

## گزارش دستورکار نهم آزمایشگاه معماری کامپیوتر

نگار موقتیان، ۹۸۳۱۰۶۲

### ماژول ROM\_16x8

در این قسمت از آزمایش می‌خواهیم یک حافظه از نوع ROM طراحی کنیم، به صورتی که ۱۶ ردیف ۸ بیتی داشته باشد (در حقیقت در این حافظه  $\text{word} = 8 \text{ bit}$ ). مدار توصیف شده چنین شکلی خواهد داشت:

16	0	0	0	0	0	1	0	1
	0	0	0	0	0	1	1	0
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	0	0	0	1	0	0	1	1
	0	0	0	1	0	1	0	0
	word = 8 bit							



در ضمن می‌دانیم حافظه ROM یک حافظه فقط خواندنی است، بنابراین در ابتدای برنامه به خانه‌های آن مقداری پیش‌فرض می‌دهیم و پس از آن (پس از کلید واژه begin و همچنین در ماژول‌های دیگری که از این حافظه استفاده خواهند کرد) اجازه تغییر مقادیر موجود داخل این حافظه را نداریم. در این آزمایش برای سادگی بررسی درستی رفتار این ماژول، محتوای داخل هر خانه مانند شکل بالا، ۵ به علاوه شماره خانه (یا همان ردیف) در نظر گرفته شده است.

این حافظه ۱۶ خانه با شماره‌ها ۰ تا ۱۵ دارد و می‌دانیم این بازه از اعداد را می‌توان توسط یک عدد دودویی ۴ بیتی نمایش داد (از  $(0000)_2$  تا  $(1111)_2$ ) بنابراین برای آدرس دهی به این حافظه یک گذرگاه ۴ بیت کافی است. به عبارتی با دریافت یک آدرس ۴ بیتی (گذرگاه address)، محتوای خانه مربوطه از این حافظه ROM را به خروجی ۸ بیتی (گذرگاه data) متصل می‌نماییم.

با توجه به این توضیحات حافظه مورد نظر را پیاده سازی کرده و در test bench مربوطه، با دادن مقادیر مختلف به گذرگاه آدرس درستی عملکرد آن را بررسی می‌کنیم.

Name	Value	0 ns	20 ns	40 ns	60 ns	80 ns	100 ns	120 ns	140 ns
address[3:0]	0000	0000	0001	0010	0011	0100	0101	0110	0111
data[7:0]	00000101	00000101	00000110	00000111	00001000	00001001	00001010	00001011	00001100

می‌دانیم آدرس یک عدد بی علامت است و فرض می‌کنیم اعداد داخل ROM به صورت بی علامت ذخیره شده‌اند. بنابراین برای سهولت بررسی خروجی، ورودی و خروجی مدار را به صورت unsigned decimal نمایش می‌دهیم.

Name	Value	0 ns	50 ns	100 ns	150 ns	200 ns	250 ns	300 ns									
 address[3:0]	0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
 data[7:0]	5	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

همانطور که مشاهده می‌شود این مدار عملکرد درستی دارد و مقادیر پیش فرض موجود در ROM را با توجه به آدرس داده شده در خروجی انعکاس می‌دهد.