STM32 Cube.AI

SPC5 Studio.AI

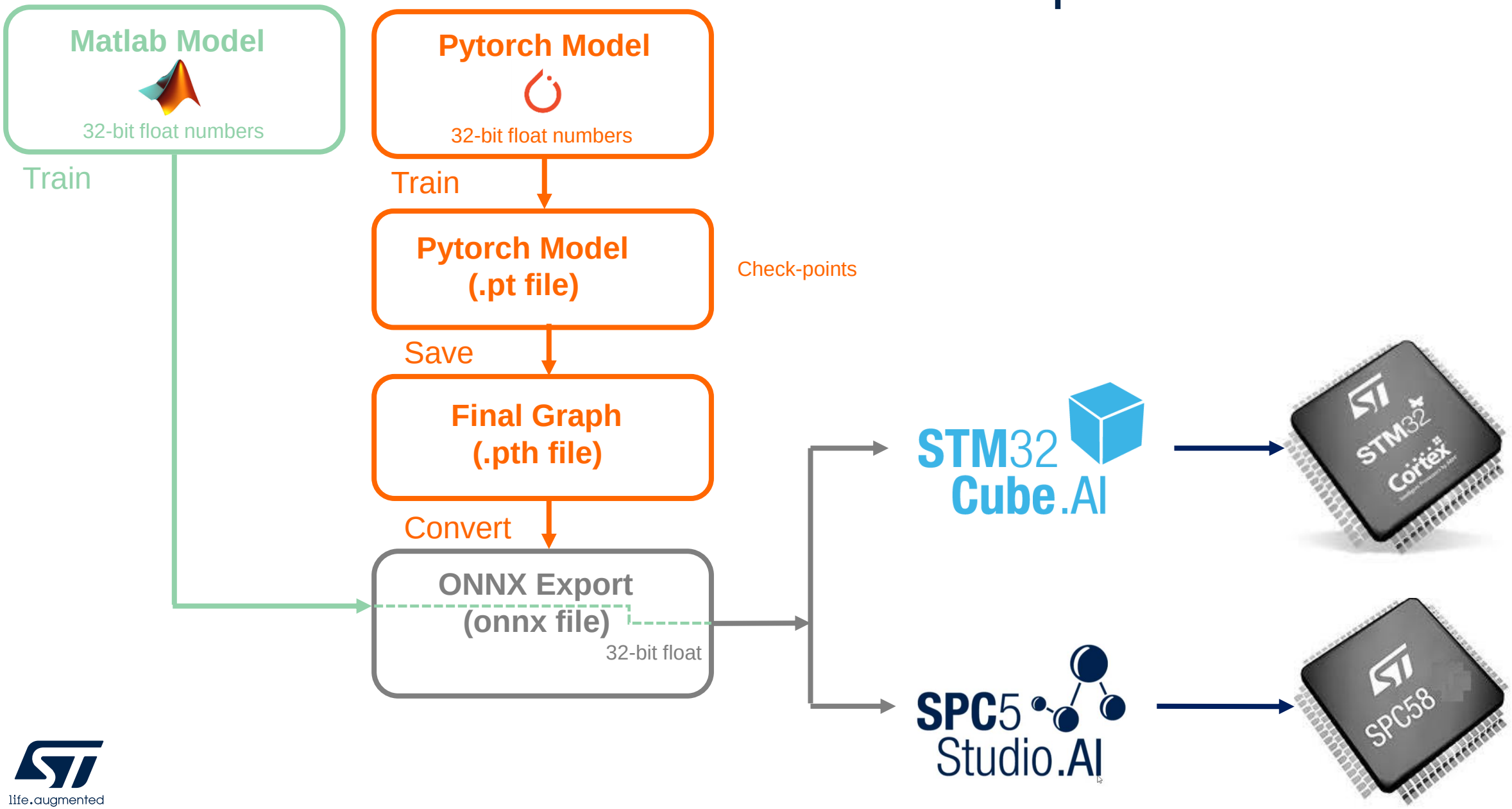
life.augmented

Flows and Tools to map ONNX Neural Networks on Micro-controllers

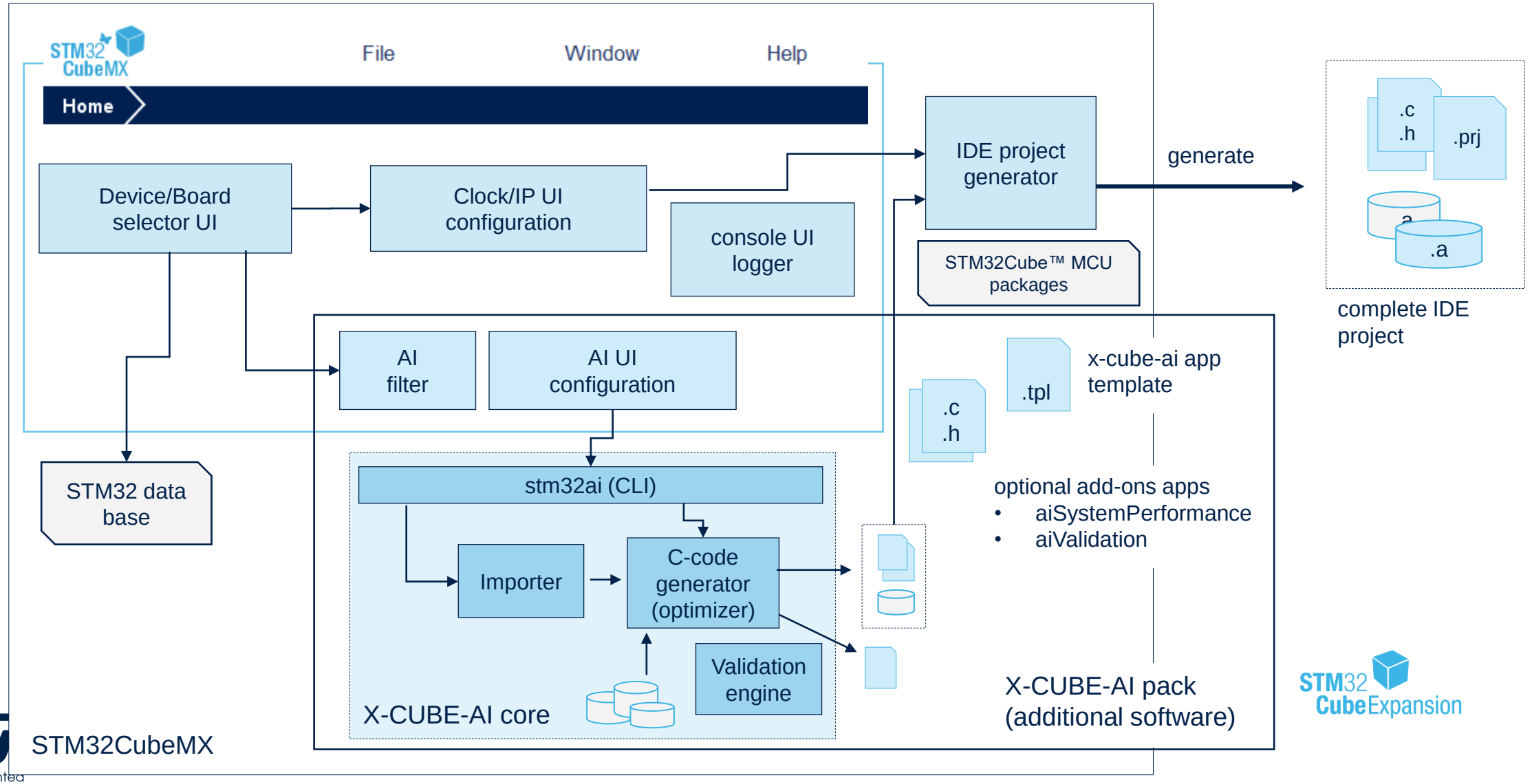
Oct 14th 2020

Danilo Pau
Technical Director, IEEE & ST Fellow
System Research and Applications
STMicroelectronics, Agrate Brianza

Development Flow in use



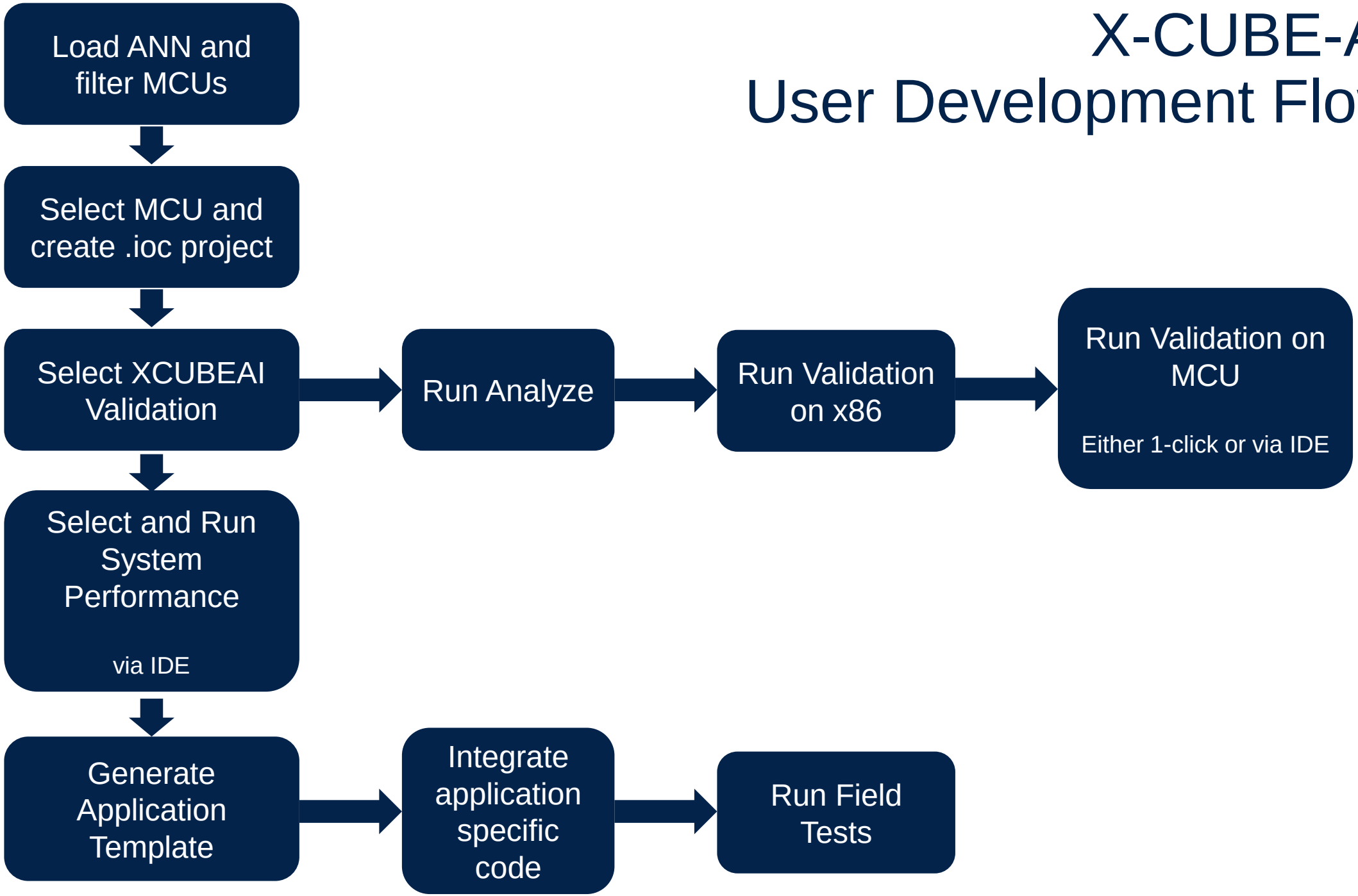
X-CUBE-AI package as STM32CubeMX cube expansion



STM32CubeMX

STM32 CubeExpansion

X-CUBE-AI
User Development Flow



life.augmented

Case Study: ESC-50 (Environmental Sound Classification)

- Dataset
  - 50 classes
  - 40 audio files, 5 sec per class
  - Sampling frequency of recordings: 44.1 KHz
  - Available @ https://github.com/karolpiczak/ESC-50
- Pre processing
  - For each recording, time-frequency spectrogram using 2048 samples windows and 512 samples stride size
  - Transformation of the frequency scale into Mel scale using 128 mel-features
  - Division of the spectrogram into 220ms intervals (128x16 matrix)
  - Ignore low energy spectra whose Frobenius norm is less than 1e-4
  - Normalization respect to maximum energy

Case Study: ESC-50 (Environmental Sound Classification)

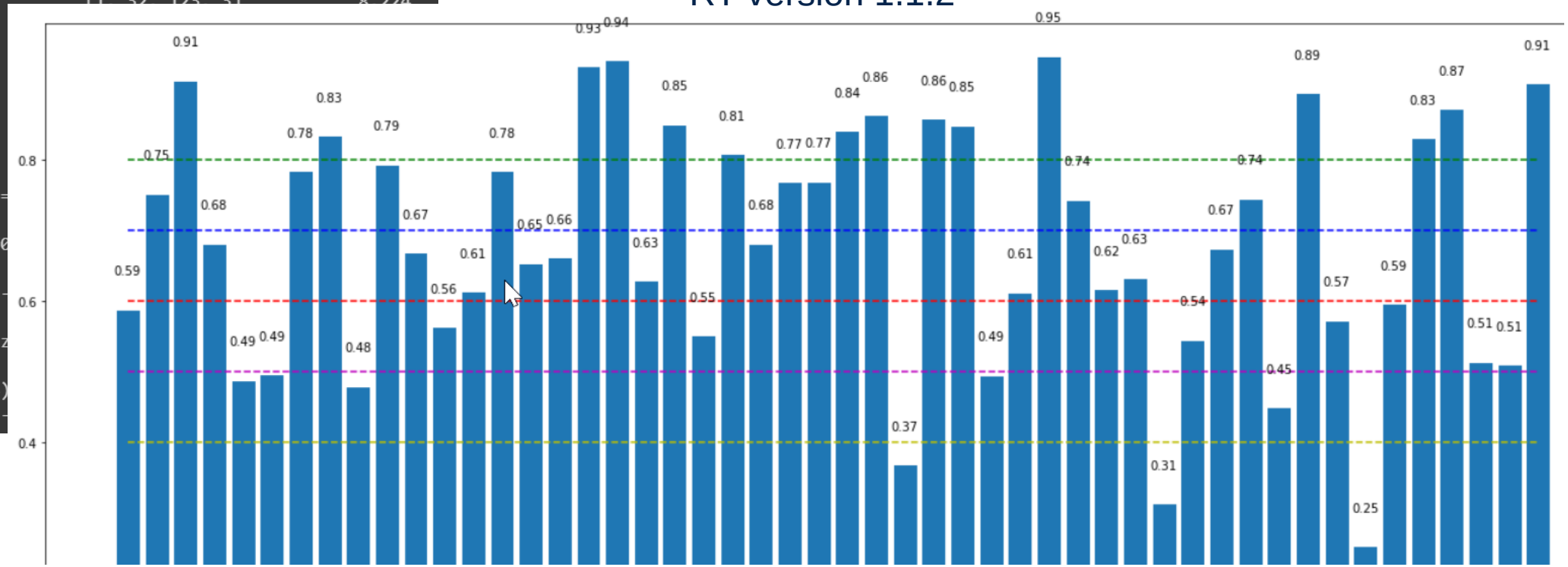
ConvNet (**Pytorch 1.6.0+cu101**)

Layer (type)	Output Shape	Param #
Conv2d-1	[1, 32, 126, 6]	320
ReLU-2	[1, 32, 126, 6]	0
Conv2d-3	[1, 64, 125, 5]	8,256
ReLU-4	[1, 64, 125, 5]	0
AvgPool2d-5	[1, 64, 124, 4]	0
Conv2d-6	[1, 32, 123, 3]	8,224
ReLU-7		
Conv2d-8		
ReLU-9		
AvgPool2d-10		
Flatten-11		
Linear-12		
Linear-13		

Total params: 150,370
Trainable params: 150,370
Non-trainable params: 0

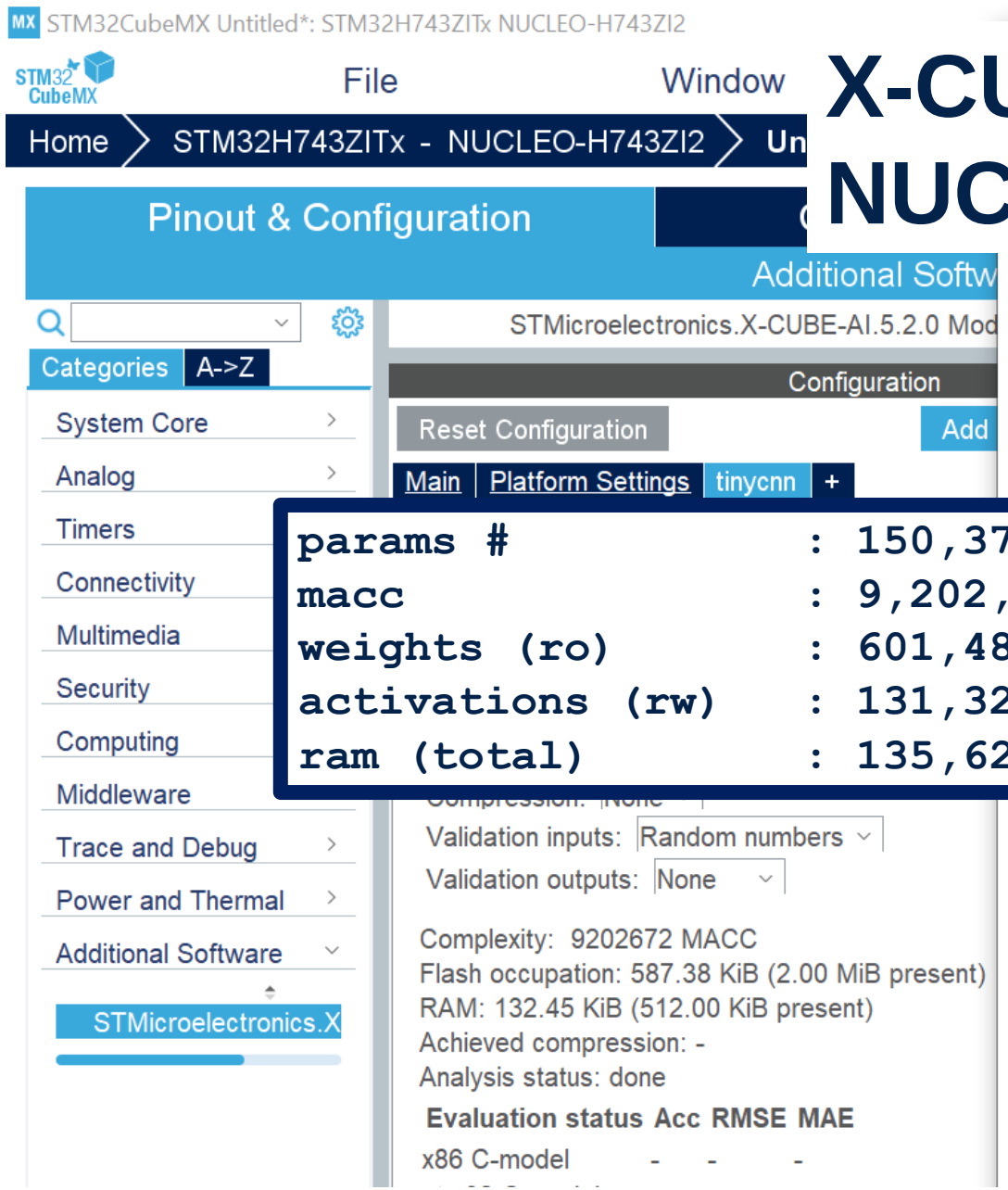
Input size (MB): 0.00
Forward/backward pass siz
Params size (MB): 0.57
Estimated Total Size (MB)

- Batch size : 100
- Epochs : 200 with early exit
- Optimizer : Adam
- Loss function : Cross Entropy
- Onnx 1.6.0
- RT version 1.1.2



life.augmented

X-CUBE-AI 5.2.0
NUCLEO-STM32H743ZI2, 480MHZ

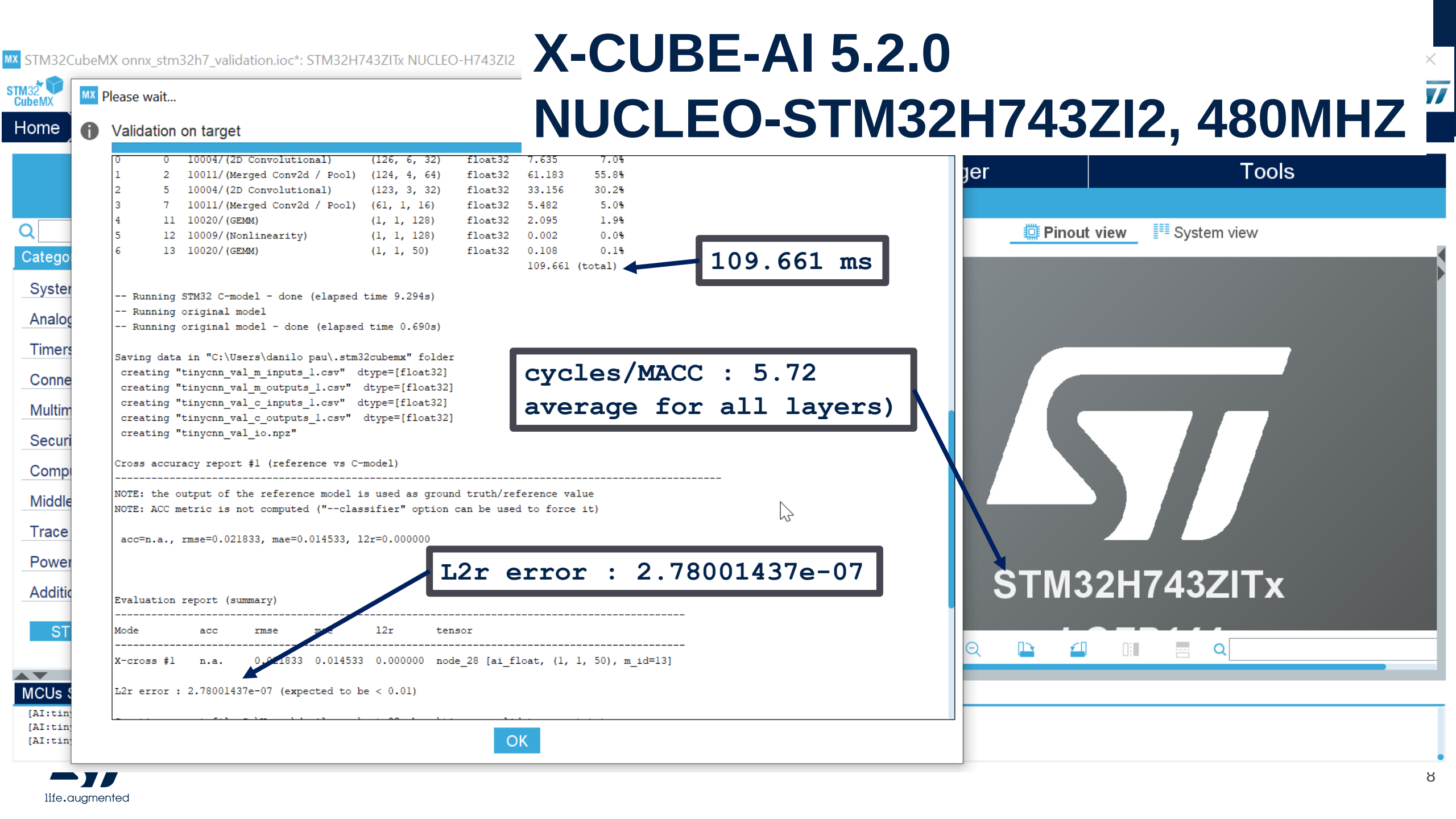



```
params #            : 150,370 items (587.38 KiB)
macc                : 9,202,672
weights (ro)        : 601,480 B (587.38 KiB)
activations (rw)    : 131,328 B (128.25 KiB)
ram (total)         : 135,624 B (132.45 KiB) = 131,328 + 4,096 + 200
```

```
ram (total)         : 135,624 B (132.45 KiB) = 131,328 + 4,096 + 200

id  layer (type)            output shape      param #     connected to              macc           rom
0   input1 (Input)          (128, 8, 1)
    node_13 (Conv2D)        (126, 6, 32)      320         input1                    241,952        1,280
1   node_14 (Nonlinearity)  (126, 6, 32)                  node_13                                  3,024
7   node_21 (Conv2D)        (122, 2, 16)      2,064       node_20                   507,536        8,256
8   node_22 (Nonlinearity)  (122, 2, 16)                  node_21
9   node_24 (Pool)          (61, 1, 16)                   node_22
10  node_25 (Reshape)       (976,)                        node_24
11  fc1weight (Placeholder) (128, 976)        124,928
    fc1bias (Placeholder)   (128,)            128
    node_26 (Gemm)          (1, 128)                      node_25                   124,928
                                                          fc1weight
                                                          fc1bias
```

X-CUBE-AI 5.2.0
NUCLEO-STM32H743ZI2, 480MHZ



Case Study: Speech Denoise

https://it.mathworks.com/help/deeplearning/ug/denoise-speech-using-deep-learning-networks.html
https://it.mathworks.com/matlabcentral/fileexchange/67296-deep-learning-toolbox-converter-for-onnx-model-format

```
params #           : 33,125 items (129.39 KiB)
macc               : 4,141,181
weights (ro)       : 132,500 B (129.39 KiB)
activations (rw)   : 16,152 B (15.77 KiB)
ram (total)        : 20,796 B (20.31 KiB) = 16,152 + 4,128 + 516
```

SPC5-AI v.2.0.0
SPC584B, 120MHZ

```
Results for 10 inference(s) @120/120MHz (macc:4141181)
 device      : 0x55AA55AA/UNKNOW @120MHz/120MHz (No FPU)
 duration    : 348.927 ms (average)
 CPU cycles  : 41871248 (average)
 cycles/MACC : 10.11 (average for all layers)
 c_nodes     : 17
```

Clayer	id	desc	oshape	fmt	ms
0	0	10022/(Container)	(129, 8, 1)	float32	0.393
1	1	10004/(2D Convolutional)	(129, 1, 18)	float32	16.768
2	3	10004/(2D Convolutional)	(129, 1, 30)	float32	28.135
3	5	10004/(2D Convolutional)	(129, 1, 8)	float32	22.520
4	7	10004/(2D Convolutional)	(129, 1, 18)	float32	16.780
5	9	10004/(2D Convolutional)	(129, 1, 30)	float32	28.122
6	11	10004/(2D Convolutional)	(129, 1, 8)	float32	22.530
7	13	10004/(2D Convolutional)	(129, 1, 18)	float32	16.779
8	15	10004/(2D Convolutional)	(129, 1, 30)	float32	28.132
9	17	10004/(2D Convolutional)	(129, 1, 8)	float32	22.522
10	19	10004/(2D Convolutional)	(129, 1, 18)	float32	16.789
11	21	10004/(2D Convolutional)	(129, 1, 30)	float32	28.123
12	23	10004/(2D Convolutional)	(129, 1, 8)	float32	22.531
13	25	10004/(2D Convolutional)	(129, 1, 18)	float32	16.792
14	27	10004/(2D Convolutional)	(129, 1, 30)	float32	28.135
15	29	10004/(2D Convolutional)	(129, 1, 8)	float32	22.521
16	31	10004/(2D Convolutional)	(129, 1, 1)	float32	11.355
					348.927 (total)

```
Complexity/l2r error per-layer - macc=4,141,181 rom=132,500
```

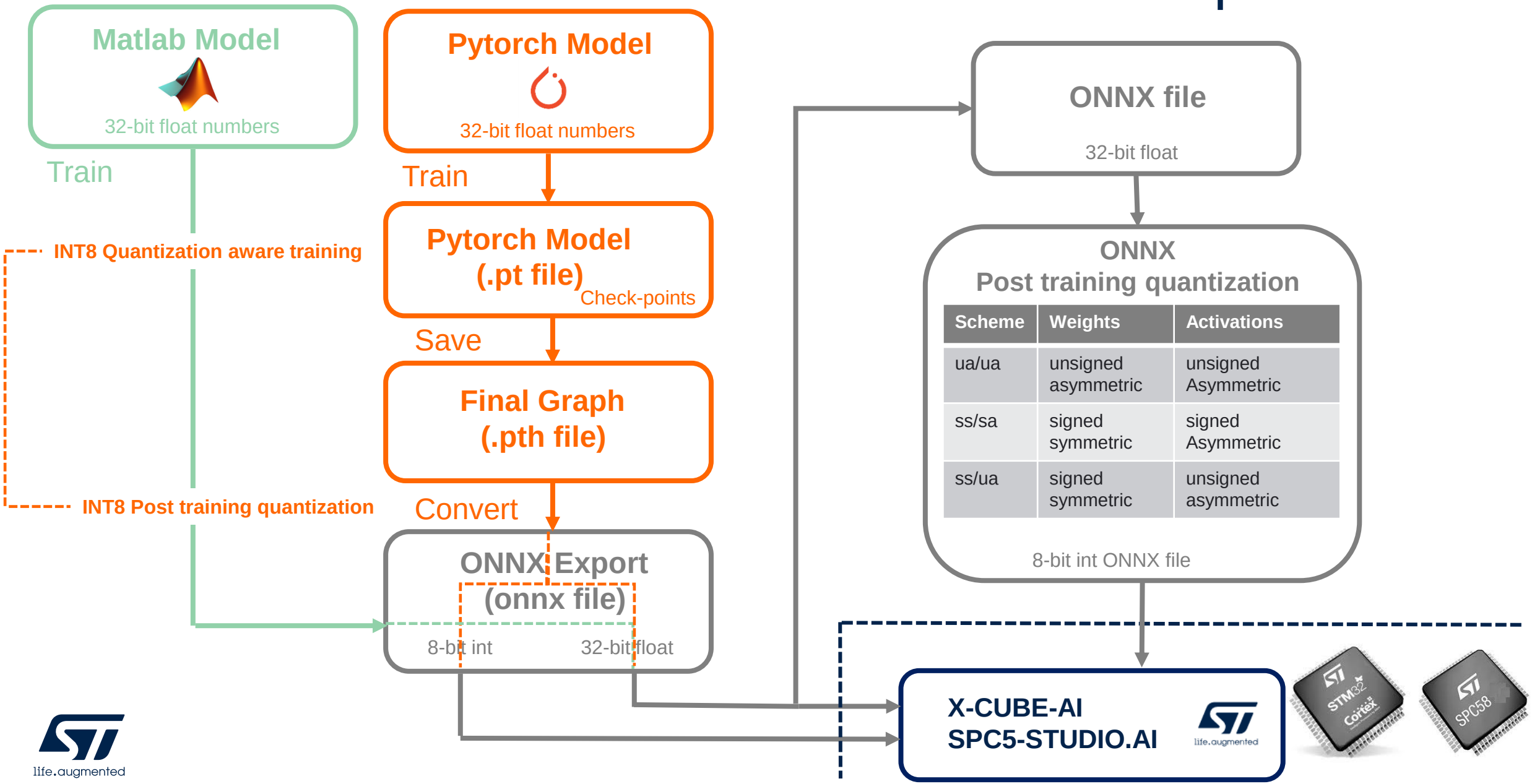
id	layer (type)	macc		rom		l2r error	
0	imageinput_Mean (Placeholder)		0.0%	||||||||	3.1%		
0	imageinput_Sub (Eltwise)		0.0%	|	0.0%		
1	conv_1 (Conv2D)	||||||||	4.1%	||||||||	4.0%		
3	conv_2 (Conv2D)	||||||||||||||||||||	8.5%	||||||||||||||||||||	8.2%		
5	conv_3 (Conv2D)	||||||||||||||||	6.8%	||||||||||||||||	6.5%		
7	conv_4 (Conv2D)	|||||||||	4.1%	|||||||||	4.0%		
9	conv_5 (Conv2D)	||||||||||||||||||||	8.5%	||||||||||||||||||||	8.2%		
11	conv_6 (Conv2D)	||||||||||||||||	6.8%	||||||||||||||||	6.5%		
13	conv_7 (Conv2D)	|||||||||	4.1%	|||||||||	4.0%		
15	conv_8 (Conv2D)	||||||||||||||||||||	8.5%	||||||||||||||||||||	8.2%		
17	conv_9 (Conv2D)	||||||||||||||||	6.8%	||||||||||||||||	6.5%		
19	conv_10 (Conv2D)	|||||||||	4.1%	|||||||||	4.0%		
21	conv_11 (Conv2D)	||||||||||||||||||||	8.5%	||||||||||||||||||||	8.2%		
23	conv_12 (Conv2D)	||||||||||||||||	6.8%	||||||||||||||||	6.5%		
25	conv_13 (Conv2D)	|||||||||	4.1%	|||||||||	4.0%		
27	conv_14 (Conv2D)	||||||||||||||||||||	8.5%	||||||||||||||||||||	8.2%		
29	conv_15 (Conv2D)	||||||||||||||||	6.8%	||||||||||||||||	6.5%		
31	conv_16 (Conv2D)	||||||||	3.2%	||||||||	3.1%	8.14623093e-07	*

L2r error 8.14623093e-07

Inference time 348.927 ms

life.augmented

Desired Development Flow



How to move forward :

- Needs
- Model zoo of Tiny networks for MCUs trained in Pytorch/Matlab/PaddlePaddle/? exported in ONNX
- Jupyter Notebook tutorials
  - Pytorch Tiny Neural Networks with int8 training aware/post training quantization procedures including exports to ONNX@int8 file format
  - ONNX@fp32 to ONNX@int8 Tiny Neural Networks with post training quantization procedures
- Support of int8 formats: ua/ua, ss/sa, ss/ua

Danilo Pau, graduated at Politecnico di Milano, on 1992 in Electronic Engineering. He joined SGS-THOMSON (now STMicroelectronics) on 1991 and worked on mpeg2 video memory reduction, then video coding, embedded graphics, computer vision, and currently on deep learning. During his career helped in transferring those developments into company products. Also funded and served as 1st Chairman of the STMicroelectronics Technical Staff Italian Community; he is currently Technical Director into System Research and Applications and a Fellow Member of ST. Since 2019 Danilo is an IEEE Fellow, serves as Industry Ambassador coordinator for IEEE Region 8 South Europe, is vice chair of the Task Force on “Intelligent Cyber-Physical Systems” within IEEE CIS and Member for the Machine learning, Deep learning and AI in CE (MDA) Technical Stream Committee IEEE Consumer Electronics Society (CESoc).

Contributed with 113 documents the development of Compact Descriptors for Visual Search (CDVS), CDVS successfully developed ISO-IEC 15938-13 MPEG standard. He was Funding Chair of MPEG Ad Hoc Group on Compact Descriptor for Video Analysis (CDVA), formerly Compact Descriptors for Video Search (CDViS). He also contributes (applications) to MPAI.community recently started by L. Chiariglione. His scientific production consists of 91 papers to date, 78 granted patents and more than 23 invited talks/seminars at various universities and conferences. He was also principal investigator into numerous funded projects at European and Italian level on embedded systems.

Danilo tutored lots of undergraduate students (till Msc graduation), Msc engineers and PhD students from various universities in Italy and India, one of the activities that he likes at most.

Thank you

© STMicroelectronics - All rights reserved.
ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.
For additional information about ST trademarks, please refer to www.st.com/trademarks.
All other product or service names are the property of their respective owners.

life.augmented