

Design for MOSIS Educational Program (Research)

A Transform Imager IC

Prepared by: Ryan Robucci and Paul Hasler

Institution: Georgia Institute of Technology

Date of Submission: Sept 24, 2004

A Transform Imager IC

Abstract

A transform imager that performs computation at the pixel plane is presented in this proposal. This imager is capable of programmable matrix operations on any image. Full matrix or block matrix operations can be implemented using this architecture. Each pixel is composed of a photodiode sensor element and a multiplier. The imager operates in the subthreshold region, and hence has extremely low power consumption. The architecture uses floating-gate technology to store the matrix coefficients on-chip. The imager is fully programmable for implementing any arbitrary block transform on any image being placed on the imager. The imager has a fill factor of 30 percent and is comparable to an Active Pixel Sensor (APS) imager.

Matrix Transform Imager Architecture

Fig. 1 shows the block diagram of the proposed block Matrix Transform Imager Architecture (MATIA). This approach allows for retina and higher-level bio-inspired computation in a programmable architecture that still possesses high fill-factor pixels characteristic of APS imagers. If the incoming voltages represent functions in time, particularly transform bases like sine and cosine, then computations analogous to matrix image transforms can be performed. The output is a continuous stream of each row of the transformed image, repeated at a desired frame rate.

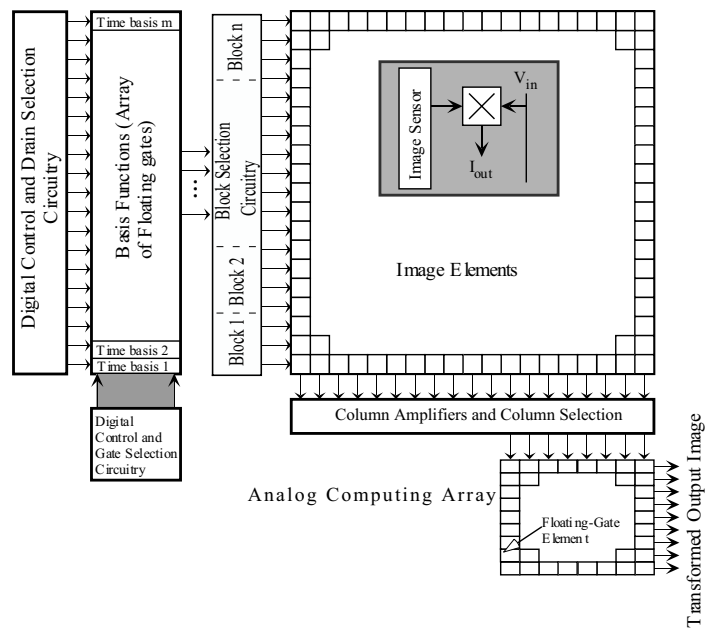


Fig 1. Block diagram of MATIA

This imaging architecture is made possible largely by advancements in analog floating-gate circuit technology and its application [22, 30]. These circuits have the added advantage that they can be built in standard CMOS or double-poly CMOS processes. In this approach, the floating-gate circuits store and reproduce arbitrary analog waveforms for image transforms, allow for correction factors to account for average device mismatch, as well as perform matrix-vector computations.

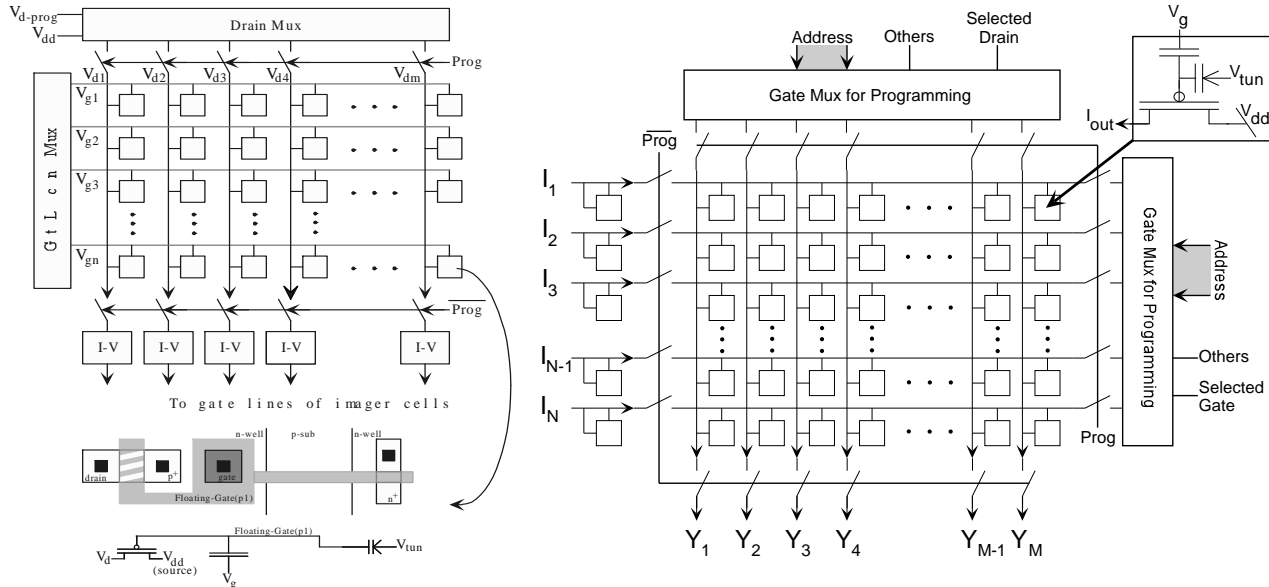


Fig 2. Floating gate systems: (a) bias generation circuit, (b) multiplication array (ACA)

The basis functions for the various block matrix transforms are stored on-chip using floating gates. Each floating gate can be individually programmed to store one coefficient. Peripheral digital control allows for individual floating gates to be selected and programmed to any arbitrary current for any gate voltage. The mechanisms involved in this programming phase are electron tunneling and hot electron injection. In the transform phase, the drains of the floating gates are connected to I-V converters that convert the respective currents to the bias voltages required for the transform. The imager array is designed such that multiplications can occur at the pixel level. The corresponding output is then placed in a parallel fashion to the Analog Computing Array (ACA) for the second matrix multiplication. The ACA consists of an array of floating gate for further multiplication. For the second multiplication floating gates are used instead of pFETs to reduce area and also perform arbitrary multiplications. These floating gates can be of minimum size ($W/L=6/4$) and still can be programmed to multiply incoming currents by any arbitrary factor depending on the transform being performed.

MATIA is both modular and programmable making it ideal for image data-flow computations. This architecture's scalability makes it feasible to compute large-scale, digital-camera resolution images. Furthermore, the image processing architecture computes on the image plane allowing for data reduction that is compatible with machine vision and biological modeling.

Floating gate systems

MATIA requires using fundamental floating-gate circuits/systems for the generation of arbitrary on-chip waveforms and for analog matrix-vector multiplication. Other floating-gate circuits are used to further enhance the circuit and signal processing performance of these systems.

We use floating-gate circuit elements to store and to generate the arbitrary basis functions needed for the matrix-vector multiplication on the imager. Fig. 2 (a) shows the top-level view of

our basis generation circuitry. This system operates in both operation (basis generation) mode and programming mode. In operation mode, we have an array of stored values that are output in sequence. In programming mode, we can easily reconfigure this circuitry on the outside edges for programming. This approach is compatible with our standard programming structure and algorithm. Each floating gate can be isolated for programming using the peripheral control circuits. The multiplication block is shown in Fig. 2 (b). We can perform vector matrix computations using our existing analog computing array (ACA) technology based upon floating-gate circuits. Using the output image stream this system will compute a transposed matrix transform.

Estimated project size and packaging

We plan to design a 1024x1024 imager with peripheral control circuitry and column parallel ADC structures for image acquisition. The chip design and layout are already near completion. The estimated size of the die required will be 6.5mmx6.5mm. We expect a need for a PGA132L package.

Simulation plans

We have simulated the pixel for speed and performance using SpectreS. We have also simulated and tested the peripheral digital control circuitry, used for floating-gate programming. Many similar circuits have been fabricated and tested using the AMI 0.5 μm technology so we have reason to believe that these should also function in TSMC 0.25 μm technology. The column parallel ADC structures have been simulated for functionality.

Test and characterization

We plan to make the individual blocks testable. We will also fabricate individual floating gates and characterize them for speed and performance. Digital circuits have been individually simulated. The pixel array will be tested using an imager test bench. This test bench allows for any image to be focused onto chip. A printed circuit board (PCB), similar to one currently used, will be designed for this chip. The chip would be interfaced to a field programmable gate array (FPGA), which will provide the necessary digital control for rapid programming of floating gates as well as image acquisition.

We plan to test the imager for image acquisition and for block matrix image transforms. The system as a whole will be tested for speed. We will perform noise and distortion analysis of the system.