

۱. یک بانک ثبات شامل ۳۲ ثبات ۳۲ بیتی پیاده‌سازی کنید. نوشتن در این بانک ثبات به صورت سنکرون انجام می‌شود به این معنی که اگر سیگنال **write** یک باشد، در لبه بالارونده کلاک، داده موجود بر روی ورودی **wrdata** در ثبات شماره **wraddr** نوشته می‌شود. در مقابل، خواندن به صورت آسنکرون انجام می‌شود به این معنی که بلافاصله بعد از صفر شدن سیگنال **write** (بدون نیاز به رسیدن لبه کلاک) محتوای ثبات‌های شماره **rdaddr1/2** به ترتیب بر روی خروجی‌های **rddata1/2** قرار می‌گیرد. زمانی که عملیات نوشتن انجام می‌شود خروجی **rddata1/2** صفر می‌شود. طول بیتی ورودی و خروجی را به درستی تنظیم کنید.

```
module regfile (input clk, input write, input rdaddr1, input rdaddr2, input wraddr, input wrdata, output rddata1, output rddata2);
```

۲. یک واحد محاسبه و منطق پیاده‌سازی کنید. عملیاتی که طبق ورودی **control** مشخص شده است بر روی دو ورودی ۳۲ بیتی **a** و **b** انجام شده و نتیجه در خروجی ۳۲ بیتی **c** درج می‌شود. اگر حاصل عملیات صفر باشد، خروجی **zero** یک می‌شود.

```
module alu (input control, input a, input b, output c, output zero);
```

Function	And	Or	Add	Xor	SLL	SRL	Subtract	SRA
Control	0000	0001	0010	0011	0100	0101	0110	0111

۳. یک حافظه شامل ۱۰۲۴ خانه یک بیتی (هشت بیتی) پیاده‌سازی کنید. نکات زیر را رعایت فرمایید.

(۱) این حافظه دو سیگنال جدا برای خواندن و نوشتن دارد؛ زمانی که یکی از این دو ورودی فعال است، عملیات مورد نظر انجام می‌شود و زمانی که هر دو فعال یا هر دو غیر فعال هستند، هیچ عملیاتی انجام نمی‌شود و تنها خروجی **rddata** برابر صفر می‌شود.

(۲) در زمان فعال بودن **write**، مقدار **rddata** صفر می‌شود.

(۳) خواندن به صورت آسنکرون و نوشتن به صورت سنکرون انجام می‌شود.

(۴) هر خانه این حافظه هشت بیت است اما پهنای نوشتن و خواندن ۳۲ بیت است. به این ترتیب اگر ورودی **addr** مقدار **x** را داشته باشد، عملیات خواندن یا نوشتن بر روی بایت‌های آدرس **x**، **x+1**، **x+2** و **x+3** انجام می‌شود. این حافظه **little endian** است و نحوه ساختن **rddata** و **wrdata** به صورت شکل زیر است.

```
module mem (input clk, input write, input read, input addr, input wrdata, output rddata);
```

mem [x+3]	mem [x+2]	mem [x+1]	mem [x]
-----------	-----------	-----------	---------

۴. یک واحد گسترش علامت پارامتری از **n** بیت به ۳۲ بیت پیاده‌سازی کنید. حداکثر مقدار **n** برابر ۳۱ است.

```
module signextend (input in, output out);
```