

# The FEIC development for the NPAL Project: Navigation for Planetary Approach and Landing

## A Core Image Processing Chip for Smart Landers Navigation Applications

- MicroElectronics Presentation Days,
- ESA/ESTEC, 4-5 February 2004
  
- Marc Souyri Astrium , Martin Dunstan University of Dundee



# Introduction- Project Context

---

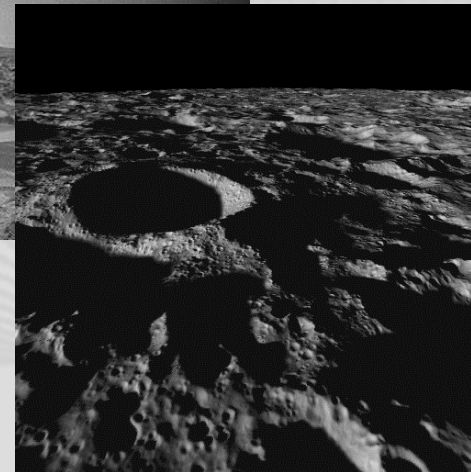
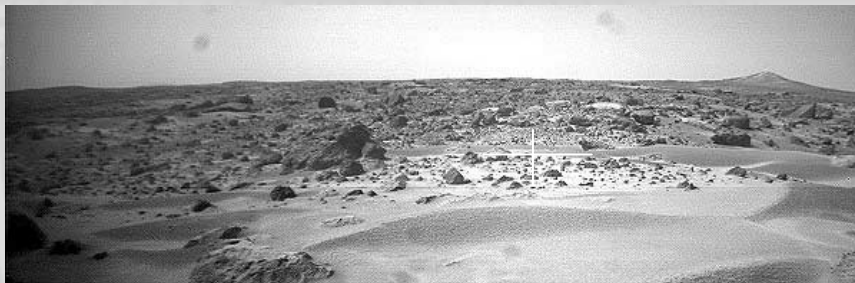


- The « Navigation for Planetary Approach and Landing » study is a complex R&D that aims to bring two major contributions to secure the concept for a planetary lander:
  - The development of a complete camera demonstrator. Compacity, low mass and large field of view are the main requirements. The camera will use an APS sensor. The camera development is ensured by Galileo Avionica.
  - The development of a FPGA implementing the most demanding image processing algorithm for on-line extraction and tracking of feature points at a high rate.
- The current presentation is focussed on the development of the FPGA called the FEIC for « Feature Extraction and Image Correlation ».

# Vision-Based Navigation

---

- **Objective: safe landing**
  - Terrain characterisation, site designation, guidance
- **Objective: soft landing**
  - Controlled static and dynamic conditions at contact
  - Enable legs and controlled touch down (large vehicles)



- **NPAL Challenge**
  - Use of vision to
    - Operate Early detection of site
    - Perform terrain characterisation with a metric accuracy
    - Make navigation converge in all cases... And demonstrate it!

# Introduction- Development Plan (1/2)

---



- The development of a FPGA as the FEIC involves many competences:
  - The study of the mission in terms of navigation concepts: Astrium Toulouse,
  - The collection and synthesis of images of Mercury for FEIC testing by upgrading the PANGU software: University of Dundee,
  - The development of a dynamic simulator: Science Systems in Bristol,
  - The comparison between the different image processing algorithms : Ineti at Lisbonne



# Introduction- Development Plan (2/2)

---



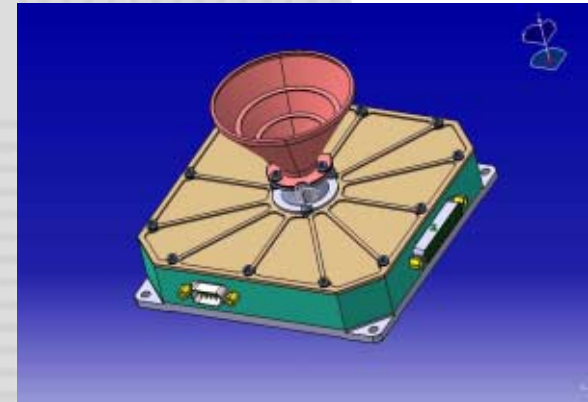
- The algorithm definition, the specification and the feasibility study of the FEIC is the responsibility of EADS Astrium
- The finalization of the Architecture, VHDL coding, Xilinx Implementation and board debug is the responsibility of the University of Dundee
- The board Test of FEIC without and with camera is conducted by EADS Astrium in Toulouse
- The camera is developed by Galileo Avionica



# Navigation Camera

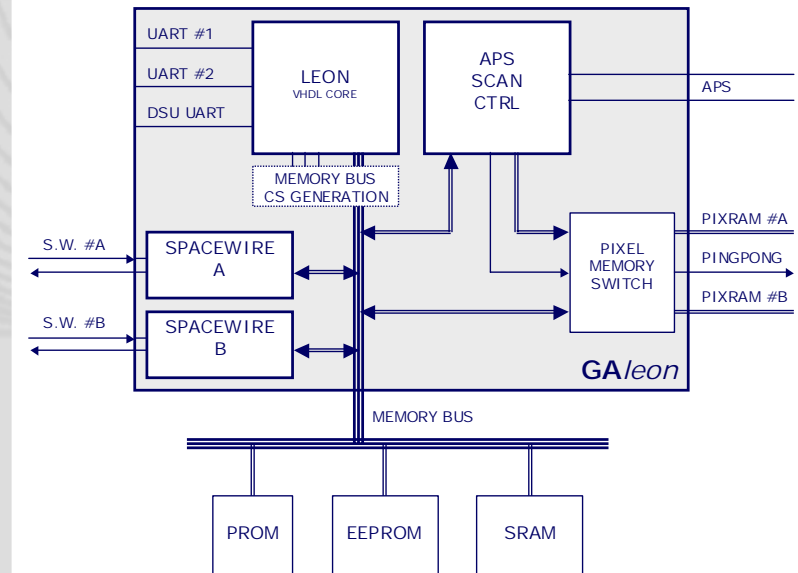
## • Navigation camera needs:

- A large field of view to support vision on a manoeuvring vehicle wrt to a fast moving environment
- A high data rate, for video speed, high resolution images
- A compact, low mass system: absence of thermal control, needs for a highly integrated electronics



## • Design features

- APS 1024x1024 detector, for a 70°FoV camera
- Compact electronics board, with a LEON processor, SpaceWire cmd/ctrl interface, for a fast, 20 Hz image rate
- **The FEIC** for «smart compression» and navigation observable extraction in line



# FEIC need specification

---



- The FEIC shall be able to perform the following functions:
  - Extract feature points from an image,
  - Track the extracted feature point between images by correlation,
  - Allow control by the On Board Computer,
  - Dialog with the camera.
- In addition, the FEIC design shall be implemented in a FPGA with a design compliant with a future ASIC implementation.

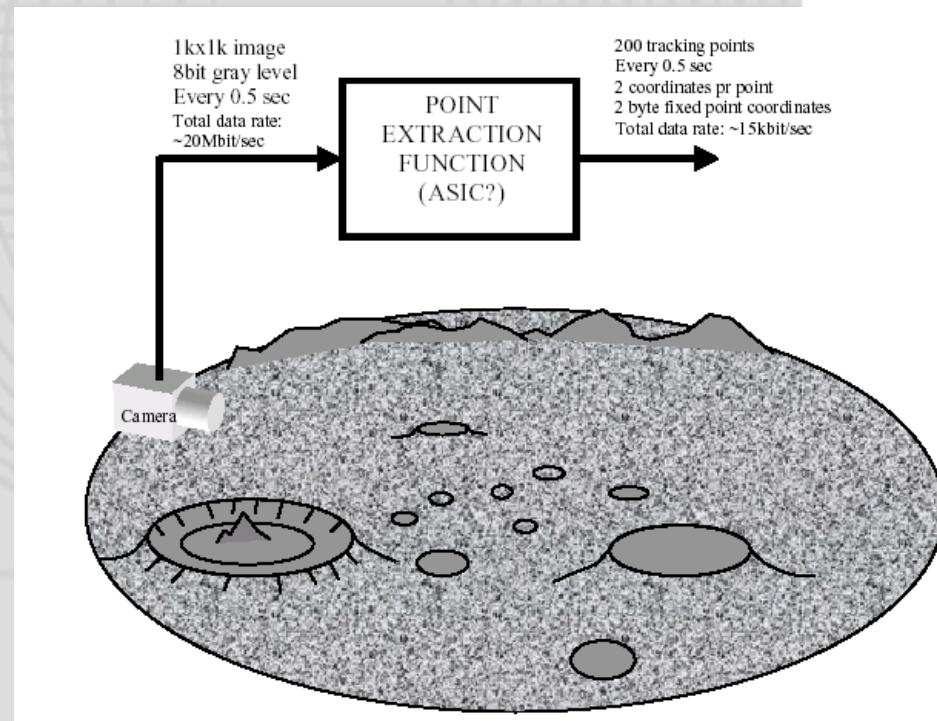
## The FEIC as a Smart Compressor

- On line Extraction of condensed navigation information

- Converts a 20 fps x 10 bits x 1 Mpixel = 200 Mbit/sec into a 200 pts x 10 words = 128 kbit /sec

- Navigation Memory

- Memorise up to 200 image textures or « vignettes » for further correlation





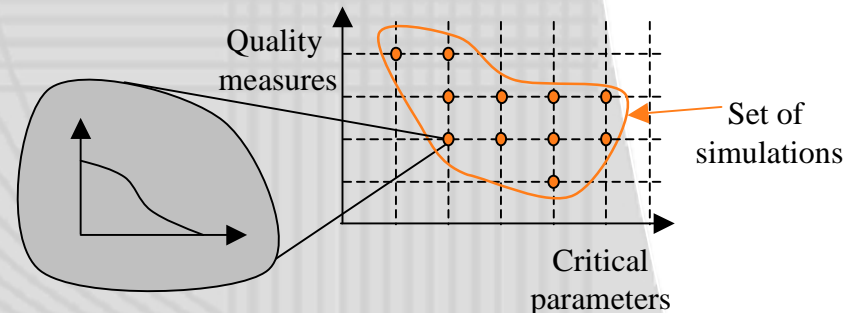
# Algorithm choice: Performance evaluation

- Image processing algorithm candidates for feature extraction and tracking:

Criterion	Selection	Tracking
Harris, Beaudet, Tomasi, Susan, Moravec, Spoke, Loupias	Global, Local	Correlation, Optical Flow

- Evaluation according to objective criterions:

- Spatial distribution of points
- Repeatability between frames
- Relative accuracy
- Probability of good match



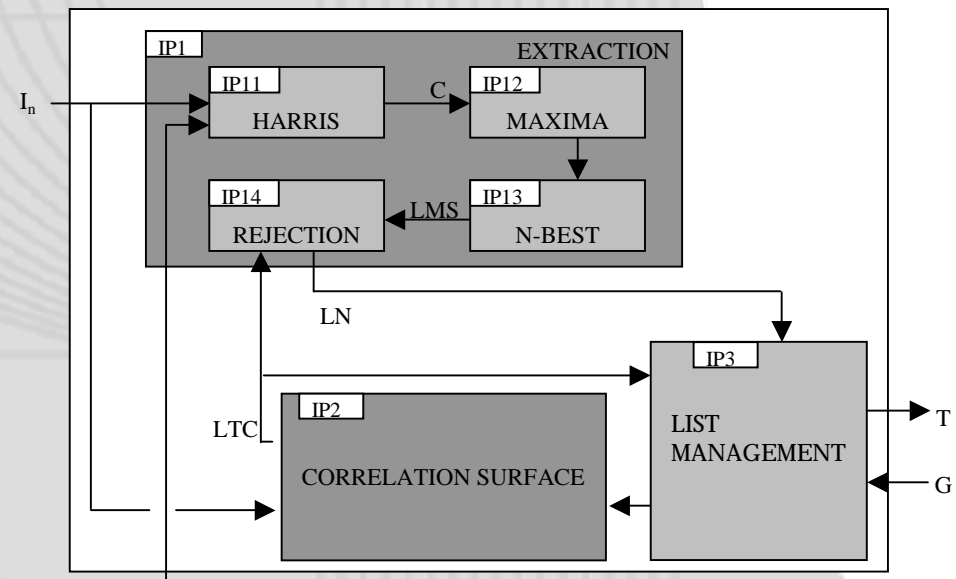
... and wrt numerous critical parameters

- Radiometric distorsion (MTF, noise, illumination)
- Kinematic distorsions (translations, rotations and approach)
- Scene content

# Algorithm choice: HW implementation

- According to performance evaluation, preselection of feature extractors for HW implementation : **Harris, Tomasi, Beaudet, Susan**
- Susan is complex to implement due to exponential computation. Harris achieves better performances than Beaudet and Tomasi. Harris is selected
- For tracking two main options :
  - A/ matching of feature points extracted on successive images
  - B/ Single feature extraction on image A then correlation between the texture around the feature point in image A and a search window around the estimated position of the feature point in the new image.

- Solution B is retained, it allows better performance in tracking.

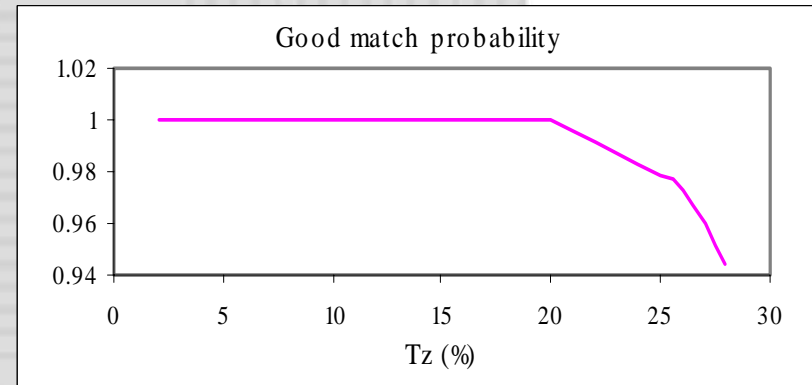


# Algorithm choice: Expected performances



- Matching accuracy around 1/10 pixel
- Robustness to distortions
  - Rotations
  - Approaching movements
  - Noise sources

	Bruit gaussien (RSB=70)	Pixels défectueux (0.1%)
P(GM)	0.995	0.979



- Reference test case

	Reference case	P(GM)
Rz	0.05°	1
Tz	5%	1
Bruit	RSB=70	0.995
Total		0.995

- Extrapolation in terms of mean length of tracks: ~200 frames

# FEIC Tasking : Feature Point Extraction

---



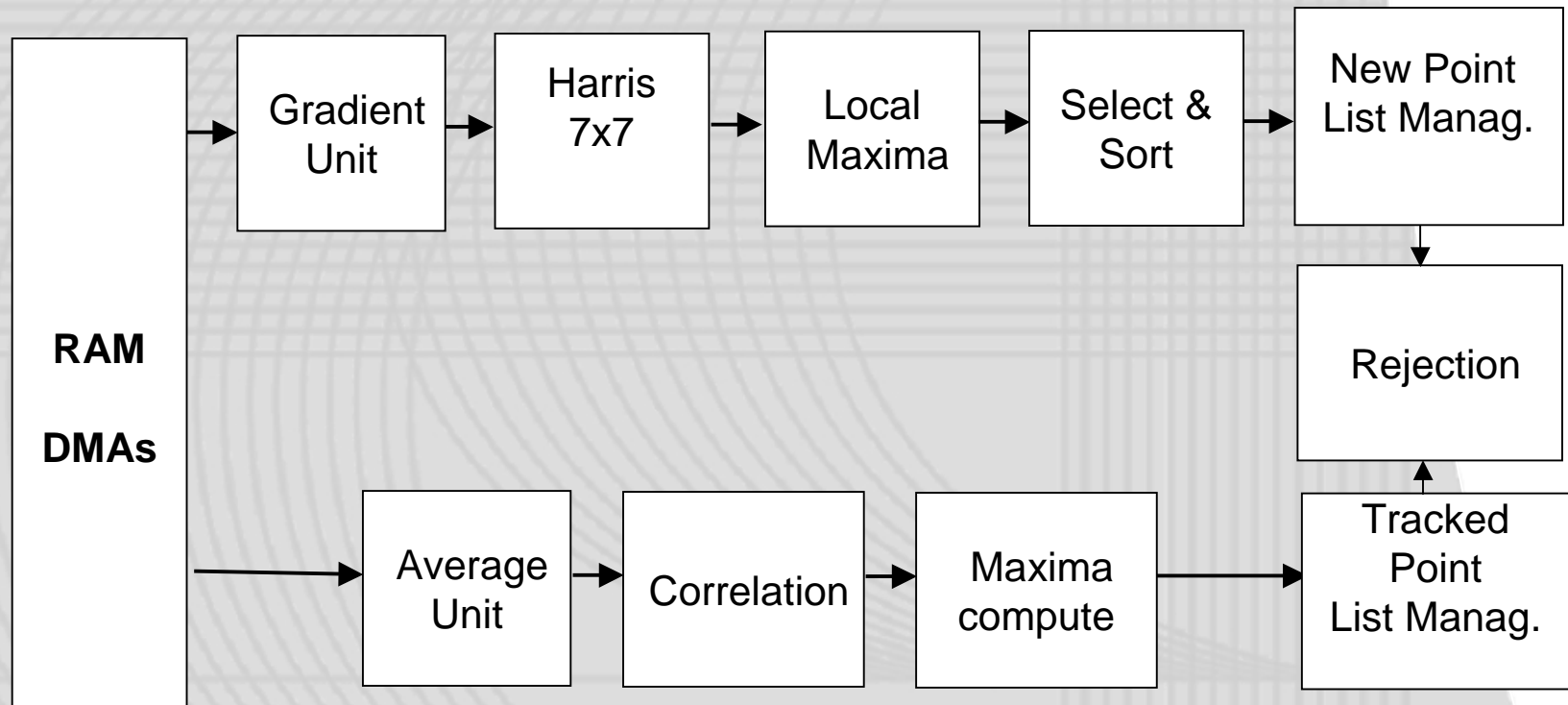
- Feature point extraction starts after toggling of the flip-flop image RAM
- Feature point are extracted with the Harris algorithm.
- Feature points too close together are rejected by the Local Maxima function
- A maximum of 200 best feature point is selected and sorted from the image.
- A rejection function suppresses points that are not new.
- The List of new feature points is transmitted to the OBC. The OBC selects the new points replacing the points that have failed to track or that disappeared from the field of view of the camera.

## FEIC Tasking : Tracking

---

- The OBC manages the List of point to correlate that is issued from the List of new feature points.
- For each feature point to correlate :
  - The OBC provides the FEIC with the displacement vector
  - The FEIC reads a search window of the new image surrounding the computed position of the feature point
  - A correlation of the search window with the reference texture kept in the FEIC is then made.
  - The best matching determines the expected position of the feature point.
  - The texture kept in FEIC can be updated if such option is selected.
- The results of tracking are sent to the OBC for decision and navigation.

# Block Diagram of the image processing



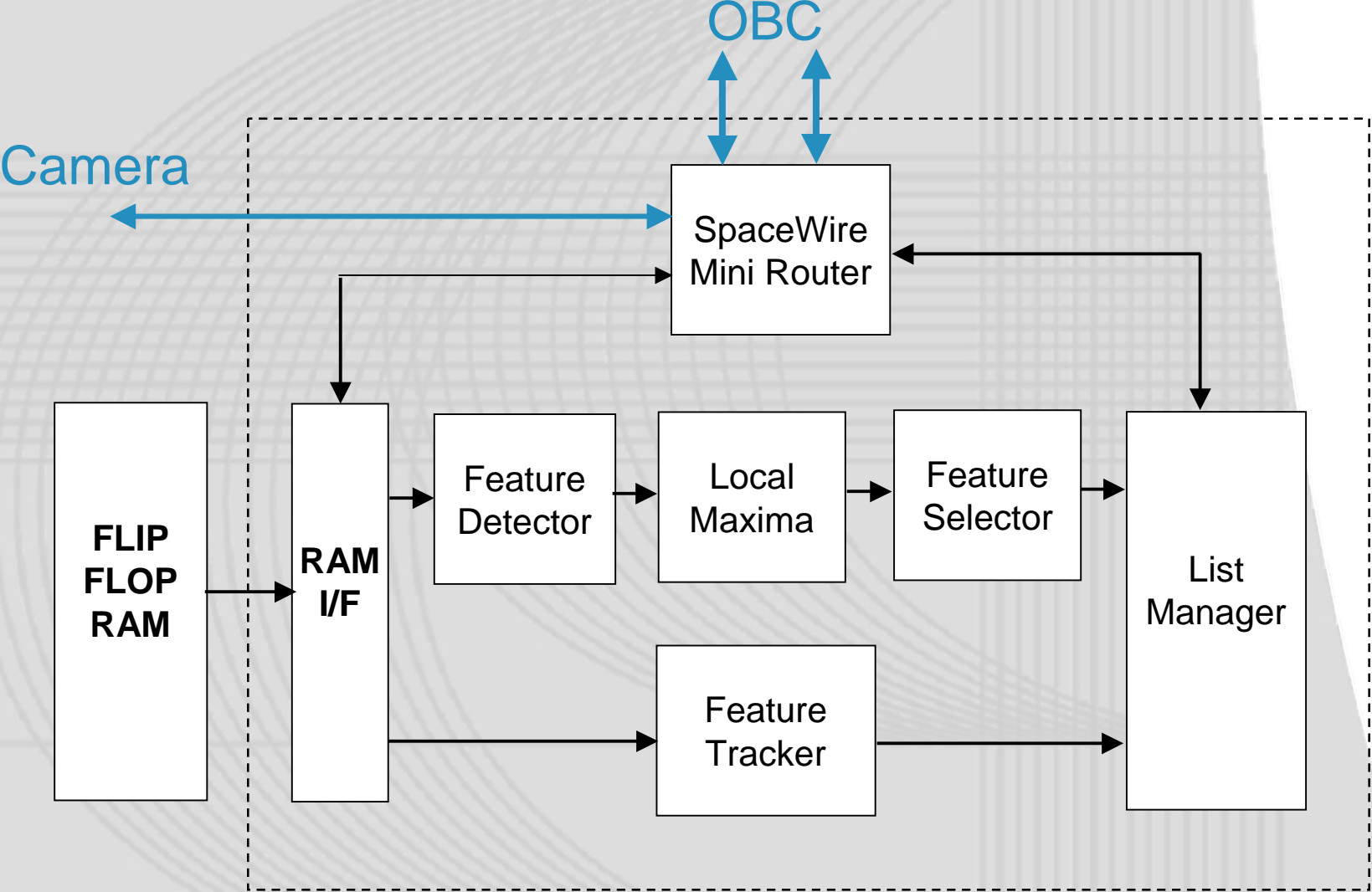
# Communication between OBC FEIC camera

---



- The SpaceWire link has been chosen since :
  - It is an ESA standard
  - VHDL IP cores are available
  - It allows to transfer commands/status and data (list and images) at a rate compliant with project needs.
- The FEIC implements a SpaceWire mini router in order to :
  - Reduce the number of cables between the OBC, the FEIC and the camera
  - Allow all the combination of exchanges
- In addition, the camera can work in two modes :
  - direct connection to OBC without FEIC via Space Wire
  - Connection to OBC via the same SpaceWire link through the mini router implemented in FEIC.
- Two SpaceWire link are implemented to link OBC and FEIC and one between FEIC and camera.

# FEIC : Functional Block Diagram



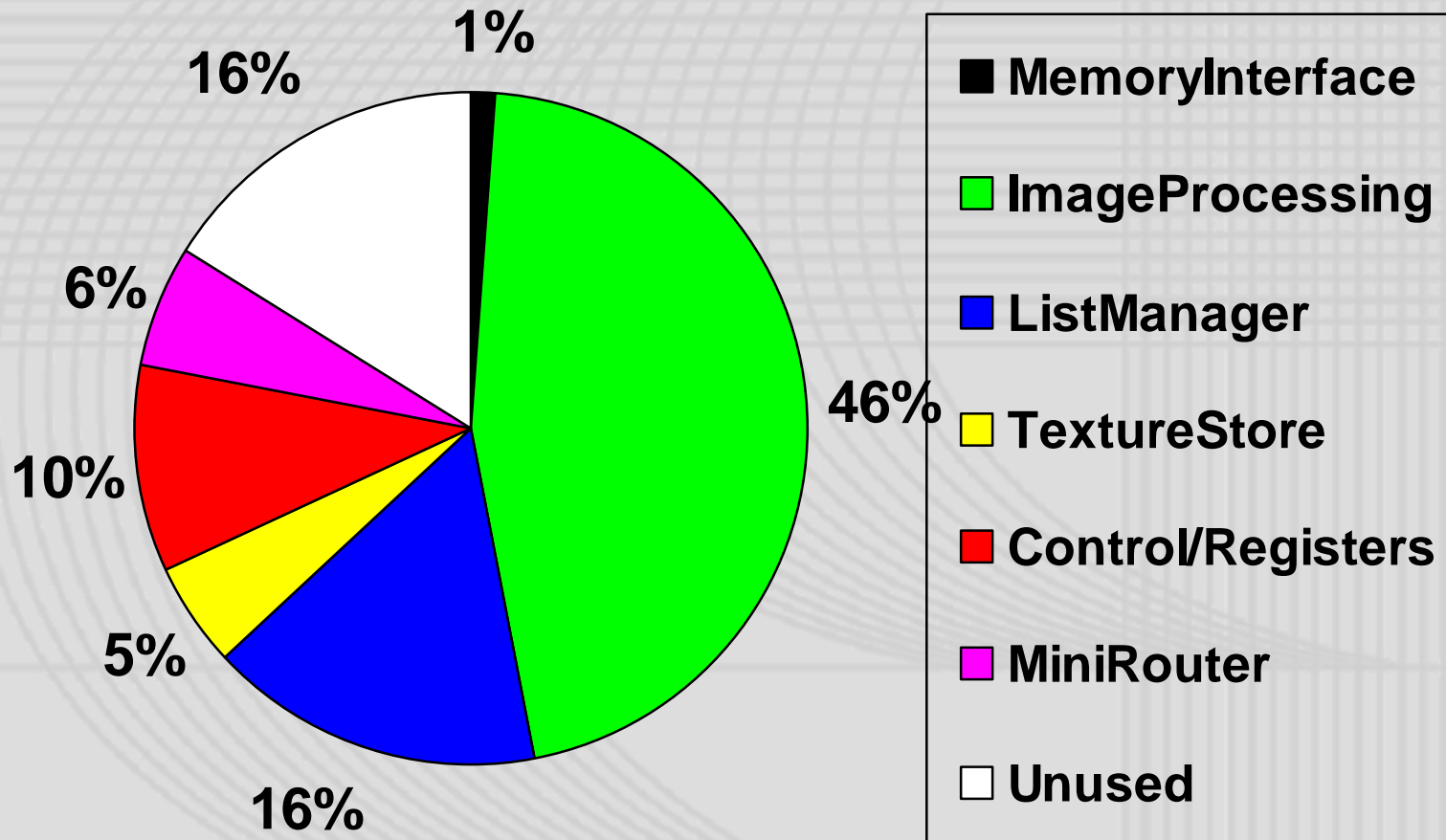


## FEIC Size

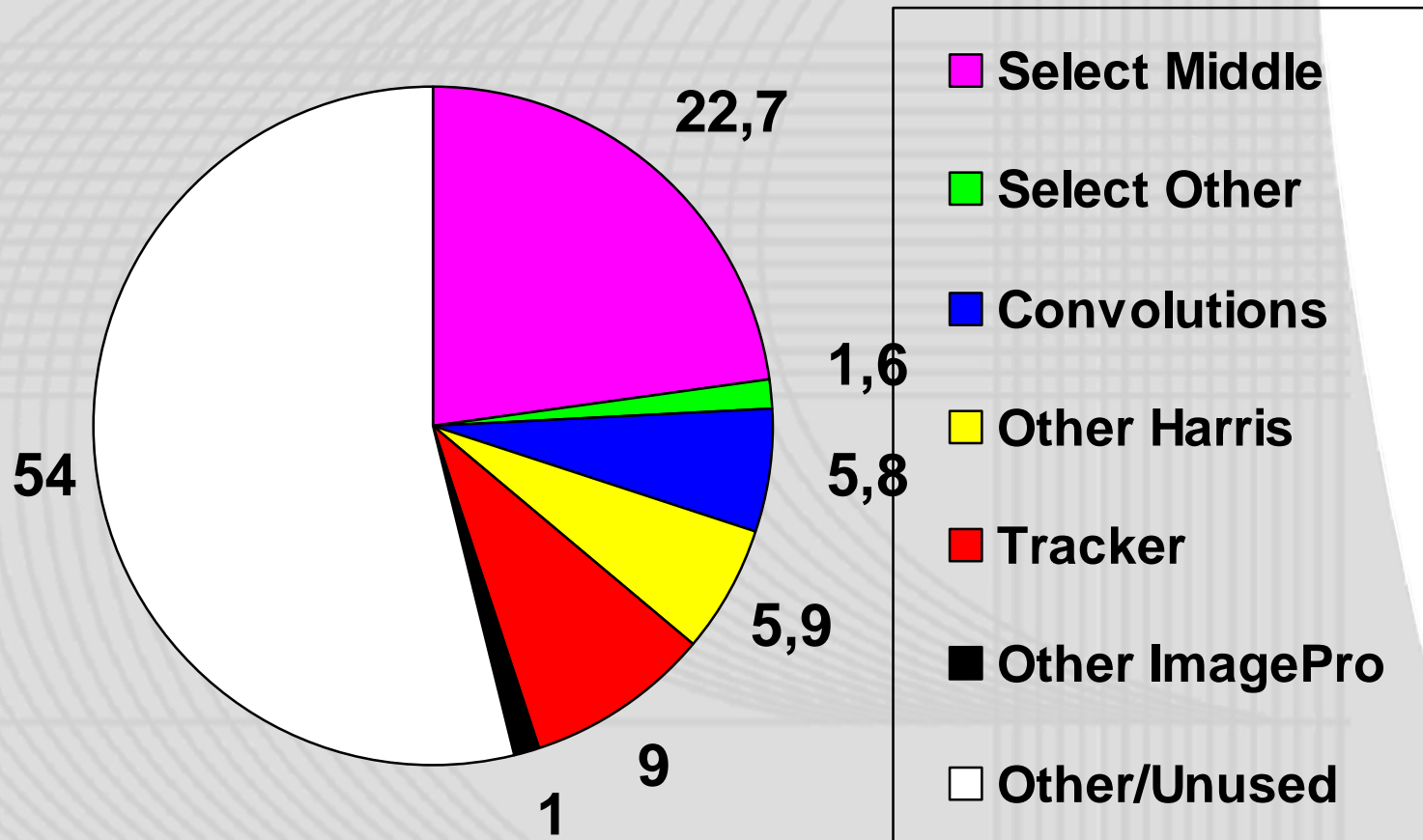
---

- FEIC do not use external RAM for image processing. Lines of the image or window for correlation are stored in internal RAM. Such Architecture is faster.
- FEIC is implemented in a XC2V6000 FPGA:
  - About 1 300 Kbit of internal RAM is used (52 %)
  - About 69 multipliers used (48 %)
  - Target is to adapt logic block usage to 80 % (by now 100%)
- About 55 % of the FPGA is used for image processing, 20 % for list management and 12 % for SpaceWire minirouter.

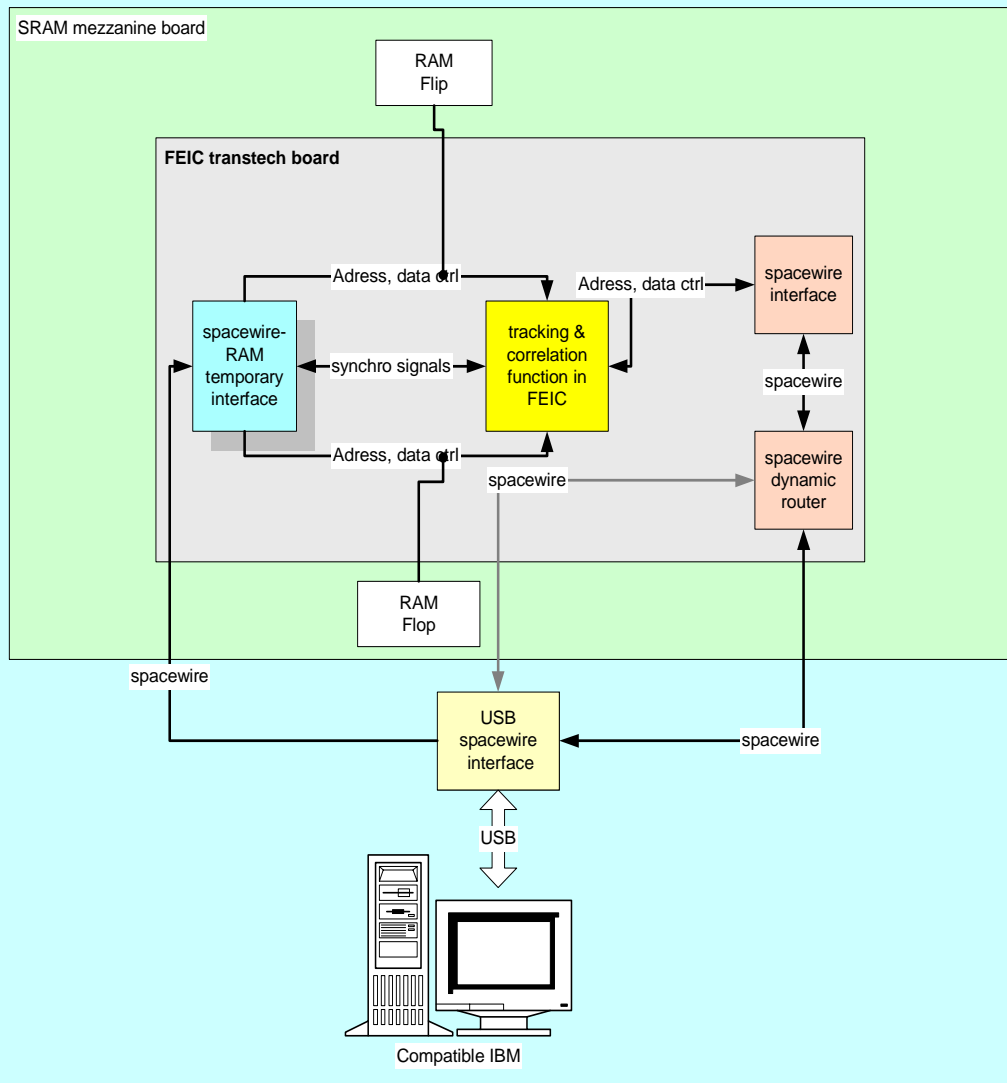
# FEIC SIZE : V8000 USAGE – Control



# FEIC SIZE : V8000 USAGE – Image Processing

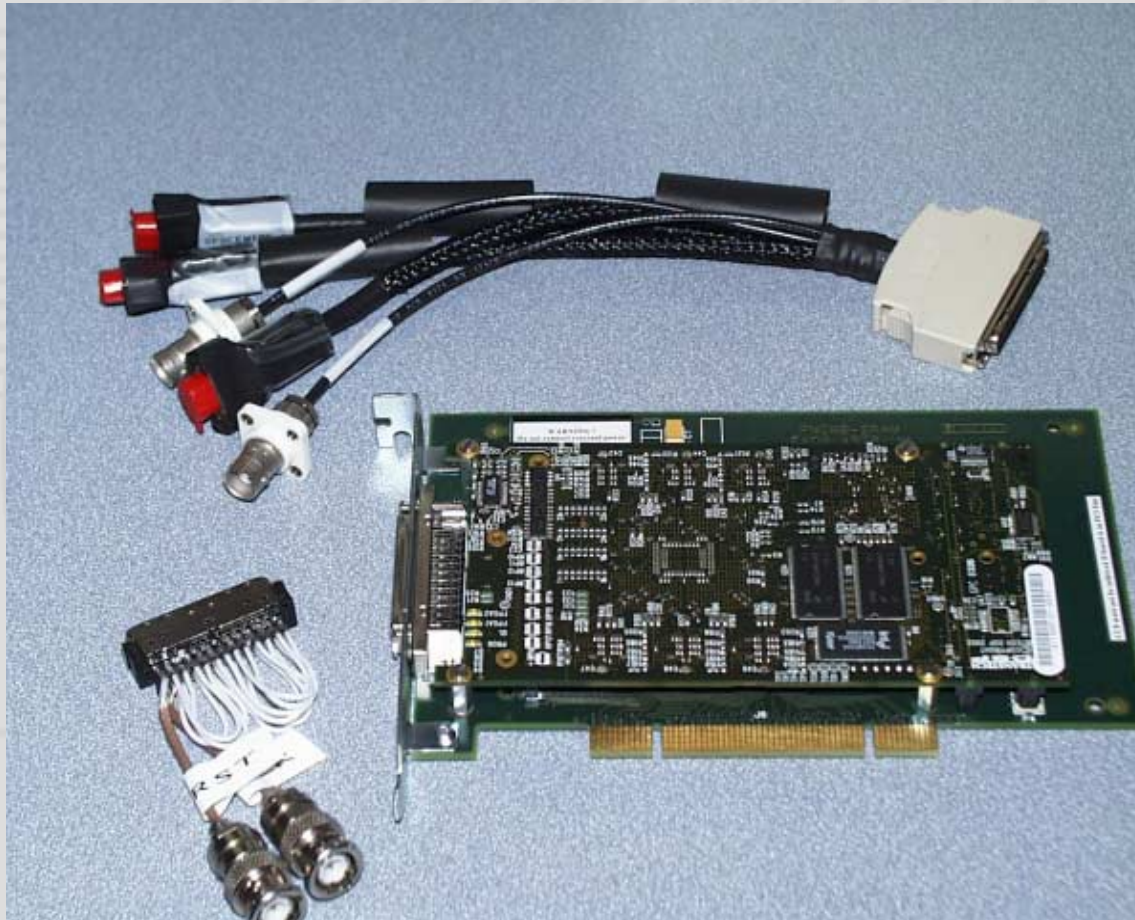


# FEIC Test bench



- Goal of test bench
  - Validation of FEIC behavior
- Delivered tools
  - USB spacewire interface
  - Linux drivers for USB Spacewire board.
  - Exemples of tests .
  - Pangu pictures will be downloaded through the spacewire link
- The test bench shall have been used for FEIC delivery acceptance test.

## FEIC SRAM board and transtech board



- The printed board set is
  - **PMC-FPGA02-6-4** includes FPGA XCV2V6000 speed grade 4
  - **SRAM board** includes flip flop buffer with **2 Mega 24 bits** words of SRAM
  - **3 spacewire links**
  - **1 input clock signal**
  - **1 input reset signal**
  - **Connector for test**

# Development Status

---



- The Navigation system has been functionally integrated and intensively tested on the VBNAT validation environment.
- The Image Processing functions have been prototyped in C++, and tested against image sequences in both open loop and closed loop.
- The FEIC is by now almost finished to be coded in VHDL. End of coding is planned for March 2004.
- The test board is ready for integration and the first tests have started at Hardware level with a partial version of the FEIC.
- Final integration of the FEIC FPGA with the camera demonstrator is planned for May 2004.

# FEIC functionalities and applications

---

- The design of the FEIC is based on a modular architecture sequentially performing:
  - **Evaluation of a criterion map:** to be used as a corner detector (NPAL need) but also edge or flat region detector (e.g. for Hazard mapping)
  - **Selection of feature points** sorted according to their Harris value. It also provides 4-neighboring criterion values (for subpixel interpolation or criterion strengthening)
  - **Tracking between stored feature points** (with local texture) and current image. Matching can be performed with older than one frame points in order to increase robustness. It also provides 8-neighboring correlation values (for subpixel interpolation or criterion strengthening)
- **One can make use of the high speed FEIC processing for each of this step in a wide set of potential applications.**

## FEIC applications

---

- **Tracking points in a structured scene:** parallax effects are compensated by high rates
- **Navigation aidings:**
  - For low relative depth and low speeds, navigation aidings are not necessary
  - Ability to proceed to an iterated estimation of a parametric motion model (e.g. for moving objects)
- **Data sources may be infrared, LIDAR maps, SAR elevation maps, DEM, ...**
- **Tracking can be performed without extraction, based on a predefined list**
- **Matching pairs of points among two consecutive sets of points** remains possible
- **Temporal filtering can be applied to tracks outside the FEIC** if the innovation remains lower than 1 pixel.



# FEIC applications

---

- Potential applications of the criterion map evaluation:
  - **Terrain classification** (flat, clustered, ...) for hazard mapping
  - Active vision: image acquisition scheduling can be adapted to image content
  - **Adaptive image processing** (adapted to local pixel information content)
- Potential applications of point extraction:
  - **Model fitting and point set matching** (e.g. pattern recognition)
  - **Global matching of images**
- Potential applications of tracking:
  - **2D image motion estimation** (online registration, mosaicking, image stabilisation, data fusion, super-resolution, attitude measurements, motion classification)
  - **3D navigation** (NPAL project, piloting and visual servoing)
  - **3D reconstruction and Calibration**