基於22nm製程實現機器學習應用的 SRAM全數位全精度內存計算宏
All-Digital SRAM-Based Full-Precision In-Memory Computing Macro in 22nm for Machine-Learning Edge Applications

組別：晶片系統組　組員：陳耀新、鄭丞鈞、陳建鈞、吳宜謙、余承哲　指導教授：王進賢教授

摘要

本專題實作一種新型的非馮紐曼架構加速器—內存計算(In-Memory Computing)。內存計算技術整合了記憶體與運算單元，核心理念是在內存中執行部分計算任務，以降低因處理器和記憶體之間數據來回傳輸而產生的延遲和功率消耗。

本專題參考了2021年台積電在ISSCC上發表的設計[1]，該論文提出了可支援1至8位元輸入及可程式化調整權重的架構。本專題在聯電22nm製程下實現並完成Post-sim驗證。採用全數位的架構，以避免類比架構可能產生的精度損失。

研究方法與設計

本專題採用循環展開(Unrolling)來設計加速器。卷積運算共有四個For迴圈[2]，包含了輸出通道數、輸入長寬、輸入通道數以及卷積核長寬。本專題藉由同時對卷積核長寬、輸入通道數以及輸出通道數做循環展開，以平行運算加速卷積運算。

本專題使用NHWC資料格式來儲存輸入與輸出資料。

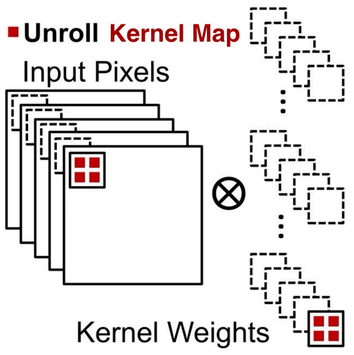

圖1 對卷積核長寬做循環展開

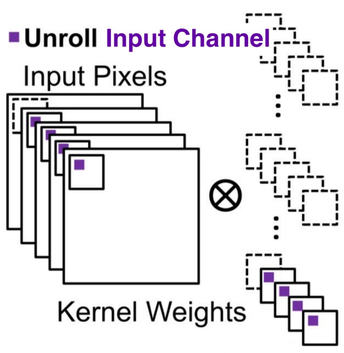

圖2 對輸入通道數做循環展開

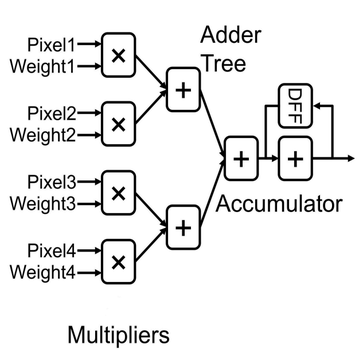

圖3 循環展開的電路架構

本專題的電路設計使用客製化的6T SRAM單元來存儲權重，並使用NOR閘實現單一位元乘法運算。將多組乘積通過加法樹相加，再通過累加器進行累加，以求得最終卷積結果。

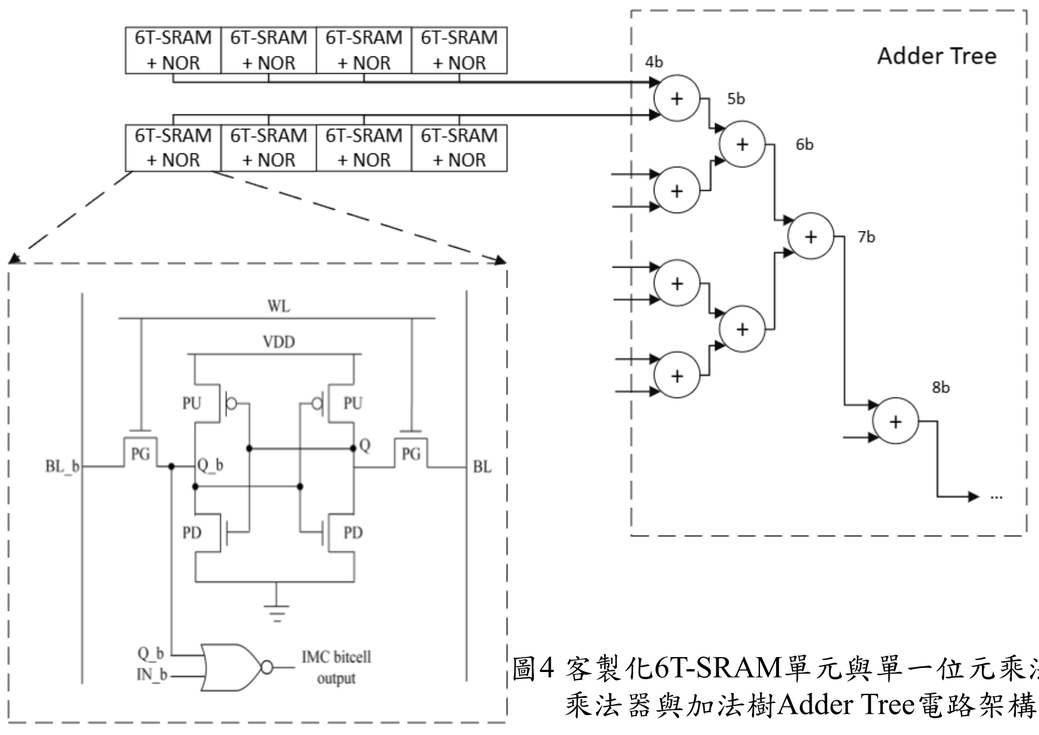

圖4 客製化6T-SRAM單元與單一位元乘法器(左圖)
乘法器與加法樹Adder Tree電路架構圖(右圖)

關於可程式化(Programable)的部分，本專題支持INT8的有號與無號數卷積運算。通過在加法樹中的加法器上新增一個用於符號擴展的全加器，以及在累加器上支援有號與無號的累加，來實現有號與無號的可程式化。

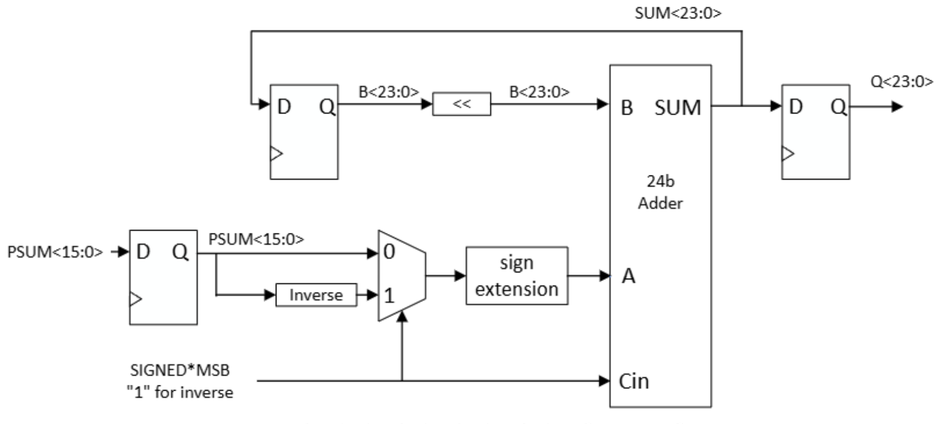

圖5 支援有號數與無號數累加的累加器

INT8 的輸入與 INT8 的權重在卷積運算後會產生 INT32 的結果。因此需要對其進行再量化(Re-Quantization)，將 INT32 的卷積結果再量化成 INT8 的資料型態。

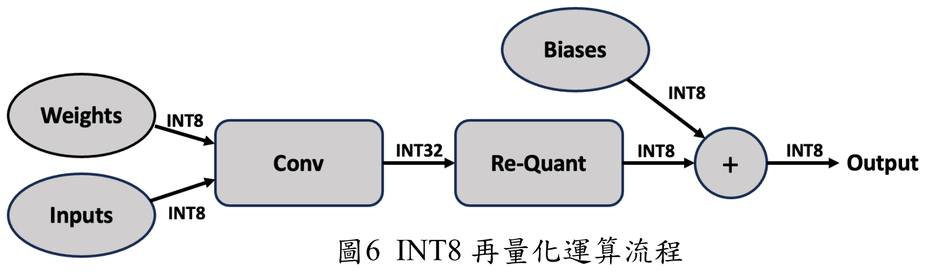

圖6 INT8 再量化運算流程

[3]提供了再量化的硬體實現方法。先將再量化的縮放因子(Scale factor)以16位精度表示，接著透過將INT32的卷積輸出(Q)與縮放因子(M)相乘，得到乘積再向右移位16位元後進行截斷，加上零點即可得到再量化後的INT8資料型態輸出。

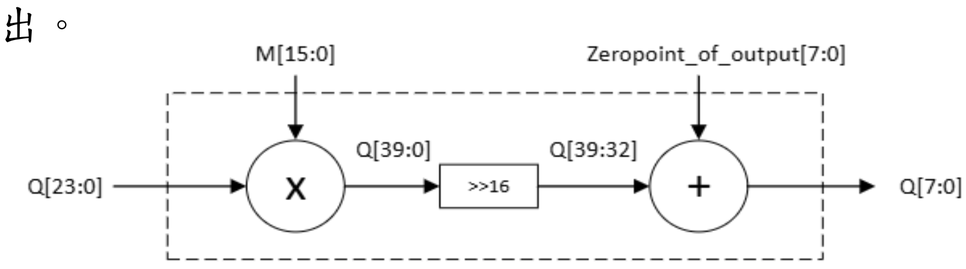

圖7 INT32到INT8的再量化硬體實現

實作結果

本專題已完成Post-Layout驗證與初步混訊模擬，並測量完成功耗效能。因著測試所用的卷積層數據有著較低的變化率(Toggle rate)，致使能效比(TOPS/W)略高於論文所示。未來將會引入更複雜的數據測試，以獲得更全面的評估。

另一方面，因爲在電路設計合成上經驗不足，結果的效能面積比(TOPS/mm2)顯著低於論文。接下來將針對此部分進行優化。

規格與功能	ISSCC'21[1]	This work
Technology	22nm	22nm
Array size	64kb	64kb
Bitcell type/area	1RW 6T/0.397um^2	1RW 6T/0.735um^2
Macro area	0.202mm^2	1.020mm^2
Power Supply(V)	0.72	0.8
Input bit	1~8	1~8
Weight bit	4/8/12/16	8
Output bit	8	8
F_{max}	100MHz@0.8/0.72V	90.9MHz@0.8V
TOPS/W	89 (18% input toggle rate, 50% 1s for weights)	91.03 (Input toggle rate <18%, 1s for weights < 50%)
TOPS/mm^2	16.3	0.365

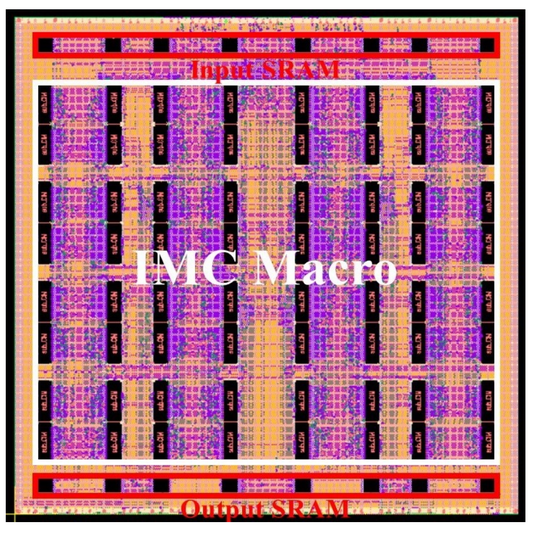

圖8 本專題的整體電路佈局

參考文獻

[1]. Y. -D. Chih et al., "16.4 An 89TOPS/W and 16.3TOPS/mm2 All-Digital SRAM-Based Full-Precision Compute-In Memory Macro in 22nm for Machine-Learning Edge Applications," 2021 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, 2021, pp. 252-254.

[2]. Y. Ma et al., "Optimizing the Convolution Operation to Accelerate Deep Neural Networks on FPGA," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 26, no. 7, pp. 1354-1367, July 2018.

[3]. Jacob, B et al., "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only inference" 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2704-2713.