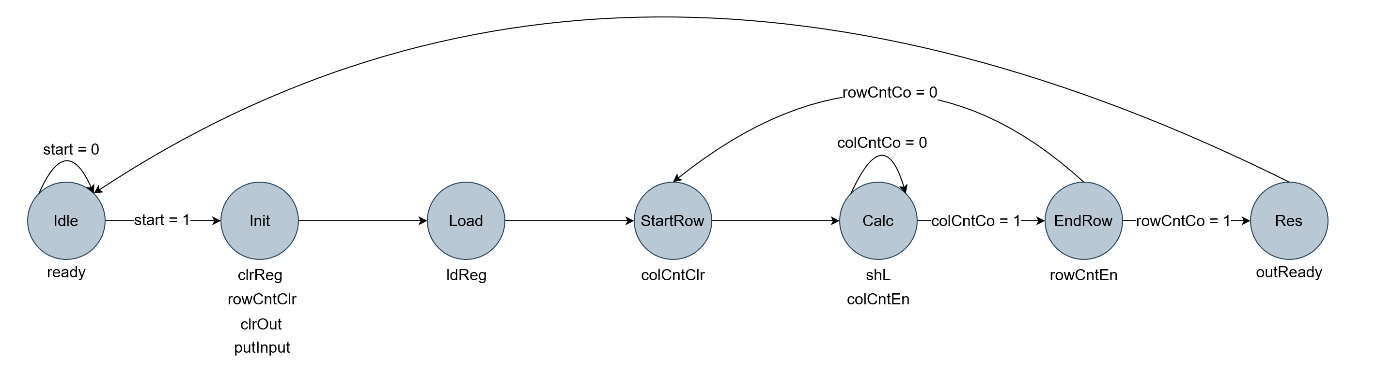
برای پیاده‌سازی این تابع غیر خطی، هر ردیف از اسلایس‌های ماتریس جدا در نظر گرفته شده و با استفاده از دو شمارنده (یکی برای ردیف‌ها و دیگری برای هر خانه از آن ردیف) پیمایش می‌شود.

مقدار محاسبه شده تابع وارد یک shift register شده که در نهایت تمام مقادیر جدید اسلایس ماتریس را خواهد داشت.

## مسیر داده



## واحد کنترل



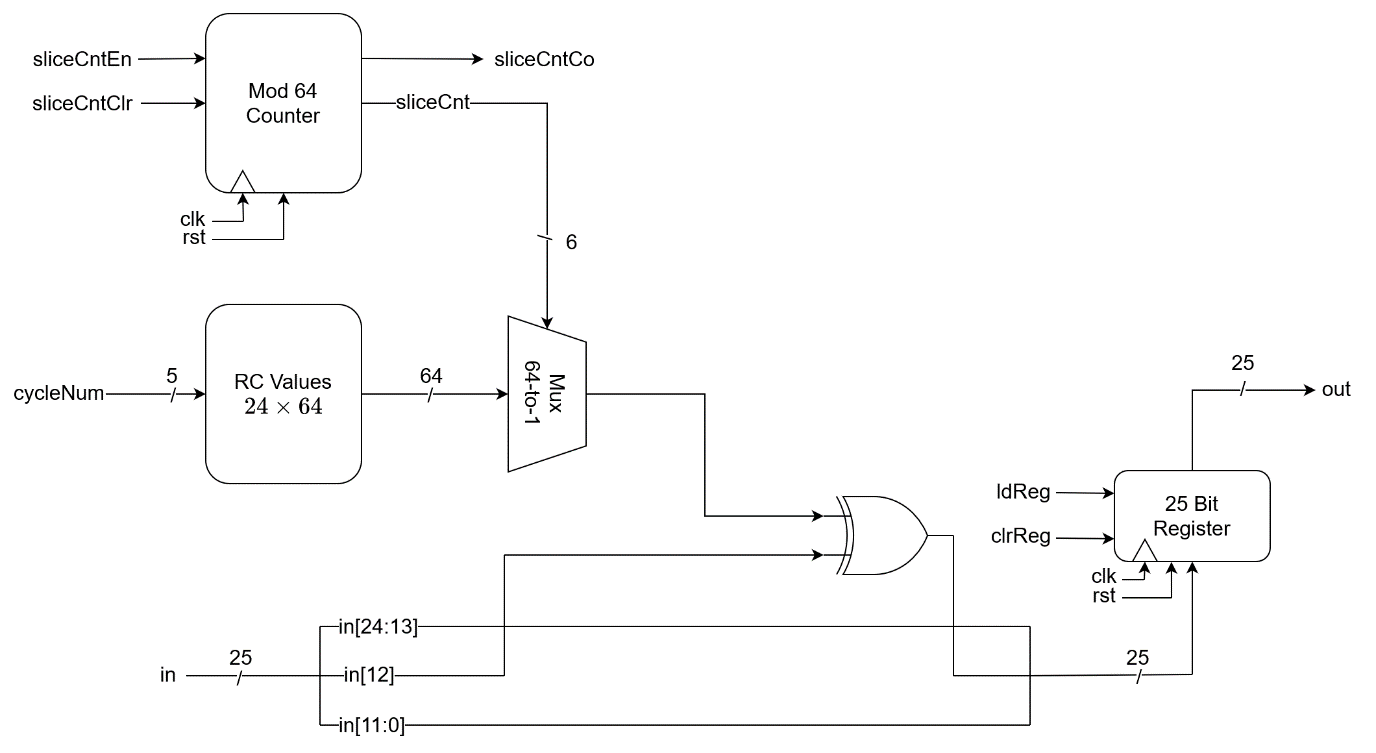
# ماژول AddRC

این ماژول lane وسط ماتریس یعنی خانه (0, 0) را با مقدار RC متناظر با دور cycleNum-ام ماژول تاپ-لول xor می‌کند.

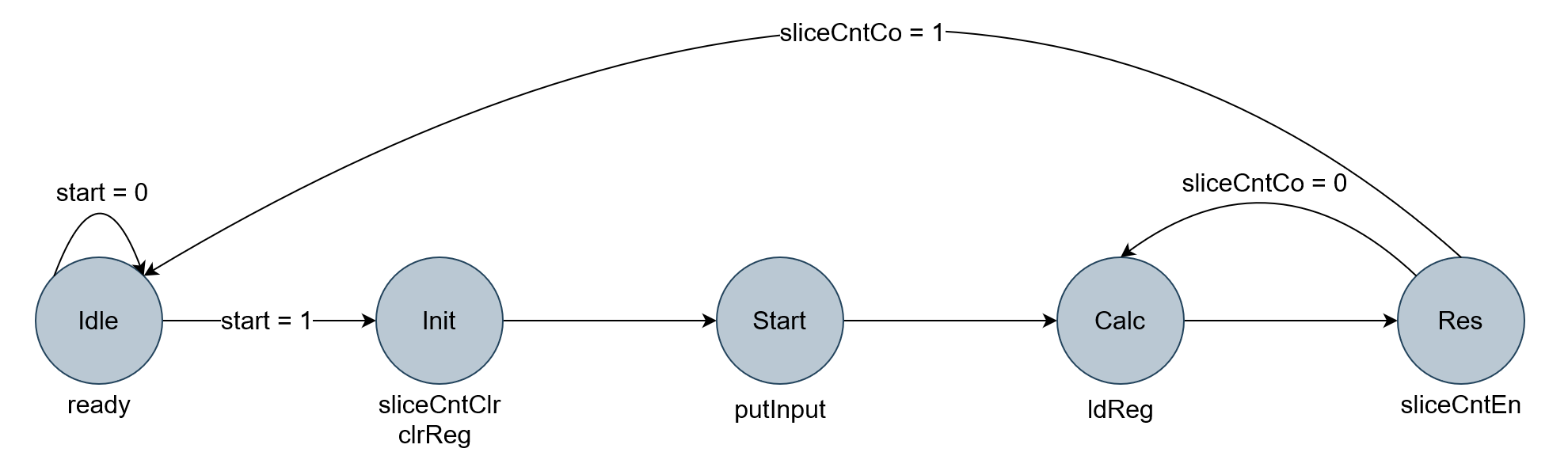
مقادیر RC در فایل rc.hex ذخیره شده و توسط مموری این ماژول خوانده می‌شود.

این ماژول 25 بیت یک slice ماتریس را دریافت کرده و خروجی تغییر یافته را در رجیستر خروجی ذخیره می‌کند. این کار تا گرفتن 64 ورودی انجام می‌شود.

## مسیر داده



## واحد کنترل



# نمونه اجرا

