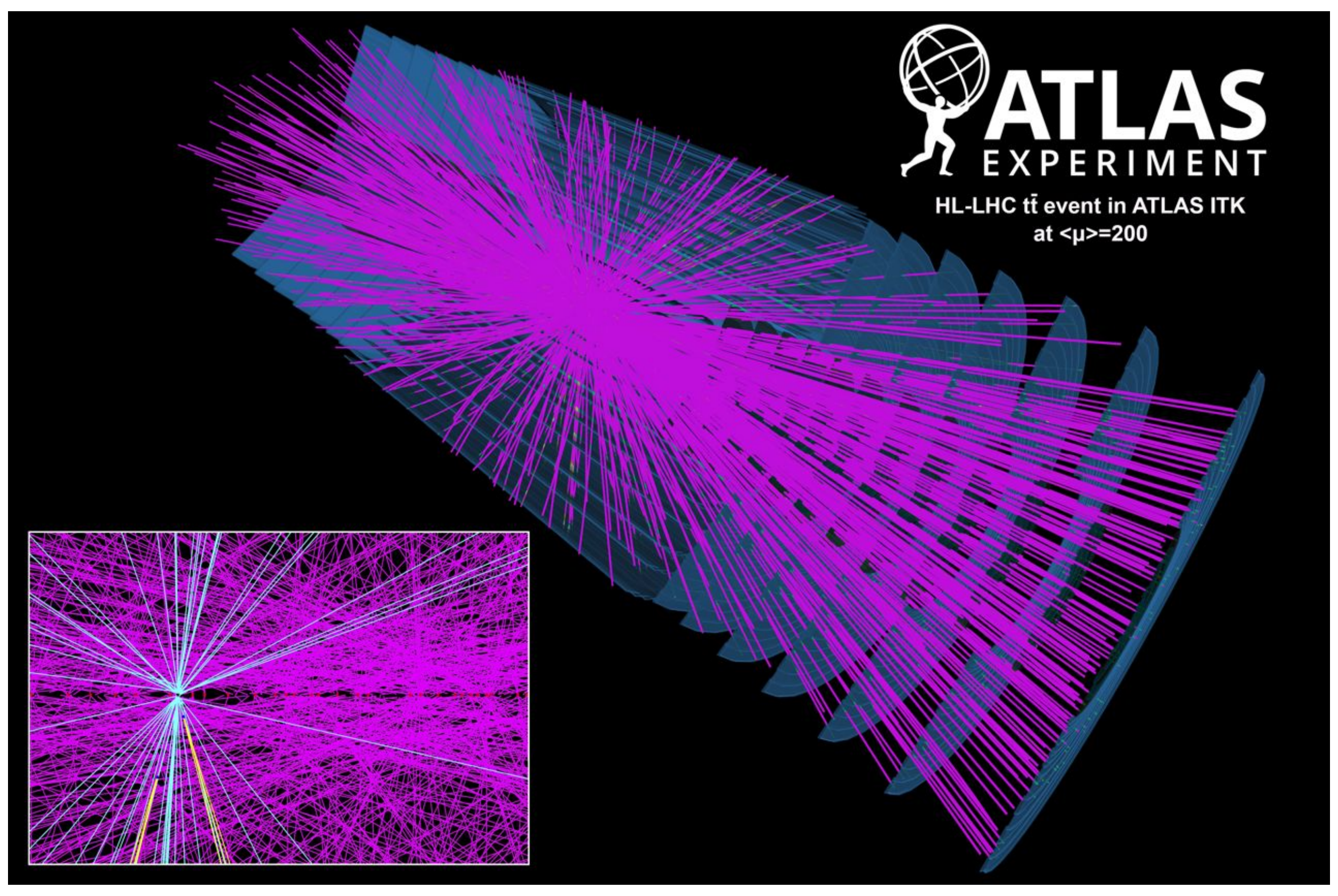
A FPGA Implementation of the Hough Transform Tracking Algorithm for the Phase II upgrade of ATLAS

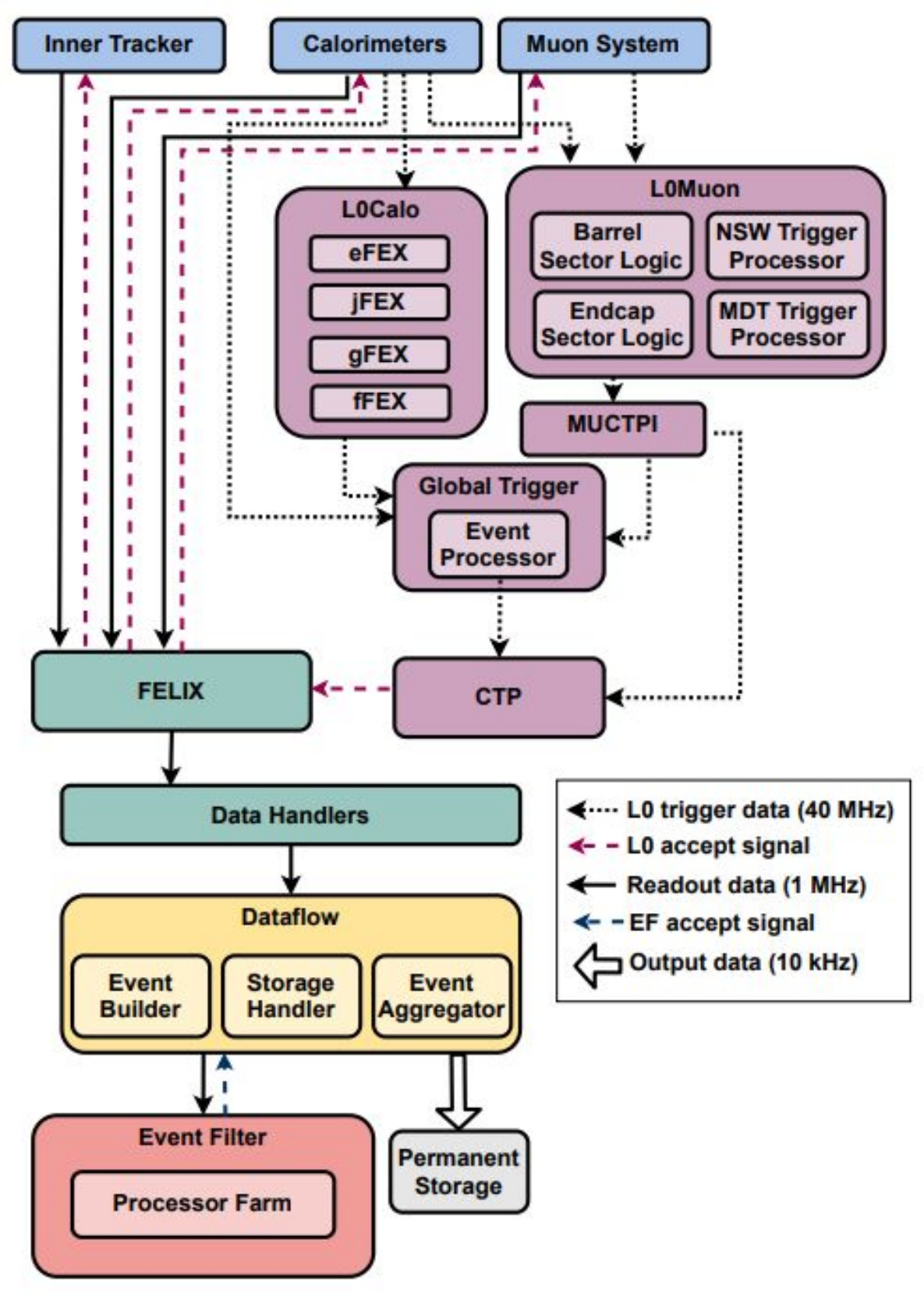
F.Alfonsi[1] on behalf of the ATLAS TDAQ Community
[1] Istituto Nazionale di Fisica Nucleare Bologna
fabrizio.alfonsi@bo.infn.it

BACKGROUND and MOTIVATION

Several machine and and detector upgrades are taking place in view of the LHC high luminosity running which is scheduled to start in 2029 after the Long Shutdown 3. The pileup for proton-proton collisions is expected to increase from the present 60 up to 140 or more. The peak luminosity is planned to reach 5 x 10^{34} cm^{-2} s^{-1}[1].



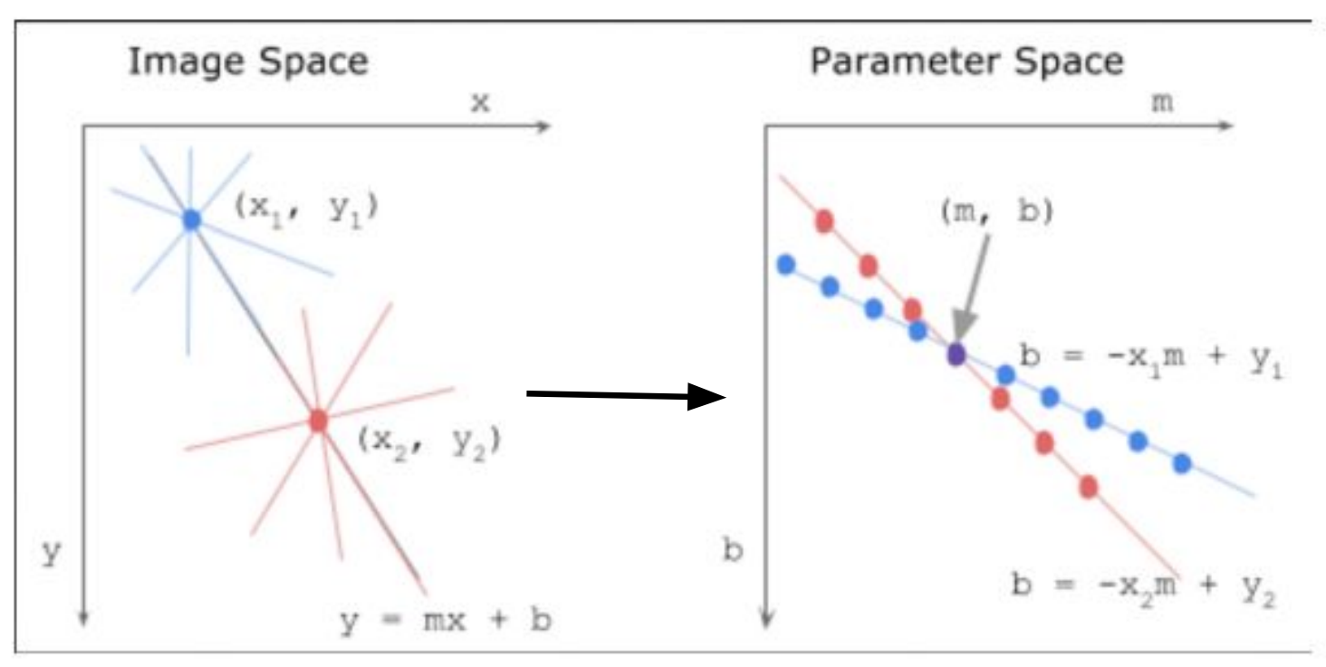
This new physics environment will need upgrades for the CERN experiments. The ATLAS experiment will upgrade its detector and consequently the Trigger and Data Acquisition system, to reach 10 kHz of output data stream with respect to the current 1.5 kHz. To achieve this a hardware accelerator-based farm can join the CPU-based Processor Farm to speed up the tracking from the inner-most detectors [2].



[1] ATLAS Collaboration,**Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System**,ATLAS-TDR-029.
[2] ATLAS Collaboration, **Technical Design Report for the Phase-II Upgrade of the ATLAS Trigger and Data Acquisition System - Event Filter Tracking Amendment**,ATLAS-TDR-029-ADD-1

Event Filter Tracking

ATLAS is studying the performance of the Hough Transform (HT) tracking algorithm to use it for the future Inner Tracker detector. To exploit it, the ATLAS environment requires that the position of the particle be defined with the polar coordinates radius "r" and azimuth angle "ϕ".



Generic example of HT algorithm: two points in the left are connected in the right thanks to a change in coordinate system

Two versions of HT implemented on FPGA are under investigation as candidates for **(raw) particle tracking for filtering**, the Flexible version and the Low-Resources version. The FPGA boards used for implementation tests are Xilinx commercial demonstrators VC709, used by both versions, and VCU1525.

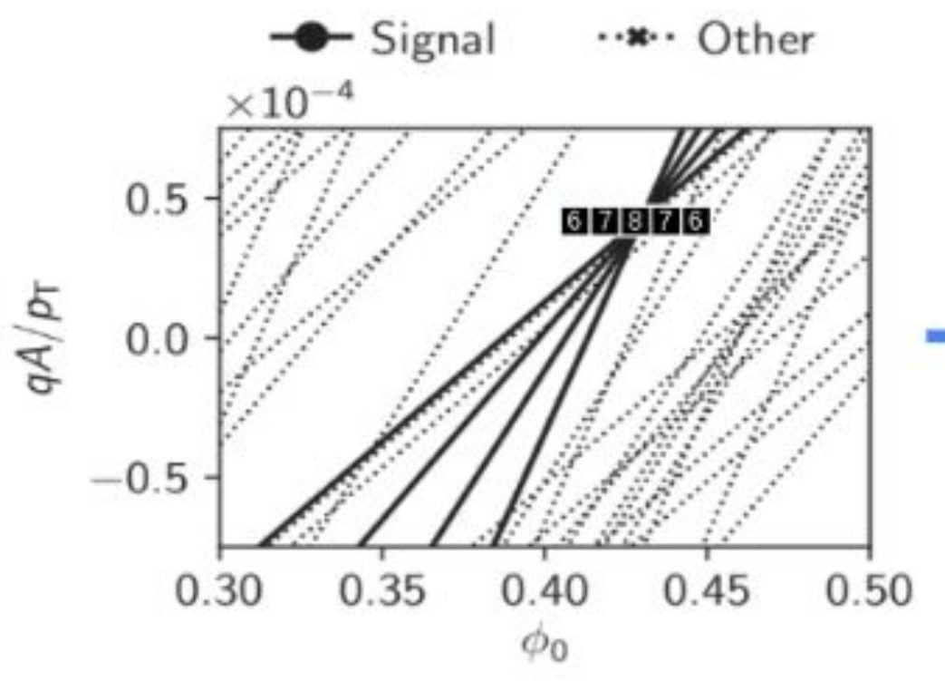
FLEXIBLE

Goal: Versatile HT that targets compute time performance versus occupied resources through high parallelization and high clock rate

APPROACH

$\phi_{0,n} = \phi + r*qA/p_{t,n}$

$qA/p_{t,n} = (\phi_{0,n} - \phi) / r$

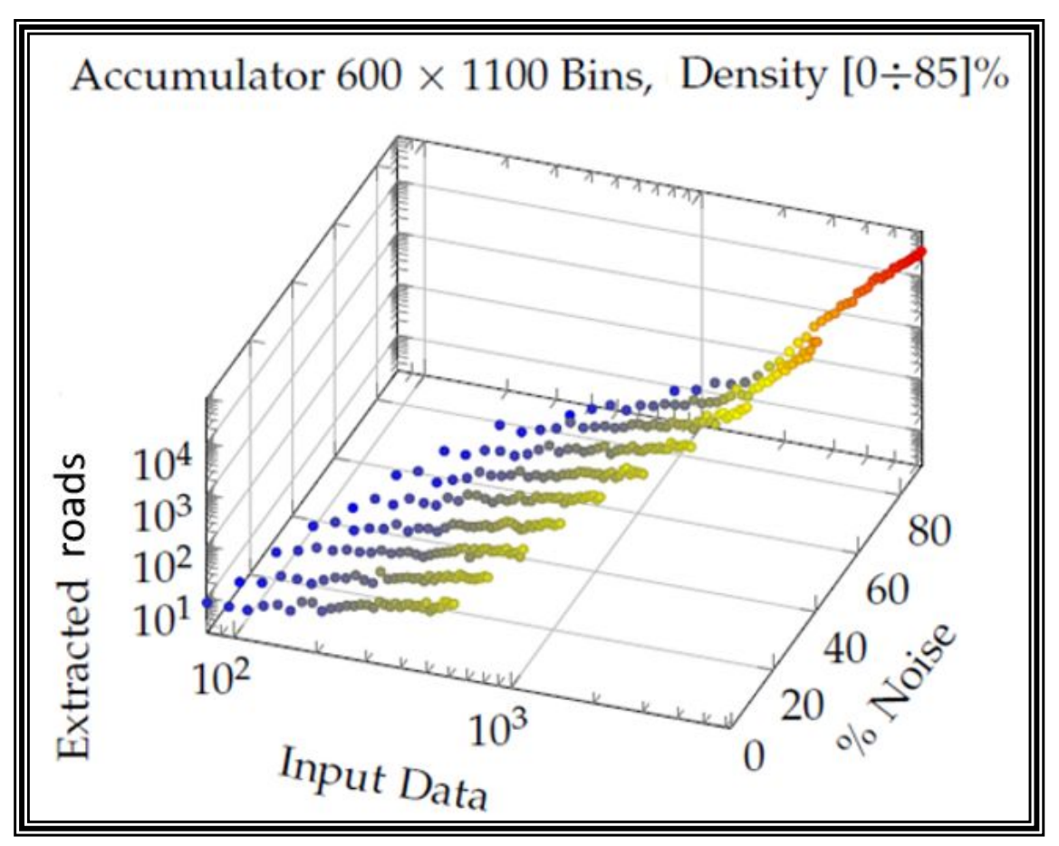


$\phi_0 = \phi_n + r_n*qA/p_t$

$qA/p_t = (\phi_0 - \phi_n) / r_n$

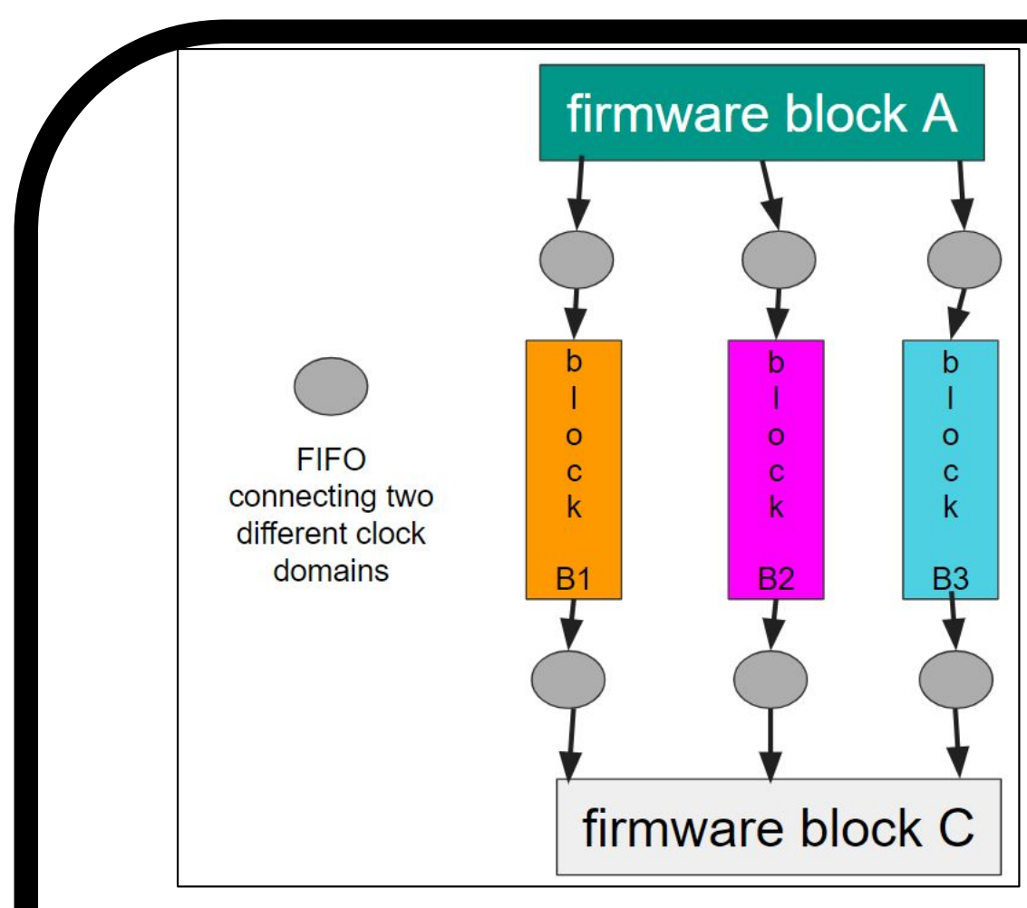
HT operations:

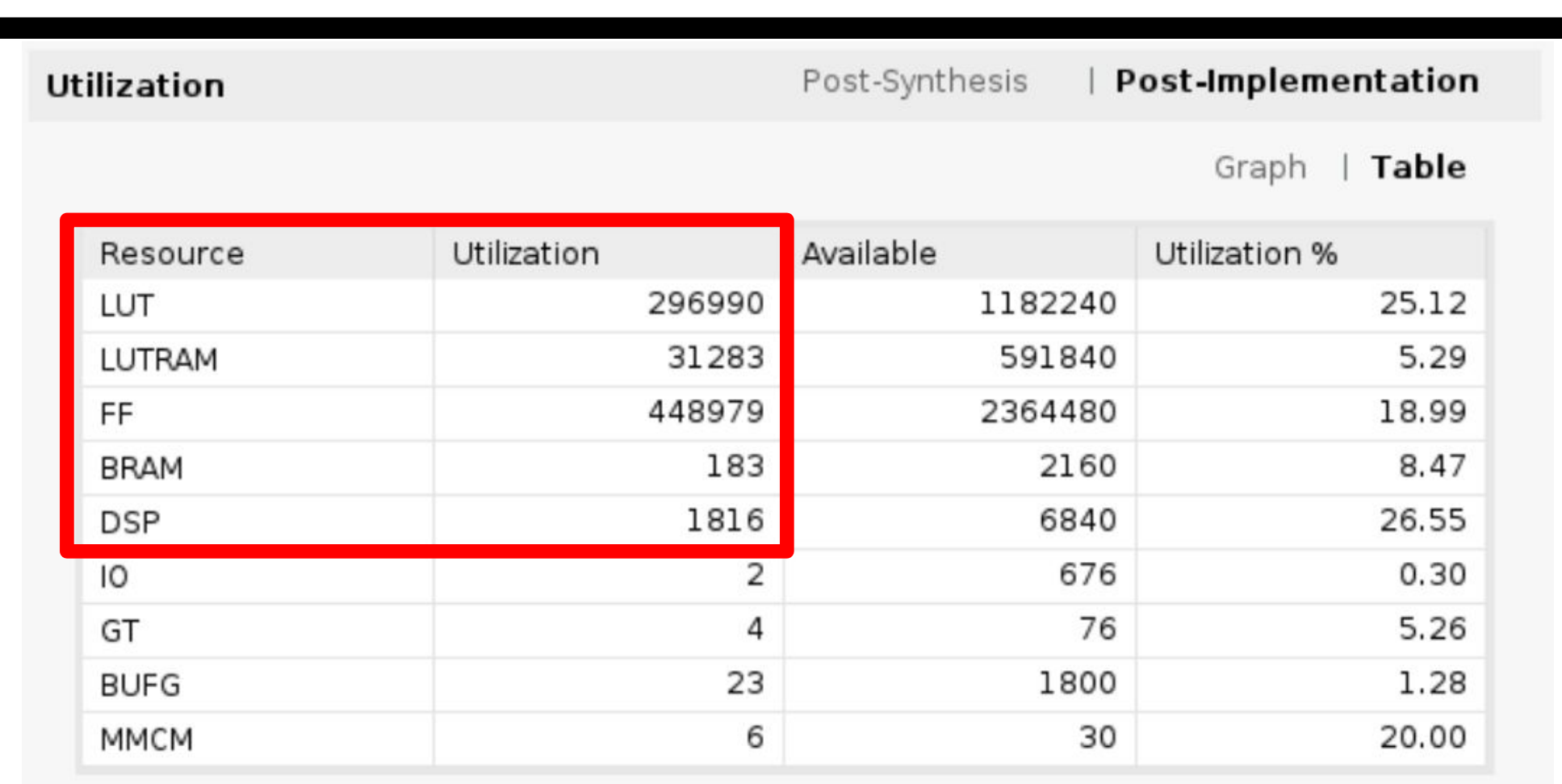
- Accumulator filling using the original HT formula in parallel across all incoming bins.
- Extraction of candidate tracks from the accumulator by applying the original HT formula "again" across all event clusters in parallel.



- First studies on the behavior of the algorithm with pure random inputs (hit tracks and % unwanted hit) with "road" as candidate tracks.

FIRMWARE DESIGN AND PERFORMANCE





Utilization Post-Synthesis | Post-Implementation

Graph | Table

Resource	Utilization	Available	Utilization %
LUT	296990	1182240	25.12
LUTRAM	31283	591840	5.29
FF	448979	2364480	18.99
BRAM	183	2160	8.47
DSP	1816	6840	26.55
IO	2	676	0.30
GT	4	76	5.26
BUFG	23	1800	1.28
MMCM	6	30	20.00

- Left: separate the firmware into different clock domains as much as possible;
- Right: firmware implementation functionality for a maximum frequency of 350 MHz on a Xilinx VCU1525 card for an accumulator of 216 qA/p_t x 32 ϕ_0 bins, 8 layers and 160 clusters stored per layer;

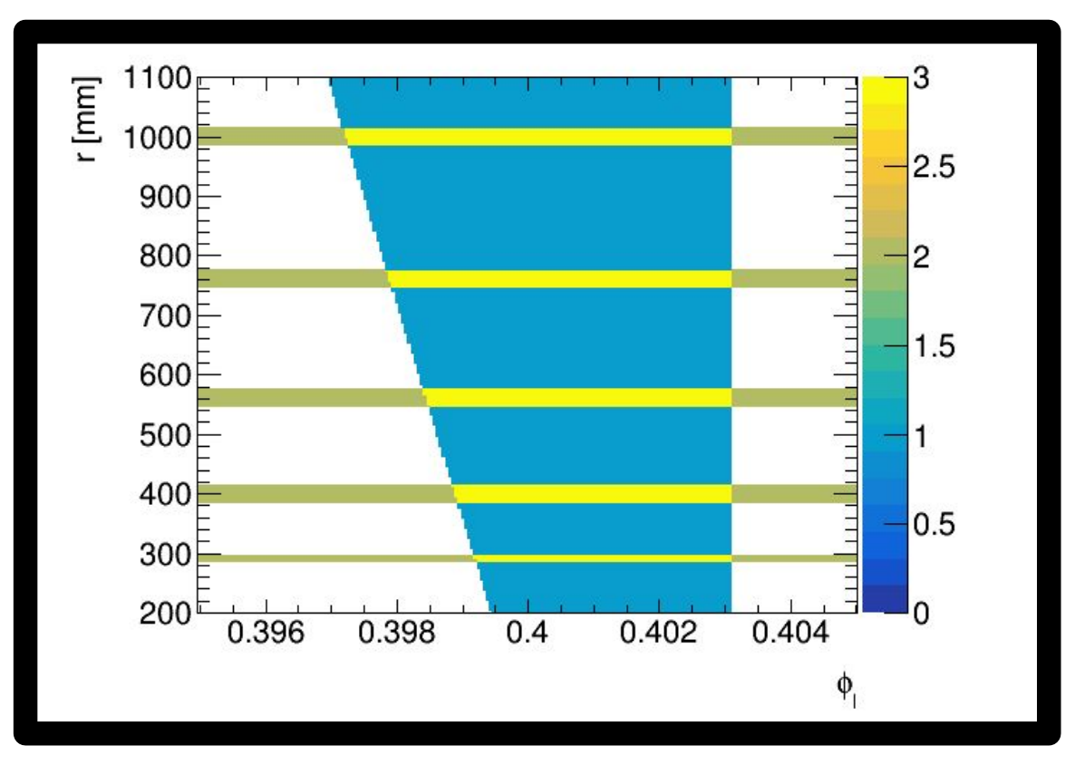
ALGORITHM PERFORMANCE AND TIME

Z-slices	μ 1-2 Gev	μ 2-4 GeV	μ > 4 GeV	π 1-2 GeV	π 2-4 GeV	π > 4 GeV
19	95.9 %	100 %	98.6 %	88.8 %	92.7 %	95.2 %
6	96.6 %	100 %	98.6	89.3 %	93 %	95.9 %

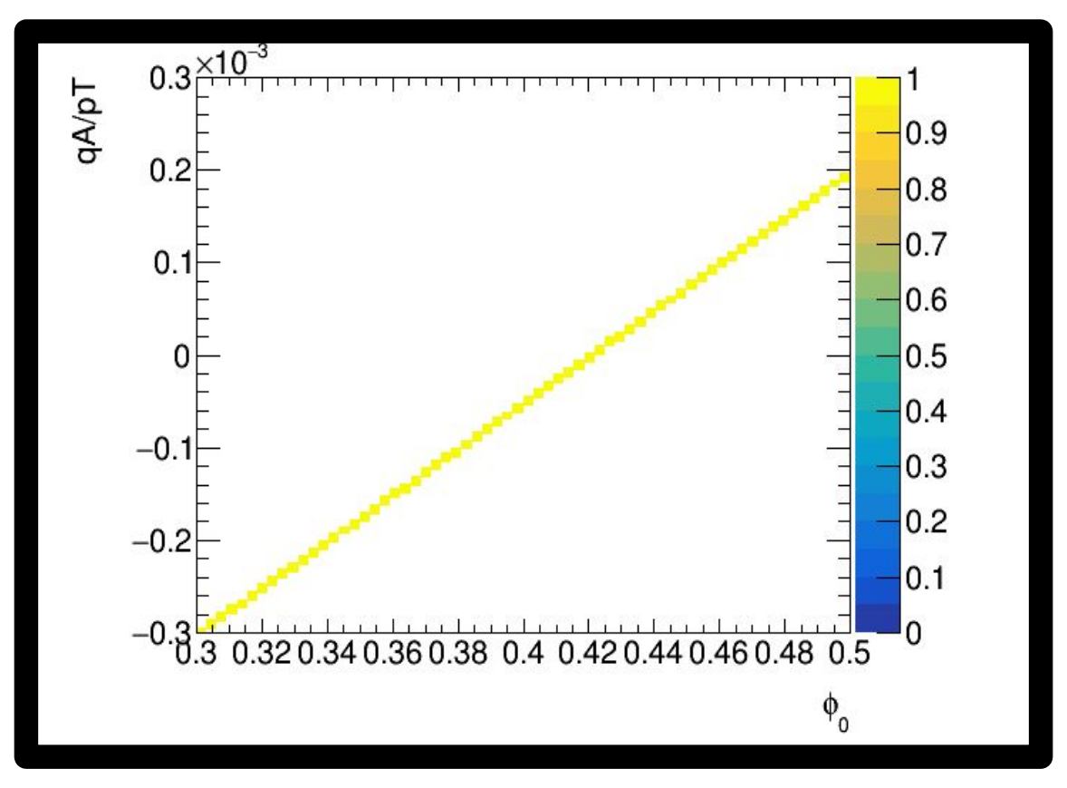
Efficiency of candidate tracks extracted with respect to truth. Tests done for muons and pions tracks. Low p_t shows low efficiency, issue in investigation. Candidate tracks and corresponding hits range between 550-950 and 10-13 respectively, leading to a processing time ranging from 3 to 6 μs. These results are related to the 8 outer-most layers of the barrel region only.

LOW-RESOURCES

Goal: to confirm the feasibility of the implementation and verify its functional compatibility



- Left: on software evaluated range of r: ϕ (in blue) to fire an accumulator bin;
- Right: on firmware checked ϕ range to fire accumulator bin (low cost solution);



APPROACH

- Idea: perform all possible mathematical calculations on software beforehand, assuming ATLAS geometry, and implement only calculated results on firmware
  - For simplification, input r is considered as a fixed value for each layer
  - No need for mathematical calculation logic on board
  - On the board, each accumulator cell needs only to check whether input phi is within the pre-calculated range or not
    - Very low-resource implementation is possible on firmware

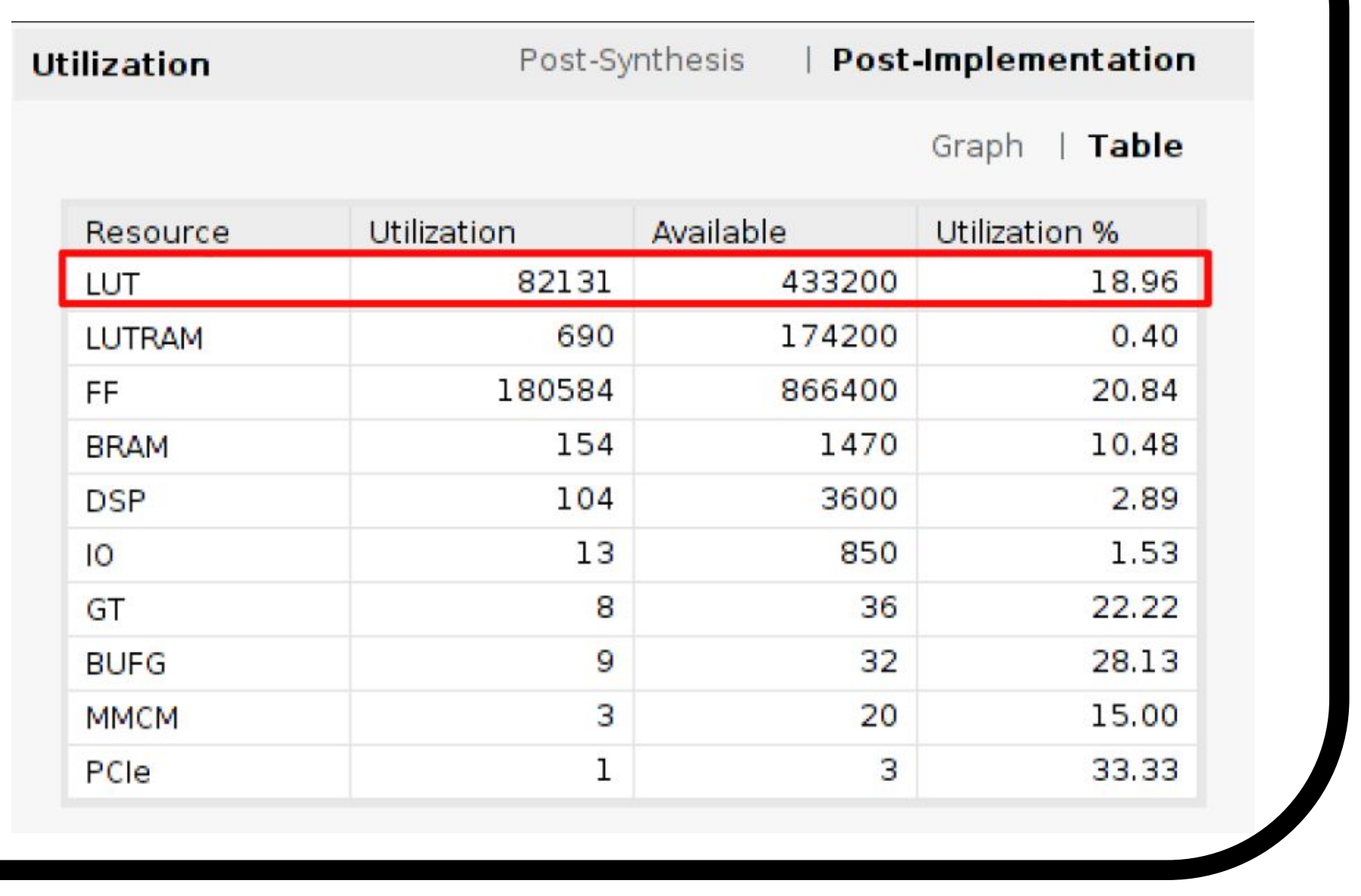
FEASIBILITY STUDY: SOFTWARE TEST

	Previous	math	Fix r	Scan r
#LUTs	400k	—	31k	108k
<#fit>	—	2375	2709	292k
<#candidate tracks>	—	242	270	4881
Inefficiency	—	4.5 %	6.5 %	0.36 %

- Logically compatible function is implemented as software
- The expected performance and utilisation are evaluated for some options
- Results:
  - Reasonably compatible efficiency is achieved using less than 8% resource of flexible version
  - By assuming input r width and accepting all possible r cases, efficiency is significantly improved
    - However, the amount of outputs is not acceptable by the downstream
    - In case duplicated outputs can be removed on board, this is promising results

FEASIBILITY STUDY: FIRMWARE TEST

- Target: confirm implementation feasibility and check functional compatibility
- Results:
  - The implemented blocks work as expected by software simulations
  - Any block works within the 250 MHz clock domain
  - The use of resources corresponds to what is evaluated by the software
    - with approximately 31k LUTs utilization compared to the 400k of the flexible version, compatible performance is achieved.
    - Warning: This approach is only valid for the barrel region due to the assumption of a constant "r" maintained.



Utilization Post-Synthesis | Post-Implementation

Graph | Table

Resource	Utilization	Available	Utilization %
LUT	82131	433200	18.96
LUTRAM	690	174200	0.40
FF	180584	866400	20.84
BRAM	154	1470	10.48
DSP	104	3600	2.89
IO	13	850	1.53
GT	8	36	22.22
BUFG	9	32	28.13
MMCM	3	20	15.00
PCIe	1	3	33.33

Conclusions and Future Steps

The ATLAS experiment is investigating commodity solutions for Event Filter Tracking, including the possible use of FPGA-based accelerators. Two FPGA implementations of the Hough Transform algorithm are proposed as a feasible and well-performing solution for fast tracking in HEP experiments. Firmware development is quite advanced, being stable for both version at 250 MHz, and has reached satisfactory and competitive functionality and frequency. Preliminary tracking performance studies are promising, showing > 95 % efficiency for reconstructing single muons with p_T > 1 GeV.

ACAT 2022, Bari, Italy