

# CMOS Architectures and Circuits for High-Speed Decision-Making from Image Flows

Ángel Rodríguez-Vázquez, Rafael Domínguez-Castro, Francisco Jiménez-Garrido, Sergio Morillas, Juan Listán, Luis Alba, Cayetana Utrera, Rafael Romay and Fernando Medeiro  
AnaFocus (Innovaciones Microelectrónicas S.L.)  
Avda Isaac Newton, Pabellón de Italia, Planta çtico  
Parque Tecnológico Isla de la Cartuja  
41092-Sevilla (SPAIN)  
[angel.rodriguez-vazquez@anafocus.com](mailto:angel.rodriguez-vazquez@anafocus.com)

## ABSTRACT

We present architectures, CMOS circuits and CMOS chips to process image flows at very high speed. This is achieved by exploiting *bio-inspiration* and performing *processing* tasks in *parallel* manner and *concurrently* with image *acquisition*. A vision system is presented which makes decisions within sub-msec range. This is very well suited for defense and security applications requiring segmentation and tracking of rapidly moving objects.

## KEYWORD LIST

Vision Chips, Smart CMOS Sensors, Smart Cameras, Bio-Inspired Chips, High-Speed Image Processing

## 1. INTRODUCTION

CMOS technologies enable on-chip embedding of optical sensors with data conversion and processing circuitry, thus making possible to incorporate intelligence into optical imagers and eventually to construct vision systems by using CMOS chips. The term *vision* herein refers to the set of tasks to interpret the environment from the information contained into images. It involves signal acquisition, signal conditioning, and information extraction and processing. *Sensor intelligence* refers to the incorporation of processing capabilities into the sensor itself.

Different types of CMOS sensors with different sensory-pixel types and different levels of intelligence have been devised during the last few years. These CMOS devices are either targeted to replace CCDs in applications where smartness is important or to make optical sensing and eventually vision feasible for applications where compactness, power consumption and cost are important.

Most of these smart CMOS optical sensors employ a conventional architecture where *sensing* is physically separated from *processing* and processing is realized by using either PCs or DSPs (see Figure 1). In the conventional architecture of Figure 1 most of the intelligence is hence far from the sensory device. It means that all input data, most of which are *useless*, must be codified in digital form and processed. On the one hand, this fact stresses the system requirements regarding memory, computing resources, etc.; on the other, it causes a significant bottleneck in the data-flow.

Such way of processing is actually quite different from what is observed in *natural vision systems*, where processing happens already at the sensor (*the retina*), and the data are largely compressed as they travel from the retina up to the visual cortex. Also, processing in retinas is realized in *topographic* manner; i.e. through the concourse of structures which are *spatially distributed* into arrangements similar to those of the sensors and which operate *concurrently* with the sensors themselves.

These architectural concepts borrowed from nature, namely:

- realization of the processing tasks by an *early processing* step followed by a *post processing* step,
- incorporation of the early processing structures right at the sensor layer,
- concurrent sensing and processing operations through the usage of either topographic or quasi-topographic

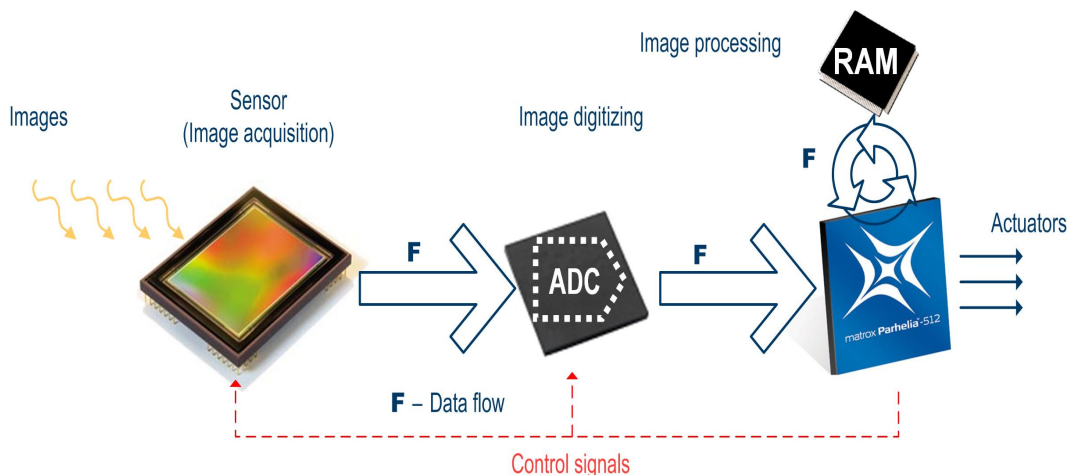


Figure 1: Conventional Smart Image Sensor Concept

early-processing architectures.

define basic attributes of the vision systems presented in this paper.

Some of these attributes, particularly the splitting of processing into pre-processing and post-processing, are also encountered in the smart camera Inca 311 from Philips. This camera embeds a *digital pre-processing* stage based on the so-called Xetal processor<sup>1</sup>. As a main difference to this approach, our approach realizes pre-processing by using *mixed-signal* circuits distributed in a pixel-wise area arrangement and embedded with the optical sensors. Because pre-processing operations are realized in truly parallel manner in the analog domain, power efficiency and processing speed of our system are both very large.

## 2. THE EYE-RIS VISION SYSTEM CONCEPT

Eye-RIS is a generic name used to denote the bio-inspired vision systems from AnaFocus. These systems are conceived for on-chip integration of all the structures needed for:

- *Capturing* (sensing) images
- *Enhancing* sensor operation, such as to enable high dynamic range acquisition
- Performing *spatial-temporal* processing
- *Extracting* and *interpreting* the information contained into images
- Supporting *decision-making* based on the outcome of that interpretation.

The Eye-RIS are *general-purpose*, fully-programmable hardware-software vision systems. They are complemented with a software layer and furnished with a library of image processing functions which are the basic instructions for algorithm development.

Three generations of these system have been already devised (Eye-RIS v1.0, v1.1 and v1.2) in a road-map towards single chip implementation (Eye-RIS v2). All these generations follow the architectural concepts depicted in Figure 2.

The main difference between the concept in Figure 2 and the conventional one depicted in Figure 1 comes from the “*retina-like*” structure placed at the front-end in Figure 2. This “*retina-like*” front-end stage is conceptually depicted as a *multi-layer* one. In practice it is a *multi-functional* structure where all the conceptual layers depicted in Figure 2 are actually realized on a common semiconductor substrate. These functions include:

- 2-D image sensing
- 2-D image processing. Programmable tasks in space (across the spatial pixel distribution) as well as in time are contemplated

- 2-D memorization of both analog and digital data.
- 2-D data-dependent task scheduling.
- Control and timing.
- Addressing and buffering the core cells.
- Input/output.
- Storage of user-selectable instructions (programs) to control the execution of operation sequences.
- Storage of user-selectable programming parameter configurations.

Note from Figure 2 that the front-end largely reduces the amount of data (from  $\mathbf{F}$  to  $\mathbf{f}$ ) which must first be codified into digital representations and then processed. At this early processing stage many useless data are hence discarded through processing and only the relevant ones are kept for ulterior processing. Quite on the contrary, in the conventional architecture of Figure 1 the whole data amount  $\mathbf{F}$  must be codified and processed. This reduction of data supports the rationale for advantages of the Eye-RIS vision system architecture.

In order to quantify the advantages let us calculate the *latency time* needed for the system to react in response to an event happening in the image. In the case of Figure 1,

$$t_{\text{LAT}}^{\text{conv}} = t_{\text{acq}} + N \times R \times t_{\text{A/D}} + N \times t_{\text{comp}} + t_{\text{proc}} \quad (1)$$

Where  $N$  is the number of pixels in the image,  $R$  is the number of bits employed for coding each pixel value,  $t_{\text{acq}}$  is the time required for the sensors to acquire the input scene,  $t_{\text{A/D}}$  is the per-bit conversion time,  $t_{\text{comp}}$  is the time needed to compare the new image with a previous one stored in memory as to detect any change, and  $t_{\text{proc}}$  is the time needed to understand the nature of the change and hence prompt a reaction. Although the exact value of the latency time above may significantly change from one case to another, let us consider for reference purposes that most conventional systems produce values in the range of 1/30 to 1/50 sec.

Similar calculations for Figure 2 yield,

$$t_{\text{LAT}}^{\text{Eye-RIS}} = t_{\text{acq}} + M \times R \times t_{\text{A/D}} + t_{\text{comp}} + t_{\text{proc}} \quad (2)$$

Where  $M$  denotes the reduced number of data obtained after early processing. Comparing the latency times for figures

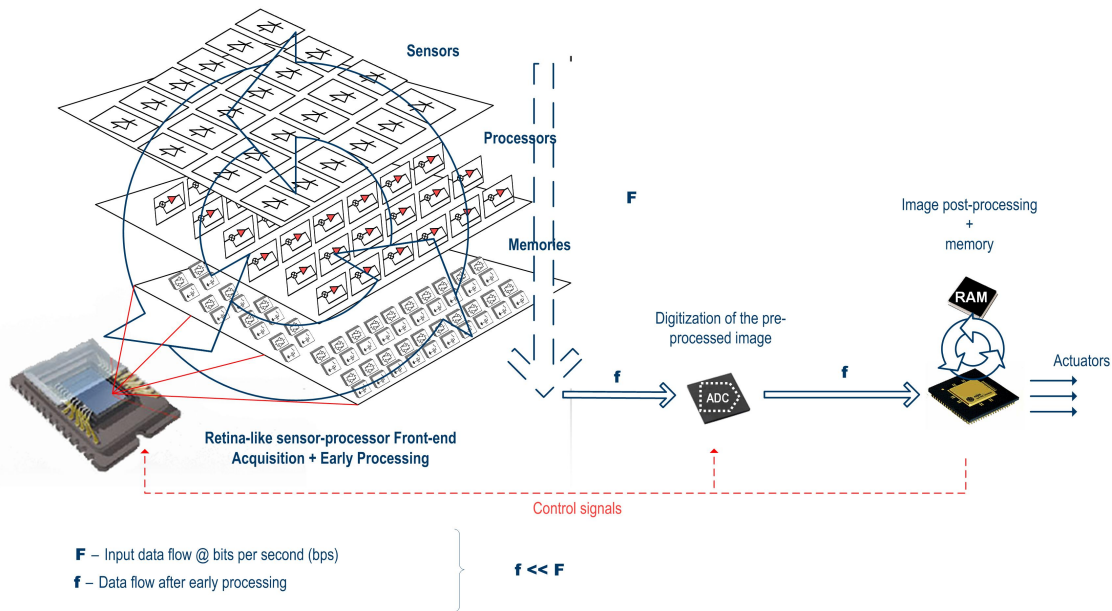


Figure 2: Eye-RIS system conceptual architecture with multi-functional retina-like front-end

yield,

$$t_{\text{LAT}}^{\text{conv}} - t_{\text{LAT}}^{\text{Eye-RIS}} = (N - M) \times R \times t_{\text{A/D}} + (N - 1) \times t_{\text{comp}} \quad (3)$$

Since typically it is  $N \gg 1$ , the equation above can be simplified as,

$$t_{\text{LAT}}^{\text{conv}} - t_{\text{LAT}}^{\text{Eye-RIS}} \approx (N - M) \times R \times t_{\text{A/D}} + N \times t_{\text{comp}} \quad (4)$$

Also, since in most cases the number of changes to be tracked will be small, we can assume that  $M \ll N$  and hence further simplify the equation above into,

$$t_{\text{LAT}}^{\text{conv}} - t_{\text{LAT}}^{\text{Eye-RIS}} \approx N \times (R \times t_{\text{A/D}} + t_{\text{comp}}) \quad (5)$$

It shows that the time saving reported by the Eye-RIS system approximately equals the sum of the time needed for conversion and the time needed in the conventional architecture for pixel-wise comparison. This time saving enables Eye-RIS system be employed for applications that require on-line operation with rapidly changing scenes. For instance, in a traffic collision at 80Km/h, the time lag for a standard driver to hit the steering wheel is 12,5m/s. In order to properly estimate the distance from the driver to the wheel as to control the triggering of the air-bag, vision systems must acquire-and-process at rates higher than 500 frames/sec which is difficult to achieve with conventional systems but which are intrinsic to the operation of the Eye-RIS vision systems.

Besides on-line operation, the data reduction featured by the Eye-RIS system relaxes the computational demands on the pots-processing structures and hence the complexity and power consumption of these structures.

Figure 3 shows a conceptual block diagram of one instance of the so-called Eye-RIS system family, namely the Eye-RIS v1.2. Its basic functional features include:

- **Early processing.** Front-end, retina-like sensor-processor. Particularly, current Eye-RIS systems employ the so-called Q-Eye front-end sensor-processor which will be described in the next section.
- Post processing:
  - 32-bit RISC uP at 70MHz – realized on a FPGA.
  - 32Mb SDRAM for program and image/data storage and 4Mb Flash for FPGA configuration.
- Interfaces:
  - 2 Serial Programming Interface (SPI).
  - UART: General purpose RS232 port.
  - 16 general-purpose 3.3V TTL I/Os.

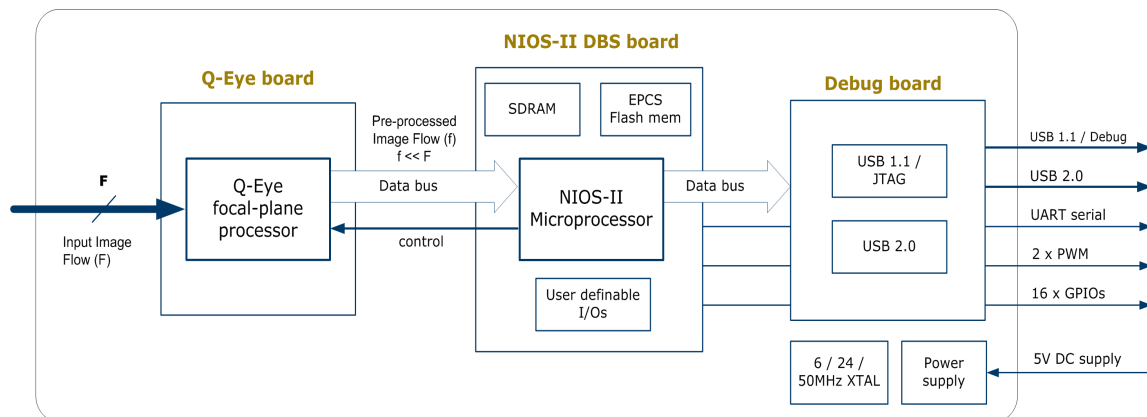


Figure 3: Conceptual block diagram of the Eye-RIS v1.2

- USB 1.1 Interface for JTAG control.
- USB 2.0 Interface for high-speed image I/O.
- System tools:
  - Eye-RIS ADK (Application Development Kit), an Eclipse-based development environment including all the tools needed for programming Eye-RIS, namely: project manager, code editor, C/C++ compiler, assembler and linker, source-level debugger and etc.
  - Image-processing library including basic routines such as: point-to-point operations, spatio-temporal filtering operations, morphological operations, statistical operations, blob analysis, etc.

### 3. THE Q-EYE CHIP

Figure 4 is a block diagram of a pixel of the Q-Eye chip. Basic analog processing operations among pixels are *linear*

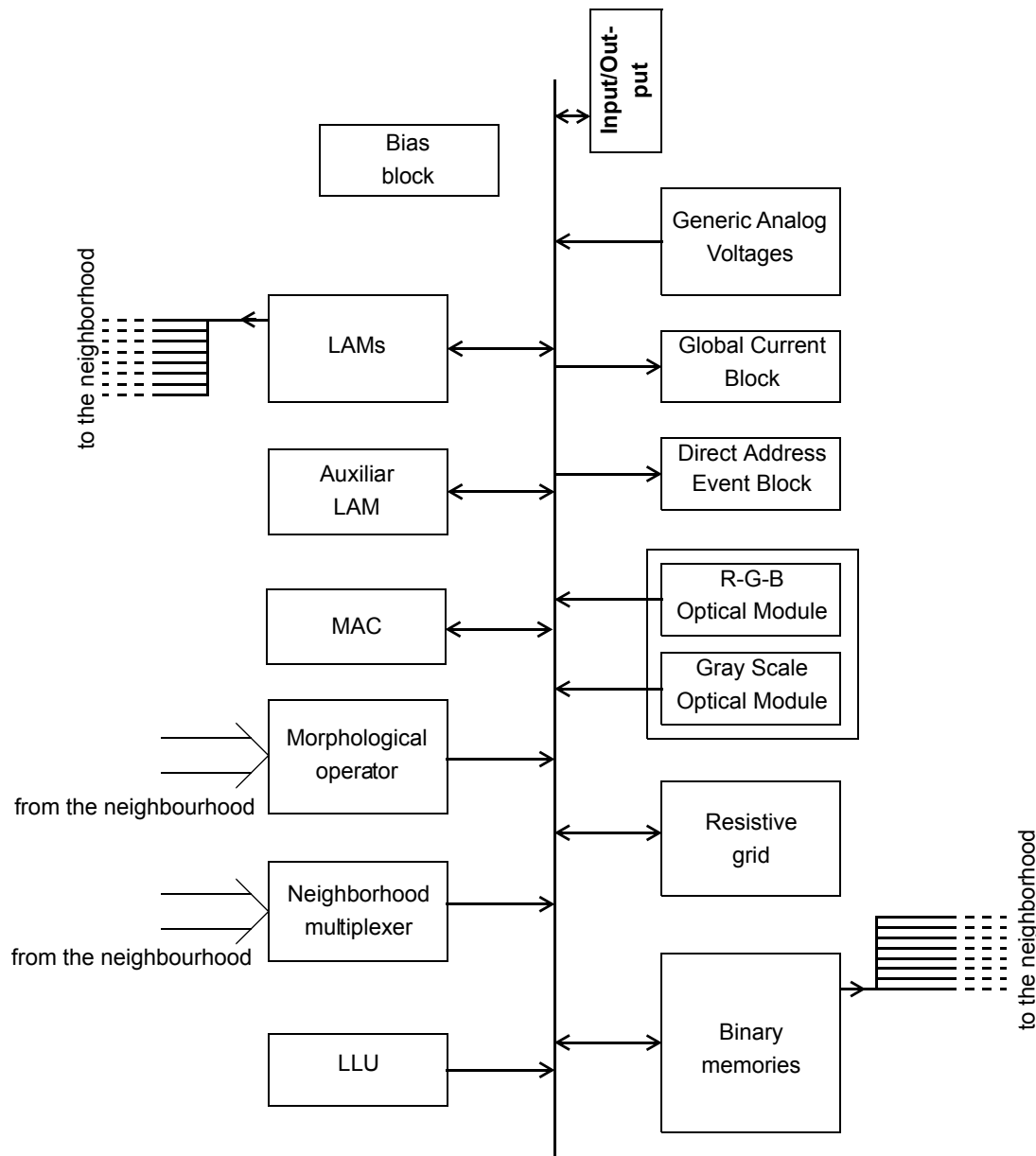


Figure 4: Block diagram for Q-eye pixel.

*convolutions* with programmable masks. However, the Q-Eye does not employ transconductance multipliers (as it happens in the previous similar chips) but a Multiplier-Accumulator Circuit unit (MAC) which processes neighbour pixels into an *algorithmic sequence*. Despite this sequential operation, computation times are similar to those obtained for other similar chips<sup>2-6</sup> since no calibration of the transconductors is needed<sup>7</sup>.

The area saving reported by the absence of spatially replicated structures (i.e. the transconductance multipliers employed for the linear convolutions) enables the incorporation at the Q-Eye pixel of functions which are not found in other similar chips<sup>2-6</sup>. These include:

- A pattern matching block to perform fast *morphological* operations on binary images and which complements the Local Logic Unit already found in the other similar chips<sup>2-6</sup>.
- An additional bank of analog memories to allow: 1) shifting of grey-scale and binary images through an analog multiplexer and 2) swapping between analog memories.
- A circuitry for analog thresholding.
- Three additional sensors for *colour* RGB sensing.

Besides these functions the Q-Eye array includes a *resistive grid* with programmable diffusion time.

Figure 5 shows a floor-plan of the Q-Eye chip. The external interface of the Q-Eye is completely digital and synchronous. It is composed of a 32-bit data bus for image I/O and two additional buses, namely a 10-bits data bus and 12-bits address bus. These latter buses are employed to program a digital control system which contains 256 control words of 60-bit and individual register for analog references and miscellaneous configuration. This system controls the array of processing-sensing cells, on the one hand, and the I/O control unit which handles all basic I/O process, on the other. The I/O interface can operate in three modes:

- loading-downloading of grey-scale images,
- loading-downloading of binary images and
- address-event mode.

Grey-scale values are coded into digital form by on chip 8-bits AD Flash converters, and decoded by on chip 8-bits DA resistors string converters.

To the purpose of improved *power management*, and hence reduced power consumption, most of the processing blocks in the Q-Eye and the analog reference buffers used for biasing have independent power up/down signals. Also, the operation speed of most blocks is programmable. Thus, the chip can be tuned to process either very high frame rates or low frame rates with optimum power consumption for each configuration.

*Robustness enhancement* is achieved through improved calibration techniques. In previous chips, offsets were stored into analog memories which experienced significant degradation specially at high temperatures. Instead, offsets in the Q-Eye chip are stored in static (non volatile) digital memories. Automatic calibration in the Q-Eye is performed by dedicated state machines which control in-loop A/D converters. Also, a temperature sensor and a temperature controlled correction loop are embedded in the Q-Eye to preclude the impact of junction temperature increases onto optical sensors and analog memories.

Main features of the Q-Eye chip are listed below:

- 176 x 144 cell array with 29.1 $\mu$ m x 29.1 $\mu$ m area per cell (cell density of 1,180 cells/mm<sup>2</sup>).
- Monochrome or color RGB (1 multi-mode grey-scale pixel plus 3 RGB pixels per cell).
- High-speed non-rolling electronic shutter. Programmable exposure time (controlling step-down to 20ns).
- Sensitivity above 1.0V/lux-sec at 550nm (with microlenses).
- Fill factor above 50% (with microlenses).
- Frame rate above 10,000 fps.
- 4 + 1 (two banks) high-retention analog and 4 binary memories.

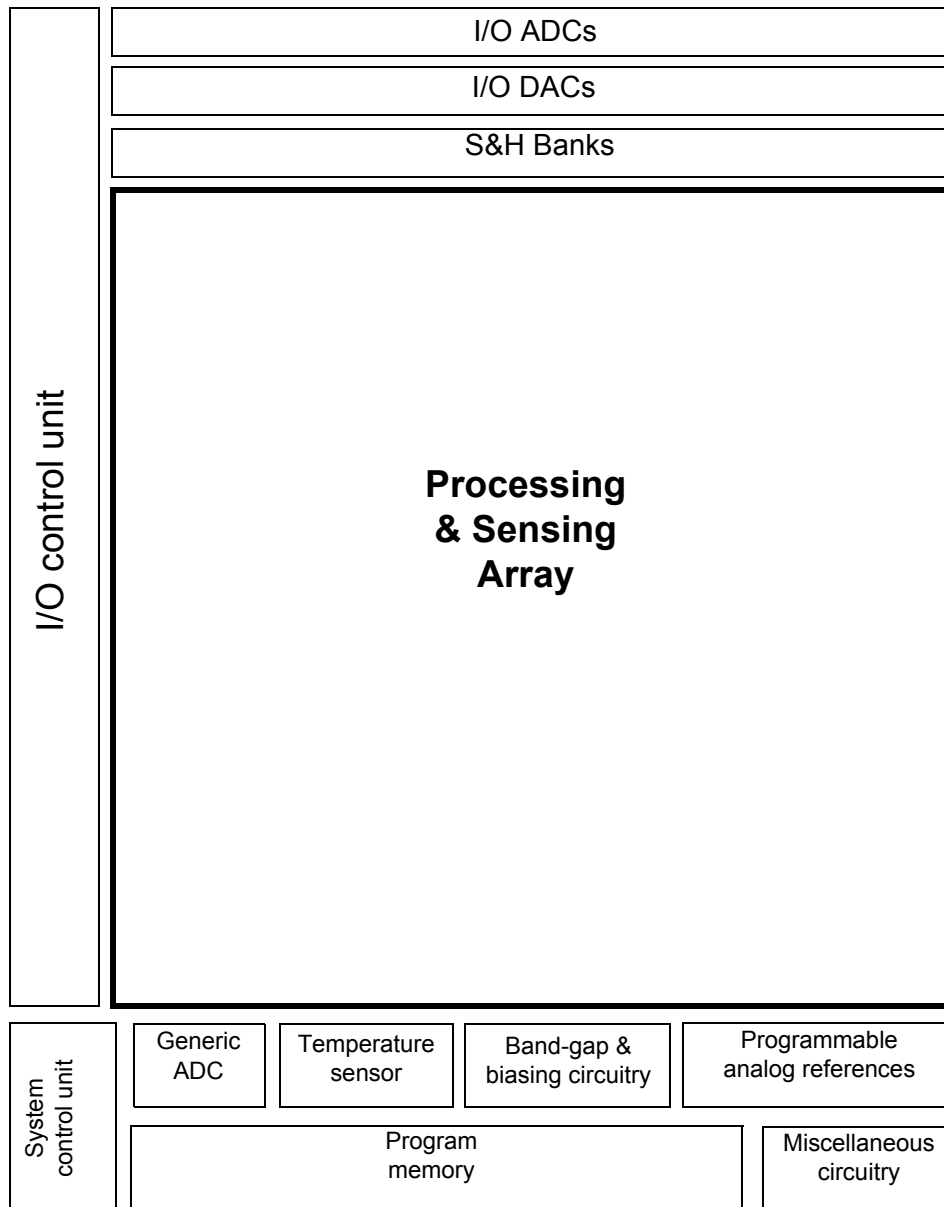


Figure 5: Block diagram of the Q-Eye chip

- Analog multiplexer for image shifting.
- Analog MAC unit.
- Programmable, 3 x 3 neighborhood pattern matching with 1/0/d.n.c. pattern definition (fast morphological functions).
- Programmable local logical unit (LUT table).
- Resistive grid for controllable image smoothing.
- 37.5mm<sup>2</sup> die area.
- 0.18μm 1.8V (core), 3.3V (I/O) CMOS technology.
- 50MHz clock frequency.
- < 100mW typical power consumption with 300mW peak during grey-scale image I/Os.
- Binary and analog image I/Os.

- On-chip bank of 4 ADCs and 4 DACs (8-bit@50MHz) for grey-scale image I/Os.

#### 4. THE EYE\_RIS SYSTEM IN OPERATION

The Eye-RIS vision systems are conceived to enable vision for applications where *compactness*, *cost*, *energy consumption efficiency* and *operation speed* define major targets. Since the Eye-RIS systems are general-purpose platforms they can be software-programmed for a large variety of applications including:

Demonstrations representative of the use of the Eye-RIS systems for these application are found at the AnaFocus web page ([www.anafocus.com](http://www.anafocus.com)). Figure 6 and Figure 7 illustrate the operation of the retina-like front-end, the Q-Eye chip. The two set of pictures at the top in Figure 6 show two different high dynamic range (HDR) images acquired in linear integration mode and HDR mode, respectively. In the latter case, HDR acquisition is achieved by processing right at the pixel level utilizing an algorithm based on the *well capacity adjustment* technique.

Images (c) to (h) in Figure 6 show inputs and outcome, respectively for different linear and nonlinear diffusion processes realized by using the on-chip embedded resistive grid whose parameters (mainly the spatial band-width of the diffusion process) can be controlled by the user. The results of performing low-pass, high-pass and band-pass spatial filtering on the input image Figure 6(c) are shown in Figure 6(d) (e) and (f), respectively. Figure 6(g) shows the output of a masked diffusion process (bottom figure) the mask being the binary figure at the top.

Figure 7 illustrate on the multi-functional capabilities of the Q-Eye chip by showing the input and output sequence of a Sobel filtering process (Figure 7(a)) and the input and outputs of the extraction of geometrical features from a binary image. Specifically, from left to right: the input binary image, the result of eliminating the single, isolated points, the borders of the latter image and the centroids of the same image. Note that all the described operations are realized directly at the pixel level, and simultaneously in all the pixels of an image, leading to extremely low processing times and consumed power.

#### 5. CONCLUSIONS

While vision in living beings handles a significant percentage of the information needed to interact with the environment, the use of vision in machines is limited due to very high cost/performance ratio. The Eye-RIS vision system overcomes this drawback by employing a bio-inspired architecture which can be realized at low cost in single chip form and which is capable of high-speed on-line operation.

#### 6. REFERENCES

- [1] Kleihorst R.P. et al., "Xetal: A Low-power High-Performance Smart Camera Processor". *Proc. of the 2001 IEEE Int. Symposium on Circuits and Systems*, Vol. 5, 215 - 218 (2001).
- [2] Carmona Galán R., [*Analysis and Design of Mixed-Signal Chips for Real-Time Image Processing*], PhD dissertation-University of Seville, (2002).
- [3] Liñán G., [*Design of Programmable Mixed-Signal Low-Power Consumption Chips for Vision Systems*], PhD dissertation-University of Seville, (2002).
- [4] Domínguez-Castro R. et al., "A 0.8 $\mu$ m CMOS 2-D Programmable Mixed-Signal Focal-Plane Array Processor with On-Chip Binary Imaging and Instructions Storage", *IEEE J. Solid-State Circuits*, 1013-1026, (1997).
- [5] Liñán G. et al., "ACE4k: An Analog I/O 64x64 Visual Microprocessor Chip with 7-bit Analog Accuracy", *Int. Journal of Circuit Theory and Applications*, Vol. 30, 89-116, (2002).
- [6] Liñán G. et al., "A 1000 FPS at 128 x 128 Vision Processor UIT 8-Bit Digitized I/O", *IEEE Journal of Solid-State Circuits*, Vol. 39, 1044-1055, (2004).
- [7] Rodríguez-Vázquez A. et al., "Mismatch-Induced Trade-Offs and Scalability of Analog Preprocessing Visual Microprocessor", *Analog Integrated Circuits and Signal Processing*, Vol. 37, 73-83, (2003).



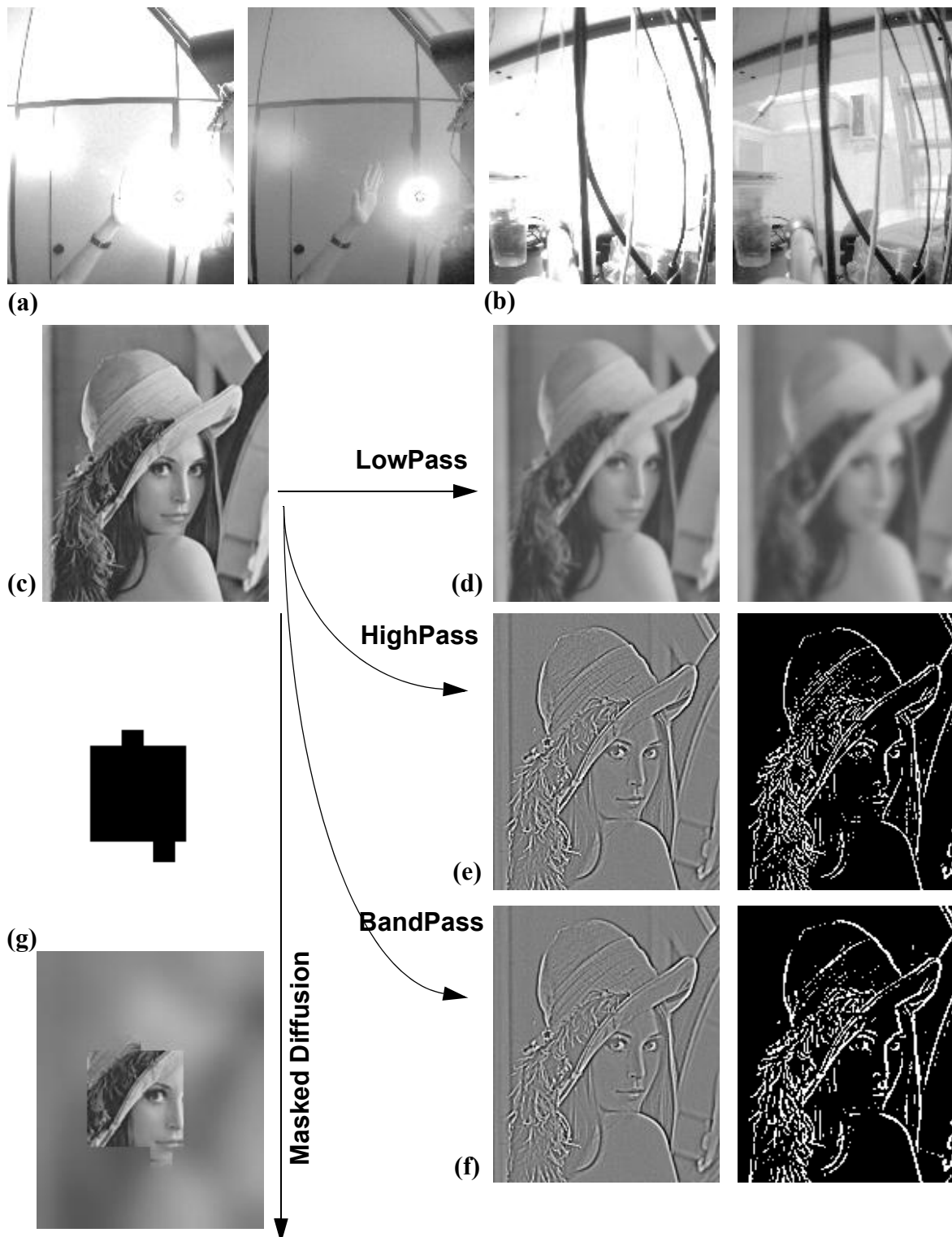
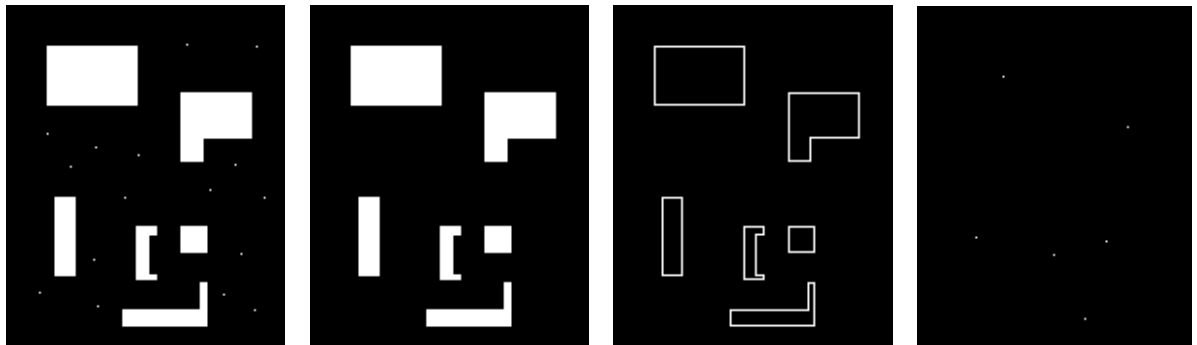


Figure 6: The Q-Eye chip in operation: (a) (b) Capturing HDR images in linear and HDR modes; (c) Input image for diffusive spatial filtering; (d) Outcome of lowpass filtering; (e) Outcome of highpass filtering; (f) Outcome of bandpass filtering; (g) Binary mask and outcome of a masked diffusion.



(a)

Sobel Filtering



(b)

Figure 7: The Q-Eye chip in operation: (a) Input and outcome of a Sobel filtering process; (b) Extraction of geometrical features from a binary image.