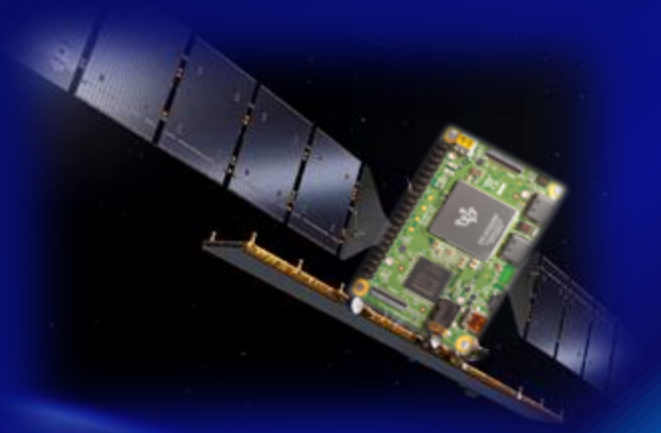
[ESA EDHPC, France 4/10/2023]

Performance and Radiation Testing of the Coral TPU Co-Processor for AI Onboard Satellites



George Lentaris, (UNIWA & NTUA, GR)
V. Leon, C. Sakos, D. Soudris (NTUA, GR)
A. Tavoularis, A. Costantino, C. Boatella (ESTEC, NL)

*acknowledgment: M. Bernou (OHB-Hellas, GR)

Informatics & Computer Engineering dpt.,
University of West Attica (UNIWA), Greece
Microlab, School of Electrical & Computer Engineering,
National Technical University of Athens (NTUA), Greece
European Space Research and Technology Centre,
European Space Agency (ESA), The Netherlands

work partially supported by ESA OSIP pr. "CAIRS21" (4000135491/21/NL/GLC/ov).)

Contents

INTRODUCTION

AI/ML in space (1/2)

- great success on Earth but limited penetration in space flight

- **AI market:** > $100B today
- **satellites:** > 100 new/year

where is AI ?

AI/ML in space (1/2)

- **great success on Earth but limited penetration in space flight**
- **increased computational requirements**
  - e.g., 1 order of magnitude vs CPU, 2 orders vs traditional rad-hard CPU
- **increased programming requirements**
  - SW complexity, libraries, frameworks. ++ability to adapt quickly to proliferation of algorithms
  - …also qualification/reliability aspects (not in this talk)

- **AI market:** > $100B today
- **satellites:** > 100 new/year

where is AI ?

AI/ML in space (1/2)

- **great success on Earth but limited penetration in space flight**
- **increased computational requirements**
  - e.g., 1 order of magnitude vs CPU, 2 orders vs traditional rad-hard CPU
- **increased programming requirements**
  - SW complexity, libraries, frameworks. ++ability to adapt quickly to proliferation of algorithms
  - …also qualification/reliability aspects (not in this talk)
- **solution: COTS HW+SW in mixed-criticality avionics architectures**

- **AI market:** > $100B today
- **satellites:** > 100 new/year

where is AI ?

AI/ML in space (2/2)

- **many potential use cases, especially in less critical tasks**
  - cloud detection (Earth Observation)
    - avoid downloading/storing useless pixels

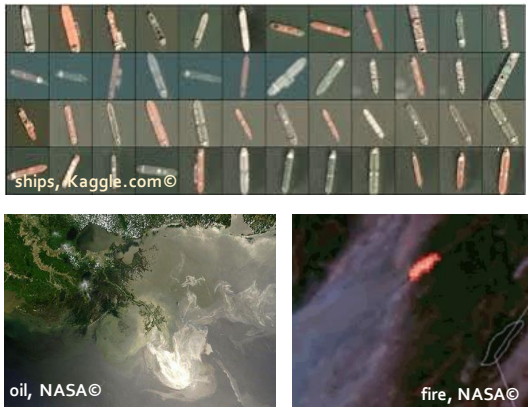
AI/ML in space (2/2)

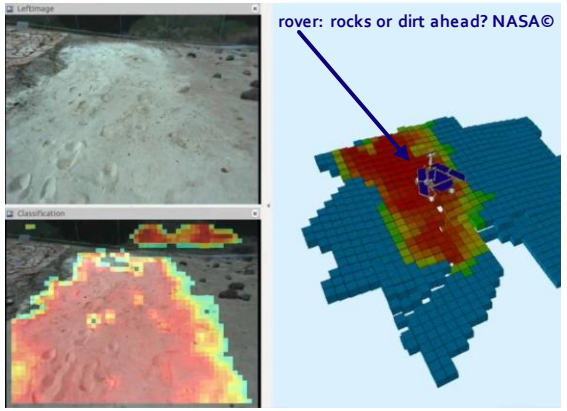
- **many potential use cases, especially in less critical tasks**
  - cloud detection (Earth Observation)
    - avoid downloading/storing useless pixels
  - object detection (Earth Observation)
    - e.g., find pirate ships/fisheries, oil spills, dangerous icebergs, forest fires, …

G. Lentaris
UNIWA & NTUA,
EDHPC 2023

AI/ML in space (2/2)

- **many potential use cases, especially in less critical tasks**
  - cloud detection (Earth Observation)
    - avoid downloading/storing useless pixels
  - object detection (Earth Observation)
    - e.g., find pirate ships/fisheries, oil spills, dangerous icebergs, forest fires, …
  - terrain classification (EO, explorers), e.g., during autonomous Mars rover navigation, …

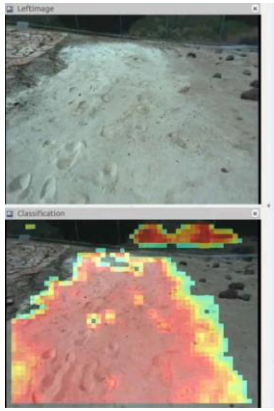


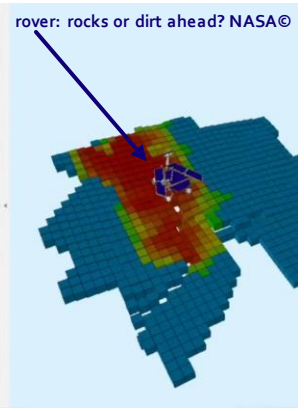


G. Lentaris
UNIWA & NTUA,
EDHPC 2023

AI/ML in space (2/2)

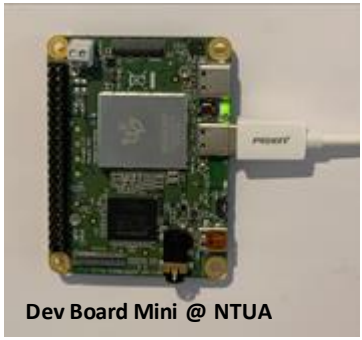
- **many potential use cases, especially in less critical tasks**
  - cloud detection (Earth Observation)
    - avoid downloading/storing useless pixels
  - object detection (Earth Observation)
    - e.g., find pirate ships/fisheries, oil spills, dangerous icebergs, forest fires, …
  - terrain classification (EO, explorers), e.g., during autonomous Mars rover navigation, …
  - anomaly detection, e.g., automatic satellite housekeeping (monitor vital signals), …
  - pose estimation of satellites, e.g., for non-cooperative in-orbit servicing (docking)
  - image super-resolution via GAN, scene change detection, AI to configure telecom,…



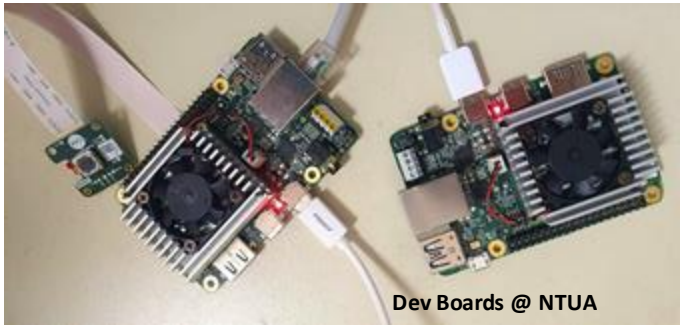


Candidate chip: Coral TPU (Google)

- **co-processor for AI tasks on Earth (…IoT, low-power,…)**
  - advertised as complete HW+SW toolkit



Dev Board Mini @ NTUA

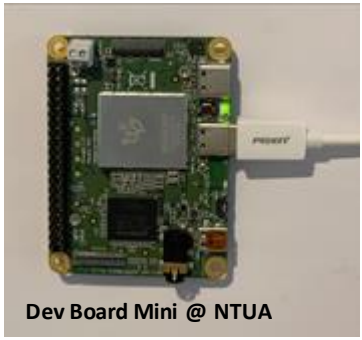


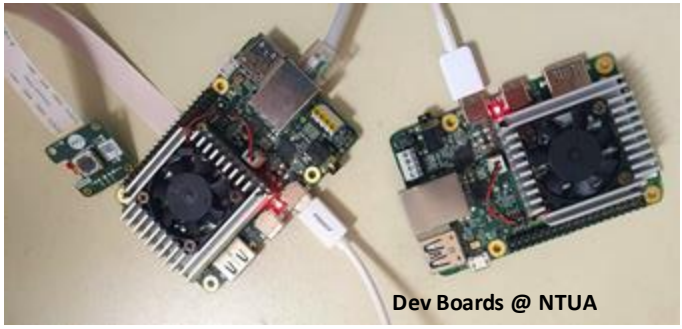
Dev Boards @ NTUA

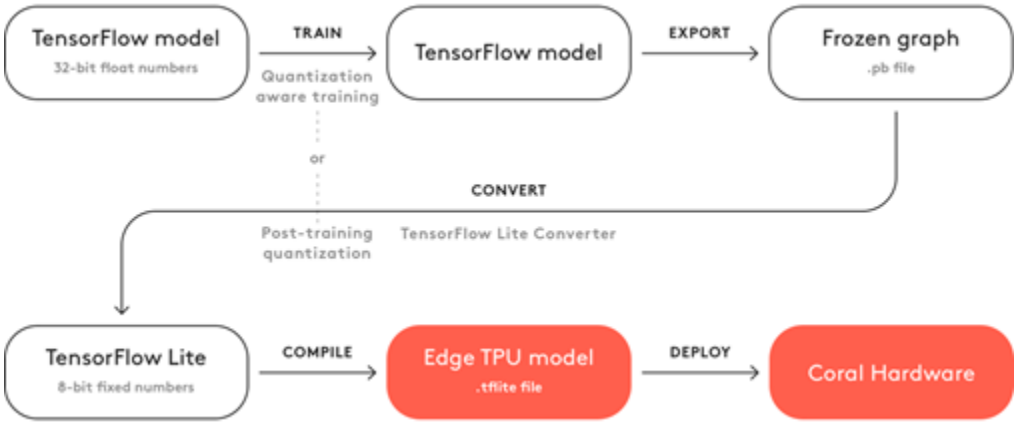
G. Lentaris
UNIWA & NTUA,
EDHPC 2023

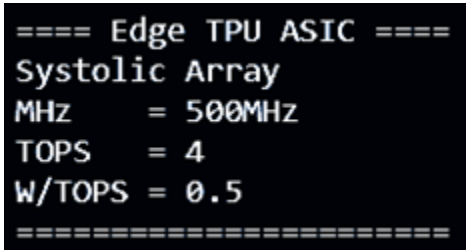
Candidate chip: Coral TPU (Google)

- **co-processor for AI tasks on Earth (…IoT, low-power,…)**
  - advertised as complete HW+SW toolkit
  - SW = framework, from Python-TF to binary
    - almost press-button ⇒ huge pool of potential users
  - HW= 64x64 systolic array of MULT-ADD
    - chip placed next to CPU, e.g., ARM A53 w/ USB2.0


Dev Board Mini @ NTUA


Dev Boards @ NTUA

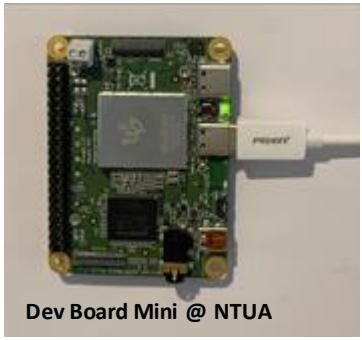


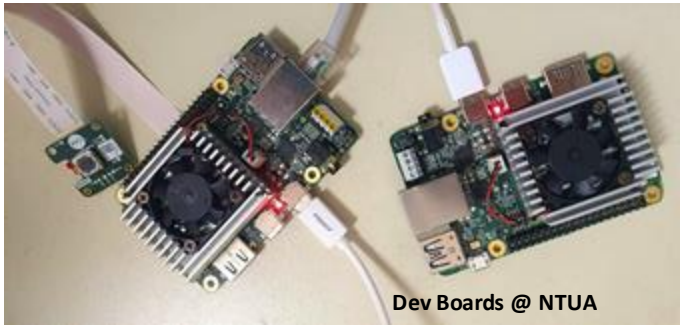


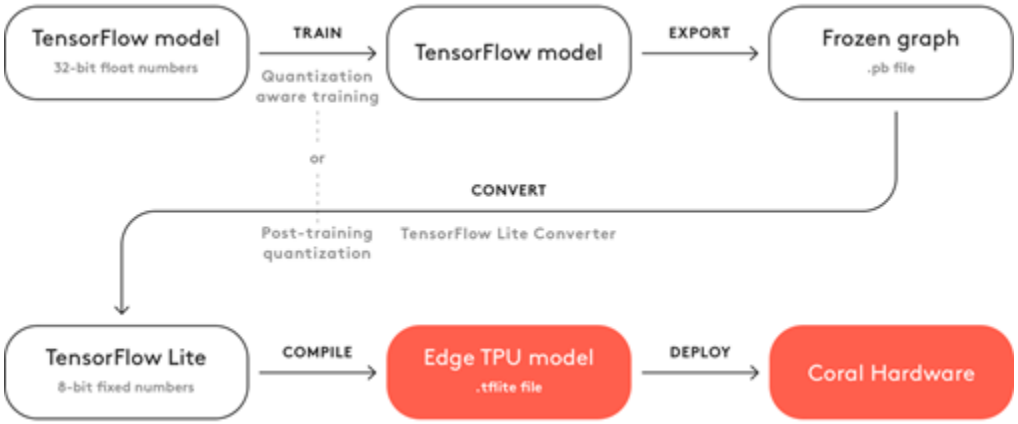
G. Lentaris
UNIWA & NTUA,
EDHPC 2023

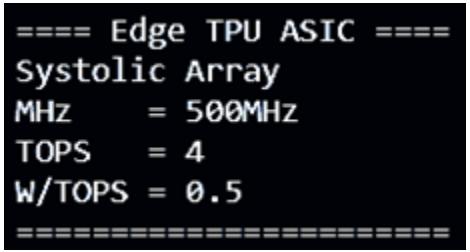
Candidate chip: Coral TPU (Google)

- **co-processor for AI tasks on Earth (…IoT, low-power,…)**
  - advertised as complete HW+SW toolkit
  - SW = framework, from Python-TF to binary
    - almost press-button ⇒ huge pool of potential users
  - HW= 64x64 systolic array of MULT-ADD
    - chip placed next to CPU, e.g., ARM A53 w/ USB2.0
- **is it good for AI in space?**
  - performance? programmability? *test…*
  - radiation tolerance? mitigations? *test…*


Dev Board Mini @ NTUA


Dev Boards @ NTUA

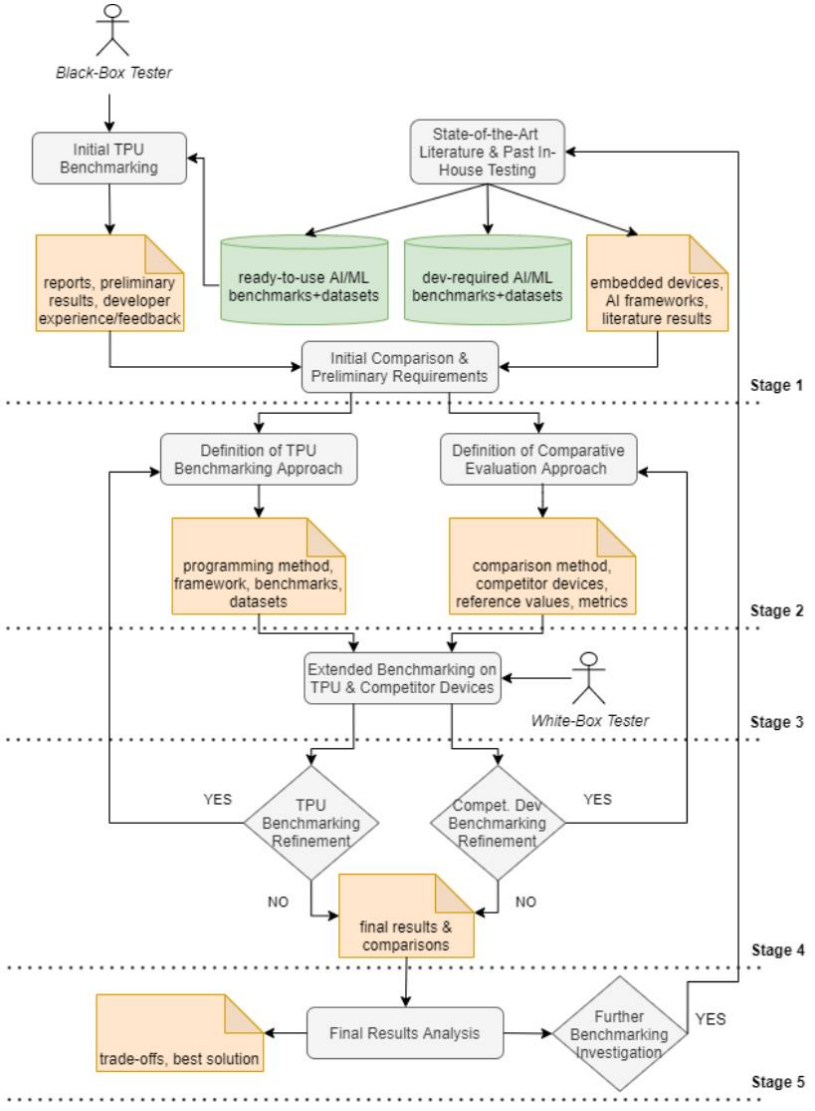




METHODOLOGY

Performance-Productivity

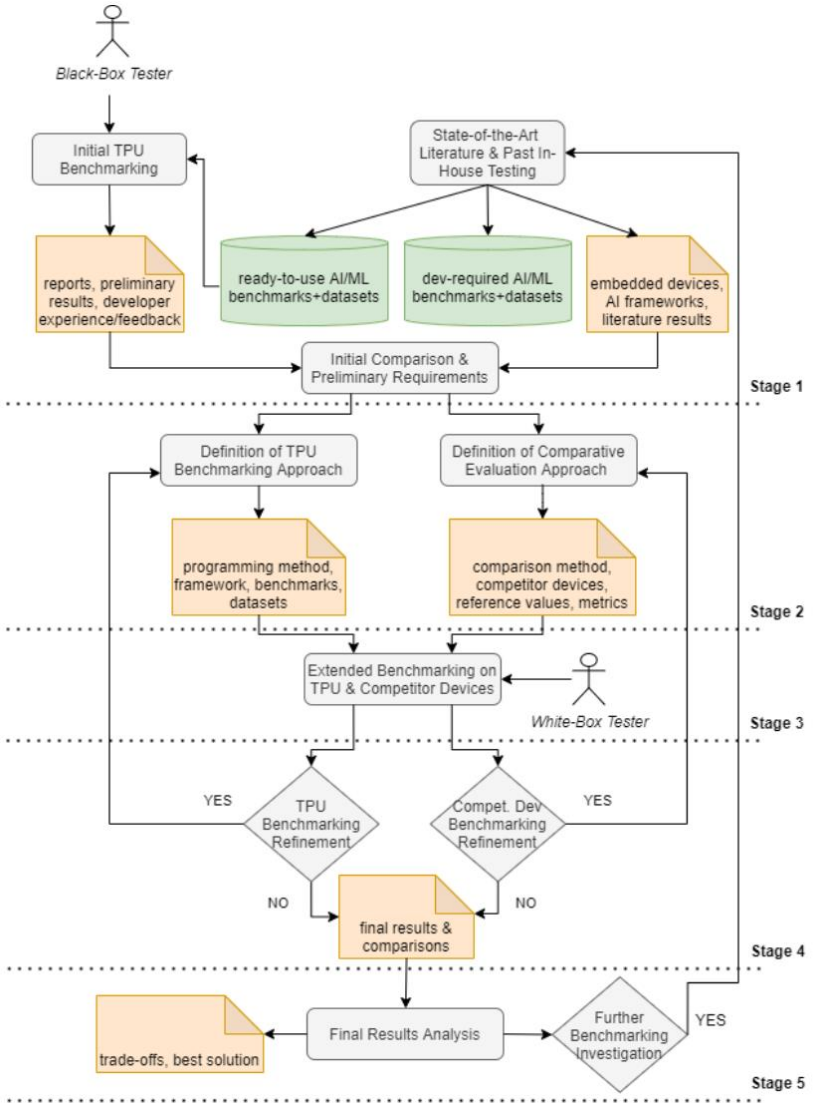
- **survey of literature/public sources**
- **extensive hands-on benchmarking**
  - AI networks (high-level)
  - TF operations (low-level)
- **comparisons to competitive devices**
  - FPGA (Zynq)
  - VPU (Myriad2/X)
  - GPU (Jetson Nano)
  - CPU (ARM A53)



G. Lentaris
UNIWA & NTUA,
EDHPC 2023

Performance-Productivity

- **survey of literature/public sources**
- **extensive hands-on benchmarking**
  - AI networks (high-level)
  - TF operations (low-level)
- **comparisons to competitive devices**
  - FPGA (Zynq)
  - VPU (Myriad2/X)
  - GPU (Jetson Nano)
  - CPU (ARM A53)
- **combine all into 1 methodology**
  - involve multiple users/developers
  - move gradually from generic to specific
    - from black- to white-box testing
    - ubiquitous AI to custom AI (for space)
    - high-level to detailed comparison
  - output = trade-off analysis, pros-cons



Radiation Testing

- **TID and SEE**
  - targeting confident results with limited budget (not full compliance to standards)
  - test parameters/plan based on experience and ESCC guidelines
    - focus on LEO environment
  - execute AI/benchmarks during irradiation (CNN, mult,...)
    - diversify functions, localize errors, assess criticality

Algorithm 1 Radiation Test SW (Benchmarking & Sampling)

```
Initialize: connect to all DUTs, load code & golden data to RAM
   TakeSample: log idle I-V, do BENCHMARK(), log active I-V

procedure BENCHMARK(PU)          // PU={ARM, TPU}
   for 1 to N do                 // arbitrary num. of iterations
      for 1 to 3 do              // TMR ("sparse" temporal)
         for f ← 1 to 3 do       // f={MULT, MATM, CLAS}
            for 1 to 3 do        // TMR ("dense" temporal)
               execute f-th function on PU, store output
            end for
            print f's Execution Time
            verify on ARM all f's outputs
            if (errors) : log input+output data, send to host PC
         end for
      end for
      compare on ARM the sparse temporal outputs of each f
      if (errors) : log input+output data, send to host PC
      for 1 to 3 do              // TMR ("dense" temporal)
         execute DETE function on PU, store output
      end for
      print DETE's Execution Time
      verify on ARM all DETE's outputs
      if (errors) : log input+output data, send to host PC
   end for
   send golden data to host PC for data integrity check
end procedure
```

Radiation Testing

- **TID and SEE**
  - targeting confident results with limited budget (not full compliance to standards)
  - test parameters/plan based on experience and ESCC guidelines
    - focus on LEO environment
  - execute AI/benchmarks during irradiation (CNN, mult,...)
    - diversify functions, localize errors, assess criticality
  - DUT= TPU on DevBoard, 5+1 COTS boards
    - focus on digital TPU chip (not PCB)
    - regular monitoring via laptop+PSU

Algorithm 1 Radiation Test SW (Benchmarking & Sampling)

Initialize: connect to all DUTs, load code & golden data to RAM
TakeSample: log idle I-V, do *BENCHMARK()*, log active I-V

procedure BENCHMARK(*PU*) // *PU*={ARM, TPU}
  for 1 to N **do** // arbitrary num. of iterations
    for 1 to 3 **do** // TMR ("sparse" temporal)
      for $f \leftarrow$ 1 to 3 **do** // f={*MULT, MATM, CLAS*}
        for 1 to 3 **do** // TMR ("dense" temporal)
          execute f-th function on *PU*, store output
        end for
        print f's Execution Time
        verify on ARM all f's outputs
        if (errors) : log input+output data, send to host PC
      end for
    end for
    compare on ARM the sparse temporal outputs of each f
    if (errors) : log input+output data, send to host PC
    for 1 to 3 **do** // TMR ("dense" temporal)
      execute *DETE* function on *PU*, store output
    end for
    print *DETE*'s Execution Time
    verify on ARM all *DETE*'s outputs
    if (errors) : log input+output data, send to host PC
  end for
  send golden data to host PC for data integrity check
end procedure

G. Lentaris
UNIWA & NTUA,
EDHPC 2023

Radiation Testing

- **TID and SEE**
  - targeting confident results with limited budget (not full compliance to standards)
  - test parameters/plan based on experience and ESCC guidelines
    - focus on LEO environment
  - execute AI/benchmarks during irradiation (CNN, mult,...)
    - diversify functions, localize errors, assess criticality
  - DUT= TPU on DevBoard, 5+1 COTS boards
    - focus on digital TPU chip (not PCB)
    - regular monitoring via laptop+PSU

- **TID: Co-60 facility at ESTEC (NL)**
  - 340 rad(Si)/hour, 1 week ⇒ 50Krad TID
  - diverse shielding, no power cycling
- **SEE: protons, PIF facility at PSI (CH)**
  - energy 16-200 MeV, flux 1–62·10^7 p/cm²/s, fluence 10^{12} p/cm²
  - beam 1x1cm² (center @ TPU package), 10's of power cycles

Algorithm 1 Radiation Test SW (Benchmarking & Sampling)

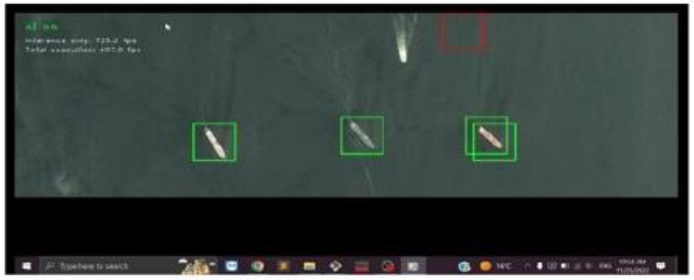
Initialize: connect to all DUTs, load code & golden data to RAM
TakeSample: log idle *I-V*, do *BENCHMARK()*, log active *I-V*

procedure BENCHMARK(*PU*) // *PU*={ARM, TPU}
  for 1 to *N* **do** // arbitrary num. of iterations
    for 1 to 3 **do** // TMR ("sparse" temporal)
      for *f* ← 1 to 3 **do** // *f*={*MULT, MATM, CLAS*}
        for 1 to 3 **do** // TMR ("dense" temporal)
          execute *f*-th function on *PU*, store output
        end for
        print *f*'s Execution Time
        verify on ARM all *f*'s outputs
        if (errors) : log input+output data, send to host PC
      end for
    end for
    compare on ARM the sparse temporal outputs of each *f*
    if (errors) : log input+output data, send to host PC
    for 1 to 3 **do** // TMR ("dense" temporal)
      execute *DETE* function on *PU*, store output
    end for
    print *DETE*'s Execution Time
    verify on ARM all *DETE*'s outputs
    if (errors) : log input+output data, send to host PC
  end for
  send golden data to host PC for data integrity check
end procedure

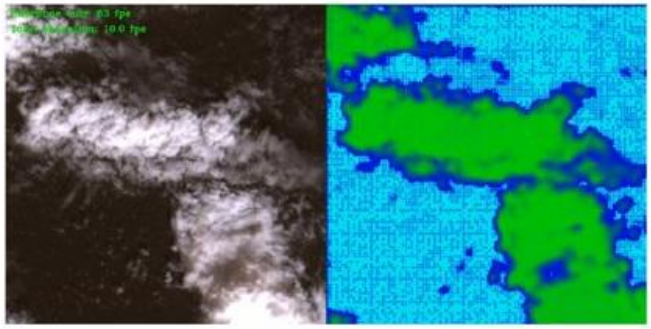
RESULTS (preliminary)

Productivity-Support

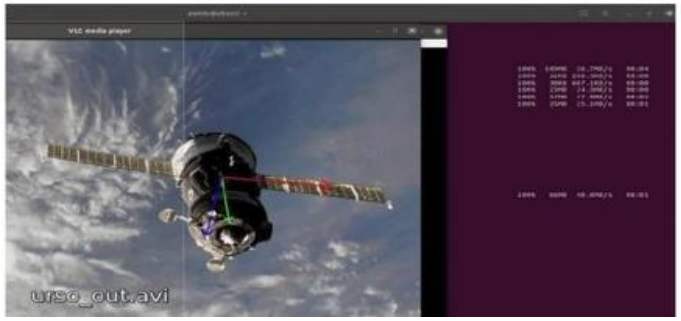
- **TPU = easiest acceleration of AI/ML**
  - practically, do TensorFlow + quantization
  - we developed various demos, sufficient accuracy



TPU ship detection



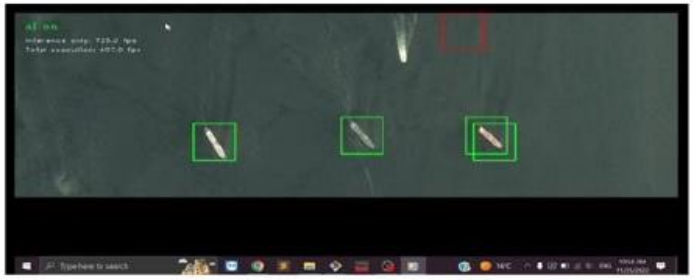
TPU cloud segmentation



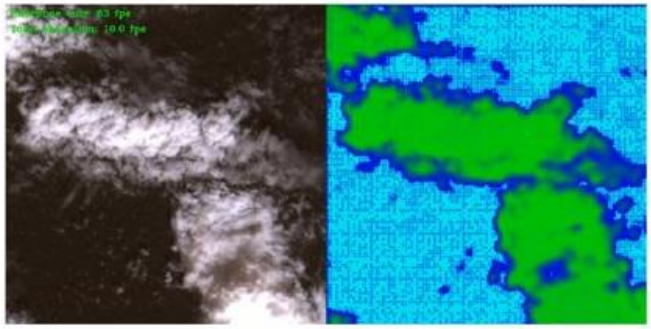
TPU pose estimation

Productivity-Support

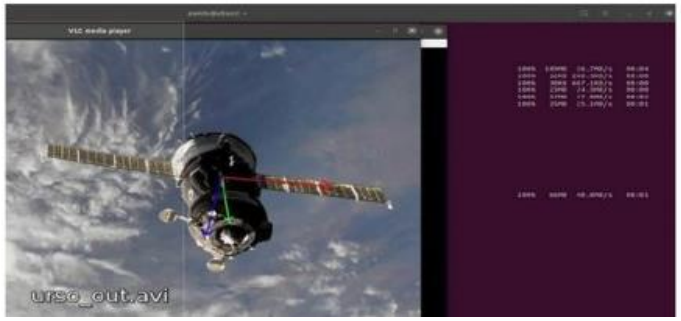
- **TPU = easiest acceleration of AI/ML**
  - practically, do TensorFlow + quantization
  - we developed various demos, sufficient accuracy
- **but with considerable limitations**
  - supports only certain layers/ops (TFLite)
    - e.g., no acceleration of classical DSP
    - e.g., not good for new/weird AI
  - has limited 8-bit accuracy
    - e.g., not good for LSTMs
  - accelerates only inference
    - e.g., no training (federated, transfer learn?)
  - no low-level coding
    - e.g., little opportunities for error mitigation



TPU ship detection



TPU cloud segmentation



TPU pose estimation

Performance

- **TPU Coral = most efficient chip for mid-sized CNN & MLP**
  - i.e., for majority of embedded AI that need acceleration
  - but, must keep model size <7.6MB (else uses off-chip memory), latency overhead 0.3msec
  - for bigger nets results vary, e.g., TPU similar or 2x worse vs GPU/VPU (Incept.v4, ResNet-50)
- **power**
  - 1.5W for chip (5W for board), e.g., like VPU but with 10x performance/watt
- **speed**
  - 2x-100x vs ARM-Axx
  - 10x vs small GPU/VPU
  - 2x vs mid-sized FPGA

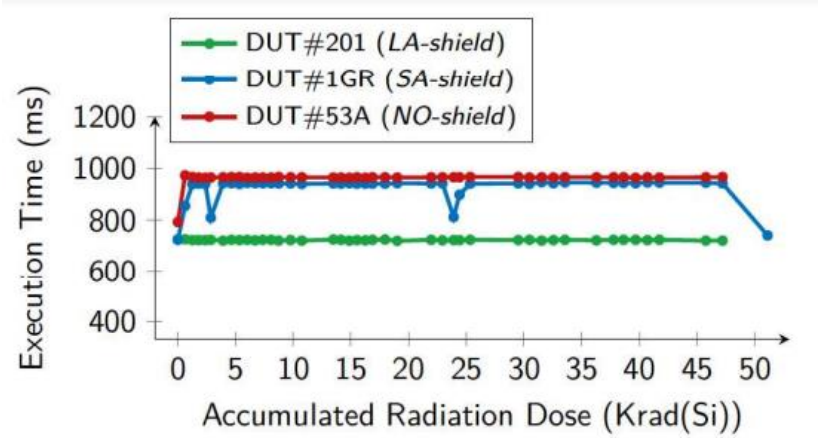
Benchmarks
Mult-Add, LSTM, FullyConnected MLP, MNIST, CIFAR, SHIPNET, Inception, Mobilenet, Mobilenet SSD, Pose ResNet 50, EfficientDet Lite3, DeepLab, Yolo, ...

TPU speedup workload	vs ARMa53@1.5G	vs MyriadX (USB+PC)	vs JetNano GPU	vs Zynq FPGA
MLP (low)	0.1–0.5x	7x	4x	1x
RNN (mid)	2–3x		X	1x
CNN (low)	2–5x	1x	5x	0.2x
MLP (mid)	5–20x	5-10x	10–20x	1-4x
CNN (mid)	20–60x	10x	8x	2x
CNN (high)	5–25x	1.5-0.5x	1–0.25x	1–0.5x

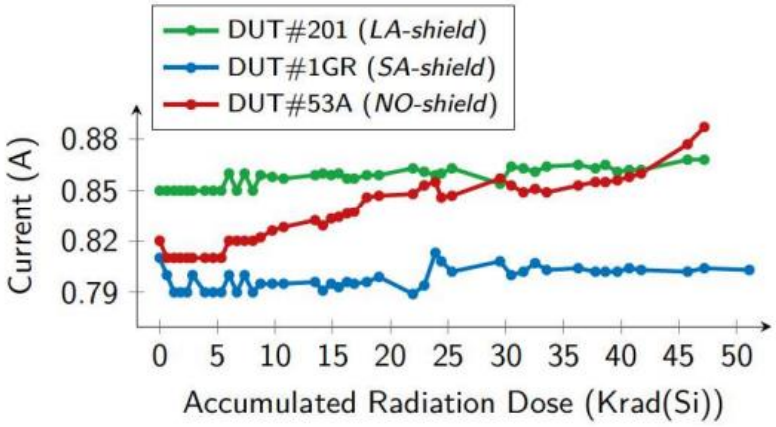
"mid" workload = 10-1000msec for CNN, 1-10msec for MLP (on ARMa53@1.5GHz)

Radiation, TID test

- **~50 Krad(Si) → digital TPU still OK**
  - all 3 chips operated correctly throughout the test (139+ hours, 40+ runs per DUT)
    - zero errors, constant performance
  - small current increase on unshielded PCB
    - 9%, attributed to analogues of PCB



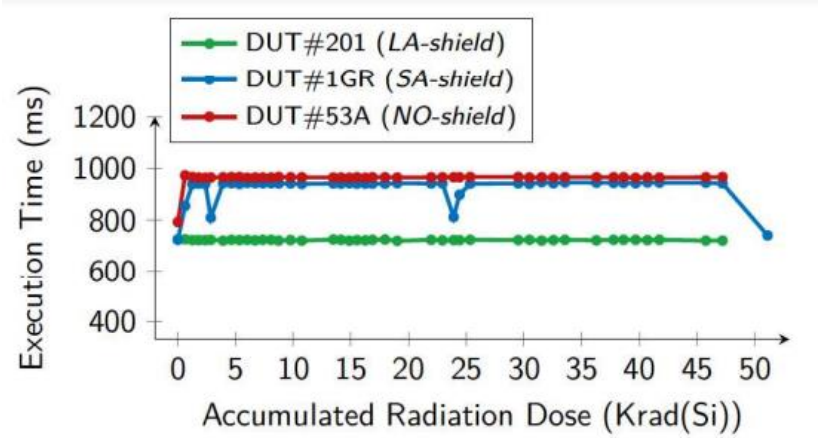
(a) Avg. execution time on TPU (object detection benchmark)



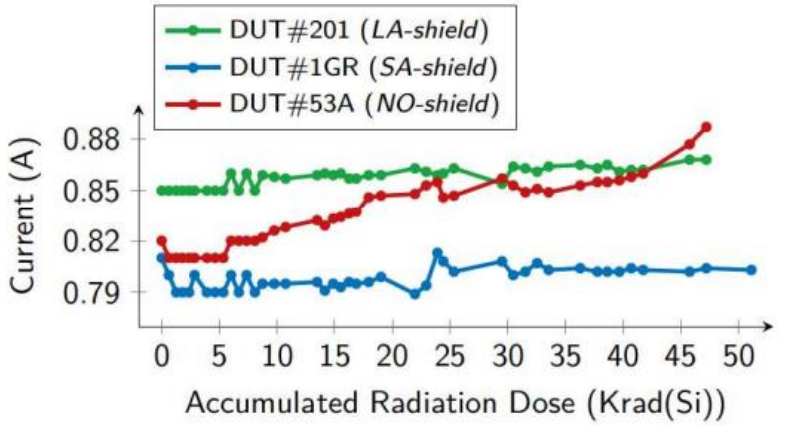
(b) Current during processing (for entire PCB, supply=5V)

Radiation, TID test

- **~50 Krad(Si) → digital TPU still OK**
  - all 3 chips operated correctly throughout the test (139+ hours, 40+ runs per DUT)
    - zero errors, constant performance
  - small current increase on unshielded PCB
    - 9%, attributed to analogues of PCB
  - two poorly-shielded PCBs became inoperable after first power cycle, at 47Krad
    - well-shielded went 51Krad (2.7Krad/h)
  - digital parts didn't reach breaking point!
    - TPU, ARM-A53, DDR, eMMC
    - failures attributed to analogue + USB
  - annealing + aging test (7-day @ 75^{0}C)
    - DUTs still good
    - failed USB restored
    - 30-50% less throughput (DVFS?)



(a) Avg. execution time on TPU (object detection benchmark)

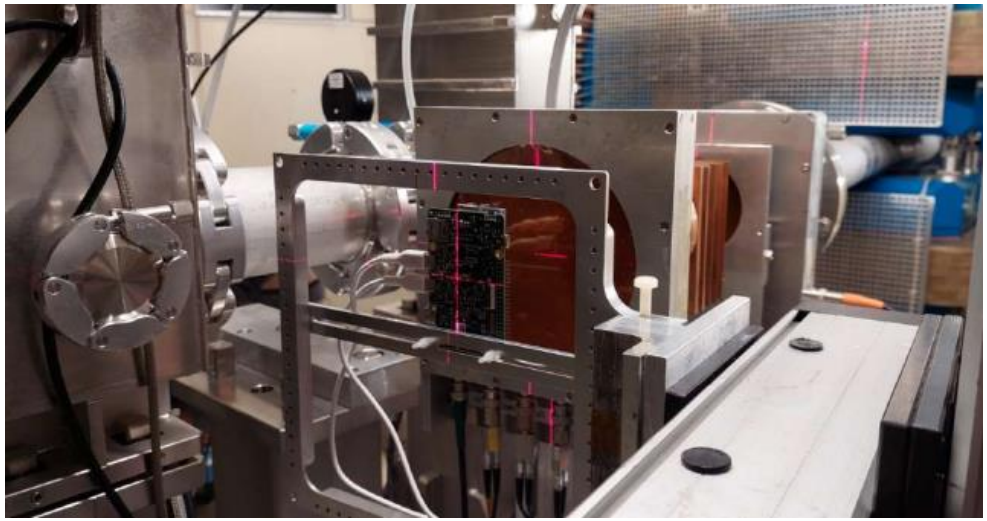


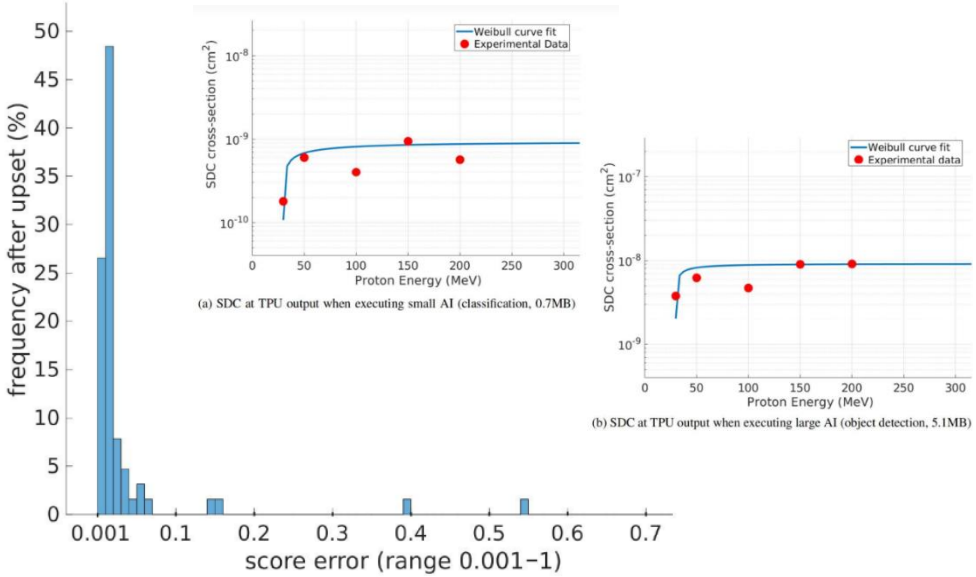
(b) Current during processing (for entire PCB, supply=5V)

G. Lentaris
UNIWA & NTUA,
EDHPC 2023

Radiation, SEE test

- **TPU no hard-errors / latch-ups**
- **SEFIs due to DevBoard / CPU**
  - always corrected via reboot
- **SEU corrected by TPU reprog.**
  - occur mainly in onchip mem/cache of TPU, not in its systolic array
- **increased SDC cross-section**
  - especially for bigger AI models
  - 10^{-8} to 10^{-9} cm² (or 10^{-10} at 30MeV)
- **decreased error magnitude**
  - vast majority of upsets are negligible for the AI (90-97%)
- **TID result partially confirmed**
  - even with 10's of power cycles
  - DUT4 lost connection at 30Krad
    - attributed to USB, restored after months room temp (unbiased)





CONCLUSIONS

Conclusion

- TPU most efficient and user-friendly ASIP for mid-sized CNN & MLP
- but has limitations w.r.t. supported SW/ops (use-cases)
- promising TID results
- increased SEE sensitivity to protons
  - but decreased importance for AI (!)
  - no hard-errors (heavy-ions test still needed)
- e.g., looks good for missions up to 800Km (OMERE simulations)

Thank you!
Q&A

George Lentaris
glentaris@microlab.ntua.gr
glentaris@uniwa.gr