

فاز ۴

ما در این فاز کمک پردازنده ای برای محاسبه ی دستورات floating point طبق استاندارد IEEE 754 پیاده سازی کردیم. در ابتدا دستورات پشتیبانی شده را مطرح می کنیم. در جلوی هر مورد دستور مورد نیاز را می نویسم. سعی کرده ام که دستورات برای assembler استاندارد باشد. دستوراتی بوده اند که معادل نداشته اند و از یک دستور استاندارد دیگر استفاده کردم که در تفسیر کد راحت تر باشیم.

- جمع: ۲ ثبات floating point با هم جمع می شوند و خروجی در ثباتی دیگر ذخیره می شود:

add.s \$f1,\$f2,\$f3

- تفریق: ۲ ثبات floating point از هم تفریق می شوند و خروجی در ثباتی دیگر ذخیره می شود.

sub.s \$f1,\$f2,\$f3

- ضرب: ۲ ثبات floating point در هم ضرب می شوند و خروجی در ثباتی دیگر ذخیره می شود.

mul.s \$f1,\$f2,\$f2

- تقسیم: ۲ ثبات floating point بر هم تقسیم می شوند و خروجی در ثباتی دیگر ذخیره می شود.

div.s \$f1,\$f2,\$f3

- قرینه: مقدار یک ثبات floating point قرینه می شود و در ثبات دیگر ذخیره می شود.

neg.s \$f1,\$f2

- انتقال از ثبات پردازنده ی اصلی به فرعی: محتویات یک ثبات اصلی به صورت خام به یکی از ثبات های floating point می رود.

mtc1 \$1,\$f2

- انتقال از ثبات پردازنده ی فرعی به اصلی: محتویات یک ثبات فرعی به صورت خام به یکی از ثبات های اصلی می رود.

mfc1 \$1,\$f2

- گرد کردن به نزدیک ترین عدد صحیح: محتویات یک ثبات فرعی که اعشاریست به نزدیک

ترین عدد صحیح گرد می شود. Round.w.s \$f1,\$f2

- مقایسه ی ۲ ثبات فرعی و در صورت برابری ۱ شدن ثبات دیگر: در این دستور ۲ ثبات فرعی برابریشان چک می شود و همانند دستور slt خروجی مقایسه در ثبات دیگر ذخیره می شود.

add.d \$f1,\$f2,\$f2

- مقایسه ی ۲ ثبات فرعی و در صورت کوچک تر مساوی بودن ۱ شدن ثبات دیگر: در این دستور ۲ ثبات فرعی کوچک تر مساوی بودن چک می شود و همانند دستور slt خروجی مقایسه در ثبات دیگر ذخیره می شود.

sub.d \$f1,\$f2,\$f3

در ابتدا سعی کردیم ALU برای این دستورات بنویسیم. از آن جا که single cycle پیاده سازی شد دیگر نیازی به stall کردن پردازنده نبود. در مرحله ی بعد alu control متناسب با دستورات بروزرسانی شد. مرحله ی بعد زدن co-processor جدا و قرار دادن واحد های alu و regfile درون آن بود. در مرحله ی آخر اتصالات برقرار شد. نکاتی که در اتصالات باید مدنظر می گرفتیم وجود pipeline بود که کار را دشوار می کرد.

در انتها واحد signal control برپرزسانی شد تا این پیاده سازی این فاز تمام شود.

برای تست کردن آن ابتدا یک کد اسمبلی حاوی تمامی دستورات زده شد که در کامنت ها مقدار واقعی و بدون تقریب نوشته شده است و به صورت دستی چک کردیم و دیدیم درست است.

و در انتها کد خواسته شده را پیاده سازی کردم. به این صورت که حاصل a/b و b/a و گرد شده ی آن ها محاسبه شد. سپس به regfile پردازنده اصلی منتقل شد چون ما دستورات پرشی روی ثبات های floating point نداشتیم. سپس روی مقادیر انتقال داده شده مقایسه و پرش انجام دادیم تا خروجی درست محاسبه شود.