

حافظه‌های ROM و BRAM

مهدی حق‌وردی

۳۰ آبان ۱۴۰۲

فهرست مطالب

۱	توضیحات فایل‌های تمرین	۱
۱	فایل rom8x8	۱.۱
۱	توضیحات موجودیت rom8x8	۱.۱.۱
۲	توضیحات معماری behave	۲.۱.۱
۲	فایل romram	۲.۱
۲	موجودیت romram	۱.۲.۱
۳	معماری structural	۲.۲.۱
۳	فایل clockDivider	۳.۱

۱ توضیحات فایل‌های تمرین

۱.۱ فایل rom8x8

این فایل، ساده‌ترین فایل این تمرین است که شامل یک موجودیت به نام rom8x8 و یک معماری به نام behave است.

۱.۱.۱ توضیحات موجودیت rom8x8

این موجودیت شامل دو پورت است:

- addr •

این پورت یک ورودی std_logic_vector ۳ بیتی است که در واقع مقدار خروجی را تعیین می‌کند.

- dout •

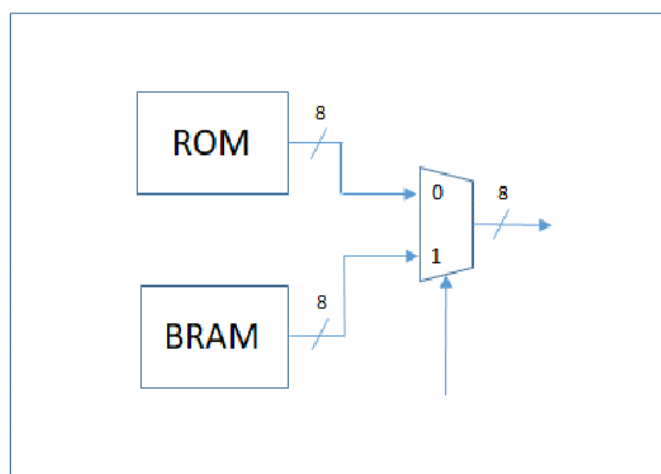
این پورت هم یک پورت خروجی از نوع std_logic_vector ۸ بیتی است که مقدار خروجی روی آن قرار می‌گیرد.

۲.۱.۱ توضیحات معماری behave

این تنها معماری حاضر برای موجودیت rom8x8 است که در واقع یک decoder ۳ به ۸ است. مقادیری که برای خروجی انتخاب شده‌اند کاملاً hard code شده‌اند و این کاملاً با نام موجودیت (که rom است) هماهنگی دارد. در معماری از language feature عی به نام select ... with استفاده شده است که با توجه به مقدار addr خروجی را روی dout قرار می‌دهد.

۲.۱ فایل romram

این فایل در واقع کل پروژه و کد این شکل است:



چرا کد این شکل است؟

چون در کد مشاهده می‌شود که مشخصاً موجودیت‌های rom8x8، clock_divide و bram8x8 در آن به عنوان یک component استفاده شده‌اند (که موجودیت‌هایی هستند که در شکل وجود دارند، البته مالتی پلکسری که در شکل است در قسمت معماری همین فایل تعریف شده است). کد شامل یک موجودیت به نام romram و یک معماری به نام structural است.

۱.۲.۱ موجودیت romram

موجودیت دارای دو ورودی و یک خروجی است:

- ورودی‌ها

– clk_100MHZ

از نوع std_logic و تبعاً یک بیت‌ست که کلاک سیستم است.

– switch

از نوع std_logic است که همان پایه‌ی انتخاب داخل شکل پروژه است.

- خروجی

– leds

یک پورت ۸ بیتی از نوع std_logic_vector است مقداری انتخاب شده روی آن قرار می‌گیرد.

۲.۲.۱ معماری structural

در قسمت معماری این فایل، ابتدا ۳ component ی که در این پروژه استفاده شده‌اند، تعریف شده‌اند و پس از آن ۵ سینگال میانی تعریف و مقداردهی اولیه شده‌اند. در قسمت begin این معماری، ابتدا ۳ component بالا اصطلاحاً port map شده و به سینگال‌های ورودی، میانی و خروجی وصل شده‌اند. معماری شامل یک process است که به سینگال clk_1Hz حساس است. سپس این فرآیند برای هر یک کلاک با مقدار ۱، با توجه به ورودی switch که ۰ باشد یا ۱، خروجی را به ترتیب با dout_rom و یا dout_bram پر می‌کند. سپس فارغ از مقدار switch یک شمارنده یا یکی اضافه می‌کند.

۳.۱ فایل clockDivider