

Computer Architecture

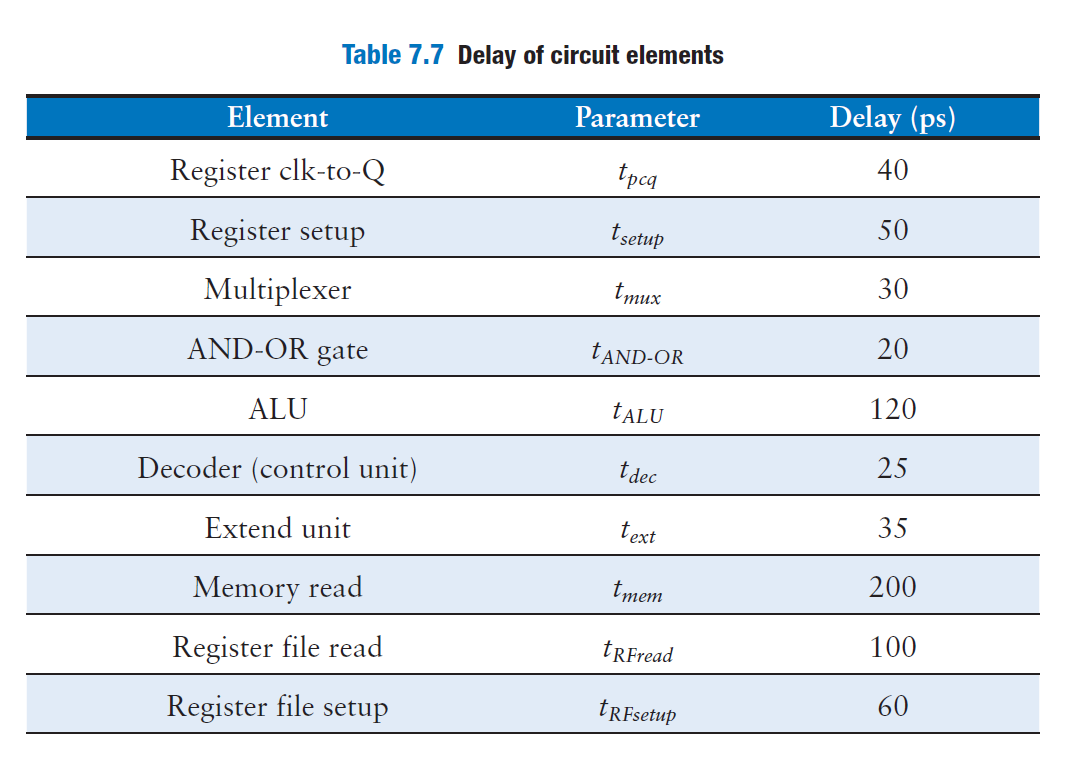
HW6 – G14

Mohammad Bahrami & Navid Raeeszadeh

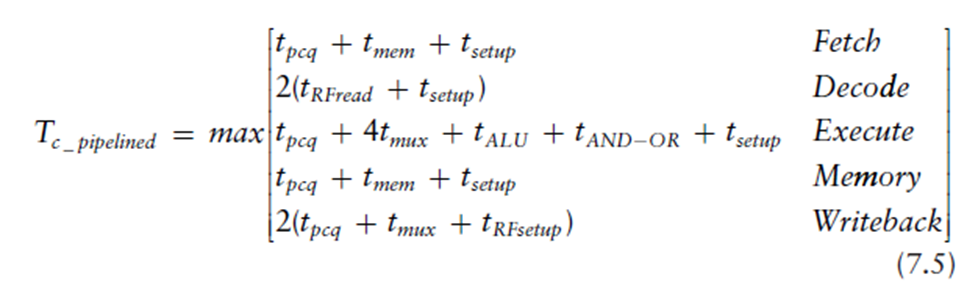
**گزارش سوال اول :**

طبق جدول 7.7 کتاب مقدار تاخیر120 پیکوثانیه است.

. الف)کاهش بیست درصدی: 96 پیکوثانیه



فرمول محاسبه کلاک پردازنده پایپ لاین:



مقدار کلاک سایکل پردازنده پایپ لاین به اندازه ی طولانی ترین استیج پایپ لاین است.

**Fetch = 40 + 200 + 50 = 290 ps**

**Decode = 2\*(100 + 50) = 300 ps**

**Execute = 40 + 4\*30 + 96 + 20 + 50 = 326 ps**

**Memory = 40 + 200 + 50 = 290 ps**

**Writeback = 2\*(40 + 30 + 60) = 260 ps**

طولانی ترین استیج همان استیج Execute باقی ماند ولی تاخیر آن 24 تا کاهش پیدا کرد و به 326 پیکوثانیه رسید که این مقدار تاخیر برابر Tc\_pipeline خواهد بود.

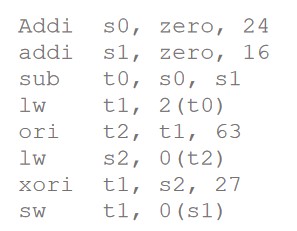
ب) افزایش 20 درصدی: 144 پیکوثانیه

تنها استیجی که تغییر خواهد کرد Execute خواهد بود چون مقدار ALU تنها در این استیج تاثیر گذار است.

**Execute = 40 + 4\*30 + 144 + 20 + 50 = 374 ps**

مقدار Tc\_pipeline برابر با 374 پیکوثانیه خواهد بود.

**گزارش سوال دوم:**



از سیکل پنجم شروع می­شود، هر کدام از دستورات addi و sub و ori و xori در طی یک سیکل ساعت انجام میشوند ولی دستور lw به دلیل stall در طی دو سیکل انجام میشود. پس مجموعا در 10 سیکل ساعت انجام میشود (هر خط یک سیکل، lw ها در دو سیکل) پس مجموعا 14 سیکل طول خواهد کشید که cpi برنامه با هشت دستور می­شود (8 / 14). Cpi = 1.75

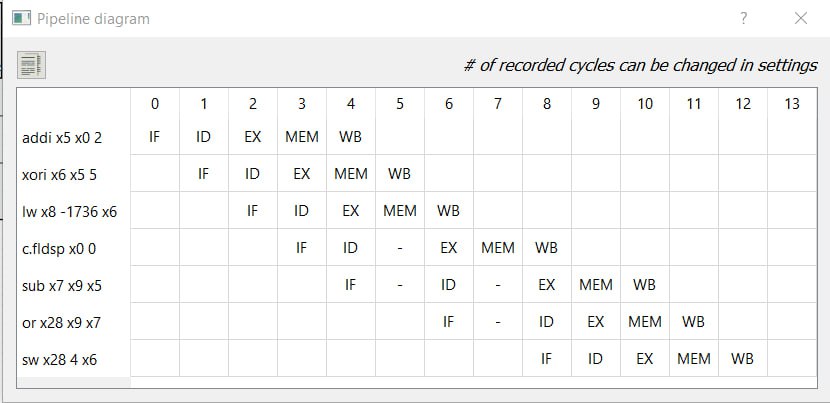
گزارش سوال سوم:

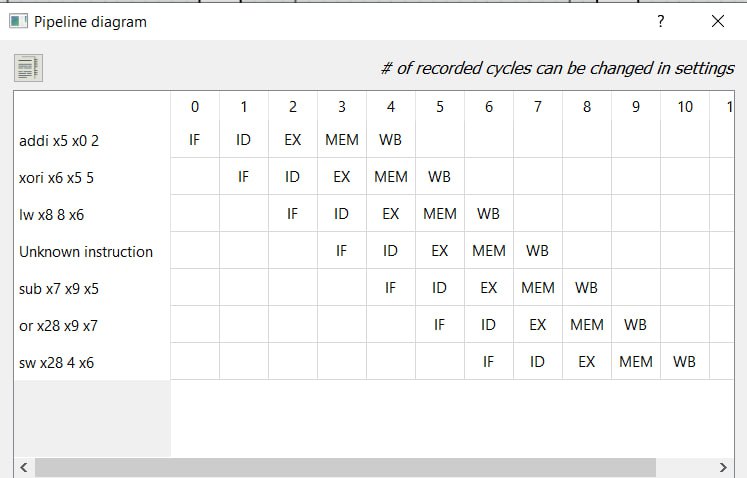
هنگامی که hazard را هندل نمی­کنیم کلا دستور اشتباه فهمیده می­شود و خروجی­ها تماما نادرست می­شوند. هزارد های بوجود آمده هنگام RAW رخ خواهند داد. بدون کنترل هزارد با فورواردینگ، در هنگام کامپایل عبارت nop قرار داده نمی­شود، بنابراین خروجی­های ما نادرست خواهند بود.

جهت هندل کردن هزارد از روش forwading استفاده کرد، به این صورت که دیتاها را از باس­های درونی به استیج­های اجرا میفرستیم. برای دستوری مانند lw هم که استفاده از روش forwarding امکان پذیر نیست از stalling استفاده میکنیم و بدین ترتیب latency آن دو سیکل خواهد شد.

در تصاویر زیر از خروجی­ها، با قابلیت رفع مخاطره و حالت درست آن با forwarding نمایش داده می­شود.



مقادیر درست و مطابق انتظار s0 و t3 با forwarding و رفع مخاطره





مقدار نادرست ذخیره شده در s0 و t3 بدون کنترل هزارد رخ داده

برای بخش ب، همان­گونه که در شکل مشخص شده در 5 جا به forwarding نیاز داشتیم، در دستورات (addi – xori – sub – or) و از آنجایی که فورواردینگ برای lw امکان­پذیر نمی­باشد، برای lw هم از stall استفاده میکنیم و چون دو دستور lw داریم بنابراین به دو دستور lw احتیاج داریم.

