JUSTUS-LIEBIG-
UNIVERSITÄT
GIESSEN

Belle II

Status of ROI Algorithm

David Münchow

Thomas Geßler, Wolfgang Kühn, Jens Sören Lange, Björn Spruck

II. Physikalisches Institut
Universität Gießen

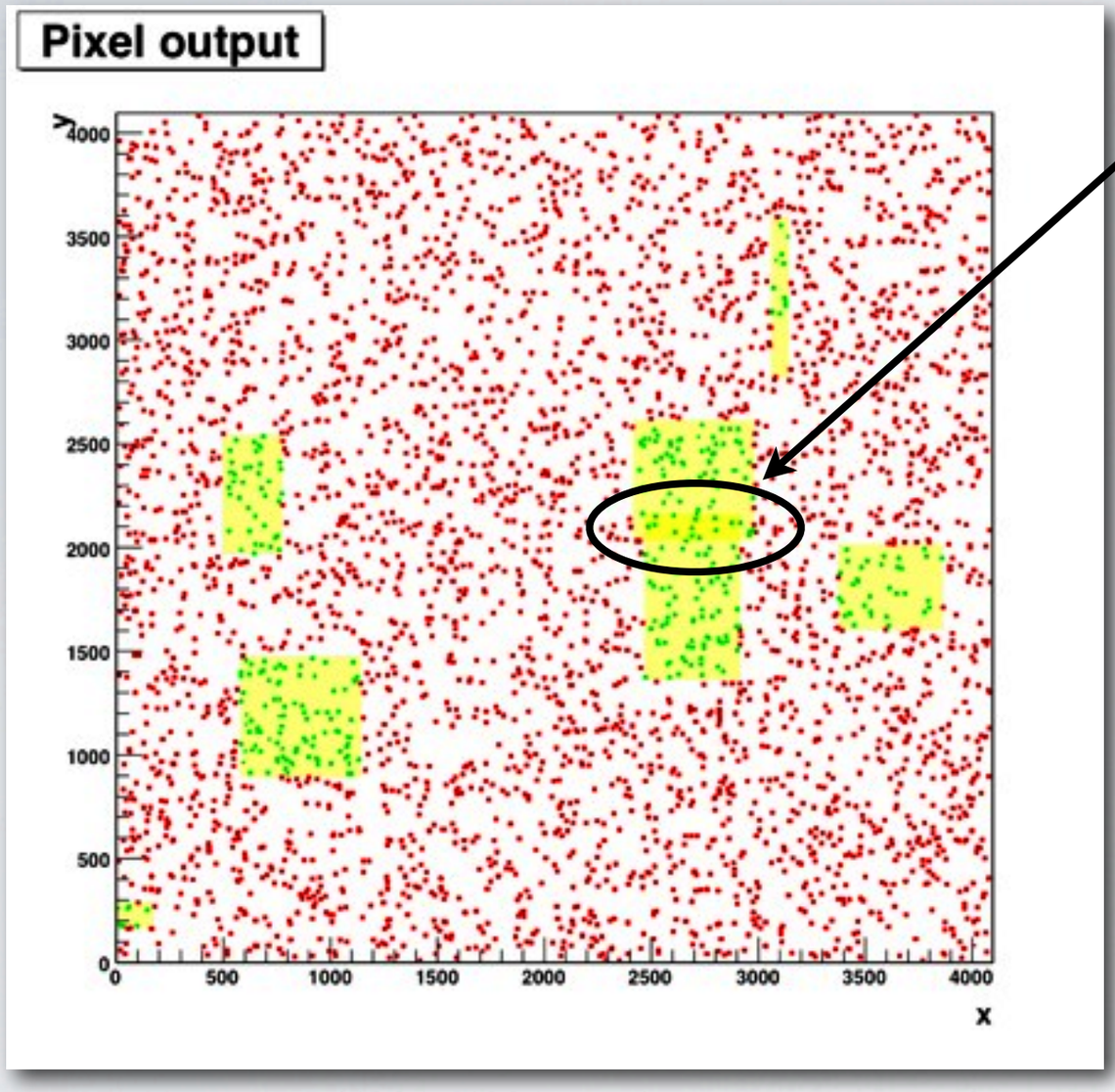
29.04.2011
DEPFET Workshop
Ringberg

II. Physikalisches
Institut

supported by BMBF under grant no. 05H10RG8

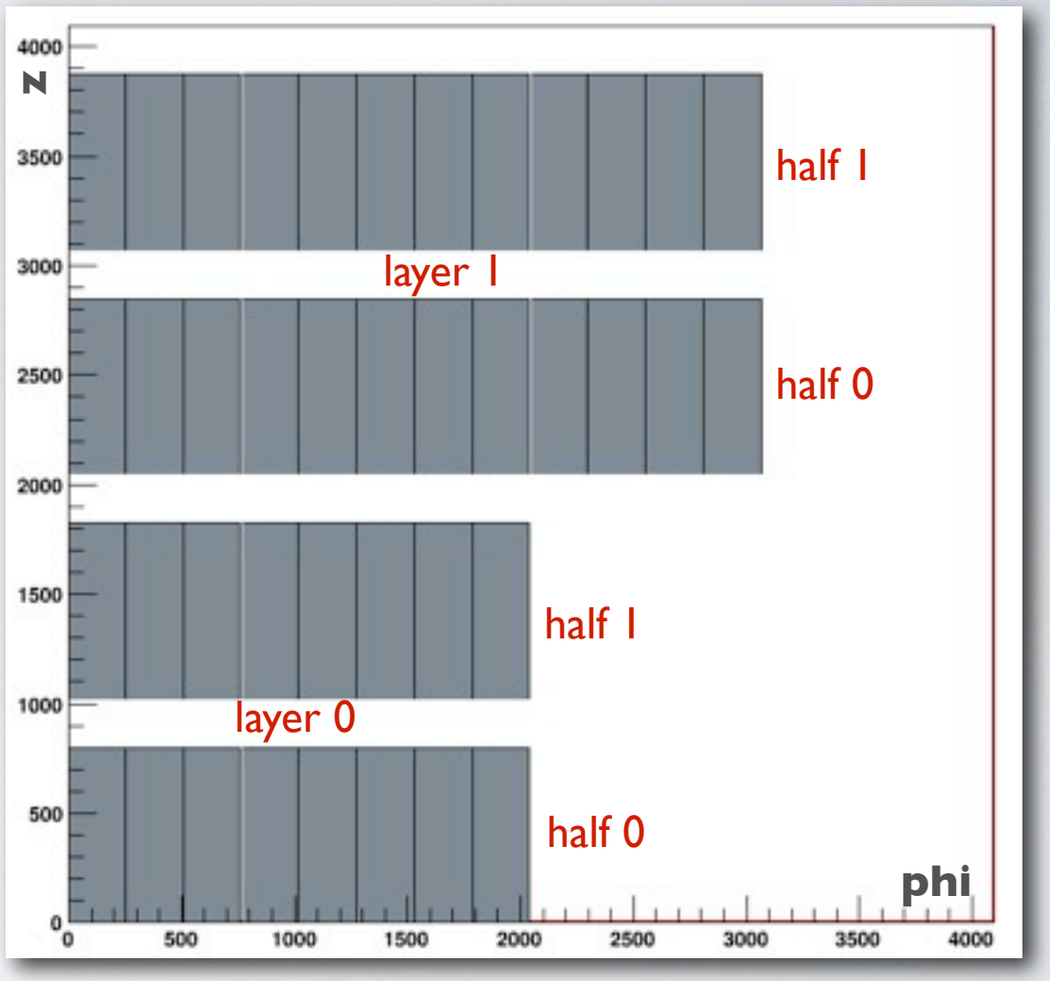
Bundesministerium
für Bildung
und Forschung

Data appearance



Full address range (4096×4096 pixel) with the result after ROI finding algorithm (green: accepted pixel; red: rejected pixel)

overlaping ROIs



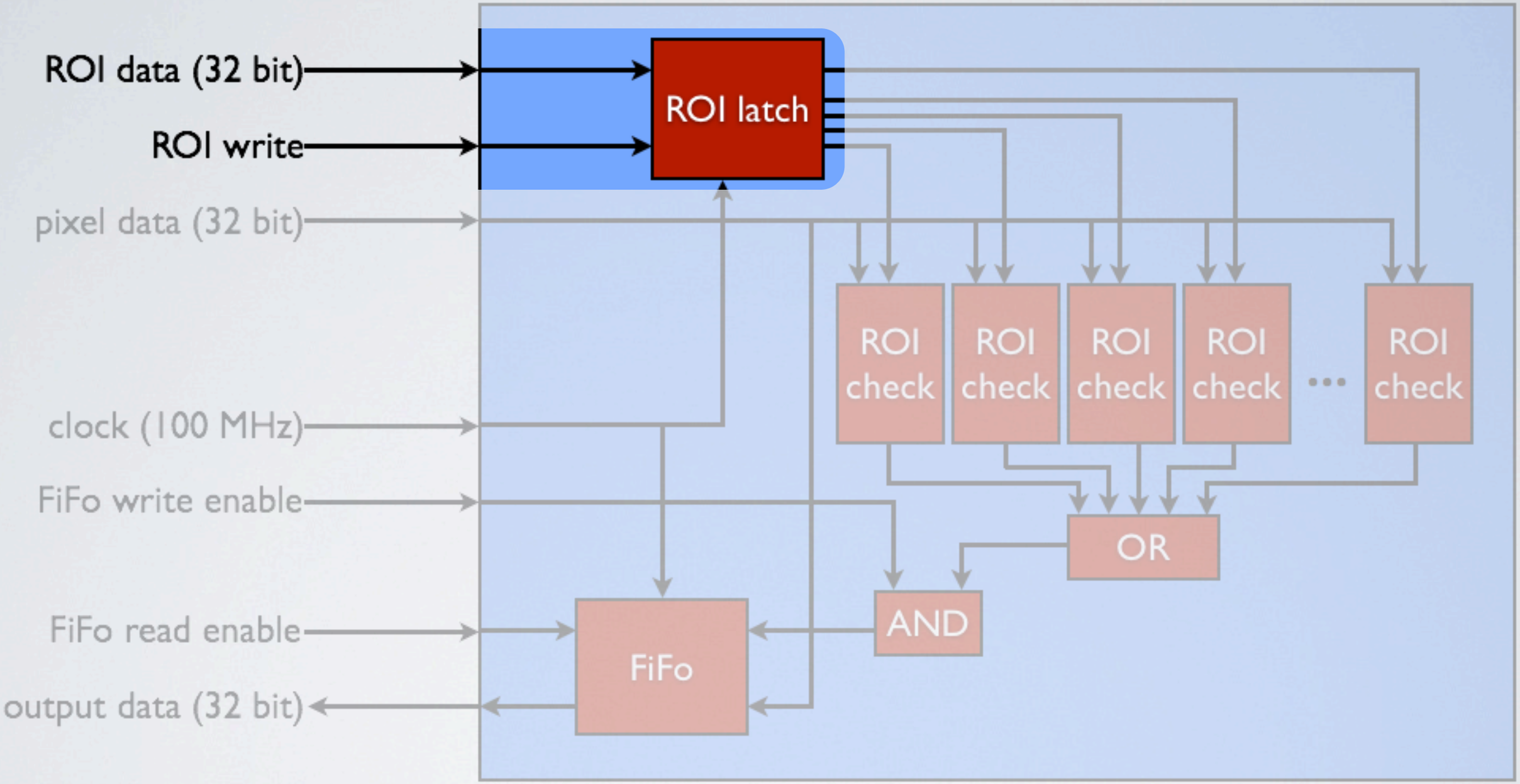
Mapping of all 40 half ladders in address range

- Simulated occupancy: $\leq 3\%$
- No cluster
- Event processed on CN

- Expected data rate: $\leq 20\ \mathrm{GB/s}$
- Definition of ROIs: $x_1 < x < x_2$

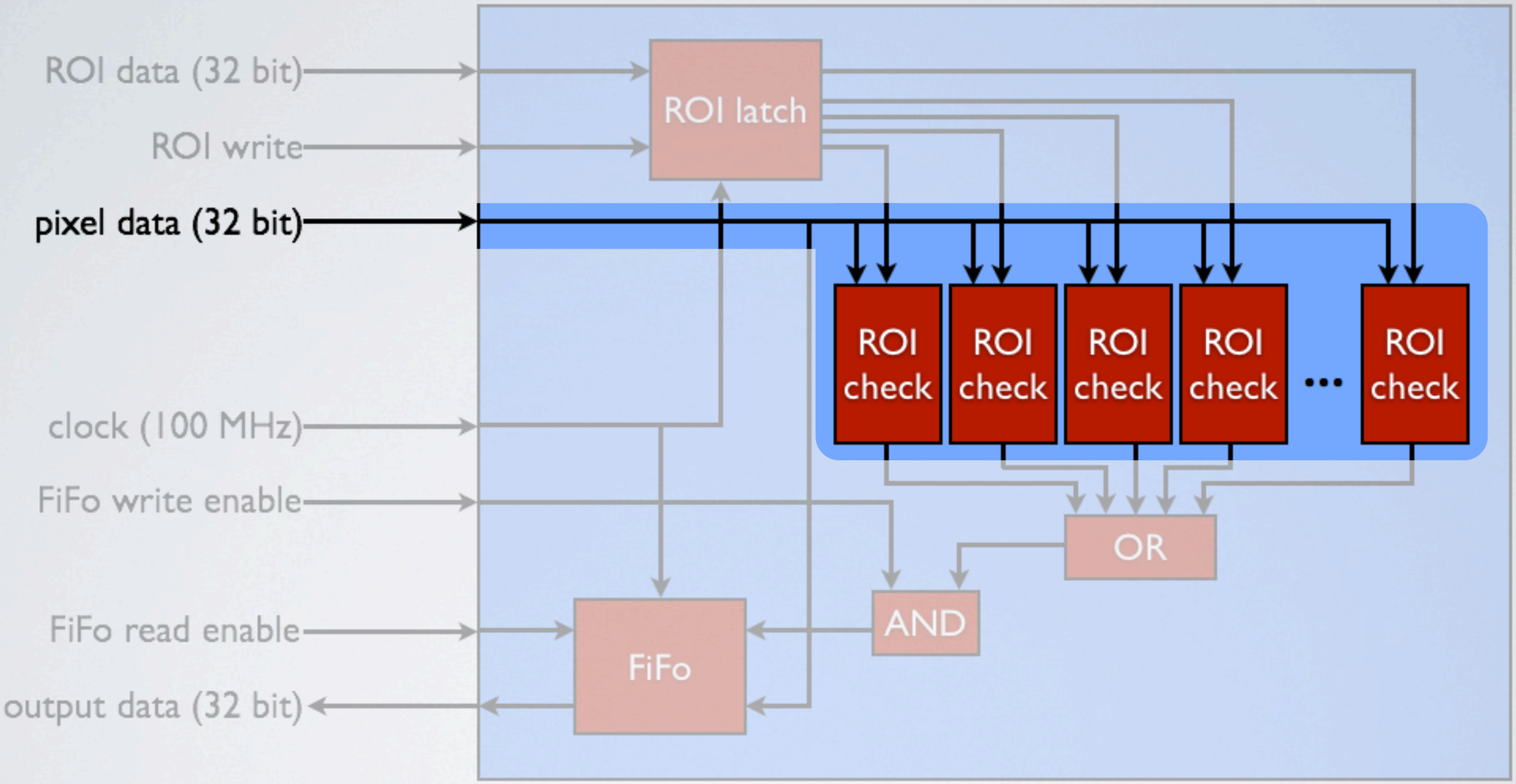
$y_1 < y < y_2$

ROI selection



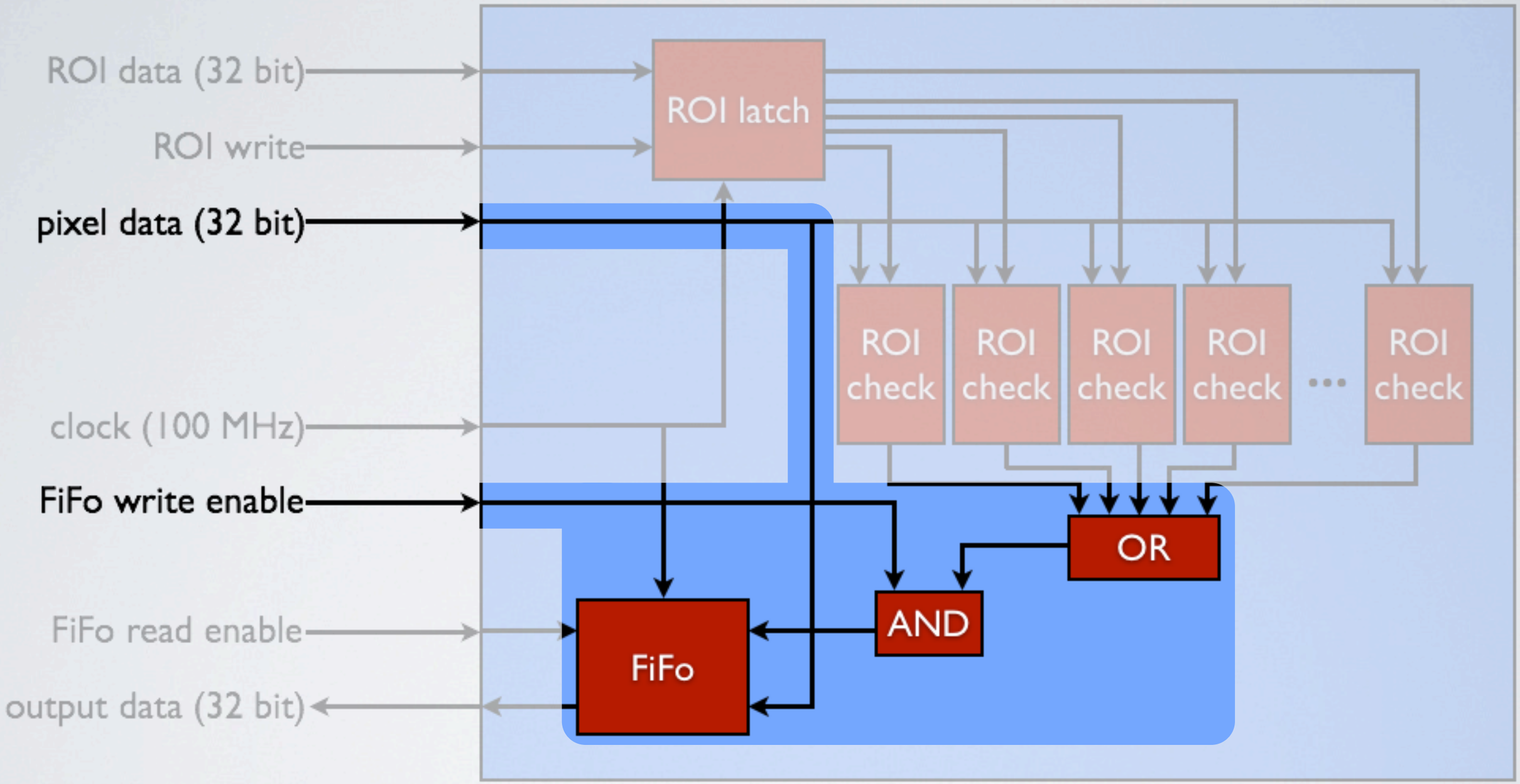
- VHDL based hardware algorithm
- ROI input with full clock speed (100MHz) and latched for further calculations

ROI selection



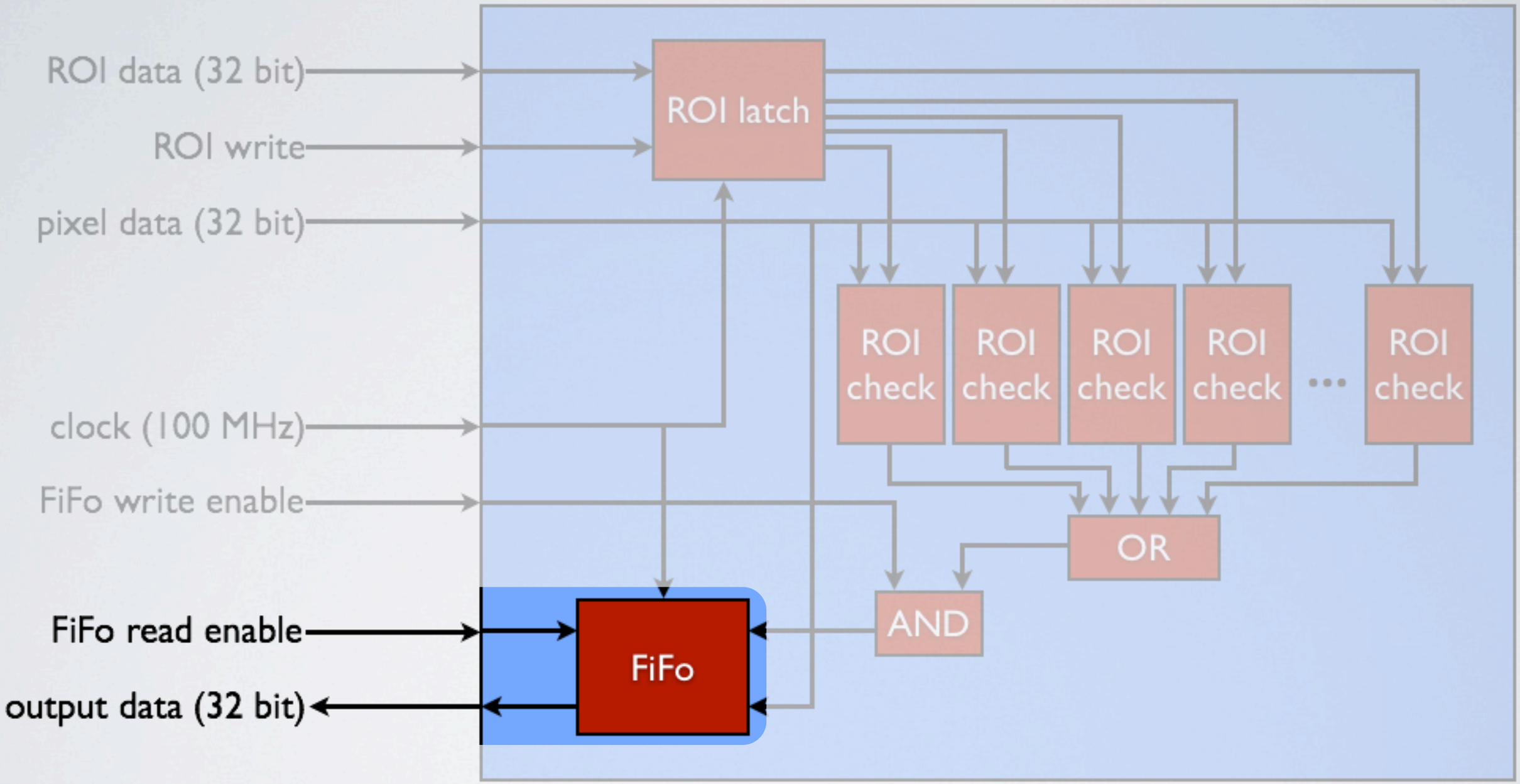
- Pixel input with full clock speed (100MHz)
- ROI-check for all ROIs in parallel (up to 31 realized)

ROI selection



- Pixel which are in at least one ROI get buffered in FiFo
- Discard all others; no buffer or storage of those

ROI selection



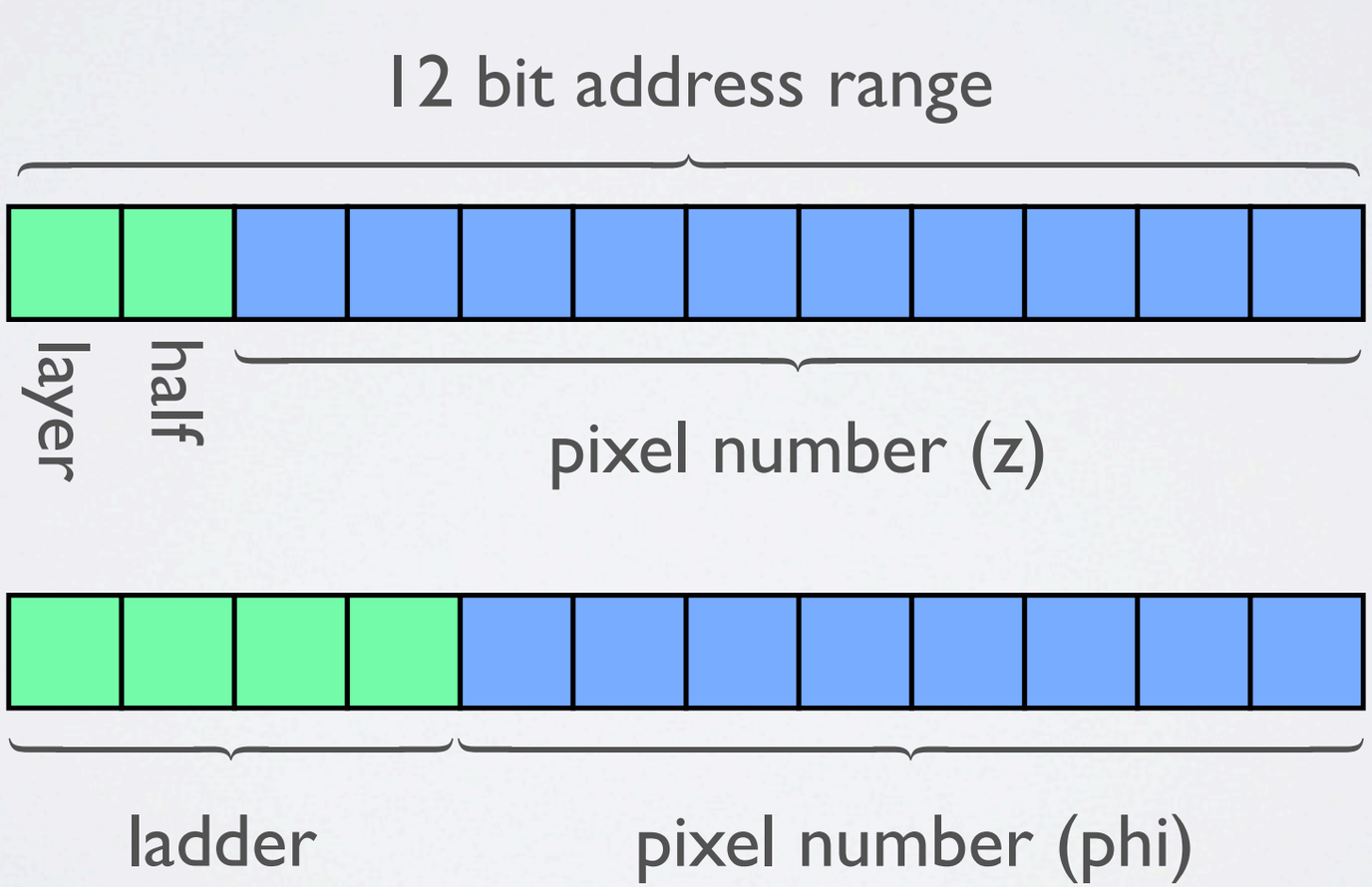
- Read out buffered pixel with full clock speed (100MHz)

Test on CN with Random Data

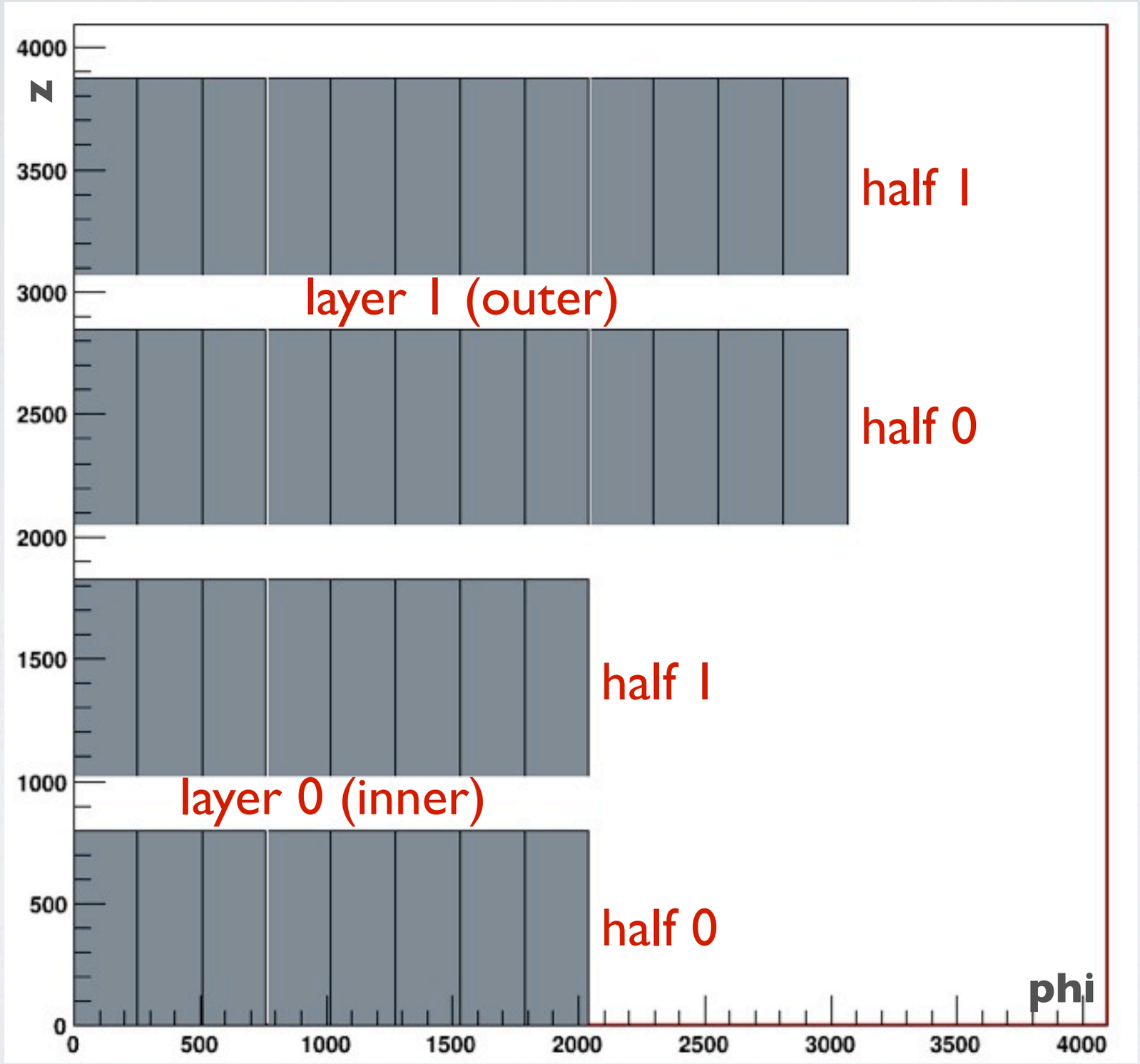
- Test code:
  - Running on PPC (Linux)
  - Generation of pixel and ROI data
  - Sending and reading data to/from core
  - Checks if core output is correct
- Test data:
  - Random generated pixel
  - Random generated ROIs (position and size)
    - including overlapping ROIs
- Bit error check:
  $1.9 \cdot 10^6$ events checked (with 31 ROIs)
  - $1.6 \cdot 10^{10}$ pixel in total
  - $\approx 40\%$ pixel in ROIs (all correct found)
  - $\approx 60\%$ pixel outside ROIs (all correct not found)
- **No errors**
- Achievment of ROI selection algorithm on ATCA-system:
  1 CN × **5** FPGAs × **4** IP-Cores × **31** ROIs
  ⇒ **620** parallel operations per CN

Test on CN with MC Data

- Address range of 4096×4096 pixel possible in actual IP-core
- Each half ladder needs only 786×250 pixel
- Upper bits of address for half ladder numbering
- Whole PXD detector fits in address range of IP-core

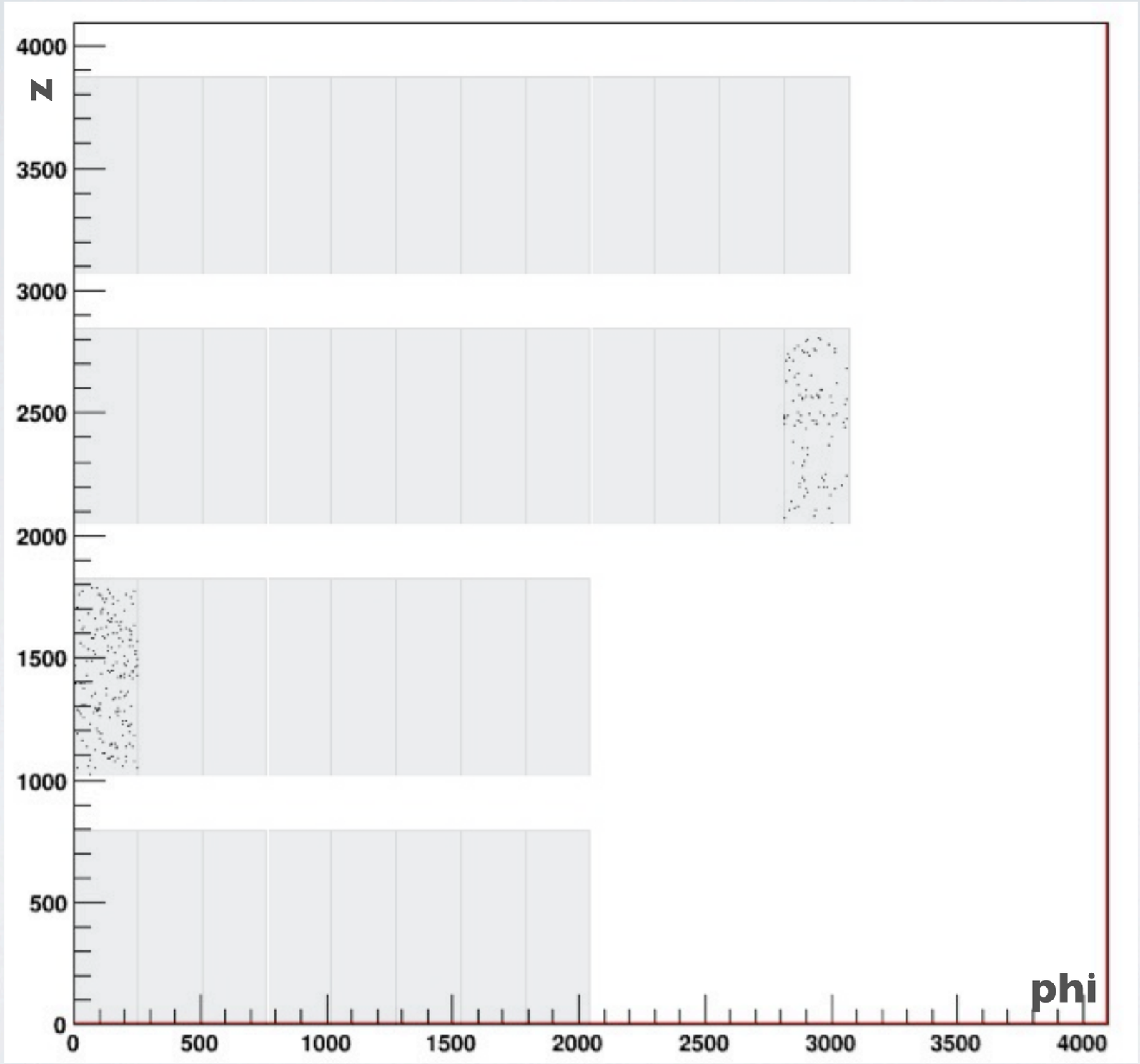


Test on CN with MC Data



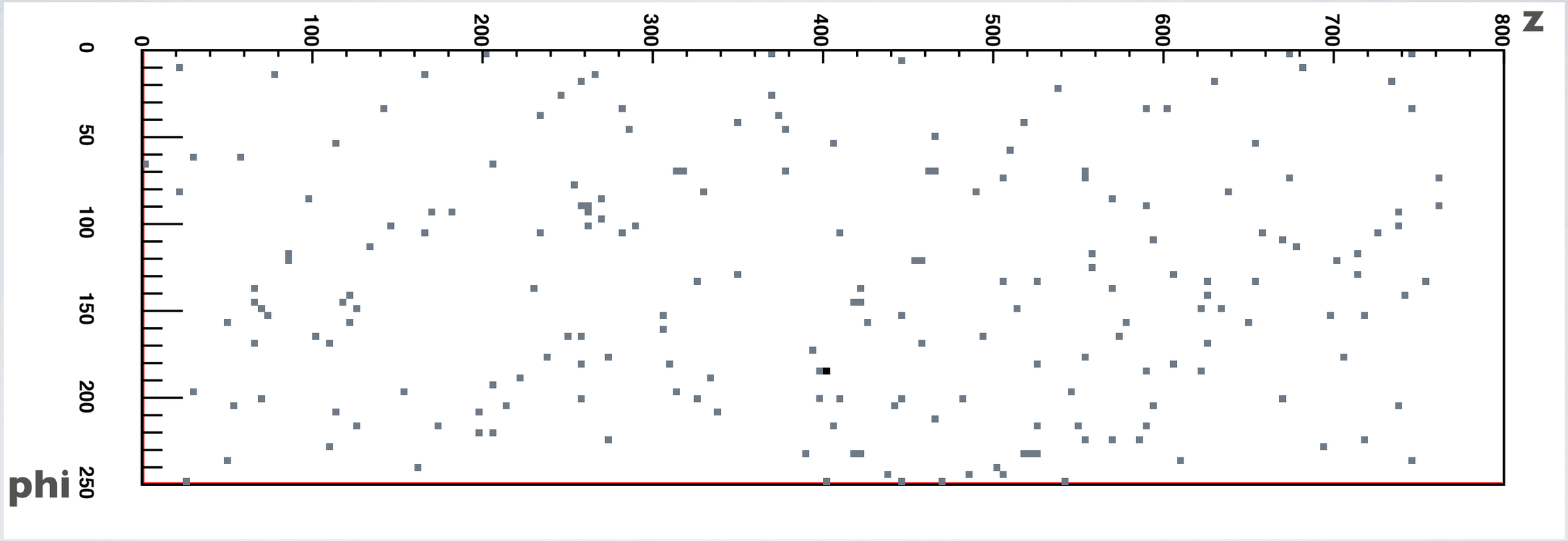
Mapping of the half ladders (800×250) in the full address range (4096×4096 pixel)

Test on CN with MC Data



Simulated data in the full address range (4096×4096 pixel)

Test on CN with MC Data



Distribution of simulated data on one half ladder (800×250)
(simulated occupancy ≈0.1%)

Simulated data generated by Zbynek Drasal, Prague

Test on CN with MC Data

- Test code:
  - Server on PPC at CN
  - Client on PC
  - Sending and reading data via network
  - Checks if core output is correct
- Test data:
  - Simulated data with and without background
  - Data contain tracks e, K, μ and π with 1GeV and 0.1GeV
    - Single PXD hits (no cluster)
    - MC PCD hits with random background PXD hits
    - ROIs generated with random size arround MC hits
- Bit error check:
  10^4 events for each type checked (with up to 31 ROIs and about 300 hits)
  $1.2 \cdot 10^5$ events in total
- **No errors**

Summary

- Data reduction for PXD needed
  (data rate: $\leq 20\ \mathrm{GB/s}$)

- Reduction algorithm realized on FPGA based Compute Node

- Algorithm realized for up 31 ROIs

- Algorithm works without errors
  - $1.9 \cdot 10^6$ random events checked ($1.6 \cdot 10^{10}$ pixel)
  - $1.2 \cdot 10^5$ physics events checked

- Full speed data throughput works ($100\ \mathrm{MHz}$)

Thank you for your attention