



به نام خدا



دانشگاه تهران
دانشکده مهندسی برق و کامپیوتر
آزمایشگاه معماری کامپیوتر

گزارش کار

سیاوش شمس، محمدرضا توکلی	نام و نام خانوادگی اعضای گروه
۸۱۰۱۹۷۴۷۷، ۸۱۰۱۹۷۶۴۴	شماره دانشجویی اعضای گروه
۰۱/۰۴/۰۳	تاریخ ارسال گزارش

فهرست گزارش

۳	Arm
۶	Forwarding
۷	SRAM
۱۰	Cache

توضیح بخش های مختلف پردازنده ARM و شیوه پیاده سازی :

۱. Instruction Fetch

آدرس از رجیستر PC آمده و از instruction memory دستور خوانده شده و در لوله به جلو ارسال می شود. همچنین یک سیگنال freeze برای hazard detection وجود دارد.

۲. Instruction Decode

که یک بخش control unit دارد که وضعیت تمام سیگنال های کنترلی مانند memoryReadEn یا WBEn را مشخص می کند.

یک بخش condition check که وضعیت سیگنال های ZCV را بررسی می کند.

رجیستر فایل که دو آدرسی که از مقدار آنها در خروجی نمایش داده می شود و یک آدرس برای wb به آن داده می شود

۳. Execution Stage

یک بخش ALU که control unit در بخش قبل سیگنالهای کنترلی آن را تولید کرده

Val2generator که ورودی دوم را با توجه به مود آن تعیین می کند.

۴. Memory Stage

Memory address و read , written و یک سیگنال دیتا و خروجی دارد. که به صورت synchronous

داده ها در خروجی قرار میگیرند.

۵. WB Stage

از روی سیگنال MEM_R_EN متوجه می شویم که در مالتی پلکسر خروجی مموری یا ALU برای WB باید ارسال شود.

۶. HAZARD DETECTION UNIT

با توجه به نوع دستور و رجیستر های مورد استفاده سیگنال hazard detected تولید می شود.

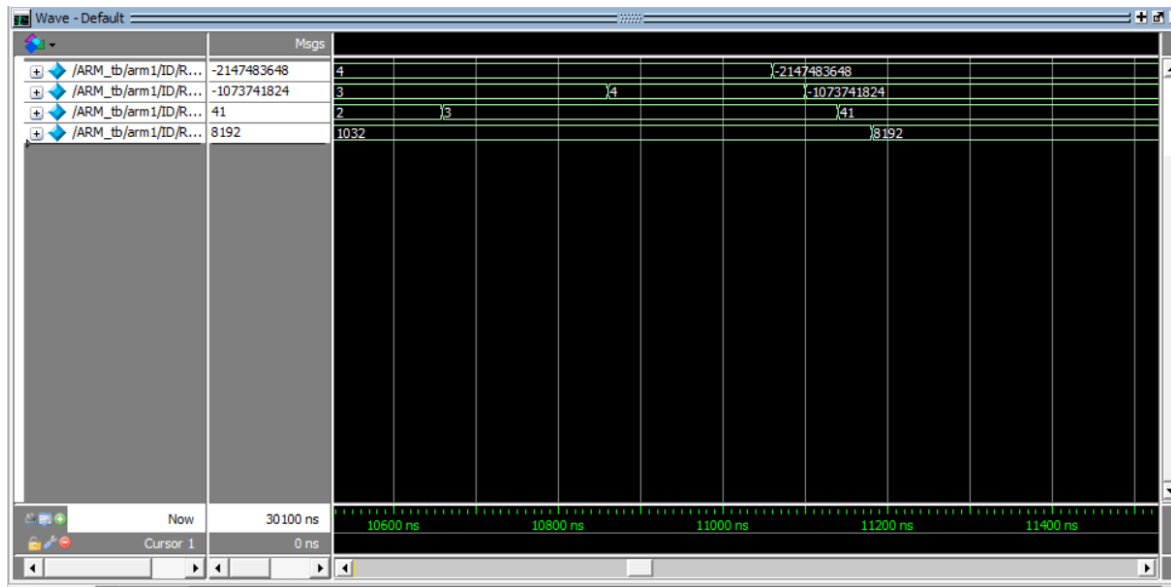
حالت های اتفاق افتادن Hazard:

Src1 = EXE_Des & EXE_WB_EN

Src1 = MEM_Dest & MEM_WB_EN

Src2 = EXE_Des & EXE_WB_EN&Two_SRC=1

Src2 = MEM_Det & MEM_WB_EN&Two_SRC=1



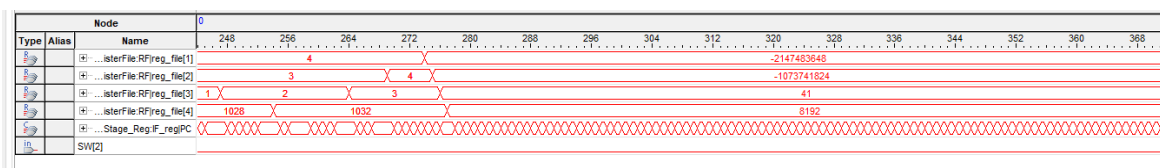
شکل شماره ۱ - خروجی arm در modelsim

$$clock\ cycles = \frac{11160}{40} = 279$$

$$CPI = \frac{clock}{instructions} = \frac{279}{182} = 1.532$$

Flow Status	Successful - Fri Jun 24 12:09:04 2022
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	test_armi
Top-level Entity Name	test_armi
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	7,873 / 33,216 (24 %)
Total combinational functions	4,723 / 33,216 (14 %)
Dedicated logic registers	5,964 / 33,216 (18 %)
Total registers	5964
Total pins	418 / 475 (88 %)
Total virtual pins	0
Total memory bits	82,944 / 483,840 (17 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

شکل شماره ۲ - نتایج سنتز در کوآرتس



شکل شماره ۳ - نتیجه تست روی برد

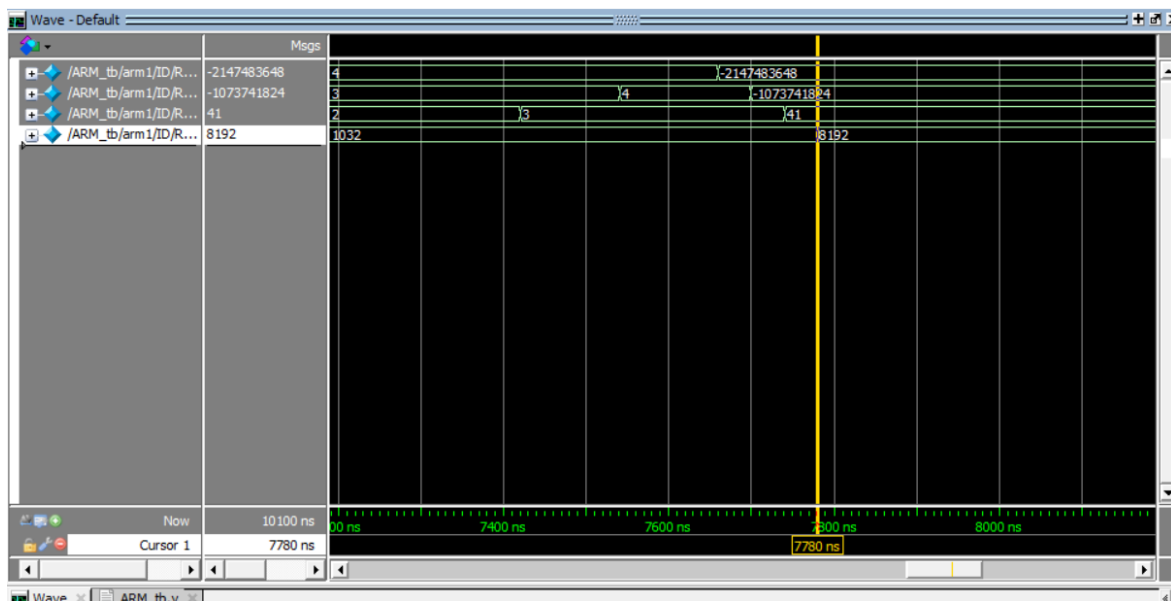
$$clock\ cycles = 279$$

$$CPI = \frac{clock}{instructions} = \frac{279}{182} = 1.532$$

Forwarding

Forwarding برای جلوگیری از stall ایجاد شده در hazard اضافه می شود با استفاده از آن می توان تاخیر را به جز در دستور لود حذف کرد حال آنکه برای دستور لود همچنان یک کلاک freeze داریم.

به این منظور ماژول forwarding unit اضافه شده که ورودی ها به WB_stage را به وسیله ی یک mux می تواند از خروجی ALU یا MEM_STAGE مستقیماً دریافت کند.



شکل شماره ۴ - خروجی modelsim forwarding

$$clock\ cycles = \frac{7780}{40} = 194.5$$

$$CPI = \frac{clock}{instructions} = \frac{194.5}{182} = 1.068$$

Flow Summary	
Flow Status	Successful - Fri Jun 24 13:23:02 2022
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	new_forwarding
Top-level Entity Name	new_forwarding
Family	Cyclone II
Device	EP2C35F672C6
Timing Models	Final
Total logic elements	6,878 / 33,216 (21 %)
Total combinational functions	4,553 / 33,216 (14 %)
Dedicated logic registers	4,883 / 33,216 (15 %)
Total registers	4883
Total pins	418 / 475 (88 %)
Total virtual pins	0
Total memory bits	50,176 / 483,840 (10 %)
Embedded Multiplier 9-bit elements	0 / 70 (0 %)
Total PLLs	0 / 4 (0 %)

شکل شماره ۵ - خروجی سنتز در کوآرتس forwarding

Type	Alias	Name	0	680	688	696	704	712	720	728	736	744	752	760	768	776	784	792	800
IO		SW[2]																	
IO		SW[3]																	
		ARM_Register1																	
		ARM_Register2																	
		ARM_Register3																	
		ARM_Register4																	
		IF_Stage_it_stagePC																	

شکل شماره ۶ - خروجی تست شده روی بورد forwarding

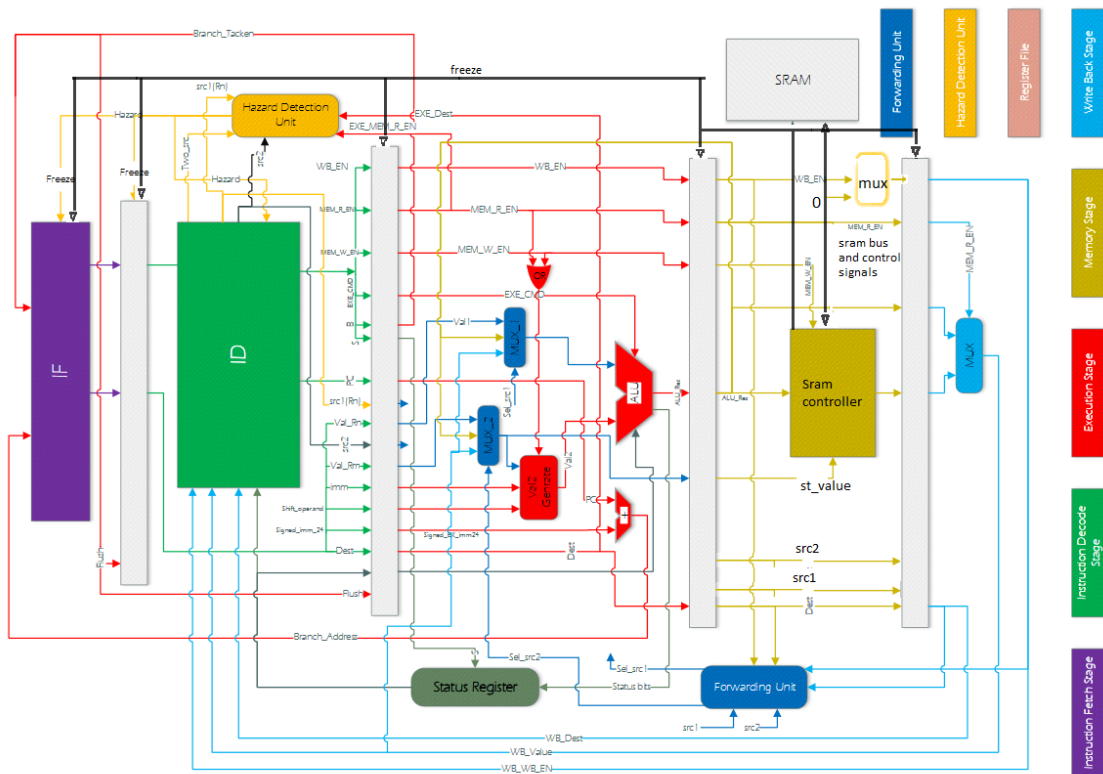
$$\text{clock cycles} = 256$$

$$CPI = \frac{\text{clock}}{\text{instructions}} = \frac{256}{182} = 1.4$$

با مقایسه ی cpi ها در بالا متوجه بهبود عملکرد نسبت به حالت arm بدون forwarding می شویم.

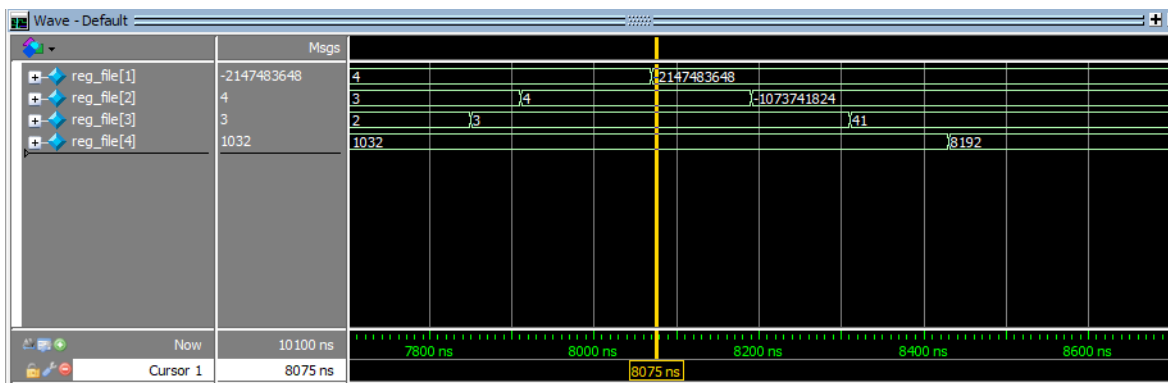
SRAM

حافظه داخلی پردازنده ما محدود است و در عمل ما از حافظه SRAM روی برد استفاده می کنیم، برای استفاده از این حافظه نیاز است تغییراتی در ساختار پردازنده اعمال کنیم که در ادامه به شرح این تغییرات می پردازیم

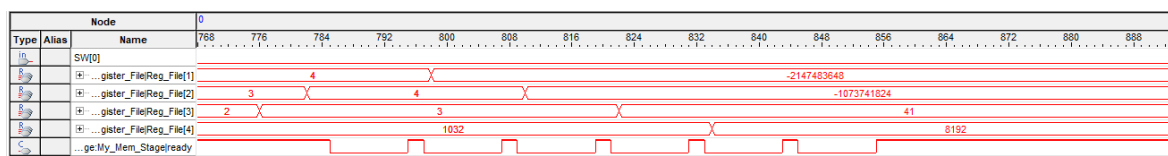


شکل ۷- مدار پردازنده به همراه SRAM در سطح RTL

ارتباط ما با حافظه SRAM از طریق سیگنال های کنترلی $SRAM_UB_N$, $SRAM_OE_N$, $SRAM_WE_EN$, $SRAM_LB_EN$, $SRAM_CE_EN$, $SRAM_WE_EN$ می باشد. که به غیر از $SRAM_WE_EN$ بقیه سیگنال های کنترلی را ۰ قرار می دهیم. این سیگنال کنترلی از طریق $SRAM_Controller$ به حافظه داده می شود. کنترلر حافظه شامل یک شمارنده می باشد که در صورت ۱ بودن سیگنال های ورودی mem_r_en یا mem_w_en شروع به شمارش می کند و اگر دستور ما از نوع write بود کنترلر سیگنال $SRAM_WE_E$ را صفر می کند و در طی دو سیکل و قرار دادن آدرس مربوطه روی باس address داده را روی SRAM می نویسد و اگر دستور ما از نوع read باشد کنترلر سیگنال $SRAM_WE_E$ را یک می کند و در طی دو سیکل و قرار دادن آدرس مربوطه روی باس address داده را از روی SRAM می خواند. همچنین در طی مراحل خواندن و نوشتن نیاز است تا سیگنال ready برابر صفر باشد، این سیگنال همان freeze است که در طی عملیات خواندن و نوشتن خط لوله را متوقف می کند.



شکل ۸- شکل موج پردازنده به همراه SRAM



شکل ۹- شکل موج پردازنده به همراه SRAM با تست روی برد

همانطور که می بینیم نتیجه برنامه در تعداد کلاک بیشتری آماده شده است و در نتیجه کارایی پردازنده کاهش می یابد.

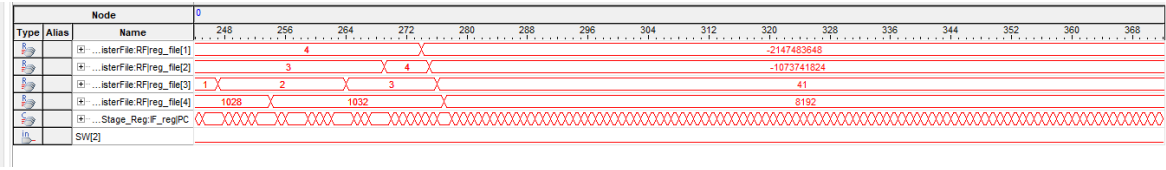
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	sram_report
Top-level Entity Name	sram_report
Family	Cyclone IV GX
Total logic elements	2,648
Total combinational functions	2,159
Dedicated logic registers	815
Total registers	815
Total pins	418
Total virtual pins	0
Total memory bits	0
Embedded Multiplier 9-bit elements	0
Total GXB Receiver Channel PCS	0
Total GXB Receiver Channel PMA	0
Total GXB Transmitter Channel PCS	0
Total GXB Transmitter Channel PMA	0
Total PLLs	0

شکل ۱۰- سخت افزار مورد استفاده

$$clock\ cycles = \frac{8430}{20} = 422$$

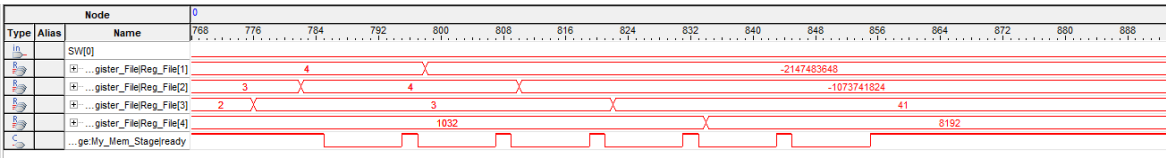
$$CPI = \frac{clock}{instructions} = \frac{422}{182} = 2.32$$

Cache



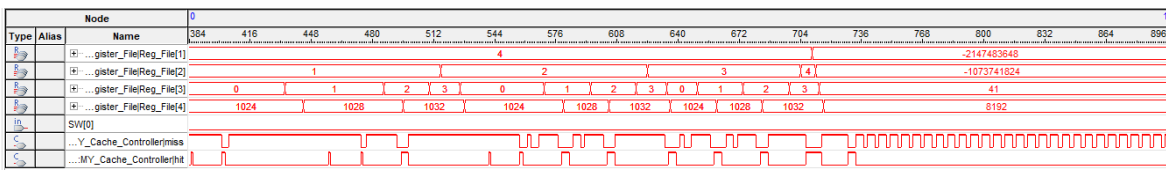
شکل ۱۱- شکل موج پردازنده ARM با تست روی برد

$$CPI = \frac{\text{clock}}{\text{instructions}} = \frac{279}{182} = 1.532$$



شکل ۱۲- شکل موج پردازنده به همراه SRAM با تست روی برد

$$CPI = \frac{\text{clock cycles}}{\text{instructions}} = \frac{834}{182} = 4.58$$



شکل ۱۳- شکل موج پردازنده به همراه SRAM و Cache با تست روی برد

$$CPI = \frac{\text{clock}}{\text{instructions}} = \frac{705}{182} = 3.87$$

Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Web Edition
Revision Name	test_cache_rep
Top-level Entity Name	test_cache_rep
Family	Cyclone IV GX
Total logic elements	19,441
Total combinational functions	18,913
Dedicated logic registers	856
Total registers	856
Total pins	418
Total virtual pins	0
Total memory bits	0
Embedded Multiplier 9-bit elements	0
Total GXB Receiver Channel PCS	0
Total GXB Receiver Channel PMA	0
Total GXB Transmitter Channel PCS	0
Total GXB Transmitter Channel PMA	0
Total PLLs	0

شکل ۱۴- سخت افزار مورد استفاده

همانطور که می بینیم با اضافه کردن cache کارایی پردازنده بهبود یافته است اما هزینه سخت افزاری افزایش داشته است. و با مقایسه ۳ حالت فوق می بینیم که کارایی پردازنده هنگام استفاده از حافظه داخلی پردازنده از بقیه حالت ها بیشتر است و همچنین سخت افزار کمتری استفاده می شود اما دیدیم که حافظه داخلی پردازنده محدودیت هایی دارد و معمولاً از حافظه بیرونی استفاده می شود. استفاده از SRAM همراه Cache سرعت بهتری نسبت به SRAM تنها دارد اما هزینه سخت افزاری آن بیشتر است.

