

پروژه نهایی تحلیل و طراحی سیستم های دیجیتال

بنفشه قلی نژاد - نیکی مجیدی فرد

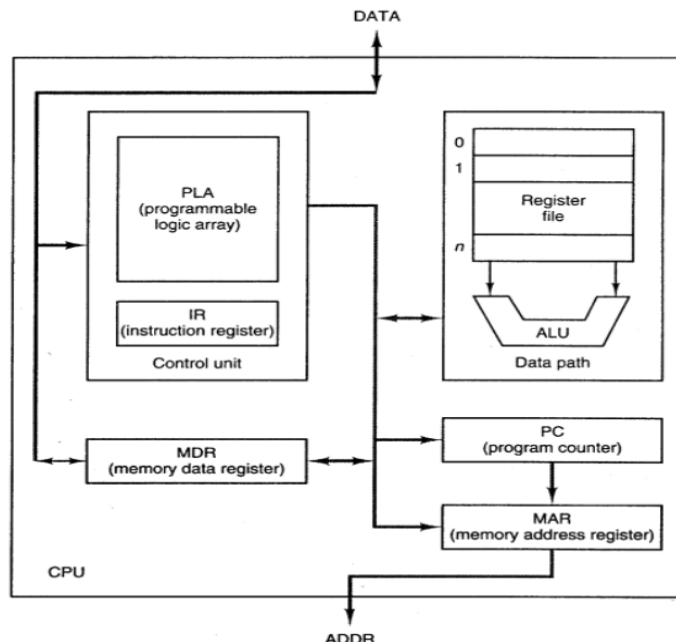
شرح پروژه

پردازنده مرکزی یا پردازنده اصلی یک مدار الکترونیکی است که با اجرای دستورالعمل هایی، یک برنامه کامپیوتری را تشکیل می دهد. این واحد عمل های منطقی، محاسباتی و کنترل تسک های مربوط به ورودی و خروجی را انجام می دهد. اجزای اصلی **cpu** عبارتند از:

- ✓ واحد منطق و محاسبه که برای انجام عملیات منطقی و محاسباتی است.
 - ✓ رجیستر که برای **ALU** عملوند فراهم می کند و نتایج عملیات را ذخیره می کند.
 - ✓ واحد کنترل که عمل واکشی دستورالعمل ها از حافظه و اجرای آنها را با هدایت کردن عملیات **ALU** به سازماندهی رجیسترها و سایر اجزا می پردازد.
- ما می خواهیم یک کامپیوتر رایپاده سازی کنیم که بخش های آن عبارتند از:
- ✓ بخش کنترل
 - ✓ بخش محاسبات
 - ✓ (بخش مموری) وظیفه نگه داشتن داده ها
- این بخش ها با **vhdl** پیاده سازی شده و برای بخش محاسبات برای عملیات موجود در آن نوشته شده است .

پیاده سازی

در بخش زیر برای هر کدام از فایل های پیاده سازی شده، شرحی نوشته شده است .
تصویر زیر مربوط به یک کامپیوتر پایه است.



فایل ALU

در این فایل با استفاده از **multiplexing** عملیات ریاضی و و منطقی پیاده سازی شده است. این ماژول دارای سه ورودی است که اولین و دومین آدرسی است که وارد ماژول می شود و سومین نیز نوع عملیاتی که باید روی آن دو پارامتر اجرا شود را مشخص می کند .

global_def

با توجه به استفاده ی زیاد از مقادیر ثابت، در این فایل تعریف شده تا قابل فهم تر باشد .

Control_Memory

در این بخش رجیستر های مربوط به دستور ها و PLA پیاده سازی شده است و به پیاده سازی حافظه موقت برای پردازنده پرداخته شده است .

Program_Counter

یک شمارنده به عنوان شمارنده دستورات پردازنده پیاده سازی شده است که نقش کنترل برنامه را عهده دار است.

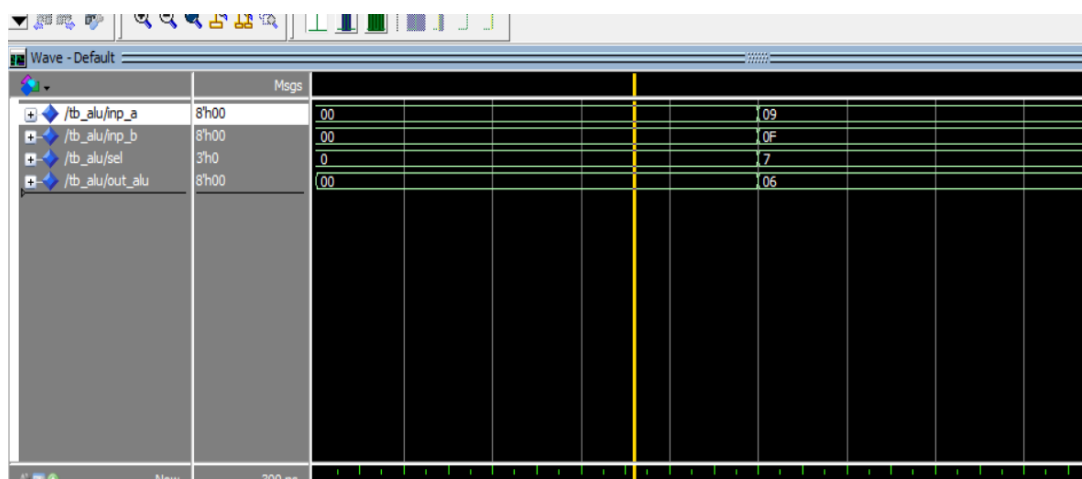
فایل Data_Path

این فایل انتقال داده از رجیستر ها به ALU را بر عهده دارد .

Main Computer

تمام اتصالات بخش های قبلی به صورت پورت مپ در این بخش ذخیره شده است.

نمایش خروجی



همان طور که مشخص است ، **R1a(first register)** مقدار 1001 ، **R1b(1111)** مقدار دوم را ذخیره می کند و در رجیستر سوم یعنی **inst** ، مقدار 111 که به معنای **Xor** هست ذخیره شده. و همان طور که مشاهده میشود پاسخ مطلوب و مورد نظر ماست.

پیاده سازی پردازنده ساده ۸ بیتی با کمک ماشین حالت

state	function	Operation	next
S0 - 000	Fetch instructions	Pc <= addreg +1 , adreg = data , oe <= 0 , data <= z	S1:Opcpde:10 , s2:Opcode :01 ,s3 : 00 : 11 ,s5, carry = 1 S0 :11,Crry = 0
S1-001	Write to memory	We <= 0 , data <= akku , adreg <= pc	S0
S2- 010	Read operand ADD	Oe <= 0 , data <= x , addreg <= pc , akku <= akku + data ,	S0
S3-011	Read operand NOR	Oe <= 0 , data <= z , adreg <= pc , akku <= akku nor data	S0
S5-101	Clear carry- Read pc	Carry <- 0, adreg <= pc	S0

نمایش خروجی :

