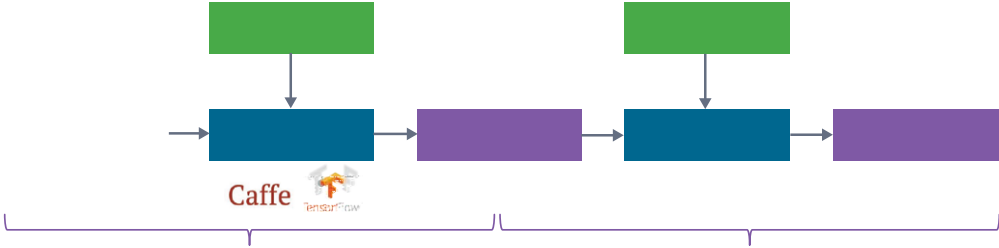


Pieter van der Wolf

Sk'abeke I' U. :

Dmitry Zakharov

Sk'abeke I' U. •



1

CSXXW TW ead' ai

(CNN)

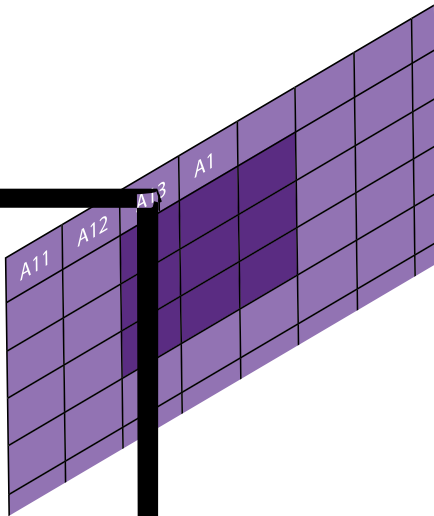
3

(M23)

(2, 3)

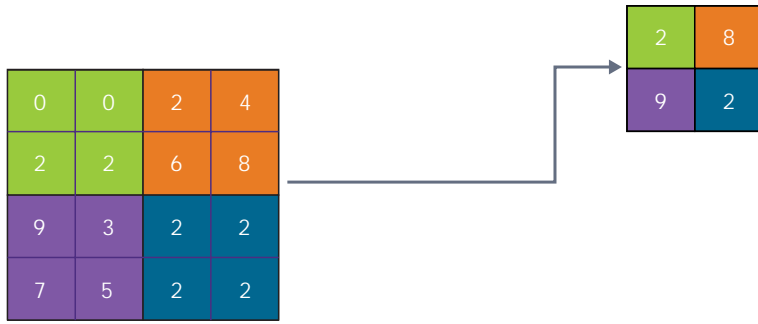
A24-A26 A34-A36 A44-A46

M24



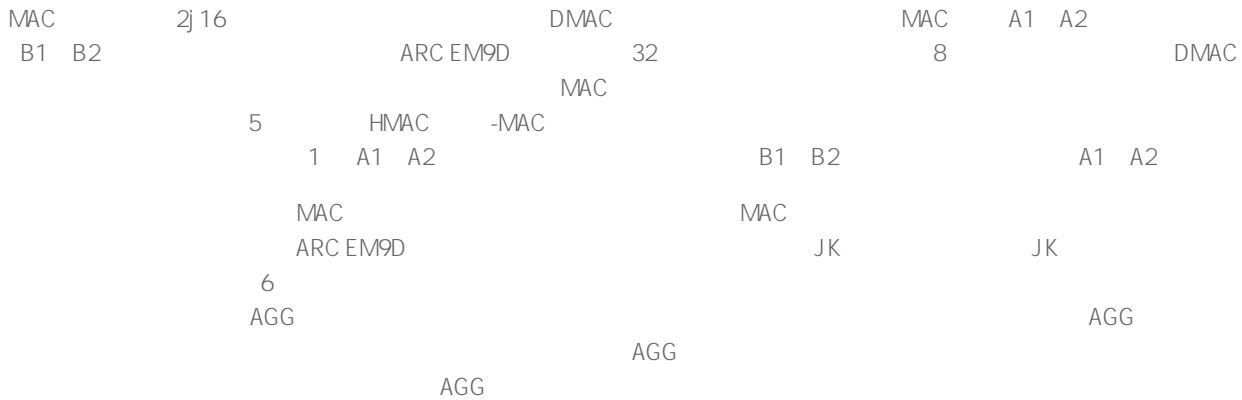
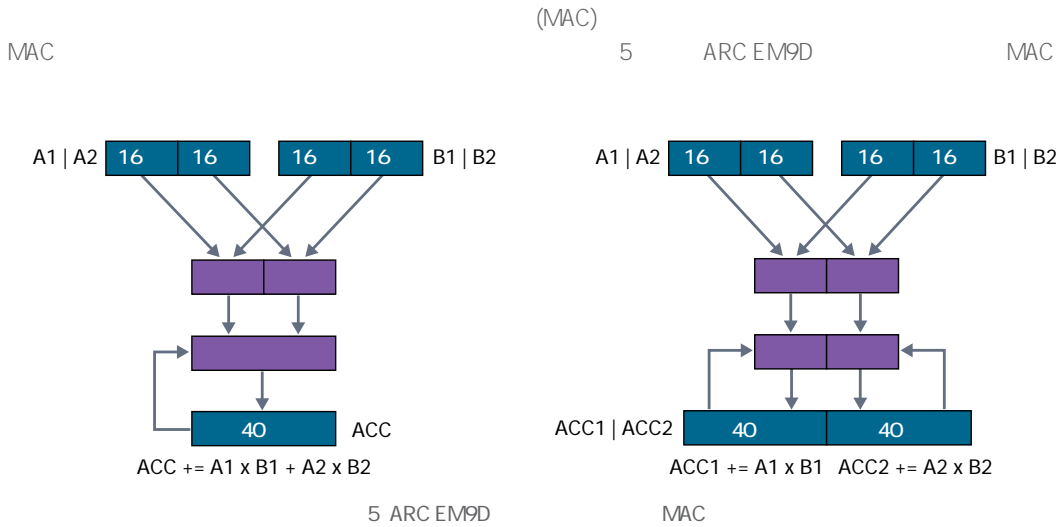
$$\begin{aligned} M23 = & W11 \times A23 + W12 \times A24 + W13 \times A25 \\ & + W21 \times A33 + W22 \times A34 + W23 \times A35 \\ & + W31 \times A43 + W32 \times A44 + W33 \times A45 \end{aligned}$$

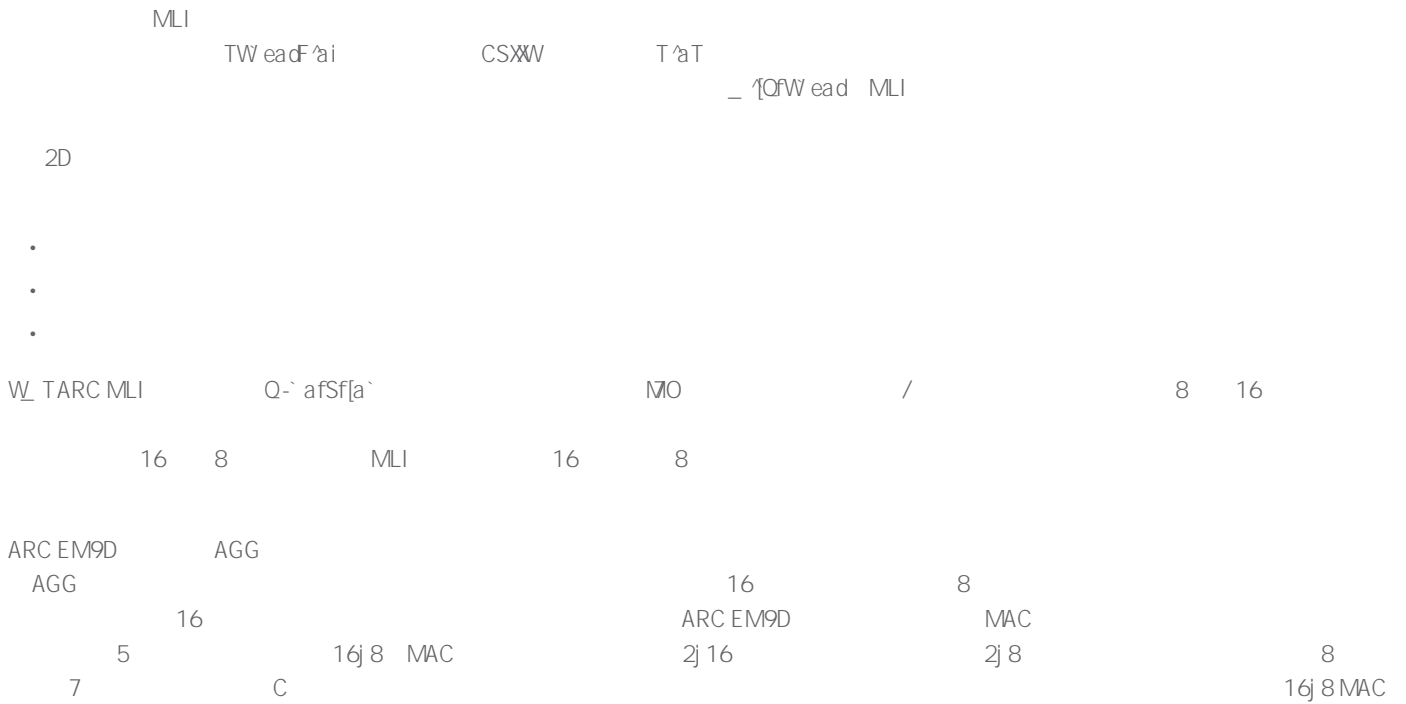




/

32 DSP ARC EM9D





```

; AGU 0 is used for input data. AGU 1 is used for weights
; setup AGU 0 for loading next 2x16-bit vector
; setup AGU 1 for loading next 2x8-bit into lower bytes of 32-bit word

A B 0, %agu_u0, %agu_u1 ; DMAC + 2 loads + 2 pointer updates
:

```

7 MLI C 16 8



```

; AGU 0 and AGU 1 are used for input data (one data pointer with two modifiers)
; AGU 2 is used for weights
; setup AGU 0 for loading next two 16-bit input values
; setup AGU 1 for loading two 16-bit input values at next row of input data
; setup AGU 2 for loading next 8-bit value with sign extension & replication
...
A 2 0, %agu_u0, %agu_u2 ; VMAC + 2 loads + sign ext + repl + 2 pointer updates
A 2 0, %agu_u0, %agu_u2 ; VMAC + 2 loads + sign ext + repl + 2 pointer updates
A 2 0, %agu_u1, %agu_u2 ; VMAC + 2 loads + sign ext + repl + 2 pointer updates
:

```

8 MLI C 16 8 2D

AGG

2D

- CS~~W~~

([2Ua)
AGG

ARC EM9D

W_ TARC MLI

W_ TARC MLI

3j 3

1

TW ea dF 'a i

SAME

`ml i_krn_depthwise_conv2d_chw_fx8w16d_k3x3_str1_krnpad(...)`

W_ TARC MLI

C++

(i dSbbWd)

W_ TARC MLI

ARC

EM9D

MWSI SdW

C/C++

DSP

ARC EM9D

W_ TARC MLI

ARC

W_ TARC MLI


```

mli_krn_permute_fx8(&input, &permute_hwc2chw_cfg, &ir_Y);

ir_X.el_params.fx.frac_bits = CONV1_OUT_FRAQ;
mli_krn_conv2d_chw_fx8_k5x5_str1_krnpad(&ir_Y, &L1_conv_wt, &L1_conv_b, &conv_cfg, &ir_X);
mli_krn_maxpool_chw_fx8_k3x3(&ir_X, &pool_cfg, &ir_Y);

ir_X.el_params.fx.frac_bits = CONV2_OUT_FRAQ;
mli_krn_conv2d_chw_fx8_k5x5_str1_krnpad(&ir_Y, &L2_conv_wt, &L2_conv_b, &conv_cfg, &ir_X);
mli_krn_avepool_chw_fx8_k3x3_krnpad(&ir_X, &pool_cfg, &ir_Y);

ir_X.el_params.fx.frac_bits = CONV3_OUT_FRAQ;
mli_krn_conv2d_chw_fx8_k5x5_str1_krnpad(&ir_Y, &L3_conv_wt, &L3_conv_b, &conv_cfg, &ir_X);
mli_krn_avepool_chw_fx8_k3x3_krnpad(&ir_X, &pool_cfg, &ir_Y);

ir_X.el_params.fx.frac_bits = FC4_OUT_FRAQ;
mli_krn_fully_connected_fx8(&ir_Y, &L4_fc_wt, &L4_fc_b, &ir_X);

ir_Y.el_params.fx.frac_bits = FC5_OUT_FRAQ;
mli_krn_fully_connected_fx8(&ir_X, &L5_fc_wt, &L5_fc_b, &ir_Y);
mli_krn_softmax_fx8(&ir_Y, &output);

```

10 CIFAR-10

MLI

10

(bVd_gfV)
[dJ [dK

RGB

W_TARC MLI
CHI

W_TARC MLI

CIFAR-10CNN

MIOO

EM9D / ARC
 JK DSP ARC EM9D MAC
 W_ TARC MLI ARC EM9D / CIFAR-10
 Sk` abeke

References

- [1] Samuel, Arthur (1959). "Some Studies in Machine Learning Using the Game of Checkers". IBM Journal of Research and Development. 3 (3): 210–229.
- [2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. "Going deeper with convolutions". In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–9, 2015.
- [3] www.synopsys.com/ev
- [4] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. "Quantization and training of neural networks for efficient integer-arithmetic-only inference". arXiv preprint arXiv:1712.05877, 2017.
- [5] <https://embarc.org/>
- [6] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al., "Deep speech 2: End-to-end speech recognition in english and mandarin", in International Conference on Machine Learning, 2016, pp. 173–182.
- [7] [https://en.wikipedia.org/wiki/Q_\(number_format\)](https://en.wikipedia.org/wiki/Q_(number_format))
- [8] Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images." 2009.
- [9]
