

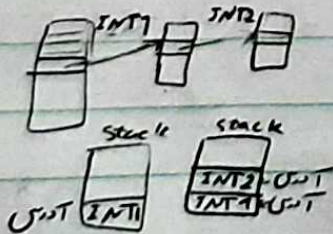
به نام خدا تمرین 1 OS

1. به نام CPU wire ای به نام interrupt-request دارد که در صورت وقوع INT، می‌فهمد. اگر CPU، INT ای دریافت کند دستور العمل فعلی را تمام می‌کند و دستور العمل واقع در شمار INT را شروع به انجام دادن می‌کند در همین حین CPU به ازای IO دستورات جدیدی را قبول می‌برد. با انجام کار دستگاه IO وقفه ای در device controller به پردازنده ارسال می‌شود و CPU باز هم دستور العمل فعلی را تمام کرده سپس ISR اجرای شود (در این هنگام CPU فعال نیست) و در نهایت با اجرای دستور IRET، CPU از اول دستور العمل هارا اجرا می‌کند.

	register	cache	M.M	SSD	Magnetic Disk
موجود	Compiler	hardware	operating system	OS	OS
بازگشت	cache	M.M	disk	disk	disk

کاربرد: register و cache و M.M، حافظه های فراد Primary (اولیه)، SSD حافظه های غیر فراد (secondary) و magnetic disk حافظه غیر فراد ثالث (tertiary) می باشد

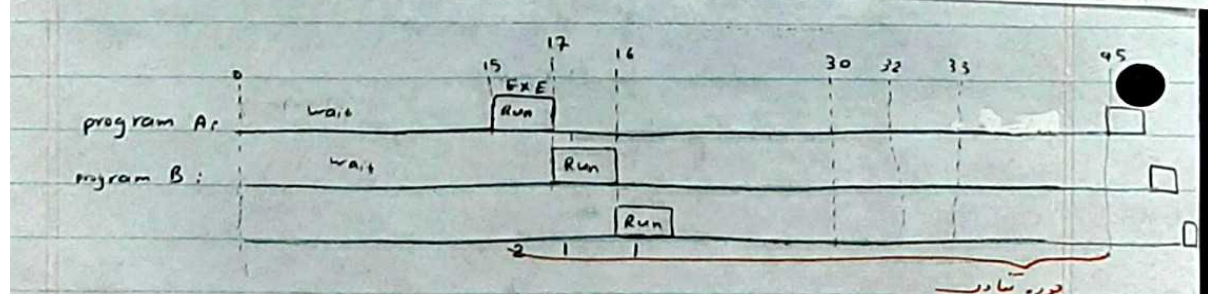
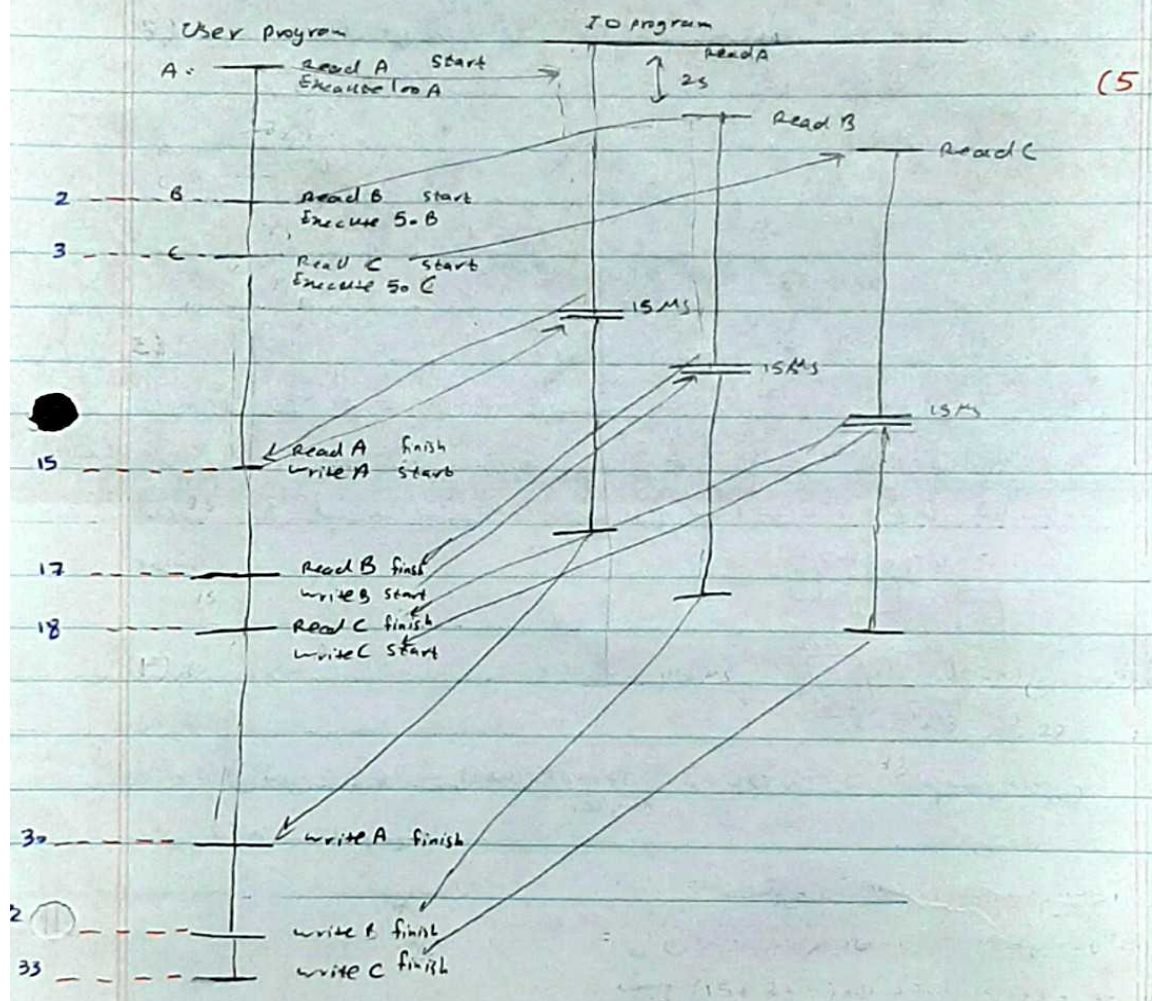
3. در هنگام وقوع وقفه program counter و status word در processor (در stack) پس می‌شود تا اطلاعات دستوری که در آن وقفه ایجاد شده است را ذخیره داشته باشیم. دلیل استفاده از زنجیره زنجیری از وقفه های تودرتو (nested interrupt) است که اگر در حین انجام عملیات مربوط به یک وقفه وقفه ای آمد و اطلاعات آن در زنجیره ذخیره شد، ISR های مربوط به هر وقفه اجرا شود.



در نتیجه در این حال با pop از زنجیره (stack) interrupt مربوط به INT2 ابتدا انجام شده و سپس برای INT2 انجام می‌شود.

4) NUMA همپوزنده ها با تاخیرهای متفاوت به حافظه دسترسی دارند که زمان دسترسی به محل حافظه نسبت به هم متفاوت دارد. به وقوع با ارتقاء فضای آدرس قطع به هم پردازنده ها اجازه می دهند مستقیماً تمام حافظه را آدرس دهی کنند که باعث جایابی سریع تر دیتا، کمتر شدن تعداد دستورات و رضای برنامه نویسی می شود. اما این روش حساب و محدودیت هایی نیز دارد از جمله هر یک rammer های سخت افزاری و تفاوت استاندارد های برنامه نویسی در سخت افزارهای بزرگ.

برای جریان آن FMS مجموعه ای از ابزارهای برنامه نویسی را فراهم کرده که برای مورد مواردی های NUMA قابل اجراست که thread binding (تخصیص ثابت هایتی اجرای thread ها در جایی که به فضای حافظه می دهند) memory placement (آر FMS فضای از حافظه را نشان کند که در آن پردازنده باشد) و طبق FMS به حداقل رساندن درخواست به دیتاهای دور از دسترسی است. در واقع سیستم عامل می تواند با زمان بندی و مدیریت حافظه محدودیت های NUMA را جریان کند.



CPU Utilization = $\frac{2+1+1}{32} = \frac{4}{32} = 12,5 \%$

(6) حفاظت صرفه نرخی برای کنترل دسترسی خرابی ها یا کار بران به منابع یک سیستم باشد. حفاظت می تواند با تشخیص خطاهای نمره در رابط بین مؤلفه ها، اطمینان را بهبود دهد. همچنین تشخیص زود هنگام خطاهای رابط می تواند اغلب از آلودگی زیر سیستم سالم توسط زیر سیستم دچار نقص جلوگیری کند. علاوه بر این یک منبع حفاظت نشده می تواند در برابر سوء استفاده کاربر غیر مجاز از خود دفاع کند.

با این حال یک سیستم می تواند از حفاظت کافی برخوردار باشد اما همچنان مستعد خرابی باشد و دسترسی نامناسب را امکان پذیر کند. وظیفه دفاع سیستم در برابر خطرات ناشی از سر دزدی و تلفات "امنیت" است.