

# FPG-AI: a Technology Independent Framework for Edge AI Deployment Onboard Satellite, and its Characterisation on NanoXplore FPGAs

*Pietro Nannipieri\**, *Tommaso Pacini\**, *Tommaso Bocchi\**, *Matteo Dadà\**, *Luca Fanucci\**, *Silvia Moranti<sup>†</sup>*

*\*Department of Information Engineering, University of Pisa, Pisa, Italy*

*<sup>†</sup> European Space Research and Technology Centre, European Space Agency (ESA), Noordwijk, The Netherlands*

---

BRAVE Days 2023 – ESTEC, The Netherlands



Pietro Nannipieri  
[pietro.nannipieri@unipi.it](mailto:pietro.nannipieri@unipi.it)



## Outline

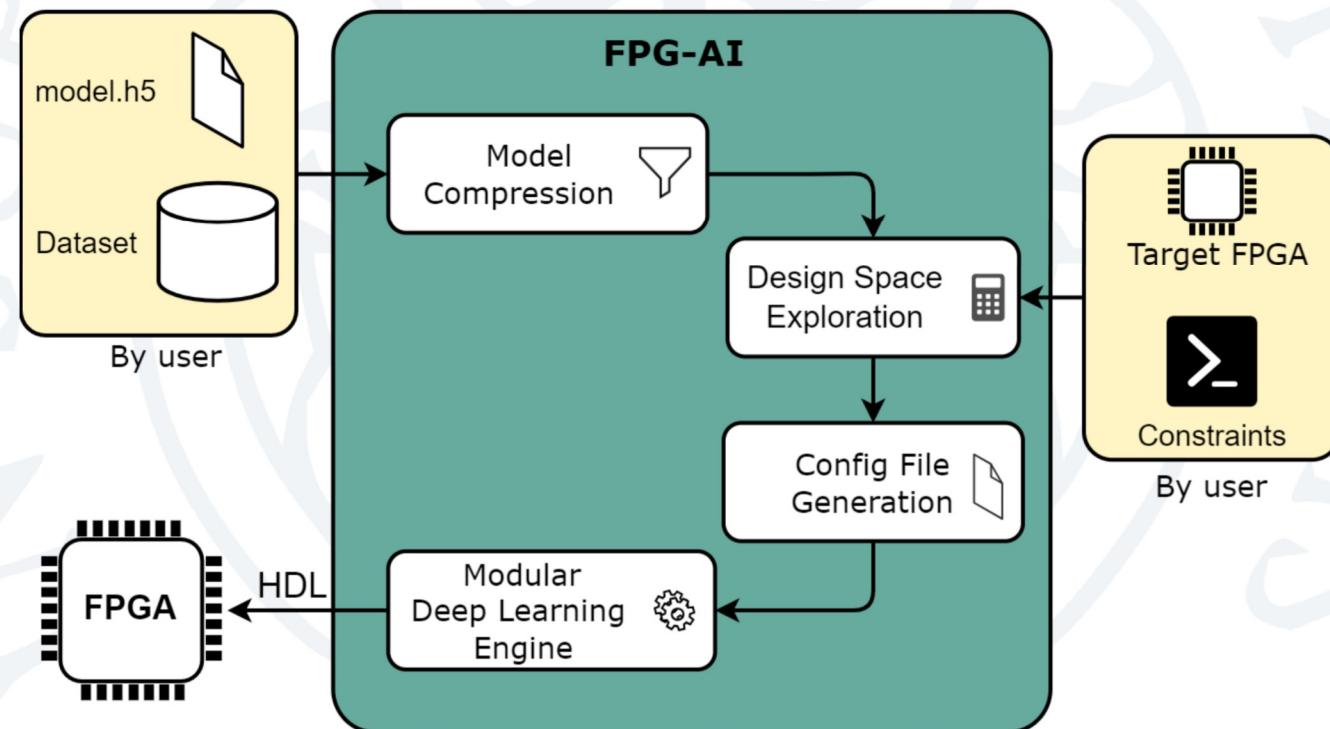
1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNN Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

## Outline

1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNN Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

# Background: FPG-AI Toolflow for CNNs

- Automation toolflow for efficient deployment of pre-trained CNN models on FPGA technology [1], [2]



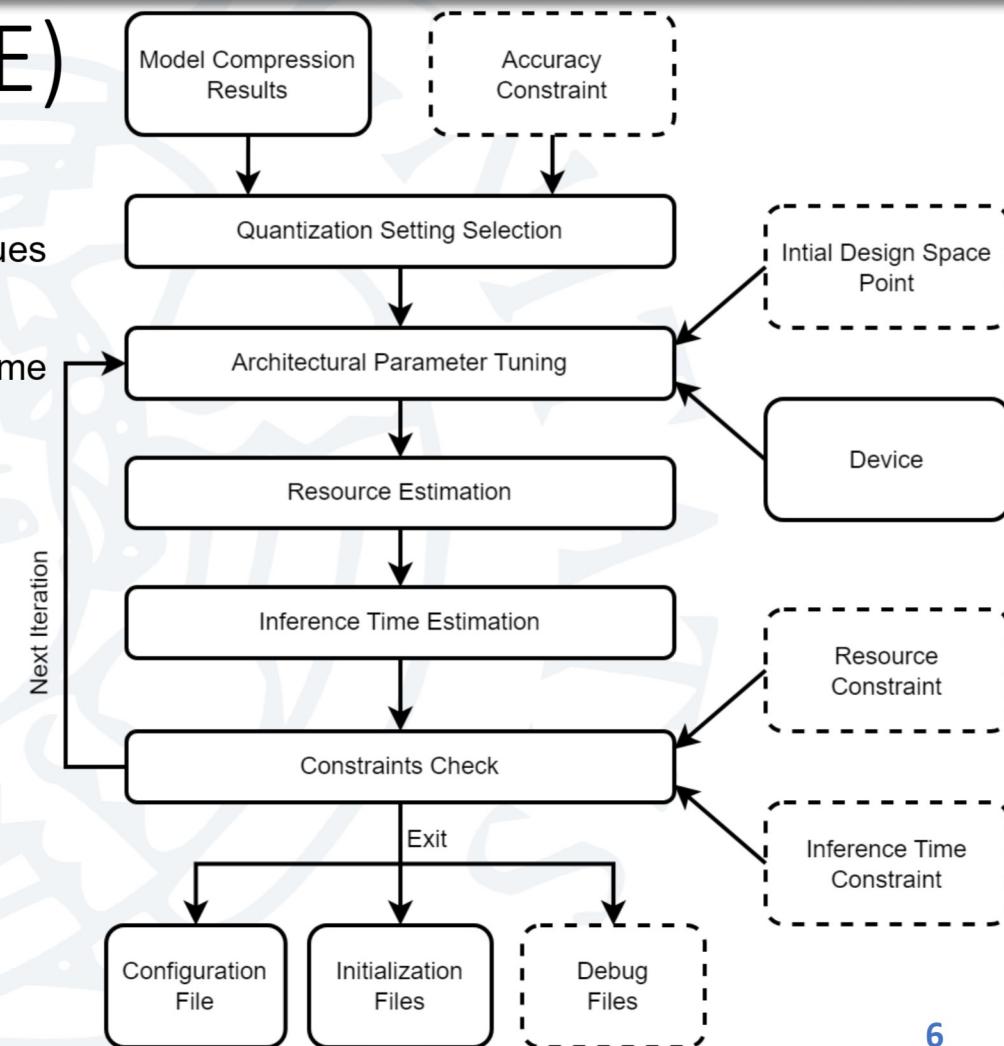
# Model Compression

- **Post-training** quantization:
  - **Layer-wise** quantization applied on layers' inputs and weights
  - **Truncation** at layers' output for further memory footprint reduction
  - From **floating-point** to **fixed-point** arithmetic for boosting hardware efficiency:
    - Timing performance
    - Area consumption
    - Power consumption
- **Layer folding:** Batch Normalization, Average Pooling



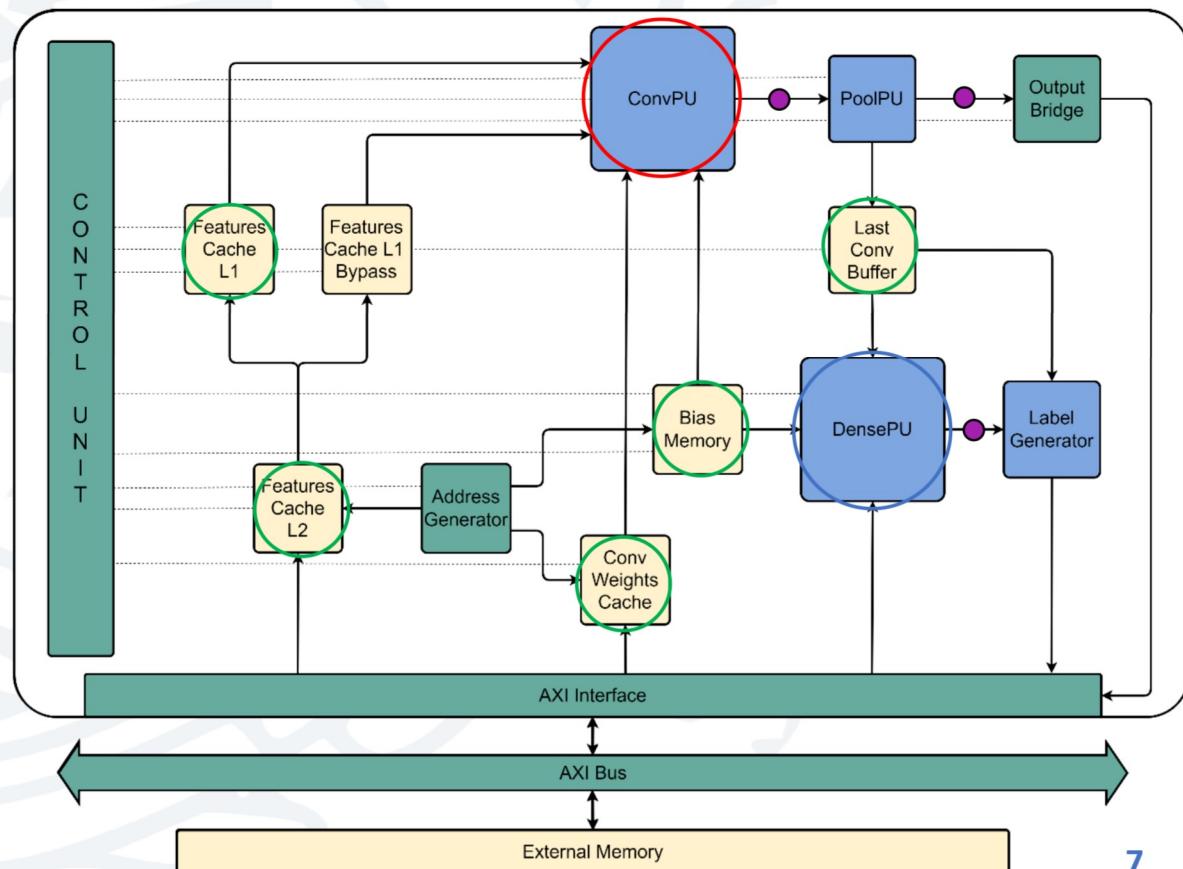
# Design Space Exploration (DSE)

- DSE inputs:
  - CNN model
  - Target FPGA
  - User's constraints (optional)
    - Parameters initial values
    - Minimum accuracy
    - Maximum inference time
    - Resources limitation
- Exploration of the architectural parameters space through an iterative algorithm
  - Detailed analytical Modular Deep-learning Engine (MDE) model for performance and resource estimation
- DSE outputs:
  - MDE configuration file
  - Initialization files for memories (Weights, Images for TBs, ...)
  - Textual debug files (optional Test Vectors for verification)



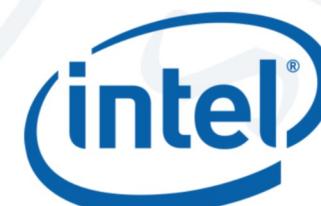
# Modular Deep Learning Engine (MDE)

- **High portability:** no third-party IPs used
- **High scalability** in terms of DSP/On-chip Memory utilization
- **Easily configurable** through a file (.vhd)
  - **Model parameters:**
    - Input shape, # Layers, Layers type, # Classes, etc.
  - **Quantization parameters:**
    - Inputs/weights bitwidths, Truncation and Saturation bits
  - **Architectural parameters:**
    - **# MAC units ( $N_{PE}$ )**, # Neurons, Memory primitives for each IP, etc.



## FPG-AI Key Features

- High degree of customization with respects to user's constraints on:
    - Resource consumption (DSP/On-chip memories)
    - Post-quantization application metric deviations
    - Inference time
  - Unmatched device portability of the Modular Deep Learning Engine (MDE) thanks to:
    - Absence of third-party IPs
    - High scalability in terms of DSP/On-chip memory usage
    - Fine-grain configurable through a .vhd file
- Enabling the implementation on FPGAs from different vendors and heterogeneous resource budgets!



## Outline

1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNN Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

## FPG-AI: Current Objectives

- Enabling the support for NanoXplore FPGA devices
- Extension of FPG-AI to Recurrent Neural Networks (RNNs)



The screenshot shows the 'ACTIVITY' page for the FPG-AI project on the European Space Agency's website. The title of the activity is 'FPG-AI: A TECHNOLOGY INDEPENDENT FRAMEWORK FOR EDGE AI DEPLOYMENT ONBOARD SATELLITE, AND ITS CHARACTERISATION ON NANOXPLORE FPGAs'. The page includes sections for 'Overview' and 'Events'. Key information displayed includes:

- Status:** RUNNING
- Prime contractor:** UNIVERSITA DI PISA
- Organisational Unit:** TEC-SF
- Start Date:** 31 March 2023
- Estimated End Date:** 01 October 2024
- DURATION:** 18 MONTHS
- Activity Type:** Early technology development
- Description:** G-AI: a Technology Independent Framework for Edge AI Deployment Onboard Satellite, and its Characterisation on NanoXplore FPGAs.
- Key Features:** AI Framework for AI-in-FPGA, Tools for Rigid-Hard FPGAs, Pool of AI Users for Space Applications.
- Objectives:** Available to the Space Community, Provide a Reference Platform for State-of-the-Art AI Libraries.
- Diagram:** A flowchart titled 'FPG-AI' showing the process from 'Code Generation' to 'Deployment' via 'Design Flow & Tools' and 'Hardware Optimisation'.

## Team Composition



- Full Prof. Luca Fanucci
  - Responsible for Management tasks



- ESA Staff Silvia Moranti
  - **Technical Officer**
  - ESA/ESTEC Microelectronics Section



- Assistant Prof. Pietro Nannipieri
  - Responsible for Hardware Prototyping and Dissemination Activity



- M. Sc Tommaso Pacini
  - PhD Candidate
  - Responsible for Extension to RNNs, Extension to NX FPGAs, Benchmark Activity



- M. Sc Matteo Dada
  - Scholarship holder
  - Working on Extension to RNNs



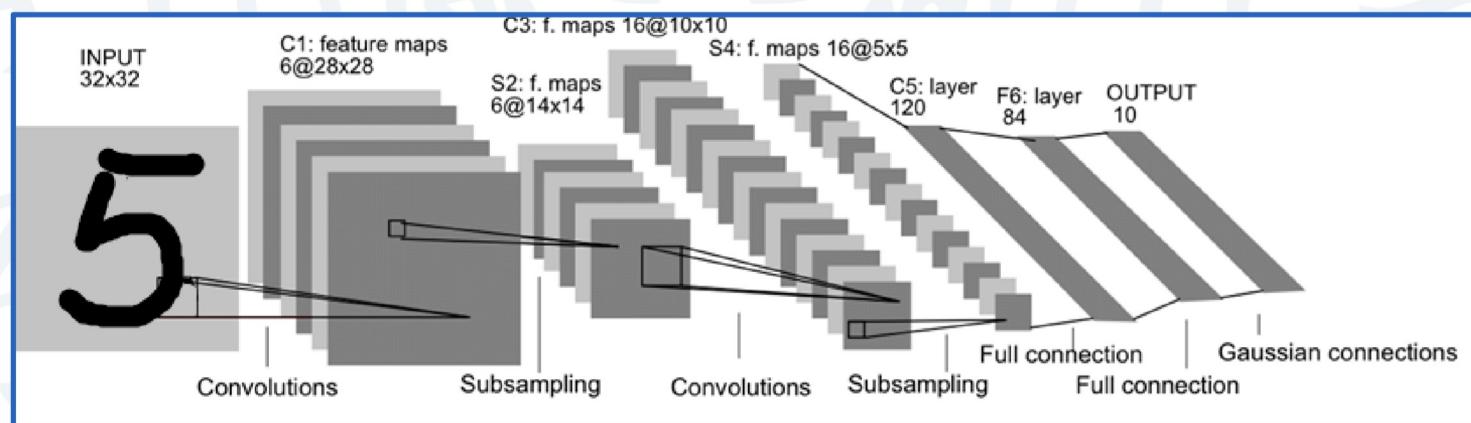
- M. Sc Student Tommaso Bocchi
  - Working on Extension to NX FPGAs, Hardware Prototyping, Benchmark Activity

## Outline

1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNNs Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

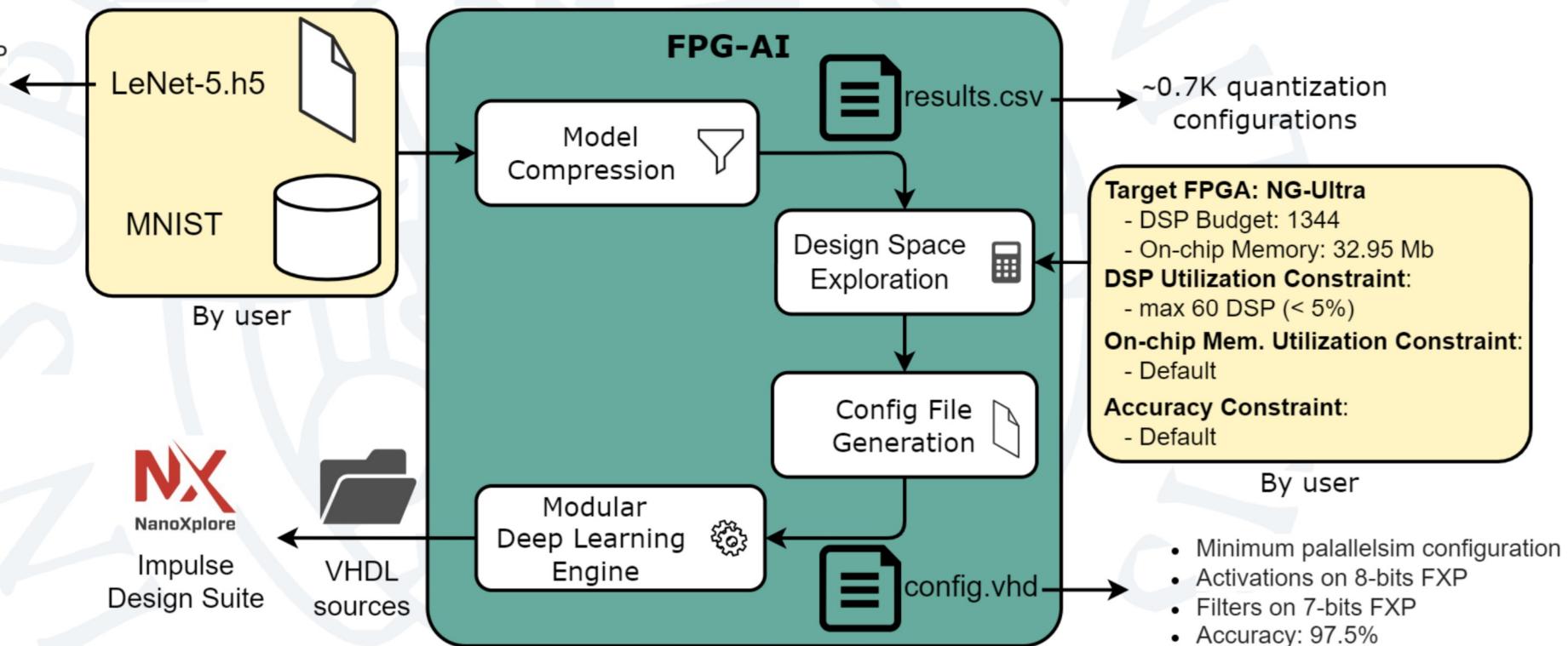
# Case Study

- Selected Convolutional Neural Network: **LeNet-5**
- Classification class: digits recognition
- Model topology: 2 Convolutional layers + 3 Fully Connected layers
- Total parameters: **44426 (~1.36 Mb)**

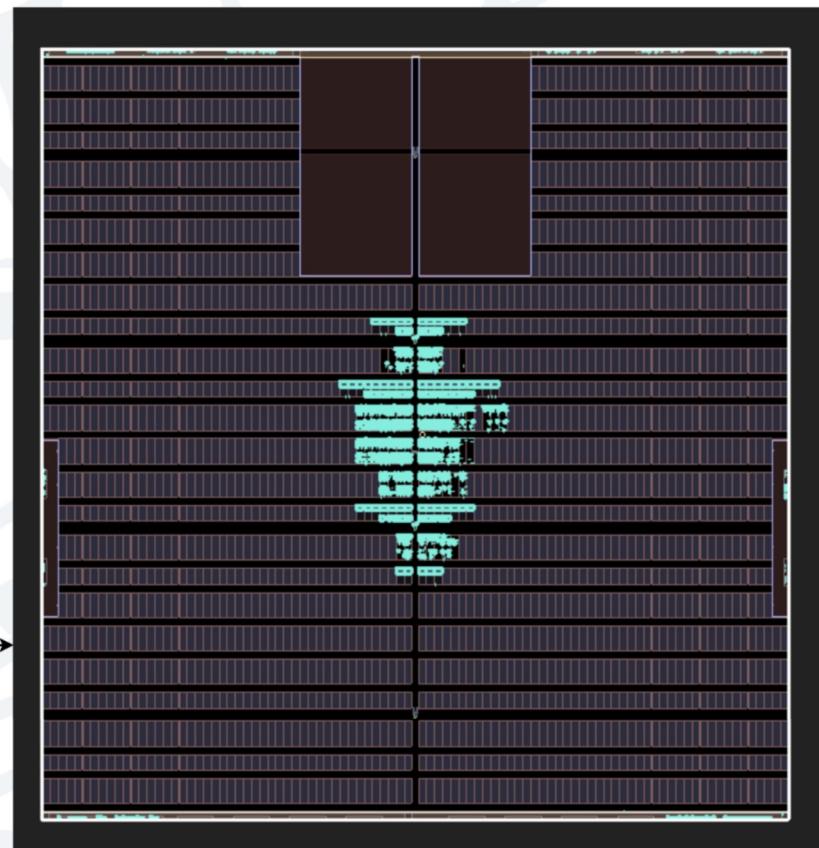
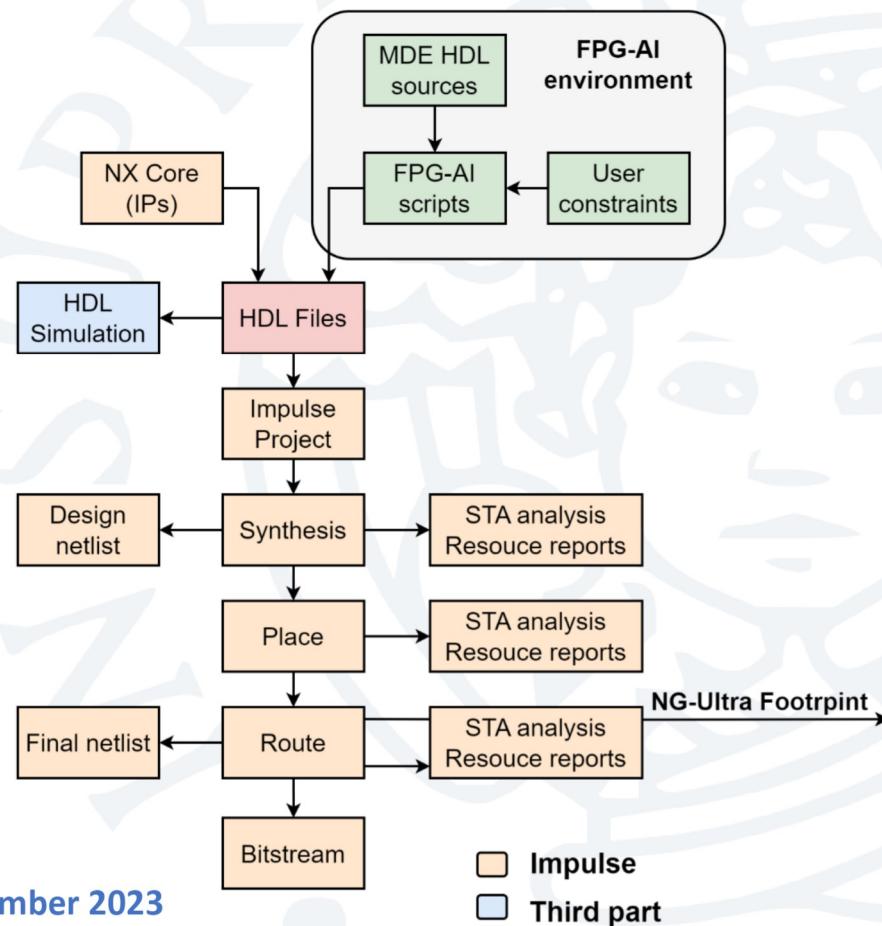


## Design Flow – FPG-AI

- Activations on 32-bits FP
- Filters on 32-bits FP
- Accuracy: 97.5%



# Design Flow – NX Impulse



## Outline

1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNN Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

# Comparison – Selected Devices



- Xilinx: **Z-7045**
- Technology: 28 nm
- Resources:
  - DSPs: 900 (DSP48E1)
  - On-chip memory: ~19.16 Mb
- Synthesis/Implementation strategy:
  - Default
- Timing corner analyzed:
  - Worst (0.922V, 100°C)



- NanoXplore: **NG-Ultra FF-1760**
- Technology: 28 nm
- Resources:
  - DSPs: 1344
  - On-chip memory: ~32.95 Mb
- Synthesis/Implementation strategy:
  - Default
- Timing corner analyzed:
  - Worst (0.95V, 125°C)

# Comparison – Resources

The diagram illustrates the resource utilization for two FPGAs: Xilinx Z-7045 and NanoXplore NG-ULTRA. It compares resources across three design steps: Synthesis, Place, and Route. Red boxes highlight specific resource types: 'LUT as Memory' and 'Memory block'.

		Design Step	Slice LUTs	Slice Registers	LUT as Memory	Block RAM	DSPs
<b>Xilinx Z-7045</b>	Synthesis	7471 (3.42%)	3015 (0.69%)	0 (0%)	5 (0.92%)	59 (6.56%)	
	Implementation	7083 (3.24%)	3015 (0.69%)	1496 (2.13%)	5 (0.92%)	59 (6.56%)	

		Design Step	4-LUT	DFF	Register file block	DSP	Memory block
<b>NanoXplore NG-ULTRA</b>	Synthesis	6194 (2%)	3540 (1%)	129 (5%)	51 (4%)	15 (3%)	
	Place	6106 (2%)	3554 (1%)	129 (5%)	51 (4%)	15 (3%)	
	Route	6106 (2%)	3578 (1%)	129 (5%)	51 (4%)	15 (3%)	

Resource utilization values:

- Xilinx Z-7045:**
  - LUT as Memory:** 0 (0%) at Synthesis, 1496 (2.13%) at Implementation.
  - Memory block:** 0.18 Mb (6.56% of total memory).
- NanoXplore NG-ULTRA:**
  - Register file block:** 129 (5%) at all stages.
  - Memory block:** 0.703 Mb (3% of total memory).

# Comparison – Timing

Xilinx Z-7045	Design Step	Required [MHz]	Actual [MHz]	Critical Path	
				Routing delay [ns]	Data delay [ns]
	Synthesis	113.64	114.44	3.242	5.017
	Implementation		113.70	2.933	5.003
NanoXplore NG-ULTRA	Design Step	Required [MHz]	Actual [MHz]	Critical Path	
	Synthesis	40.000	81.726	7.329	4.194
	Place		47.239*	15.899	4.319
	Route		40.393*	19.988	3.833

\*default seed option

## Outline

1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNN Implementation Flow
4. Benchmarking
5. Hardware Prototyping