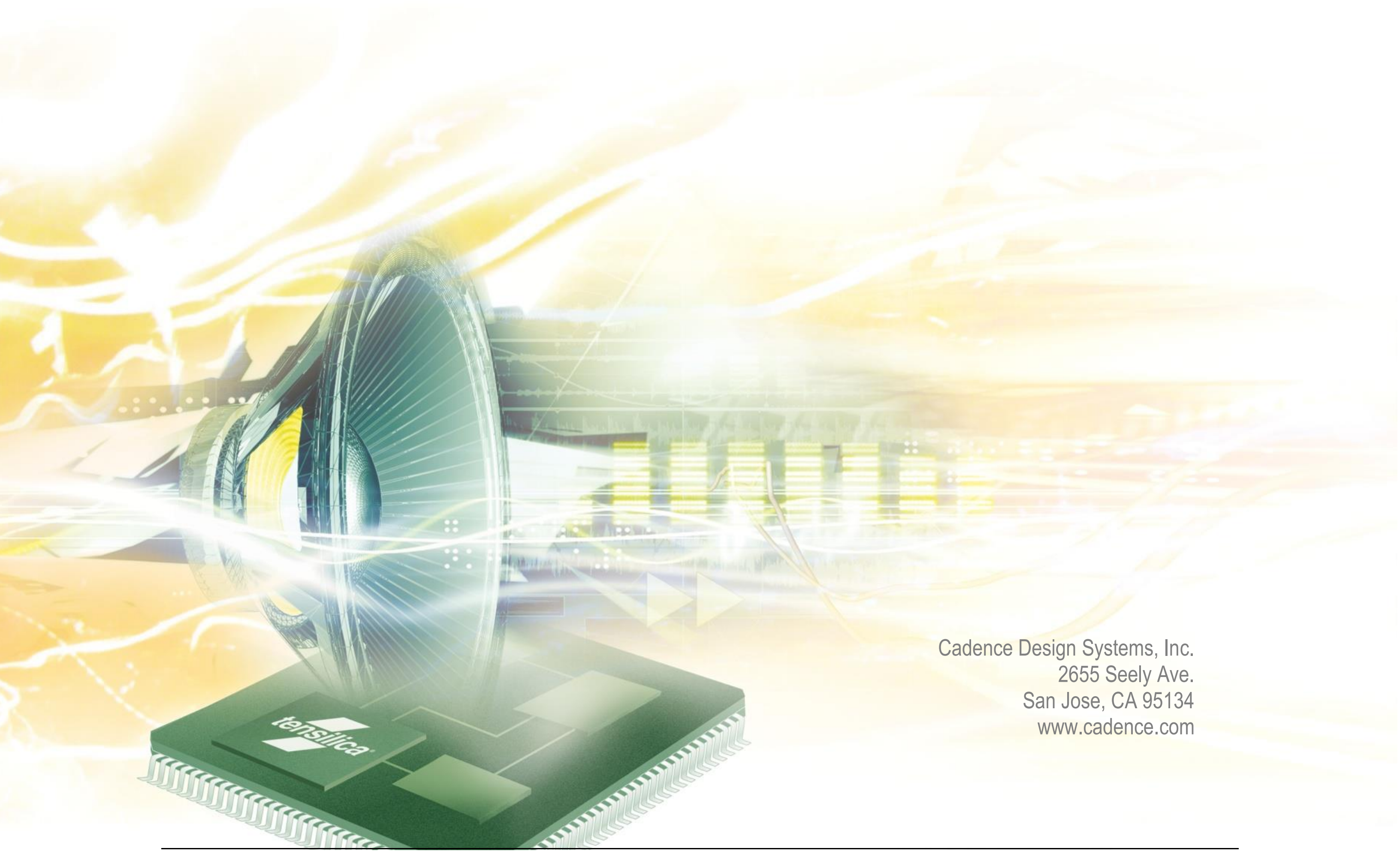
Fusion G3 Neural Network Library

Test report



Cadence Design Systems, Inc.
2655 Seely Ave.
San Jose, CA 95134
www.cadence.com

© 2024 Cadence Design Systems, Inc. All rights reserved.
Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. (Cadence) contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 1-800-862-4522.

All other trademarks are the property of their respective holders.

Patents: Licensed under U.S. Patent Nos. 7,526,739; 8,032,857; 8,209,649; 8,266,560; 8,650,516

Restricted Print Permission: This publication is protected by copyright and any unauthorized use of this publication may violate copyright, trademark, and other laws. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. This statement grants you permission to print one (1) hard copy of this publication subject to the following conditions:

- The publication may be used solely for personal, informational, and noncommercial purposes;
- The publication may not be modified in any way;
- Any copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement,
- The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration; and
- Cadence reserves the right to revoke this authorization at any time, and any such use shall be discontinued immediately upon written notice from Cadence.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. The information contained herein is the proprietary and confidential information of Cadence or its licensors, and is supplied subject to, and may be used only by Cadence's customer in accordance with, a written agreement between Cadence and its customer. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor.

For further assistance, contact Cadence Online Support at https://support.cadence.com/.
Copyright © 2024, Cadence Design Systems, Inc. All rights reserved.

Version 1.0
December 2024

Contents

Document Change History

Version	Changes
1.0	Initial version

1.Introduction

The Fusion G3 Neural Network (NN) Library is an optimized implementation of various low-level NN kernels. The low-level NN kernels are the basic building blocks for operators and networks in neural network frameworks with a generic and simple interface.

The Fusion G3 NN Library package includes the source code containing low-level kernel implementations. The current version of the library implements activation, basic operation, normalization and reorg functions as low-level kernels.

This document details the tests performed on Fusion G3 NN library kernels.

Note This version of the library supports Fusion G3 DSPs with the SP-VFPU (Single Precision Vector Floating Point Unit).

Note This version of the Fusion G3 NN Library is tested with the xt-clang/xt-clang++ compilers using Xtensa Software Tools from RI-2022.10 release.

2. Fusion G3 NN Library tests

The details of the tests performed on the FusionG3 NN library is provided in the next sections.

The following table provides details of the library version, core information and build target used for verification of the kernels.

Table 2-1 Details of Setup Used for tests

Library Name	FusionG3 Neural Network Library
Library Version	1.0
Library API Version	1.0
Core Name	FusionG3
Tool Chain	RI-2022.10
Build Target	Release

2.1 Accuracy tests

Executorch on x86 platform is used as reference to verify accuracy of Fusion G3 NN library. The kernels are divided into 2 categories – signle precision float kernels and integer kernels. For validating kernels with single precision float, ULP (Unit of Least Precision) error is used as metric. For validating kernels with integer data types, bit error (how many LSBs have error) is used as metric.

The following sections shows the results in the form of tables. The first row specifies the different input ranges used for verification of the kernel. The next rows specify the maximum bit error observed in the case of kernels with integer datatypes or maximum ULP error in the case of kernels with single precision floating point.

2.1.1 Kernels with integer datatypes

- **Addition** - 32-bit

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[0, 65535]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max error (bits)	0	0	0	0	0	0	0

- **Multiplication** - 32-bit

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max error (bits)	0	0	0	0	0	0	0

- **Quantization** - 4-bit, 8-bit, 16-bit

Input range	[-2^31, 2^31)	[-2^15, 2^15)	[-2^7, 2^7)	[-1, 1]	[-2^15, 1]	[1, 2^15)	[-100, 100]
Max error (bits)	0	0	0	0	0	0	0

- **Division** - 32-bit

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max error (bits)	0	0	0	0	0	0	0

- **Subtraction** - 32-bit

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max error (bits)	0	0	0	0	0	0	0

- **Concatenation** - 8-bit, 16-bit, 32-bit

Input range	Signed 8-bit	Unsigned 8-bit	Signed 16-bit	Unsigned 16-bit	Signed 32-bit	Unsigned 32-bit
	[-2^7, 2^7-1]	[0, 2^8)	[-2^15, 2^15)	[0, 2^16)	[-2^31, 2^31)	[0, 2^32)

Max error (bits)	0	0	0	0	0	0

- **Transpose** - 8-bit, 16-bit, 32-bit

Input range	Signed 8-bit	Unsigned 8-bit	Signed 16-bit	Unsigned 16-bit	Signed 32-bit	Unsigned 32-bit
	[-2^7, 2^7-1]	[0, 2^8)	[-2^15, 2^15)	[0, 2^16)	[-2^31, 2^31)	[0, 2^32)
Max error (bits)	0	0	0	0	0	0

- **Permute** - 8-bit, 16-bit, 32-bit

Input range	Signed 8-bit	Unsigned 8-bit	Signed 16-bit	Unsigned 16-bit	Signed 32-bit	Unsigned 32-bit
	[-2^7, 2^7-1]	[0, 2^8)	[-2^15, 2^15)	[0, 2^16)	[-2^31, 2^31)	[0, 2^32)
Max error (bits)	0	0	0	0	0	0

- **Slice_copy** - 8-bit, 16-bit, 32-bit

Input range	Signed 8-bit	Unsigned 8-bit	Signed 16-bit	Unsigned 16-bit	Signed 32-bit	Unsigned 32-bit
	[-2^7, 2^7-1]	[0, 2^8)	[-2^15, 2^15)	[0, 2^16)	[-2^31, 2^31)	[0, 2^32)
Max error (bits)	0	0	0	0	0	0

- **Clamp** - 8-bit, 16-bit

Input range	Signed 8-bit	Unsigned 8-bit	Signed 16-bit

	[-2^7, 2^7-1]	[0, 2^8)	[-2^15, 2^15)
Max error (bits)	0	0	0

2.1.2 Kernels with single precision float datatypes

- **Addition**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[0, 65535]	[-2^15, 2^15)	[0, 255]	[-1, 1]	[-128, 127]
Max ULP error	26	1	1	1	1	1	324194	3

Below are the reference and Fusion G3 kernel output values where the maximum ULP is seen in each of the test case mentioned above.

Reference output	**Fusion G3 kernel output**	**ULP error**
0.0000004172325134277343750	0.0000004080183657606539782140	324194
29058432.0	29058484.0	26
271818624.0	271818688.0	26

Reason - In ADD kernel, MULA (multiply and accumulate) is used to perform multiply and accumulate operation. Upon replacing the MULA instruction with separate MUL and ADD instructions, observed ULP error as mentioned in the below table. However, according to the FusionG3 instruction manual, the MULA instruction is designed to generate more precise output compared to using the individual MUL and ADD operations. This suggests that Executorch on x86 is generating lower precision results in these cases. These results are manually verified by using data that gave maximum ULP error, confirming this behavior. As a result, opting to retain the MULA instruction in the code.

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[0, 65535]	[-2^15, 2^15)	[0, 255]	[-1, 1]	[-128, 127]

Max ULP error	0	0	0	0	0	0	0	0

- **Multiplication**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[0, 65535]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	0	0	0	0	0	0	0

- **Subtraction**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[0, 65535]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	6	22	66	1	1	50	91

Below are the reference and Fusion G3 kernel output values where the maximum ULP is seen in each of the test case mentioned above.

Reference output	**Fusion G3 kernel output**	**ULP error**
-104738432.0	-104738480.0	6
12570752.0	12570730.0	22
-1784256.0	-1784247.75	66
1.4210662841796875	1.4210722446441650391	50
0.455230712890625	0.4552280008792877197	91

In SUB kernel, MULS (multiply and accumulate) is used to perform multiply and accumulate operation. Upon replacing the MULS instruction with separate MUL and SUB instructions, observed ULP error as mentioned in the below table. However, according to the FusionG3 instruction manual, the MULS instruction is designed to generate more precise output compared to using the individual MUL and SUB operations. This suggests that Executorch on x86 is generating lower precision results in these cases. These results are manually verified by using data that gave maximum ULP error, confirming this behavior. As a result, opting to retain the MULS instruction in the code.

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[0, 65535]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	0	0	0	0	0	0	0

Note: The maximum ULP error seen with MULS instruction might increase when more unit test cases are added to test all the kernels in the operator.

- **Division**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	0	0	0	0	0	0	0

- **Tanh**

Input range	[-5, 5]	[-1, 1]	[0, 5]	[-5, 0]	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-2^15, 2^15)
Max ULP error	1	1	1	1	0	0	0	0

- **Sigmoid**

Input range	[-5, 5]	[-1, 1]	[0, 5]	[-5, 0]	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-2^15, 2^15)
Max ULP error	1	1	1	1	0	0	0	0

- **Dequantize** (4-bit, 8-bit,16-bit)

Input range	[-2^31, 2^31)	[-2^15, 2^15)	[-2^7, 2^7)	[-1, 1]	[-2^15, 1]	[1, 2^15]	[-100, 100]
Max ULP error	0	0	0	0	0	0	0

- **Sqrt** – Square root

Input range	[0, 2^31)	[0, 2^15)	[0, 1]	[0, 255]	[2500, 500000]	[0, 272024]
Max ULP error	1	1	2	2	0	1

- **Rsqrt** – Inverse square root

Input range	[0, 2^31)	[0, 2^15)	[0, 1]	[0, 255]	[2500, 500000]	[0, 272024]
Max ULP error	1	0	1	1	1	1

- **Exponent**

Input range	[-2^7, 2^7)	[0, 103]	[-103, 0]	[-1, 1]	[-2^6, 2^6]	[-15, 15]	[-2^5, 2^5]
Max ULP error	1	1	1	1	1	1	1

- **Softmax**

Input range	[-5, 5]	[-1, 1]	[0, 5]	[-5, 0]	[-2^31, 2^31)	[0, 2^31)	[2^31, 0]	[-2^15, 2^15)
Max ULP error	3	4	4	2	0	0	0	0

- **Native layernorm**

Input range / Max ULP error	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-1024, 0]
W: [-2^15, 2^15) B:[-1, 1]	2073	3960	847	67	1	754	13481
W: [-2^8, 2^8] B: [-2^6, 2^6]	1104	2347	31689	185	123	98	5373
W: (-1, 1) B: (-2^10, 2^10)	0	1	1	1	1610	1	1

Below table lists the reference and Fusion G3 kernel output values for each of the test case where the maximum ULP error seen is more than 3.

For case 1:

Reference output	**Fusion G3 kernel output**	**ULP error**
-0.058274686336517333984380	-0.058282408863306045532230	2073
6.29890298843383789060	6.29701471328735351560	3960
34.97669982910156250	34.9734687805175781250	847
99.78234863281250	99.781837463378906250	67
-13052.355468750	-13052.34863281250	7
-23874.964843750	-23874.9511718750	7
12.79699802398681640630	12.79771709442138671880	754
13.83341884613037109380	13.82056236267089843750	13481

For case 2:

Reference output	Fusion G3 kernel output	ULP error
-0.085777282714843750	-0.08578550815582275390	1104
0.03700447082519531250	0.03701321408152580261230	2347
0.006672859191894531250	0.006687615532428026199340	31689
-0.934104919433593750	-0.93409389257431030270	185
0.653556823730468750	0.6535641551017761230	123
3.540779113769531250	3.54075574874877929690	98
0.456047058105468750	0.45588693022727966310	5373

For case 3:

Reference output	Fusion G3 kernel output	ULP error
-0.785789489746093750	- 0.78569352626800537110	1610

Below table lists the ULP error when high precision Layer norm kernel is enabled. The High precision layer norm kernel can be enabled by enabling the macro ENABLE_HIGH_PRECISION in the library make file. Please note that there will be a compromise on the performance in terms of cycles when high precision layer norm is enabled. By default, layer norm is enabled for high performance.

Input range/Max ULP error	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-1024, 0]
W: [-2^15, 2^15) B:[-1, 1]	0	0	0	0	0	0	0

W: [-2^8, 2^8] B: [-2^6, 2^6]	0	0	0	0	0	0	0
W: (-1,1) B: (-2^10, 2^10)	0	0	0	0	0	0	0

- **Mean**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	3	6	1	2	0	0	5

Below table lists the reference and Fusion G3 kernel output values for each of the test case where the maximum ULP error seen.

Reference output	Fusion G3 kernel output	ULP error
-100699120.00	-100699144.00	3
1059314048.00	1059313664.00	6
-505.50518798828125	-505.505340576171875	5

When the additions are serialized, below is the Maximum ULP error seen.

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	0	0	0	0	0	0	0

- **Where**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max ULP error	0	0	0	0	0	0	0

- **Less than**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max error	0	0	0	0	0	0	0

- **Clamp**

Input range	[-2^31, 2^31)	[0, 2^31)	[-2^31, 0]	[-1, 1]	[-2^15, 2^15)	[0, 255]	[-128, 127]
Max error	0	0	0	0	0	0	0

2.2 Unit tests

The below table lists input parameters with which kernels are tested and ULP error or bit error is generated which is mentioned in the previous section. “Kernel name” column specifies the kernel name, “Parameters” column specifies the input shapes used for testing.

Kernel name	Parameters
xa_nn_elm_add_32x32_32	Inp1: 3, 2, 2, 1 Inp2: 3, 2, 2, 1
	Inp1: 7 3 1 5 Inp2: 7 3 1 5
	Inp1: 10 10 5 4 4 Inp2: 10 10 5 4 4
	Inp1: 8 5 10 2 10 Inp2: 8 5 10 2 10
	Inp1: 7 3 5 Inp2: 7 3 5
	Inp1: 5 3 10 Inp2: 5 3 10
xa_nn_elm_add_scalar_32x32_32	Inp1: 4, 4, 1 Inp2: 1
	Inp1: 3, 10, 7 Inp2: 1
	Inp1: 2, 6, 6, 7 Inp2: 1
	Inp1: 8, 8, 7, 9 Inp2: 1
	Inp1: 10, 7, 5, 10, 2 Inp2: 1
	Inp1: 4, 4, 10 Inp2: 1
	Inp1: 1, 2, 8, 9

Kernel name	Parameters
	Inp2: 1
	Inp1: 7, 7, 7, 7 Inp2: 1
	Inp1: 3, 2, 8, 5 Inp2: 1
xa_nn_elm_add_broadcast_5D_32x32_32	Inp1: 3, 1, 3, 5 Inp2: 3, 5, 3, 5
	Inp1: 15, 15, 1, 10, 13 Inp2: 15, 1, 16, 10, 13
	Inp1: 1, 1, 11 Inp2: 8, 8, 11
	Inp1: 10, 48, 12, 54, 17 Inp2: 10, 1, 12, 54, 17
	Inp1: 65, 1 inp2: 65, 43
	Inp1: 5, 14, 71, 11, 1 Inp2: 5, 14, 71, 1, 23
	Inp1: 1, 160, 7, 7 Inp2: 34, 160, 7, 7
xa_nn_elm_mul_32x32_32	Inp1: 3,8,8,2 Inp2: 3,8,8,2
	Inp1: 2,8,8,4 Inp2: 2,8,8,4
	Inp1: 9,10,4 Inp2: 9,10,4
	Inp1: 7,8,9,6 Inp2: 7,8,9,6
	Inp1: 10,8,2,6 Inp2: 10,8,2,6
	Inp1: 9,7,3,8 Inp2: 9,7,3,8
	Inp1: 4,10,2,4 Inp2: 4,10,2,4
	Inp1: 7,7,9 Inp2: 7,7,9
	Inp1: 6,6,4 Inp2: 6,6,4
	Inp1: 7,7,8 Inp2: 7,7,8
	Inp1: 9,2,4,5 Inp2: 9,2,4,5
	Inp1: 17,4,8 Inp2: 17,4,8
xa_nn_elm_mul_scalar_32x32_32	Inp1: 3, 6 Inp2: 1
	Inp1: 1, 2, 7, 4 Inp2: 1
	Inp1: 6, 5, 9, 4

Kernel name	Parameters
	Inp2: 1
	Inp1: 10, 2, 9, 8 Inp2: 1
	Inp1: 10, 10, 7, 1, 10 Inp2: 1
	Inp1: 8, 1, 6, 2, 7 Inp2: 1
	Inp1: 4, 8, 9 Inp2: 1
	Inp1: 10, 3, 3 Inp2: 1
	Inp1: 4, 7, 4 Inp2: 1
	Inp1: 5, 7, 1, 9 Inp2: 1
	Inp1: 7, 6, 9 Inp2: 1
xa_nn_elm_mul_broadcast_5D_32x32_32	Inp1: 13, 15, 1, 19 Inp2: 13, 15, 65, 19
	Inp1: 15, 10, 5, 30, 18 Inp2: 15, 1, 5, 30, 18
	Inp1: 18, 8 , 14 Inp2: 18, 8, 1
	Inp1: 12, 35, 1, 17, 17 Inp2: 12, 35, 52, 17, 17
	Inp1: 8, 9, 15, 1 Inp2: 8, 9, 15, 5
	Inp1: 1, 1, 1, 1, 1 Inp2: 1, 21, 224, 224, 17
xa_nn_elm_add_f32xf32_f32	Inp1: 1, 2, 4, 6 Inp2: 1, 2, 4, 6
	Inp1: 6, 8, 8 Inp2: 6, 8, 8
	Inp1: 5, 4, 4, 7 Inp2: 5, 4, 4, 7
	Inp1: 3, 9, 6, 1, 10 Inp2: 3, 9, 6, 1, 10
	Inp1: 7, 3, 10 Inp2: 7, 3, 10
	Inp1: 9, 2, 6, 10, 3 Inp2: 9, 2, 6, 10, 3
	Inp1: 10, 2, 6 Inp2: 10, 2, 6
	Inp1: 7, 8, 4 Inp2: 7, 8, 4
	Inp1: 4, 1, 6, 2, 9 Inp2: 4, 1, 6, 2, 9

Kernel name	Parameters
	Inp1: 7, 2, 2, 7, 10 Inp2: 7, 2, 2, 7, 10
	Inp1: 2, 3, 4 Inp2: 2, 3, 4
	Inp1: 2, 7, 2, 2 Inp2: 2, 7, 2, 2
xa_nn_elm_add_scalar_f32xf32_f32	Inp1: 4 1 4 9 Inp2: 1
	Inp1: 6 5 10 6 Inp2: 1
	Inp1: 1 10 10 6 2 Inp2: 1
	Inp1: 10 6 1 9 Inp2: 1
	Inp1: 1 4 1 10 Inp2: 1
	Inp1: 8 6 7 2 9 Inp2: 1
	Inp1: 6 6 6 Inp2: 1
xa_nn_elm_add_broadcast_5D_f32xf32_f32	Inp1: 32, 45 Inp2: 32, 1
	Inp1: 87, 9, 15, 1 Inp2: 87, 9, 15, 5
	Inp1: 92 13 3 19 28 Inp2: 92 13 3 19 28
	Inp1: 18, 8, 14 Inp2: 18, 8, 1
	Inp1: 15, 10, 5, 30, 18 Inp2: 15, 1, 5, 30, 18
	Inp1: 13, 15, 1, 19 Inp2: 13, 15, 65, 19
xa_nn_elm_mul_f32xf32_f32	Inp1: 3, 5, 7, 9 Inp2: 3, 5, 7, 9
	Inp1: 9, 9, 1, 10 Inp2: 9, 9, 1, 10
	Inp1: 8, 1, 10, 6 Inp2: 8, 1, 10, 6
	Inp1: 5, 6, 6, 6 Inp2: 5, 6, 6, 6
	Inp1: 2, 7, 5, 8 Inp2: 2, 7, 5, 8
	Inp1: 8, 3, 2, 6 Inp2: 8, 3, 2, 6
	Inp1: 5, 9, 3 Inp2: 5, 9, 3
	Inp1: 10, 7, 7

Kernel name	Parameters
	Inp2: 10, 7, 7
	Inp1: 8, 12, 7 Inp2: 8, 12, 7
	Inp1: 6, 9, 8 Inp2: 6, 9, 8
	Inp1: 10, 9, 14, 1 Inp2: 10, 9, 14, 1
	Inp1: 7, 6, 2 Inp2: 7, 6, 2
xa_nn_elm_mul_scalar_f32xf32_f32	Inp1: 6, 9, 2, 2 Inp2: 1
	Inp1: 2, 1, 8 Inp2: 1
	Inp1: 2, 9, 3, 8 Inp2: 1
	Inp1: 4, 5, 7, 2 Inp2: 1
	Inp1: 5, 7, 5, 2 Inp2: 1
	Inp1: 10, 6, 4, 8, 7 Inp2: 1
	Inp1: 7, 9, 1 Inp2: 1
	Inp1: 7, 4, 4, 10, 6 Inp2: 1
xa_nn_elm_mul_broadcast_5D_f32xf32_f32	Inp1: 1, 160, 7, 7 Inp2: 34, 160, 7, 7
	Inp1: 5, 4, 71, 11, 1 Inp2: 5, 4, 71, 11, 23
	Inp1: 1, 64, 56, 45, 34 Inp2: 1, 1, 1, 1, 1
	Inp1: 65, 1 Inp2: 65, 43
	Inp1: 8, 8, 11 Inp2: 1, 1, 11
	Inp1: 3, 1, 35 Inp2: 3, 5, 35
xa_nn_vec_softmax_dim_f32_f32	Axis = 1 Inp: 6, 6, 8, 4
	Axis = 2 Inp: 6, 10, 5, 5, 9
	Axis = 3 Inp: 10, 5, 4, 6, 5
	Axis = 2 Inp: 7, 7, 6, 4, 5
	Axis = 1 Inp: 9, 8, 9, 7

Kernel name	Parameters
	Axis = 4 Inp: 4, 10, 10, 5, 5
	Axis = 3 Inp: 9, 8, 4, 9, 8
	Axis = 0 Inp: 5, 9, 8, 4
	Axis = 1 Inp: 6, 10, 10, 7
	Axis = 3 Inp: 5, 7, 5, 8
xa_nn_layernorm_f32_f32	Axis: 2 Inp: 10, 3, 10, 7
	Axis: 0 Inp: 4, 6, 1, 8
	Axis:3 Inp: 6, 9, 6, 6
	Axis: 1 Inp: 8, 7, 8, 1
	Axis: 1 Inp: 2, 1, 2
	Axis: 4 Inp: 3, 4, 8, 3, 4
	Axis: 0 Inp: 4, 6, 8, 6, 2
	Axis: 2 Inp: 8, 8, 4, 10
	Axis: 3 Inp: 4, 6, 2, 5
	Axis: 1 Inp: 6, 7, 4, 6
	Axis: 2 Inp: 8, 9, 7
	Axis: 0 Inp: 5, 5, 6
	Axis: 1 Inp: 4, 4, 4
	Axis: 0 Inp: 5, 1, 8, 7, 5
	Axis: 0 Inp: 9, 3, 4
	Axis: 1 Inp: 9, 7, 3
	Axis: 2 Inp: 9, 2, 8, 2, 9
	Axis: 1 Inp: 2, 6, 4, 9

Kernel name	Parameters
	Axis: 1 Inp: 6, 2, 2, 10
	Axis: 0 Inp: 3, 2, 6, 9
	Axis: 4 Inp: 3, 4, 8, 3, 4
xa_nn_cat_8_8	Inp1: 5, 4, 6 Inp2: 5, 8, 6 Inp3: 5, 2, 6 Inp4: 5, 7, 6 Axis: 1
	Inp1:4, 5, 6, 7 Inp2: 4, 5, 6, 3 Inp3: 4, 5, 6, 1 Inp4: 4, 5, 6, 2 Axis: 3
	Inp1: 8, 8, 8 Inp2: 8, 8, 7 Axis: 2
	Inp1: 2, 3, 4, 8 Inp2: 2, 3, 4, 6 Axis: 3
	Inp1: 2, 4, 3, 1, 8 Inp2: 2, 4, 6, 1, 8 Inp3: 2, 4, 7, 1, 8 Axis: 2
	Inp1: 3, 7, 5, 2 Inp2: 4, 7, 5, 2 Inp3: 5, 7, 5, 2 Inp4: 6, 7, 5, 2 Axis: 0
	Inp1: 6, 5, 6 Inp2: 7, 5, 6 Inp3: 5, 5, 6 Axis: 0
	Inp1: 3, 6, 9, Inp2: 3, 6, 12 Axis: 2
	Inp1: 5, 4, 3, 6 Inp2: 5, 4, 3, 9 Axis:3
	Inp1: 2, 1, 6 Inp2: 2, 5, 6 Inp3: 2, 6, 6 Axis: 1
xa_nn_cat_16_16	Inp1: 4, 3, 8, 1, 2 Inp2: 4, 3, 7, 1, 2 Inp3: 4, 3, 6, 1, 2 Inp4: 4, 3, 5, 1, 2 Axis: 2

Kernel name	Parameters
	Inp1: 8, 5 Inp2: 9, 5 Inp3: 10, 5 Axis: 0
	Inp1: 5, 4 Inp2: 4, 4 Inp3: 9, 4 Axis: 0
	Inp1: 8, 8, 3 Inp2: 3, 8, 3 Axis: 0
	Inp1: 5, 3, 2, 3, 4 Inp2: 5, 4, 2, 3, 4 Inp3: 5, 6, 2, 3, 4 Inp4: 5, 1, 2, 3, 4 Axis: 1
	Inp1: 6, 4, 8 Inp2: 6, 4, 12 Inp3: 6, 4, 6 Inp4: 6, 4, 2 Axis: 2
	Inp1: 7, 6, 7 Inp2: 9, 6, 7 Inp3: 3, 6, 7 Axis: 0
	Inp1: 6, 11, 6, 12 Inp2: 6, 13, 6, 12 Axis: 1
	Inp1: 2, 2, 15 Inp2: 2, 2, 14 Inp3: 2, 2, 13 Inp4: 2, 2, 7 Axis: 2
	Inp1: 9, 5 Inp2: 9, 7 Inp3: 9, 3 Inp4: 9, 11 Axis: 1
	Inp1: 2, 2, 3, 4, 11 Inp2: 2, 2, 3, 4, 12 Axis: 4
xa_nn_cat_32_32	Inp1: 7, 9, 9, 2 Inp2: 7, 9, 2, 2 Inp3: 7, 9, 1, 2 Inp4: 7, 9, 3, 2 Axis: 2

Kernel name	Parameters
	Inp1: 2, 8, 6, 7, 8 Inp2: 2, 1, 6, 7, 8 Axis: 1
	Inp1: 8, 6 Inp2: 8, 7 Inp3: 8, 8 Axis: 1
	Inp1: 6, 6, 5 Inp2: 7, 6, 5 Inp3: 9, 6, 5 Inp4: 2, 6, 5 Axis: 0
	Inp1: 6, 2, 3, 4 Inp2: 7, 2, 3, 4 Inp3: 2, 2, 3, 4 Axis: 0
	Inp1: 6, 5, 4 Inp2: 6, 5, 4 Inp3: 6, 5, 4 Axis: 0
	Inp1: 3, 6, 8, 3, 4 Inp2: 3, 6, 2, 3, 4 Inp3: 3, 6, 1, 3, 4 Axis: 2
	Inp1: 2, 1, 4, 5 Inp2: 2, 3, 4, 5 Inp3: 2, 2, 4, 5 Axis: 1
	Inp1: 9, 8, 7 Inp2: 9, 2, 7 Axis: 1
	Inp1: 2, 2, 3, 3, 4 Inp2: 2, 2, 4, 3, 4 Inp3: 2, 2, 2, 3, 4 Axis: 2
	Inp1: 6, 7, 2 Inp2: 6, 7, 2 Inp1 1, 7, 2 Axis: 0
xa_nn_cat_8u_8u	Inp1: 5, 4, 6 Inp2: 5, 8, 6 Inp3: 5, 2, 6 Inp4: 5, 7, 6 Axis: 1

Kernel name	Parameters
	Inp1: 4, 5, 6, 7 Inp2: 4, 5, 6, 3 Inp3: 4, 5, 6, 1 Inp4: 4, 5, 6, 2 Axis: 3
	Inp1: 5, 7, 8 Inp2: 5, 7, 2 Axis: 2
	Inp1: 8, 2, 3, 4 Inp2: 6, 2, 3, 4 Axis: 0
	Inp1: 3, 5, 6, 4, 8 Inp2: 3, 5, 6, 6, 8 Inp3: 3, 5, 6, 1, 8 Axis: 3
	Inp1: 1, 2, 7, 5 Inp2: 3, 2, 7, 5 Inp3: 5, 2, 7, 5 Inp4: 7, 2, 7, 5 Axis: 0
	Inp1: 6, 5, 6 Inp2: 7, 5, 6 Inp3: 5, 5, 6 Axis: 0
	Inp1: 3, 6, 9 Inp2: 3, 6, 12 Axis: 2
	Inp1: 5, 4, 3, 6 Inp2: 5, 4, 3, 9 Axis: 3
	Inp1: 2, 1, 6 Inp2: 2, 5, 6 Inp3: 2, 6, 6 Axis: 1
xa_nn_cat_16u_16u	Inp1: 2, 1, 9, 3, 4 Inp2: 2, 1, 7, 3, 4 Inp3: 2, 1, 5, 3, 4 Inp4: 2, 1, 3, 3, 4 Axis: 2
	Inp1: 2, 7 Inp2: 4, 7 Inp3: 6, 7 Axis: 0
	Inp1: 5, 9 Inp2: 5, 4 Inp3: 5, 7 Axis: 1
	Inp1: 8, 8, 3 Inp2: 3, 8, 3 Axis: 0
	Inp1: 5, 3, 2, 3, 4

Kernel name	Parameters
	Inp2: 5, 4, 2, 3, 4 Inp3: 5, 6, 2, 3, 4 Inp4: 5, 1, 2, 3, 4 Axis: 1
	Inp1: 6, 4, 8 Inp2: 6, 4, 12 Inp3: 6, 4, 6 Inp4: 6, 4, 2 Axis: 2
	Inp1: 7, 6, 7 Inp2: 9, 6, 7 Inp3: 3, 6, 7 Axis: 0
	Inp1: 8, 1, 6, 14 Inp2: 8, 3, 6, 14 Axis: 1
	Inp1: 4, 11, 5 Inp2: 4, 11, 4 Inp3: 4, 11, 3 Inp4: 4, 11, 7 Axis: 2
	Inp1: 3, 5 Inp2: 3, 7 Inp3: 3, 3 Inp4: 3, 11 Axis: 1
	Inp1: 2, 2, 3, 4, 11 Inp2: 2, 2, 3, 4, 12 Axis: 4
xa_nn_cat_32u_32u	Inp1: 9, 7, 3, 4 Inp2: 9, 7, 1, 4 Inp3: 9, 7, 2, 4 Inp4: 9, 7, 7, 4 Axis: 2
	Inp1 :2, 7, 6, 4, 8 Inp2: 2, 7, 6, 1, 8 Axis: 3
	Inp1: 3, 6 Inp2: 3, 7 Inp3: 3, 8
	Inp1: 2, 6, 6, 5 Inp2: 2, 7, 6, 5 Inp3: 2, 9, 6, 5 Inp4: 2, 2, 6, 5 Axis: 1

Kernel name	Parameters
	Inp1: 6, 2, 3, 4 Inp2: 7, 2, 3, 4 Inp3: 2, 2, 3, 4 Axis: 0
	Inp1: 6, 5, 4 Inp2: 6, 5, 4 Inp3: 6, 5, 4
	Inp1: 8, 3, 5, 2, 5 Inp2: 8, 3, 7, 2, 5 Inp3: 8, 3, 1, 2, 5 Axis: 2
	Inp1: 4, 1, 2, 7 Inp2: 4, 3, 2, 7 Inp3: 4, 2, 2, 7 Axis: 1
	Inp1: 10, 2, 7 Inp2: 3, 2, 7 Axis: 0
	Inp1: 2, 3, 3, 4 Inp2: 2, 4, 3, 4 Inp3: 2, 2, 3, 4 Axis: 1
	Inp1: 5, 9, 2, 2 Inp2: 8, 9, 2, 2 Inp3: 1, 9, 2, 2 Axis: 0
xa_nn_elm_quantize_f32_asym4	Inp: 11, 4, 5, 11
	Inp: 8, 4, 8, 9
	Inp: 8, 7, 5, 5
	Inp: 11, 5, 9, 7
	Inp: 7, 6, 10, 5, 5
	Inp: 7, 9, 11, 11
	Inp: 5, 10, 9, 8
	Axis=2 Inp: 9, 5, 9, 10
	Axis=4 Inp: 4, 10, 6, 10, 6
	Axis=2 Inp: 8, 5, 10, 8
	Axis=4 Inp: 7, 11, 11, 6, 5
	Axis=3 Inp: 8, 5, 4, 7

Kernel name	Parameters
	Axis=1 Inp: 4, 8, 5, 9, 11
	Axis=0 Inp: 4, 9, 4, 8
xa_nn_elm_quantize_f32_asym8	Inp: 6, 11, 11, 10, 5
	Inp: 4, 7, 4, 7
	Inp: 9, 8, 6, 7
	Inp: 10, 6, 4, 10, 10
	Inp: 6, 5, 10, 11, 8
	Inp: 4, 5, 6, 4
	Inp: 8, 7, 7, 5
	Axis=1 Inp: 11, 10, 5, 10
	Axis=0 Inp: 5, 10, 8, 10, 8
	Axis=2 Inp: 11, 10, 8, 4, 9
	Axis=2 Inp: 5, 8, 4, 8, 10
	Axis=3 Inp: 11, 4, 5, 6
	Axis=1 Inp: 8, 10, 11, 4
	Axis=4 Inp: 6, 7, 6, 10, 7
xa_nn_elm_quantize_f32_asym16	Inp: 5, 6, 8, 7, 7
	Inp: 6, 7, 6, 10, 5
	Inp: 5, 10, 9, 11
	Inp: 4, 9, 4, 9, 6
	Inp: 9, 11, 9, 9, 7
	Inp: 7, 9, 7, 10
	Inp: 5, 11, 9, 6
	Axis=1 Inp: 6, 10, 10, 5
	Axis=2 Inp: 11, 6, 10, 11
	Axis=4 Inp: 7, 6, 6, 4, 5
	Axis=2 Inp: 4, 11, 4, 8, 8

Kernel name	Parameters
	Axis=1 Inp: 6, 7, 7, 9, 8
	Axis=2 Inp: 7, 7, 11, 9
	Axis=1 Inp: 8, 8, 9, 7
xa_nn_elm_quantize_f32_asym4u	Inp: 7, 4, 6, 6
	Inp: 9, 5, 11, 4, 9
	Inp: 5, 7, 4, 11, 11
	Inp: 11, 4, 7, 10
	Inp: 11, 10, 6, 9, 9
	Inp: 8, 4, 6, 10
	Inp: 8, 8, 5, 4, 10
	Axis=4 Inp: 8, 9, 9, 11, 10
	Axis=0 Inp: 11, 4, 6, 6
	Axis=1 Inp: 10, 8, 5, 4
	Axis=0 Inp: 8, 4, 9, 4, 8
	Axis=4 Inp: 4, 8, 5, 8, 4
	Axis=4 Inp: 7, 10, 5, 11, 5
	Axis=1 Inp: 9, 7, 7, 8, 5
xa_nn_elm_quantize_f32_asym8u	Inp: 7, 5, 6, 11
	Inp: 9, 4, 11, 4
	Inp: 9, 6, 7, 10
	Inp: 5, 4, 5, 6, 4
	Inp: 5, 9, 5, 9, 9
	Inp: 6, 11, 5, 6
	Inp: 6, 6, 4, 7, 11
	Axis=1 Inp: 10, 7, 10, 4
	Axis=0 Inp: 7, 10, 6, 10, 9
	Axis=2 Inp: 7, 10, 11, 4

Kernel name	Parameters
	Axis=3 Inp: 4, 6, 5, 6, 11
	Axis=2 Inp: 10, 8, 7, 11, 9
	Axis=2 Inp: 8, 5, 5, 5
	Axis=4 Inp: 4, 11, 6, 10, 9
xa_nn_elm_quantize_f32_asym16u	Inp: 10, 10, 8, 4, 6
	Inp: 5, 9, 8, 11
	Inp: 4, 5, 11, 10, 9
	Inp: 9, 4, 4, 5
	Inp: 6, 11, 9, 5
	Inp: 6, 8, 4, 5, 8
	Inp: 9, 8, 7, 7, 7
	Axis=3 Inp: 10, 8, 4, 7
	Axis=2 Inp: 7, 10, 6, 10, 9
	Axis=1 Inp: 10, 4, 10, 10
	Axis=1 Inp: 6, 5, 8, 8, 9
	Axis=1 Inp: 5, 5, 9, 5, 5
	Axis=0 Inp: 5, 8, 8, 6, 11
	Axis=2 Inp: 6, 8, 11, 5, 9
xa_nn_elm_quantize_f32_sym4	Inp: 6, 6, 4, 10, 5
	Inp: 9, 6, 9, 7
	Inp: 9, 4, 6, 11, 10
	Inp: 11, 8, 4, 7, 4
	Inp: 8, 5, 5, 5
	Inp: 6, 7, 11, 11, 10
	Inp: 9, 4, 8, 11
	Axis=2 Inp: 6, 8, 5, 4
	Axis=1 Inp: 4, 7, 10, 10, 4

Kernel name	Parameters
	Axis=1 Inp: 5, 7, 10, 4
	Axis=4 Inp: 4, 6, 9, 6, 7
	Axis=0 Inp: 11, 7, 9, 7
	Axis=0 Inp: 7, 8, 8, 10
	Axis=0 Inp: 10, 11, 10, 7, 9
xa_nn_elm_quantize_f32_sym8	Inp: 7, 9, 10, 5, 7
	Inp: 4, 10, 11, 11
	Inp: 5, 8, 10, 9, 8
	Inp: 10, 8, 8, 8
	Inp: 4, 9, 6, 10
	Inp: 9, 9, 7, 9
	Inp: 6, 6, 5, 11, 4
	Axis=4 Inp: 6, 6, 4, 10, 7
	Axis=2 Inp: 5, 5, 7, 7
	Axis=2 Inp: 6, 9, 8, 10
	Axis=3 Inp: 8, 10, 6, 7, 10
	Axis=0 Inp: 5, 6, 5, 8, 11
	Axis=3 Inp: 5, 10, 6, 11, 11
	Axis=4 Inp: 4, 10, 6, 6, 6
xa_nn_elm_quantize_f32_sym16	Inp: 11, 6, 9, 6, 9
	Inp: 11, 6, 7, 6, 10
	Inp: 7, 11, 11, 11
	Inp: 10, 8, 8, 7, 11
	Inp: 10, 11, 6, 4
	Inp: 7, 11, 8, 6
	Inp: 8, 7, 9, 10
	Axis=1 Inp: 11, 4, 11, 5, 5

Kernel name	Parameters
	Axis=0 Inp: 5, 7, 7, 10
	Axis=3 Inp: 7, 5, 5, 8, 5
	Axis=3 Inp: 7, 9, 9, 10
	Axis=1 Inp: 4, 6, 10, 11
	Axis=0 Inp: 6, 6, 10, 11, 9
	Axis=1 Inp: 6, 4, 8, 6
xa_nn_elm_quantize_f32_sym4u	Inp: 10, 10, 5, 5
	Inp: 10, 10, 7, 6
	Inp: 5, 7, 10, 7
	Inp: 10, 4, 8, 6
	Inp: 10, 10, 9, 10, 7
	Inp: 8, 10, 6, 10
	Inp: 6, 10, 7, 8
	Axis=0 Inp: 11, 8, 10, 6
	Axis=2 Inp: 9, 10, 8, 8
	Axis=2 Inp: 7, 11, 4, 5
	Axis=3 Inp: 7, 5, 9, 10
	Axis=4 Inp: 10, 5, 7, 8, 10
	Axis=2 Inp: 8, 6, 5, 7, 8
	Axis=1 Inp: 8, 10, 5, 8
xa_nn_elm_quantize_f32_sym8u	Inp: 8, 7, 7, 8, 10
	Inp: 8, 6, 8, 7
	Inp: 6, 11, 4, 11, 10
	Inp: 11, 10, 11, 7
	Inp: 7, 7, 8, 6, 10
	Inp: 8, 5, 4, 7, 11
	Inp: 10, 6, 5, 11, 6

Kernel name	Parameters
	Axis=1 Inp: 10, 7, 10, 4
	Axis=0 Inp: 7, 10, 6, 10, 9
	Axis=2 Inp: 7, 10, 11, 4
	Axis=3 Inp: 4, 6, 5, 6, 11
	Axis=2 Inp: 10, 8, 7, 11, 9
	Axis=2 Inp: 8, 5, 5, 5
	Axis=4 Inp: 4, 11, 6, 10, 9
xa_nn_elm_quantize_f32_sym16u	Inp: 10, 10, 7, 9
	Inp: 7, 5, 6, 9
	Inp: 11, 6, 11, 11
	Inp: 5, 10, 8, 8, 10
	Inp: 9, 6, 11, 4
	Inp: 10, 4, 6, 7
	Inp: 11, 10, 7, 11, 8
	Axis=2 Inp: 8, 9, 5, 5
	Axis=0 Inp: 7, 7, 8, 11, 11
	Axis=0 Inp: 8, 6, 10, 11
	Axis=4 Inp: 9, 6, 10, 5, 10
	Axis=1 Inp: 8, 5, 10, 4, 5
	Axis=1 Inp: 7, 5, 4, 11
	Axis=2 Inp: 7, 11, 10, 11, 7
xa_nn_elm_dequantize_f32_asym4	Inp: 11, 4, 5, 11
	Inp: 8, 4, 8, 9
	Inp: 8, 7, 5, 5
	Inp: 11, 5, 9, 7
	Inp: 7, 6, 10, 5, 5
	Inp: 7, 9, 11, 11

Kernel name	Parameters
	Inp: 5, 10, 9, 8
	Axis=2 Inp: 9, 5, 9, 10
	Axis=4 Inp: 4, 10, 6, 10, 6
	Axis=2 Inp: 8, 5, 10, 8
	Axis=4 Inp: 7, 11, 11, 6, 5
	Axis=3 Inp: 8, 5, 4, 7
	Axis=1 Inp: 4, 8, 5, 9, 11
	Axis=0 Inp: 4, 9, 4, 8
xa_nn_elm_dequantize_f32_asym8	Inp: 6, 11, 11, 10, 5
	Inp: 4, 7, 4, 7
	Inp: 9, 8, 6, 7
	Inp: 10, 6, 4, 10, 10
	Inp: 6, 5, 10, 11, 8
	Inp: 4, 5, 6, 4
	Inp: 8, 7, 7, 5
	Axis=1 1 Inp: 1, 10, 5, 10
	Axis=0 Inp: 5, 10, 8, 10, 8
	Axis=2 Inp: 11, 10, 8, 4, 9
	Axis=2 Inp: 5, 8, 4, 8, 10
	Axis=3 Inp: 11, 4, 5, 6
	Axis=1 Inp: 8, 10, 11, 4
	Axis=4 Inp: 6, 7, 6, 10, 7
xa_nn_elm_dequantize_f32_asym16	Inp: 5, 6, 8, 7, 7
	Inp: 6, 7, 6, 10, 5
	Inp: 5, 10, 9, 11
	Inp: 4, 9, 4, 9, 6
	Inp: 9, 11, 9, 9, 7

Kernel name	Parameters
	Inp: 7, 9, 7, 10
	Inp: 5, 11, 9, 6
	Axis=1 Inp: 6, 10, 10, 5
	Axis=2 Inp: 11, 6, 10, 11
	Axis=4 Inp: 7, 6, 6, 4, 5
	Axis=2 Inp: 4, 11, 4, 8, 8
	Axis=1 Inp: 6, 7, 7, 9, 8
	Axis=2 Inp: 7, 7, 11, 9
	Axis=1 Inp: 8, 8, 9, 7
xa_nn_elm_dequantize_f32_asym4u	Inp: 7, 4, 6, 6
	Inp: 9, 5, 11, 4, 9
	Inp: 5, 7, 4, 11, 11
	Inp: 11, 4, 7, 10
	Inp: 11, 10, 6, 9, 9
	Inp: 8, 4, 6, 10
	Inp: 8, 8, 5, 4, 10
	Axis=4 Inp: 8, 9, 9, 11, 10
	Axis=0 Inp: 11, 4, 6, 6
	Axis=1 Inp: 10, 8, 5, 4
	Axis=0 Inp: 8, 4, 9, 4, 8
	Axis=4 Inp: 4, 8, 5, 8, 4
	Axis=4 Inp: 7, 10, 5, 11, 5
	Axis=1 Inp: 9, 7, 7, 8, 5
xa_nn_elm_dequantize_f32_asym8u	Inp: 7, 5, 6, 11
	Inp: 9, 4, 11, 4
	Inp: 9, 6, 7, 10
	Inp: 5, 4, 5, 6, 4

Kernel name	Parameters
	Inp: 5, 9, 5, 9, 9
	Inp: 6, 11, 5, 6
	Inp: 6, 6, 4, 7, 11
	Axis=1 Inp: 10, 7, 10, 4
	Axis=0 Inp: 7, 10, 6, 10, 9
	Axis=2 Inp: 7, 10, 11, 4
	Axis=3 Inp: 4, 6, 5, 6, 11
	Axis=2 Inp: 10, 8, 7, 11, 9
	Axis=2 Inp: 8, 5, 5, 5
	Axis=4 Inp: 4, 11, 6, 10, 9
xa_nn_elm_dequantize_f32_asym16u	Inp: 10, 10, 8, 4, 6
	Inp: 5, 9, 8, 11
	Inp: 4, 5, 11, 10, 9
	Inp: 9, 4, 4, 5
	Inp: 6, 11, 9, 5
	Inp: 6, 8, 4, 5, 8
	Inp: 9, 8, 7, 7, 7
	Axis=3 Inp: 10, 8, 4, 7
	Axis=2 Inp: 7, 10, 6, 10, 9
	Axis=1 Inp: 10, 4, 10, 10
	Axis=1 Inp: 6, 5, 8, 8, 9
	Axis=1 Inp: 5, 5, 9, 5, 5
	Axis=0 Inp: 5, 8, 8, 6, 11
	Axis=2 Inp: 6, 8, 11, 5, 9
xa_nn_elm_dequantize_f32_sym4	Inp: 6, 6, 4, 10, 5
	Inp: 9, 6, 9, 7
	Inp: 9, 4, 6, 11, 10

Kernel name	Parameters
	Inp: 11, 8, 4, 7, 4
	Inp: 8, 5, 5, 5
	Inp: 6, 7, 11, 11, 10
	Inp: 9, 4, 8, 11
	Axis=2 Inp: 6, 8, 5, 4
	Axis=1 Inp: 4, 7, 10, 10, 4
	Axis=1 Inp: 5, 7, 10, 4
	Axis=4 Inp: 4, 6, 9, 6, 7
	Axis=0 Inp: 11, 7, 9, 7
	Axis=0 Inp: 7, 8, 8, 10
	Axis=0 Inp: 10, 11, 10, 7, 9
xa_nn_elm_dequantize_f32_sym8	Inp: 7, 9, 10, 5, 7
	Inp: 4, 10, 11, 11
	Inp: 5, 8, 10, 9, 8
	Inp: 10, 8, 8, 8
	Inp: 4, 9, 6, 10
	Inp: 9, 9, 7, 9
	Inp: 6, 6, 5, 11, 4
	Axis=4 Inp: 6, 6, 4, 10, 7
	Axis=2 Inp: 5, 5, 7, 7
	Axis=2 Inp: 6, 9, 8, 10
	Axis=3 Inp: 8, 10, 6, 7, 10
	Axis=0 Inp: 5, 6, 5, 8, 11
	Axis=3 Inp: 5, 10, 6, 11, 11
	Axis=4 Inp: 4, 10, 6, 6, 6
xa_nn_elm_dequantize_f32_sym16	Inp: 11, 6, 9, 6, 9
	Inp: 11, 6, 7, 6, 10

Kernel name	Parameters
	Inp: 7, 11, 11, 11
	Inp: 10, 8, 8, 7, 11
	Inp: 10, 11, 6, 4
	Inp: 7, 11, 8, 6
	Inp: 8, 7, 9, 10
	Axis=1 Inp: 11, 4, 11, 5, 5
	Axis=0 Inp: 5, 7, 7, 10
	Axis=3 Inp: 7, 5, 5, 8, 5
	Axis=3 Inp: 7, 9, 9, 10
	Axis=1 Inp: 4, 6, 10, 11
	Axis=0 Inp: 6, 6, 10, 11, 9
	Axis=1 Inp: 6, 4, 8, 6
xa_nn_elm_dequantize_f32_sym4u	Inp: 10, 10, 5, 5
	Inp: 10, 10, 7, 6
	Inp: 5, 7, 10, 7
	Inp: 10, 4, 8, 6
	Inp: 10, 10, 9, 10, 7
	Inp: 8, 10, 6, 10
	Inp: 6, 10, 7, 8
	Axis=0 Inp: 11, 8, 10, 6
	Axis=2 Inp: 9, 10, 8, 8
	Axis=2 Inp: 7, 11, 4, 5
	Axis=3 Inp: 7, 5, 9, 10
	Axis=4 Inp: 10, 5, 7, 8, 10
	Axis=2 Inp: 8, 6, 5, 7, 8
	Axis=1 Inp: 8, 10, 5, 8
xa_nn_elm_dequantize_f32_sym8u	Inp: 8, 7, 7, 8, 10

Kernel name	Parameters
	Inp: 8, 6, 8, 7
	Inp: 6, 11, 4, 11, 10
	Inp: 11, 10, 11, 7
	Inp: 7, 7, 8, 6, 10
	Inp: 8, 5, 4, 7, 11
	Inp: 10, 6, 5, 11, 6
	Axis=1 Inp: 10, 7, 10, 4
	Axis=0 Inp: 7, 10, 6, 10, 9
	Axis=2 Inp: 7, 10, 11, 4
	Axis=3 Inp: 4, 6, 5, 6, 11
	Axis=2 Inp: 10, 8, 7, 11, 9
	Axis=2 Inp: 8, 5, 5, 5
	Axis=4 Inp: 4, 11, 6, 10, 9
xa_nn_elm_dequantize_f32_sym16u	Inp: 10, 10, 7, 9
	Inp: 7, 5, 6, 9
	Inp: 11, 6, 11, 11
	Inp: 5, 10, 8, 8, 10
	Inp: 9, 6, 11, 4
	Inp: 10, 4, 6, 7
	Inp: 11, 10, 7, 11, 8
	Axis=2 Inp: 8, 9, 5, 5
	Axis=0 Inp: 7, 7, 8, 11, 11
	Axis=0 Inp: 8, 6, 10, 11
	Axis=4 Inp: 9, 6, 10, 5, 10
	Axis=1 Inp: 8, 5, 10, 4, 5
	Axis=1 Inp: 7, 5, 4, 11
	Axis=2 Inp: 7, 11, 10, 11, 7

3. References

[1] FusionG3-NNLib-API.pdf