دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر

درس فناوری های حافظه
دکتر حامد فربه

رضا آدینه پور
۴۰۲۱۳۱۰۵۵

تمرین شبیه‌سازی چهارم

تدریسیار:

مرتضی عادلخانی (madelkhani@aut.ac.ir)
سارا زمانی (sara.zamani۷۳@aut.ac.ir)

CE۵۴۳۱ | بهار ۱۴۰۳

سوالات تئوری

۱. به سوالات زیر پاسخ دهید:

۱. PUM چیست و کدام نوع حافظه‌ها برای آن بیشتر استفاده می‌شوند؟ توضیح دهید چرا هر نوع حافظه استفاده می‌شود.

پاسخ: پردازش در حافظه (PUM) یک کانسپت محاسباتی است که در آن برخی از محاسبات ساده مانند جمع و ضرب به جای انتقال داده‌ها بین CPU و حافظه، مستقیما در حافظه انجام می‌شوند.

معمولا از DRAM، SRAM و NVM ها در PUM استفاده می‌شود. که در ادامه به بررسی مزایا و معایب استفاده از هرکدام می‌پردازیم.

DRAM ها به دلیل اینکه رایج‌ترین نوع حافظه فرار با تراکم بالا و هزینه کم به ازای هر بیت هستند، به طور گسترده استفاده می‌شود. ویژگی‌های خازنی سلول‌های DRAM امکان انجام تکنیک‌های محاسباتی درون حافظه مانند عملیات منطقی و حسابی را فراهم می‌کند.

مزایا: تراکم بالا، ارزان است.

معایب: فرار، نیاز به تازه‌سازی دوره‌ای و معمولاً تأخیر بیشتر نسبت به SRAM.

اما در مقابل SRAM زمان‌های تأخیر کمتری و زمان دسترسی سریعتری نسبت به DRAM دارد و در مواردی که سرعت برای ما بسیار مهم است، (مانند Cache)، استفاده می‌شود. توانایی حفظ حالت بدون نیاز به تازه‌سازی، آن را برای عملیات‌های PUM مناسب می‌سازد.

مزایا: زمان‌های دسترسی سریع نسبت به DRAM، عدم نیاز به تازه‌سازی.

معایب: تراکم کمتر و هزینه بیشتر به ازای هر بیت نسبت به DRAM.

درمقابل حافظه‌های فرار، انواع حافظه‌های غیر فرار مانند Flash، PCM و ReRAM به دلیل نگه داشتن داده بدون برق، برای ذخیره‌سازی پایدار و محاسبات مناسب هستند. این حافظه‌ها می‌توانند برخی عملیات منطقی را درون سلول‌های حافظه انجام دهند.

مزایا: غیر فرار بودن.

معایب: عموماً سرعت نوشتن کندتر و دوام کمتر نسبت به DRAM و SRAM.

۲. نقاط ضعف UPMEM چیست؟

پاسخ: از نقاط ضعف UPMEM ها می‌توان به موارد زیر اشاره کرد:

(آ) انعطاف‌پذیری و قابلیت برنامه‌ریزی محدود:
معماری PIM UPMEM برای انواع خاصی از عملیات (مانند وظایف داده‌محور مانند جست‌وجو در پایگاه داده و تحلیل) است. ممکن است به اندازه CPU یا هایGPU سنتی عمومی و همه‌منظوره نباشد، که کاربرد آن را به بارهای کاری خاص محدود می‌کند.

(ب) یکپارچه‌سازی و سازگاری:
یکپارچه‌سازی ماژول‌های PIM UPMEM با سیستم‌های موجود می‌تواند چالش‌برانگیز باشد. ممکن است مشکلات سازگاری با معماری‌های حافظه و پردازنده فعلی به وجود بیاید که نیاز به اصلاحات در کانفیگ نرم‌افزار و سخت‌افزار دارد.

(ج) مسائل مربوط به کارایی انرژی:
در حالی که PIM هدفش کاهش مصرف انرژی با حداقل‌کردن حرکت داده‌ها بین حافظه و CPU است، صرفه‌جویی واقعی در انرژی می‌تواند وابسته به بار کاری باشد. برخی عملیات ممکن است همچنان مصرف انرژی قابل توجهی داشته باشند، به خصوص اگر منطق PIM به‌طور کامل برای آن کاربرد به‌خصوص بهینه‌سازی نشده باشد.

(د) توسعه و اشکال‌زدایی:
همانطور که در کلاس هم بررسی شد، توسعه برنامه‌ها برای PIM نیاز به مدل‌های برنامه‌نویسی و

ابزارهای جدید دارد. دیباگ و پروفایل‌کردن برنامه‌های PIM می‌تواند به دلیل طبیعت توزیع‌شده و درون حافظه‌ای محاسبات سخت‌تر از برنامه‌نویسی CPU/GPU سنتی باشد.

۳. ساختار Ambit را معرفی کرده و مزایا و معایب آن را توضیح دهید.

پاسخ: Ambit یک معماری PIM است که از بستر DRAM موجود برای انجام عملیات بیتی (مانند AND، OR، NOT) به طور مستقیم درون حافظه استفاده می‌کند. این معماری از ویژگی‌های آنالوگ سلول‌های DRAM و Sense Amplifier برای اجرای این عملیات استفاده می‌کند.

از مزایای آن می‌توان به موارد زیر اشاره کرد:

(آ) کاهش حرکت داده‌ها:
با انجام محاسبات مستقیما درون DRAM به طور قابل توجهی نیاز به حرکت داده بین CPU و حافظه را کاهش می‌دهد، که منجر به کاهش تاخیر و مصرف انرژی می‌شود.

(ب) توان محاسباتی بالا:
Ambit می‌تواند عملیات بیتی را بر روی حجم زیادی از داده‌ها به طور همزمان انجام دهد، که توان محاسباتی بالایی برای وظایف داده‌محور مانند جستجوهای پایگاه داده، رمزنگاری و شبکه‌های عصبی فراهم می‌کند.

(ج) تغییرات سخت‌افزاری حداقلی:
Ambit از زیرساخت DRAM موجود با تغییرات حداقلی استفاده می‌کند، که ادغام آن را با سیستم‌های فعلی نسبت به معماری‌های PIM آسان‌تر می‌کند.

همچنین از معایب Ambit می‌توان به موارد زیر اشاره کرد:

(آ) محدودیت در انواع عملیات:
Ambit به طور عمده برای عملیات بیتی طراحی شده است. نمی‌تواند به طور کارآمد محاسبات پیچیده‌تر ریاضی یا اعشاری را انجام دهد، که کاربرد آن را به انواع خاصی از عملیات ها محدود می‌کند.

(ب) پیچیدگی در برنامه‌نویسی:
برنامه‌نویسی برای Ambit نیاز به درک مدل عملیاتی خاص و محدودیت‌های آن دارد. برنامه‌نویس ها باید الگوریتم‌های خود را برای استفاده موثر از عملیات بیتی تطبیق دهند، که می‌تواند پیچیدگی نرم‌افزاری را افزایش دهد.

(ج) چالش‌های مقیاس‌پذیری:
در حالی که Ambit توان محاسباتی بالایی برای عملیات بیتی ارائه می‌دهد، مقیاس‌پذیری آن به سیستم‌های حافظه بزرگ‌تر یا ادغام آن با واحدهای پردازشی دیگر ممکن است چالش‌هایی از نظر هماهنگی و مدیریت ایجاد کند.

سوالات شبیه‌سازی

۲. در این بخش از تکلیف خود، شما با شبیه‌ساز MNSIM 2.0 برای پیاده‌سازی یک شتاب‌دهنده شبکه عصبی سر و کار خواهید داشت. بنابراین، ابتدا باید این شبیه‌ساز را دانلود کرده و نتایج را طبق درخواست ارائه دهید.

پیکربندی پایه:

۱. پارامترهای شبکه‌عصبی VGG8 که بر روی مجموعه داده CIFAR-10 آموزش دیده است را دانلود کنید، همانطور که در راهنمای MNSIM 2.0 توضیح داده شده است.

۲. برای هر اجرا، باید VGG8 را به عنوان شبکه عصبی مورد نظر خود انتخاب کنید.

پاسخ: وزن ها را از اینجا دانلود می‌کنیم.

سوال ۱:

همانطور که در کلاس یاد گرفتید، ساختار PIM شامل تعدادی کاشی است و هر کاشی شامل تعدادی PE است. در هر PE، ما مدارهای ضروری و ساختار ضربدری سلول‌های حافظه را داریم. اگر اندازه ضربدری را کاهش دهیم چه اتفاقی می‌افتد؟ به عنوان مثال، آیا باید انتظار داشته باشیم که توان، تأخیر و دقت کاهش، افزایش یا تغییر نکند؟ پاسخ خود را به تفصیل توضیح دهید.

پاسخ: کاهش اندازه‌ی ساختار کراسبار در معماری PIM شامل چندین موازنه می‌شود و بر پارامترهای مختلفی مانند توان مصرفی، تأخیر و دقت تأثیر می‌گذارد. در ادامه به بررسی و تاثیر هر کدام می‌پردازیم:

۱ توان مصرفی

- **کاهش توان مصرفی:** کراسبارهای کوچکتر معمولاً به کاهش توان مصرفی منجر می‌شوند. این امر به این دلیل است که کراسبار کوچکتر تعداد سلول‌های حافظه و اتصالات کمتری دارد، که منجر به کاهش کلی ظرفیت خازنی می‌شود. ظرفیت کمتر به معنی شارژ/دشارژ کمتر در طول عملیات است که به توان مصرفی دینامیک کمتری منجر می‌شود.
- **ملاحظات توان استاتیک:** با این حال، توان مصرفی استاتیک ممکن است به همان اندازه کاهش نیابد، به ویژه اگر به همان تعداد مدارهای جانبی (مانند Sense amplifier و درایورها) برای کراسبار کوچکتر نیاز باشد. کاهش در توان استاتیک عموماً کمتر از کاهش در توان دینامیک است.

۲ تأخیر

- **کاهش تأخیر:** کراسبارهای کوچکتر می‌توانند تأخیر را بهبود بخشند. زمان خواندن یا نوشتن داده در یک ساختار کراسبار به طول اتصالات و تعداد سلول‌های حافظه وابسته است. با کراسبار کوچکتر، سیگنال‌ها باید مسافت کمتری را طی کنند و سلول‌های کمتری برای شارژ یا دشارژ وجود دارند، که منجر به عملیات سریعتر می‌شود.

۳ دقت

- **بهبود بالقوه در دقت:** کراسبارهای کوچکتر می‌توانند منجر به بهبود دقت شوند. کراسبارهای بزرگتر با مشکلاتی مانند افزایش مقاومت و ظرفیت خازنی در طول اتصالات مواجه می‌شوند، که می‌تواند باعث تخریب سیگنال و افزایش حساسیت به نویز شود. با کاهش اندازه‌ی کراسبار، این اثرات به حداقل می‌رسند، که منجر به انتقال سیگنال و حسگری قابل اعتمادتر می‌شود و می‌تواند دقت را بهبود بخشد.
- **نرخ خطا:** احتمال خطا به دلیل تداخل و سایر اثرات نیز در کراسبارهای کوچکتر کمتر است، که به بهبود دقت در عملیات‌هایی مانند ضرب ماتریسی و عملیات‌های برداری که به طور معمول در معماری PIM استفاده می‌شوند، کمک می‌کند.

سوال ۲:

اگر بخواهیم PUM را به این شبیه‌ساز اضافه کنیم، کدام قسمت باید تغییر کند؟

پاسخ: تغییرات مورد نیاز برای اضاف کردن PUM به شبیه‌ساز NVSim به‌صورت زیر است:

قسمت‌های کلیدی برای تغییر

۱. **پیکربندی سخت‌افزار (`SimConfig.ini`):**

- می‌بایست معماری PUM را در فایل پیکربندی سخت‌افزار توصیف کنیم.
- بر اساس معماری مورد نیاز خودمان می‌توانیم بخش‌هایی را به فایل کانفیگ اضافه نموده. این بخش‌ها شامل اضاف نمودن پارامترهای سخت افزاری PUM مانند نوع حافظه، الگوهای دسترسی حافظه و ... باشد

برای اعمال این تغییرات، فایل `SimConfig.ini` را باز کرده و کانفیگ‌های PUM را به صورت زیر اضافه می‌کنیم.

```
[PUM]
memory_type = "RRAM"
access_pattern = "row-major"
pum_specific_param1 = value1
pum_specific_param2 = value2
```

۲. **توصیف شبکه (`network.py`):**

- فایل `network.py` را برای اضاف کردن PUM به لایه‌ها تغییر می‌دهیم.
- این شامل تعریف نحوه تعامل عناصر PUM با لایه‌های شبکه عصبی و هر پیکربندی خاص مورد نیاز برای عملیات PUM می‌باشد.

۳. **ماژول‌های شبیه‌سازی:**

- ماژول‌های شبیه‌سازی را برای شامل کردن رفتار و عملکرد PUM تغییر دهید.
- این شامل تغییرات در `Model/ MNSIM/Hardware` و `Model/ MNSIM/Mapping` برای انعکاس ویژگی‌های PUM از نظر توان، مساحت، تأخیر و مصرف انرژی می‌باشد.

۴. **ماژول‌های جدید برای :PUM**

- ماژول‌ها یا کلاس‌های پایتون جدیدی را برای نمایش سخت‌افزار PUM و رفتار آن پیاده‌سازی کنید.
- این ممکن است شامل ایجاد فایل‌های جدید یا تغییرات اساسی در فایل‌های موجود در `MNSIM/Hardware /Model` و `Model/ MNSIM/Mapping` باشد.

۵. **مدل‌سازی دقت و عملکرد:**

- اطمینان حاصل کنید که مدل‌سازی دقت و عملکرد ویژگی‌های خاص PUM را در نظر می‌گیرد.
- فایل `MNSIM/Interface/` را تغییر دهید تا جریان داده و الگوهای پردازش خاص PUM را مدیریت کند.

سوال ۳:

در اولین پیاده‌سازی، ابعاد ضربدری (Xbar) را به ۲۵۶x۲۵۶ تنظیم کنید. در پیاده‌سازی دوم، ابعاد ضربدری را به ۱۲۸x۱۲۸ تغییر دهید. مجموع تأخیر، توان و انرژی را گزارش دهید و جدول را پر کنید. چه اتفاقی افتاد؟ چرا؟ (برای هر پارامتر به تفصیل توضیح دهید.)

	۱۲۸*۱۲۸	۲۵۶*۲۵۶	وضعیت
تأخیر			کاهش / افزایش
توان			کاهش / افزایش
انرژی			کاهش / افزایش

جدول ۱: جدول I