# 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†*

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

*† 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

# Outline

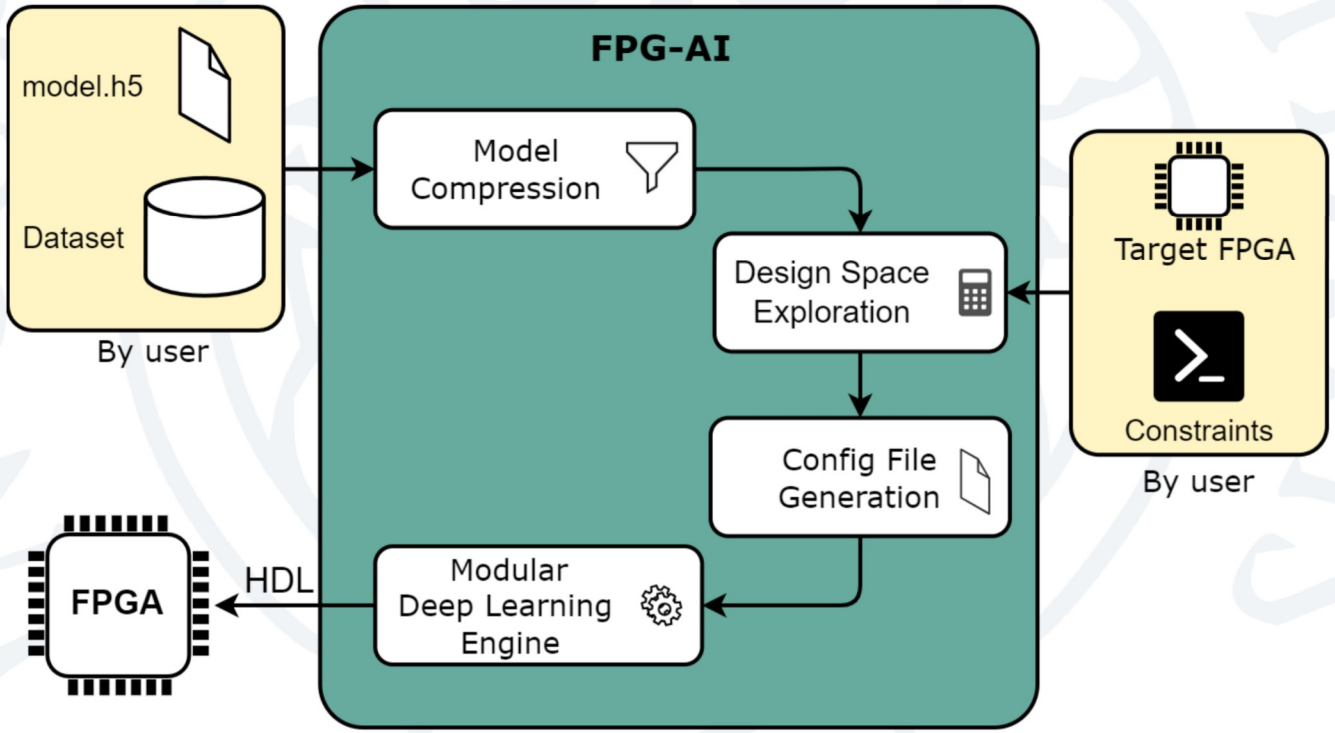
1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNN Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

# 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

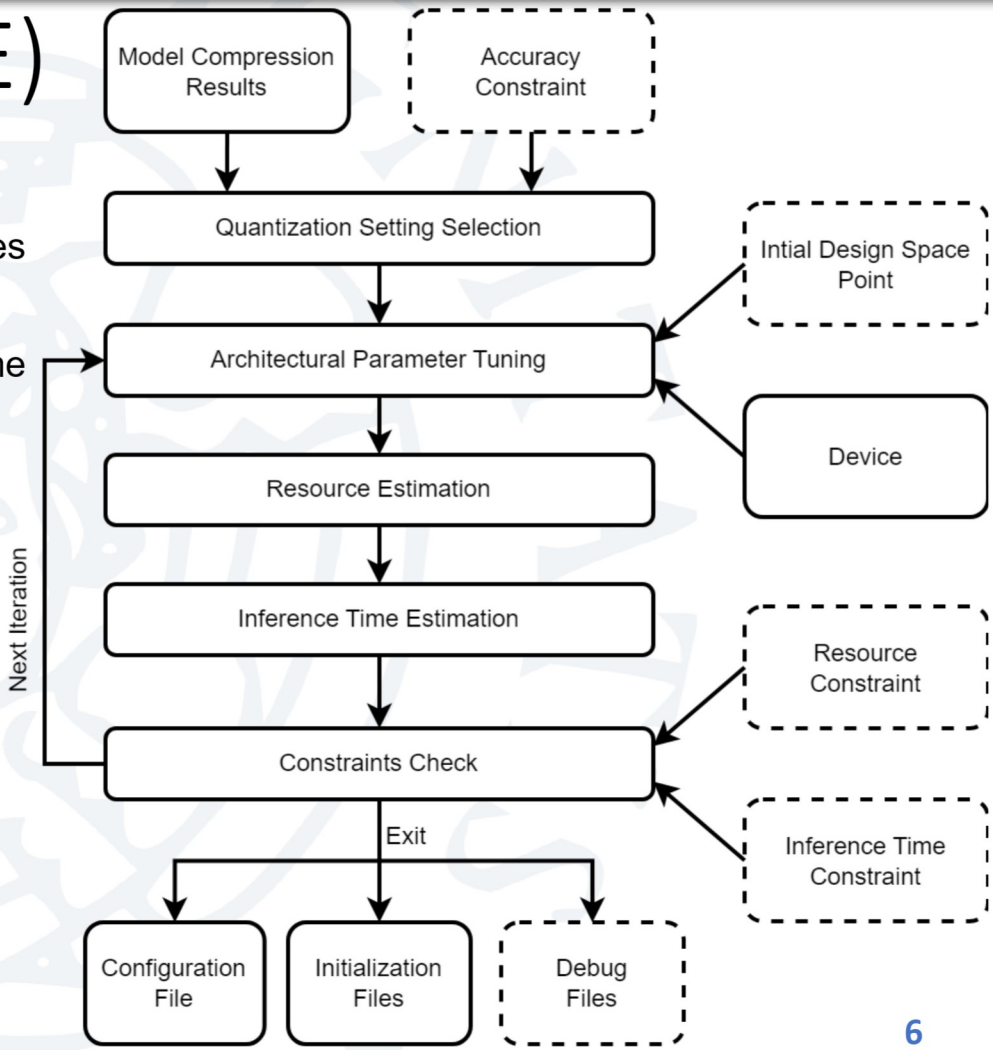
# Design Space Exploration (DSE)

- DSE inputs:
  - CNN model
  - Target FPGA
  - User's constraints (optional)

  → Parameters initial values
  → Minimum accuracy
  → Maximum inference time
  → Resources limitation

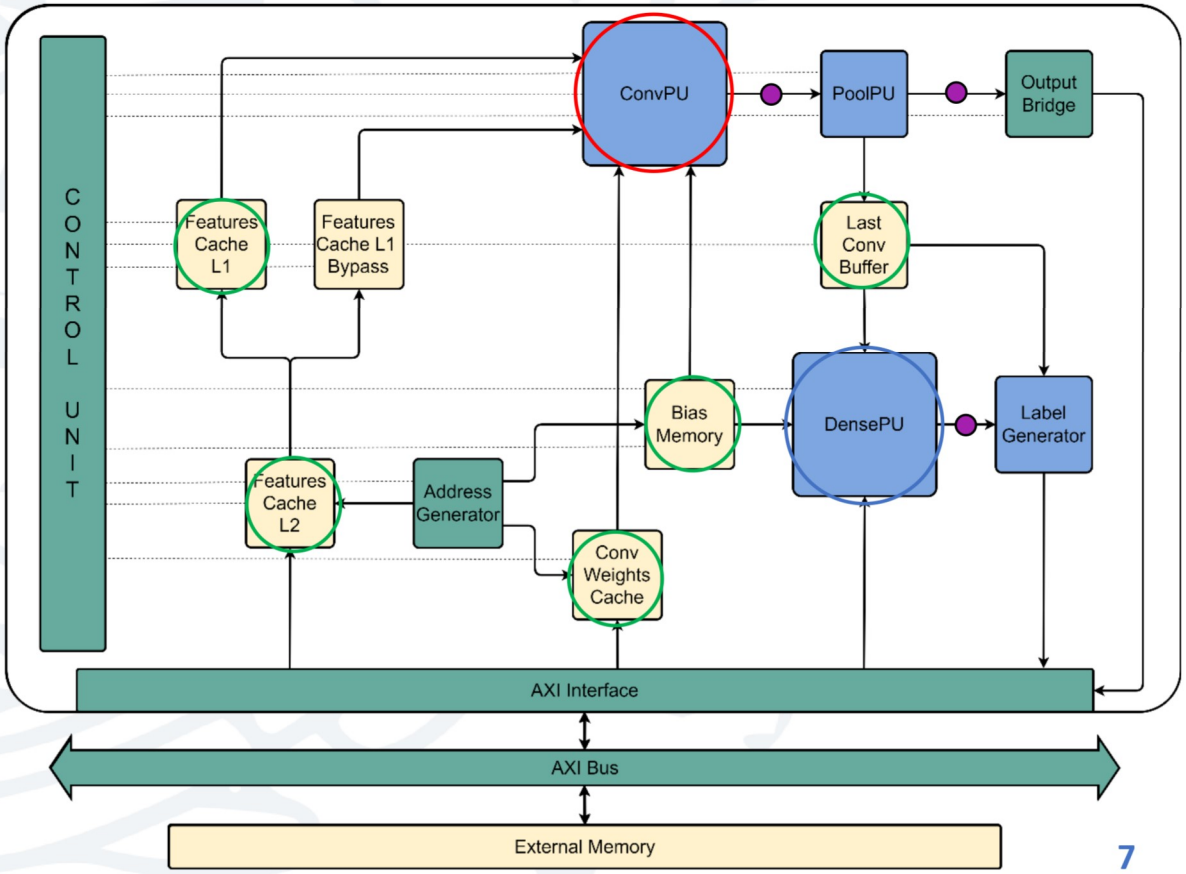
- Exploration of the architectural parameters space through an iterative algorithm
  - Detailed analytical Modular Deep-learning Engine (MDE) model for performance and resource estimation

- 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!

XILINX MICROCHIP intel

# 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

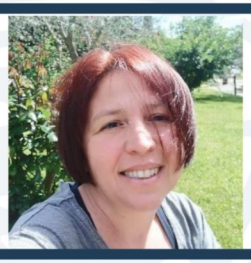
# FPG-AI: Current Objectives

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


https://activities.esa.int/index.php/4000141108

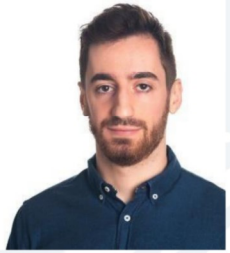
# Team Composition

- 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

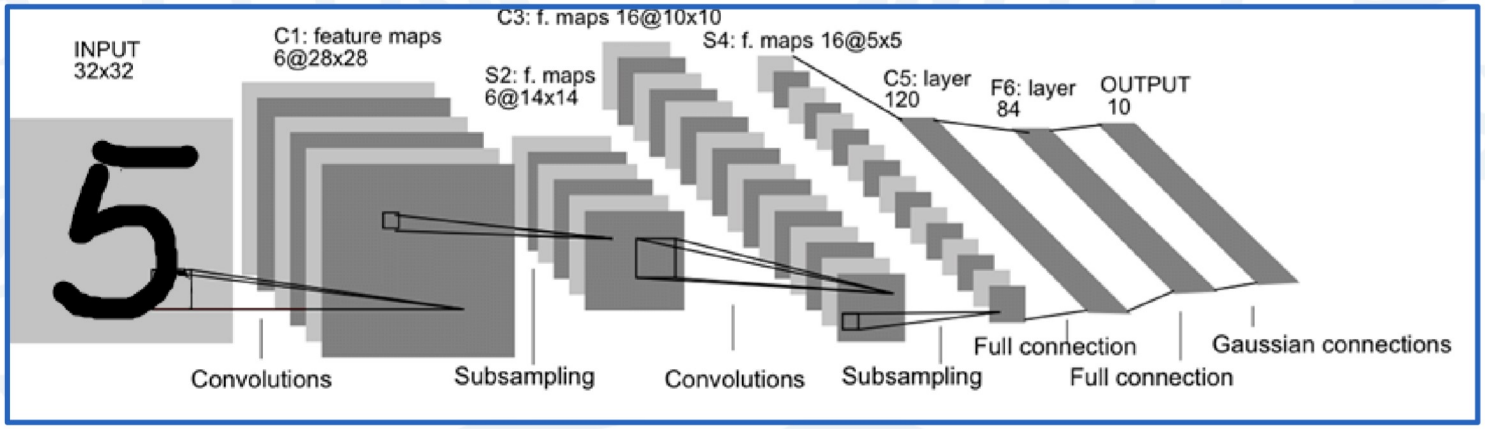
- M. Sc Student Tommaso Bocchi
  - Working on Extension to NX FPGAs, Hardware Prototyping, Benchmark Activity

# Outline

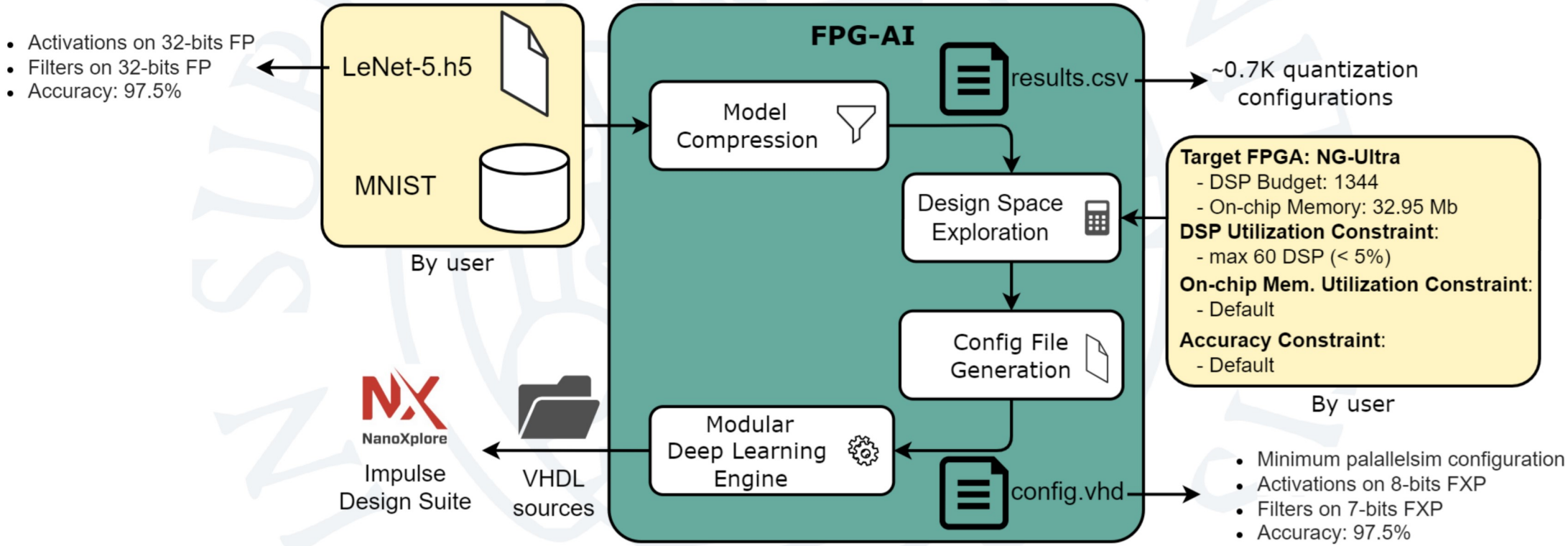
1. Background: FPG-AI Toolflow for CNNs
2. On-going Design Activities
3. CNNs Implementation Flow for NX
4. Benchmarking
5. Next Steps – Hardware Prototyping

# Case Study

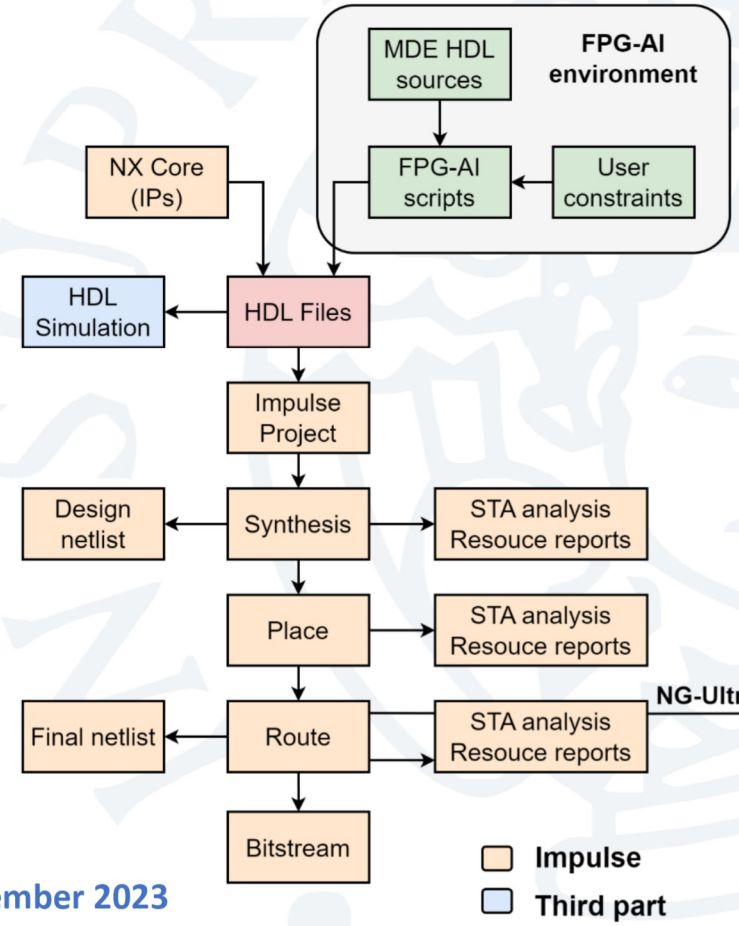
- Selected Convolutional Neural Network: **LeNet-5**
- Classification class: digits recognition
- Model topology: 2 Convolutional layers + 3 Fully Connected layers
- Total parameters: **44426 (~1.36 Mb)**



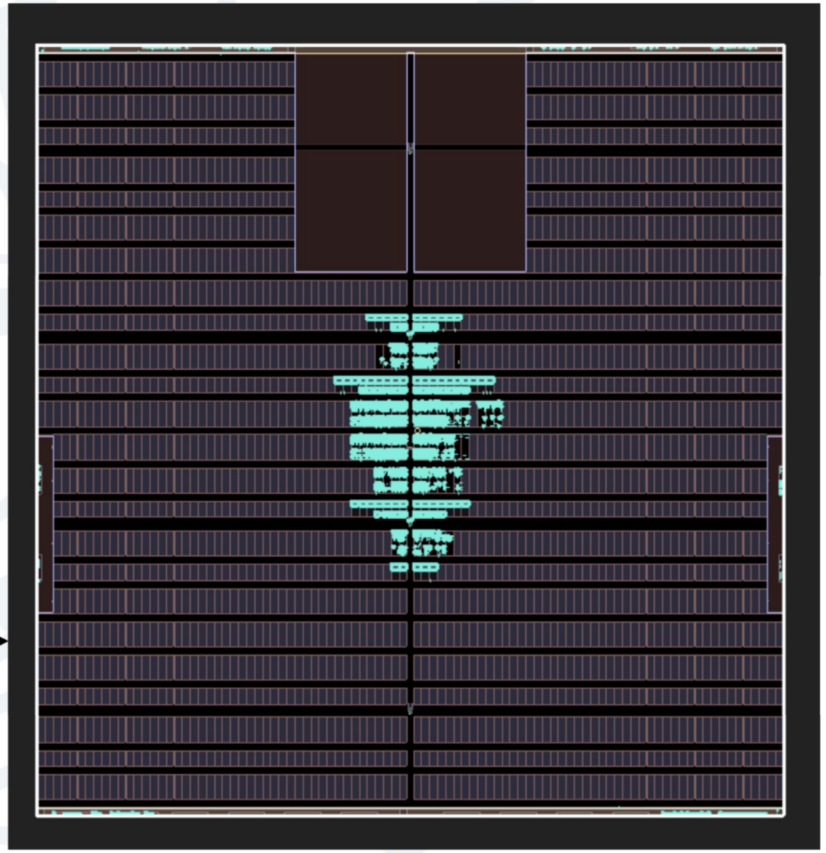
# Design Flow – FPG-AI



# Design Flow – NX Impulse




NG-Ultra Footrpint



# 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

64.25 Kb

0.18 Mb

| Xilinx Z-7045 | Design Step | Slice LUTs | Slice Registers | LUT as Memory | Block RAM | DSPs |
|---|---|---|---|---|---|---|
| | 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%) |

| NanoXplore NG-ULTRA | Design Step | 4-LUT | DFF | Register file block | DSP | Memory block |
|---|---|---|---|---|---|---|
| | 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%) |

72.56 Kb

0.703 Mb

# 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 | |
|---|---|---|---|---|---|
| | | | | Routing delay [ns] | Data delay [ns] |
| | 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