

Real-Time Onboard Data Accumulation and Pre-Processing for the Photospheric Magnetic field Imager (PMI) on ESA's Vigil mission

Deepa Muraleedharan

PhD Student (IMPRS)

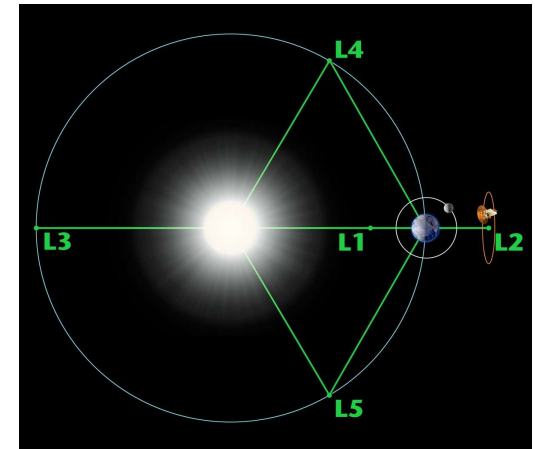
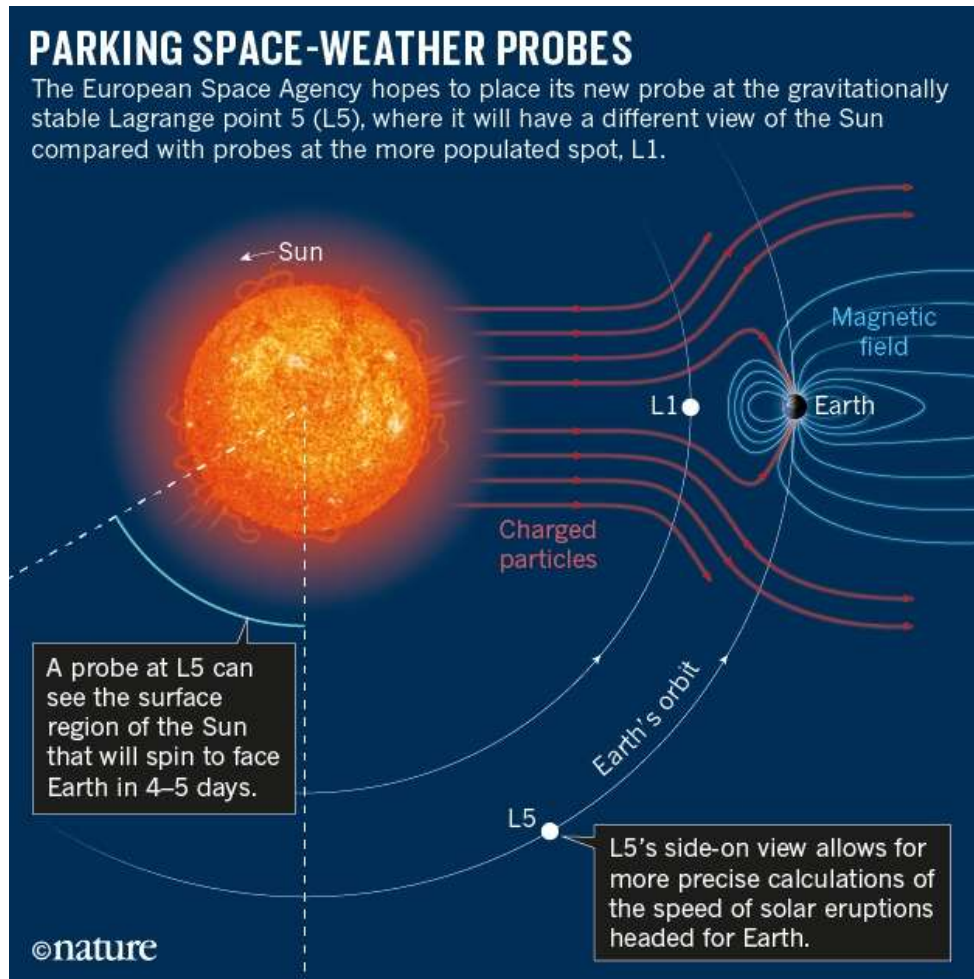
Max Planck Institute for Solar System Research

Göttingen, Germany

EDHPC 2025

15-10-2025

THE VIGIL MISSION



- Vigil is part of ESA's Space Safety Programme
- Monitors Sun from Lagrange point L5



PHOTOSPHERIC MAGNETIC FIELD IMAGER (PMI)

PMI is a solar full-disk imaging magnetograph designed to continuously monitor the Sun's photospheric magnetic field and line-of-sight velocity fields.

Heritage

Builds on the heritage of SO/PHI (Polarimetric and Helioseismic Imager onboard ESA/NASA Solar Orbiter).

Upgrade

Unlike SO/PHI (limited by telemetry & pointing constraints), **PMI is optimized for continuous full-disk observations.**

Mission support

Supports **space weather forecasts** and **scientific studies of solar activity & evolution**

Real-time onboard data reduction is essential due to:

- **Telemetry limitations** at 1 Astronomical Unit in the L5 Lagrange point
- Need for **uninterrupted monitoring** of the solar photospheric vector magnetic field and line-of-sight velocity
- Requirement for **low-latency (~20 min)** data products with **~30 min cadence**



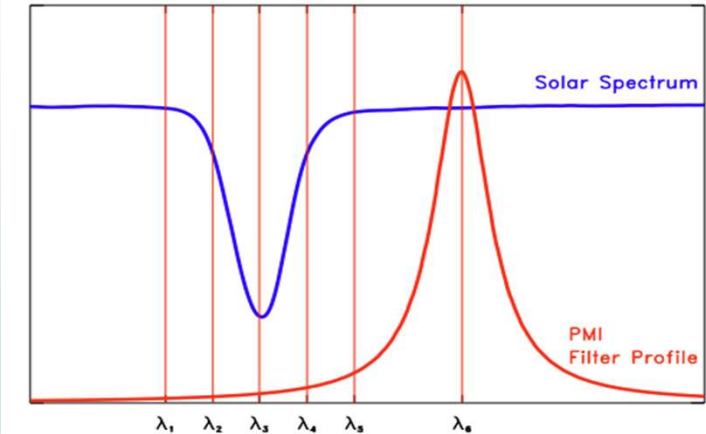
PMI – MEASUREMENT SCHEME

Magnetic field measurement: PMI uses the **Zeeman effect**; field direction appears as a wavelength-dependent polarization pattern.

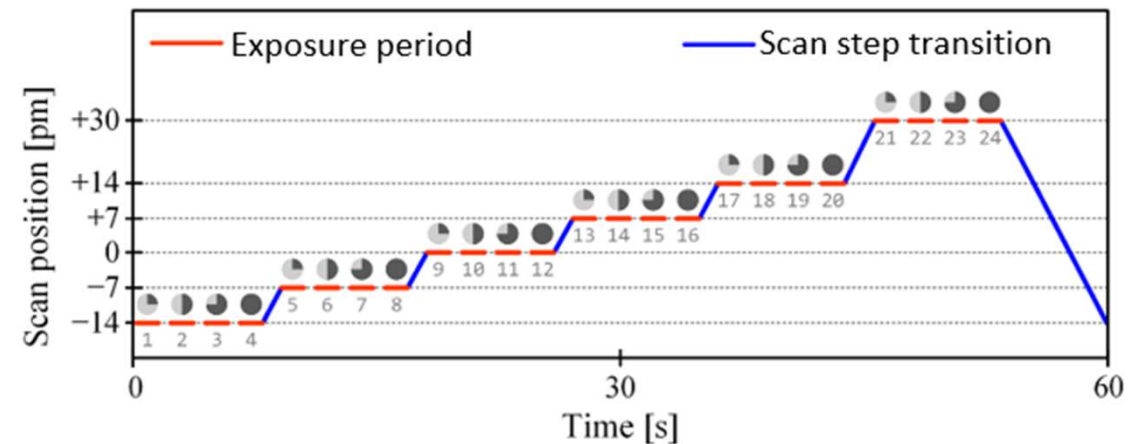
Spectral scanning: 6 wavelength positions across Fe I line at 617.3 nm.

Polarization states: 4 modulation states per wavelength → **24 raw images per dataset.**

Focal Plane Assembly (FPA): 2048×2048 pixels, 12-bit depth, 10 fps.



Up to 20 exposures per spectral–polarization combination are accumulated **to improve signal-to-noise ratio.**

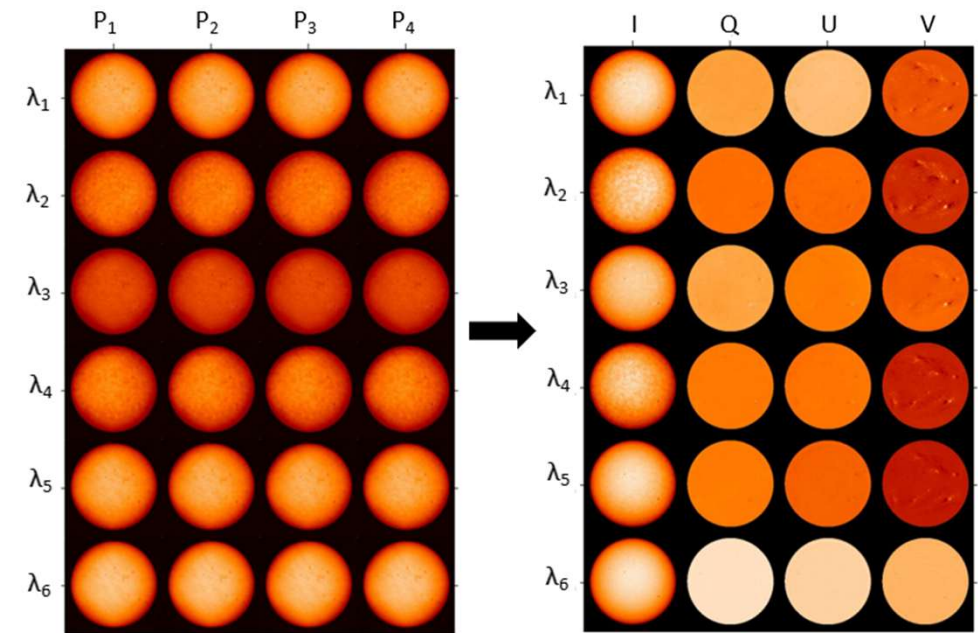


PMI – ONBOARD DATA PROCESSING



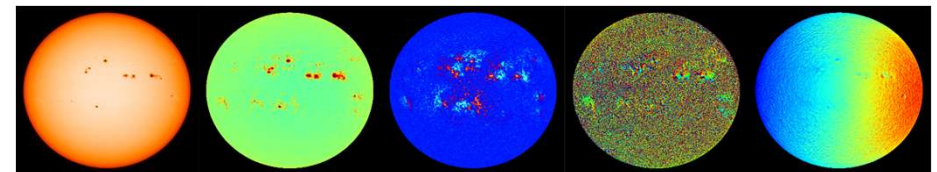
24 raw filtergrams → image corrections + polarimetric demodulation

Produces Stokes images (I, Q, U, V) at each wavelength position



Onboard RTE inversion retrieves **5 physical parameters**

- Vector magnetic field
- Line-of-sight plasma velocity





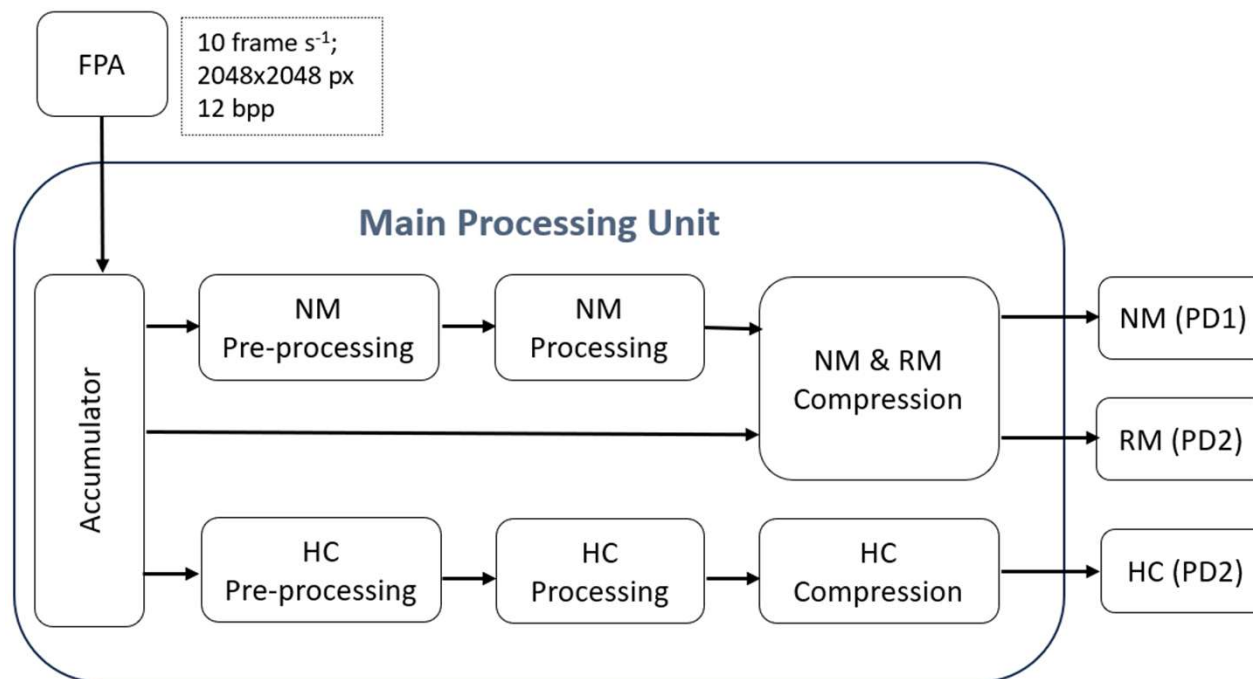
PMI – DATA PRODUCTS

- **Nominal (Priority 1)** - computed by **RTE Inversion**, **Cadence=30 min**, **Latency=20 min**
 - ✓ **Continuum Intensity** (I_c),
 - ✓ **Vector magnetic field** - (strength B , inclination γ and azimuth ϕ)
 - ✓ **Plasma Line-of-Sight velocity** (v_{LOS})
 - ✓ *For each dataset, 24.159 Gbit of raw data is converted onboard into the five parameter maps and finally compressed to ~6 bpp, corresponding to 0.125829 Gbit per product set, a **192x** reduction relative to the raw data.*
- **High Cadence (Priority 2)** – **Cadence = 1-2 min**
 - ✓ I_c , v_{LOS}
 - ✓ *Provides timely indicators of rapidly evolving magnetic features*
- **Raw (Priority 2)** – **Cadence = 30-120 min**
 - ✓ **Raw filtergrams**
 - ✓ Used for calibration, verification and algorithm refinement on the ground



MAIN PROCESSING UNIT (MPU) – NOMINAL MODE PROCESSING PIPELINE

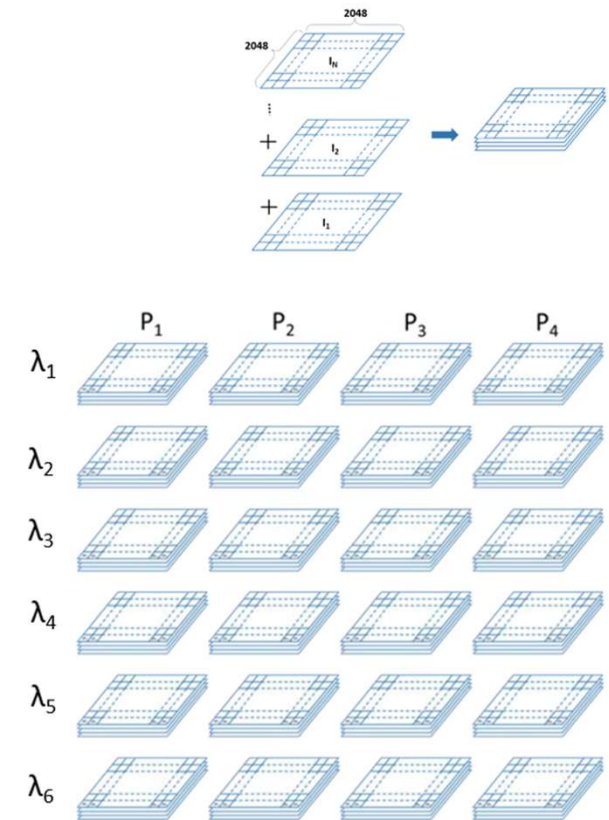
- Transforms raw detector readouts into **science-ready data products** for downlink.
- Implemented on a **radiation-tolerant AMD Kintex XQRKU060 FPGA**
- Each stage is realized as **custom FPGA Intellectual Property (IP) cores**





ACCUMULATOR IP CORE

- Raw frame acquisition from FPA (**2048 x 2048 pixels, 12 bpp, 10 fps** → that must be sustained **in real-time**)
- Frames streamed via **channel-link interface** to Accumulator IP core
- **Pixel re-ordering**: reconstructs layout from 2 multiplexed FPA channels
- **Up to 20 accumulations** per polarization state & wavelength position to improve the SNR
- **Sequencing of accumulation** across wavelength–polarization grid (λ , P), controlled by programmable parameters
- **Final result stored as 32-bit floating-point (IEEE-754) in memory**
- **Galactic Cosmic Ray (GCR) detection and correction** - each pixel compared to running mean, If the difference exceeds a programmable threshold, accumulated mean is added instead of the outlier → prevents propagation of artefacts



NOMINAL MODE PRE-PROCESSING IP CORE



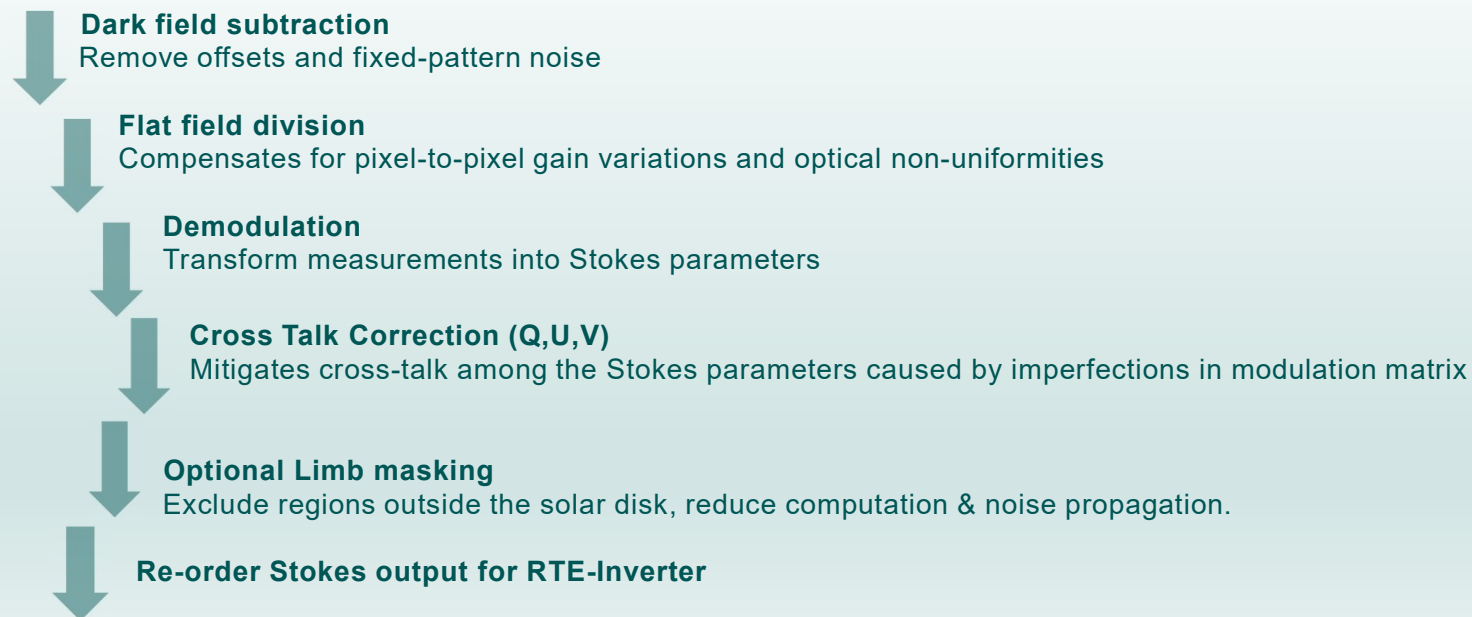
- **Computes :**

- ✓ **Stokes parameters** (I, Q, U, V) at 6 wavelengths
- ✓ **Continuum image** (I_c) - Stokes I at first or sixth wavelength (programmable)

- **Auxiliary inputs:**

- ✓ 1 dark field image (independent of p and λ)
- ✓ 6 -24 flat field images (independent of p)
- ✓ 4 x 4 demodulation matrix (16 elements, each 2K x 2K pixels)
- ✓ Scalar calibration coefficients for cross-talk correction

- **Pre-processing Functions**



All mathematical operations are carried out in 32-bit IEEE-754 single-precision floating-point format.

DESIGN CONSTRAINTS AND IMPLEMENTATION REQUIREMENTS



- **Throughput and timing:**

- ✓ Accumulator: 500 Mbps input in real time, 20 frames $\times 6 \lambda \times 4 P$ accumulation $\leq 60s$
- ✓ Preprocessing: $\leq 120 s$ (Nominal), $\leq 60 s$ (High-Cadence)

- **Data Representation:**

- ✓ Accumulator: 12-bit integer input $\rightarrow$ 32-bit IEEE-754 floating point
- ✓ Pre-processing: all operations in the 32-bit IEEE-754 floating point domain

- **Parallel & Modular Design:**

- ✓ IP cores operate in parallel
- ✓ Scalable resource allocation & deterministic scheduling
- ✓ Simplifies verification and IP reuse

- **Resource budget:**

- ✓ Accumulator: $\sim 1\%$ LUTs; (LUT - Look Up Tables)
- ✓ Pre-processing: $\sim 2.5\%$ LUTs
- ✓ Fit within FPGA LUT, BRAM, DSP limits

- **Deterministic Operation:**

- ✓ Strictly deterministic timing and data flow
- ✓ Starts processing when commanded, and signals completion with an interrupt.

- **Interface Requirements**

- ✓ Channel-link input $\rightarrow$ memory $\rightarrow$ Preprocessing $\rightarrow$ RTE-ready Stokes vectors $\rightarrow$ memory

- **System Level Scheduling**

- ✓ In NM 30 minute observation cycle, Accumulation: 0–1 min, Preprocessing: 2–3 min, RTE inversion: 4–18 min, Compression: ~ 19 min $\rightarrow$ data ready by ~ 20 min



VERIFICATION APPROACH

1

Simulation based Verification

- Verified VHDL IP cores using **Open Source VHDL Verification Methodology (OSVVM)** using **QuestaSim**
- Directed test cases + constrained-random stimuli
- Scoreboard comparison in an automated manner: DUT outputs vs. expected results
- Simulation limited to datasets up to 2048×8 pixels



2

Hardware-in-the-Loop Validation

- IP cores synthesized and implemented on XCKU040 FPGA
- System level tests on **AMD Kintex UltraScale KCU105 board**
- PCIe 4.0 interface streams 2048×2048 pixel datasets
- **MicroBlaze processor** manages commands & configuration
- Validates throughput and timing under realistic conditions

3

Software Model Comparison

- Same input datasets as in hardware validation used; all programmable parameters tested
- IP core outputs compared to floating-point reference model in Python
- Confirms functional consistency at every processing stage



RESULTS

- **Hardware-in-the-Loop + Python Reference:**
 - ✓ Confirms throughput and timing requirements

Accumulator IP Core performance

13.75 fps

processing rate, exceeding
the 10 fps requirement

27.92 s

meets <60 s per dataset
requirement

Processing of 2048×2048 px dataset
(16 accumulations $\times$ 6 λ $\times$ 4 P)

Nominal Mode Pre-processing IP core performance

15.63 s

Well within 120 s limit

- **Accuracy and timing**
 - ✓ Hardware outputs bit-accurate vs. software reference
 - ✓ Achieves timing closure at 100 MHz

- **Simulation Verification (OSVVM):**
 - ✓ Bit-accurate agreement with expected outputs
 - ✓ >99.9% code coverage $\rightarrow$ robust testbench & design

RESULTS



- **Resource Utilization:**

- ✓ Accumulator & Nominal Mode Pre-processing fit within FPGA resource budget (LUT, BRAM, DSP)

Resource Utilization of Accumulator IP Core in XQRKU060 FPGA

Resource	Utilization	Available	Utilization %
LUT	1415	331680	0.43
FF	2204	663360	0.33
BRAM	20.50	1080	1.90
DSP	4	2760	0.14

Resource Utilization of Pre-processing IP Core in XQRKU060 FPGA

Resource	Utilization	Available	Utilization %
LUT	8374	331680	2.52
FF	12983	663360	1.96
BRAM	68.50	1080	6.34
DSP	92	2760	3.33

CONCLUSION



- **Both Accumulator and Preprocessing IP cores:**
 - ✓ Achieve bit-accurate correctness
 - ✓ Meet throughput & timing requirements
 - ✓ Conform to the FPGA resource budget
- ***The presented solution is therefore validated as suitable for spaceborne deployment in the Vigil mission context.***



THANK YOU !!!