A Processing Element for an Analogue SIMD Vision Chip

Piotr Dudek*

Abstract – **This paper describes an analogue processing element (APE) suitable for high-density image sensor/processor array integrated circuits. The design trade-offs between area, power consumption, speed and accuracy are discussed and the architecture of the APE is presented. The design follows a switched-current "analogue microprocessor" approach while the implementation of arithmetic operations is simplified by introducing a register-based current division method. The circuit has been implemented in a 0.35μm single-poly 3-metal layer CMOS technology. The APE measures below 50μm×50μm, operates with a 1 MHz clock and consumes less than 12μW of power (simulation results).**

1. INTRODUCTION

Many computer vision applications, particularly where cost, size and power dissipation of the system are important, can benefit from integrating photodetection and image processing onto a single integrated circuit. Exploiting the fact that most of the low-level image processing tasks exhibit pixel-level parallelism (identical operations are performed for each pixel in the image), efficient image processing systems can be designed by associating image processing circuitry with each pixel in the image sensor array, as illustrated in Fig.1.

The processing is sometimes implemented using dedicated analogue circuits, however, due to the very limited area available to each processor it is difficult to implement a large number of different operations in a single chip. Yet, typical computer vision algorithms usually require several low-level image processing operations to be executed on a single image. Furthermore, from the applications perspective, it is beneficial to ensure complete programmability of the system so that various algorithms can be implemented using the same hardware. These goals can be achieved if the processing elements (PEs) in the array are software-programmable entities, so that the hardware comprised in the limited area designated for a PE is re-used during the execution of the algorithm. The processor-per-pixel array can be then operated as an SIMD (Single Instruction Multiple Data) parallel computer, where a single controller issues instructions that are simultaneously executed in all PEs in the array. Still though, the silicon area occupied by the PE has to be minimised if reasonable image resolutions are to be achieved.

Several vision chips have been recently proposed that implement the fine-grained processor-per-pixel SIMD array. A vision chip with bit-serial PEs is presented in [1]. Another vision chip, described in [2], is designed with particular attention to minimising the size of the PE, however, the functionality is somewhat compromised by limiting the memory associated with each PE. Another interesting approach is to combine the algorithmic programmability of the SIMD computer with the continuous-time operation of the Cellular Neural Network [3].

We suggested an "analogue microprocessor" approach to the design of compact and efficient processors [4]. We proposed the design of general-purpose analogue processing elements (APEs) implemented using switched-current analogue techniques and demonstrated the SCAMP vision chip, with APEs measuring below 100μm×100μm in a 0.6μm technology [5]. This compact size was achieved while speed, accuracy and power dissipation were still comparable or better than those of the alternative approaches.

In this paper we present the design of APEs implemented on the new SCAMP-2 vision chip, highlighting the differences between the present design and the design of the APEs implemented on our previous vision chip. First, the trade-offs between area, speed, accuracy and power dissipation are discussed and the APE specifications are presented. Next, the architecture of the APE is overviewed. Finally, circuit design issues are briefly discussed.

2. DESIGN TRADE-OFFS

The SCAMP-2 chip has been designed in a 0.35μm single-poly 3-metal layer CMOS technology. The major design goal was to achieve an APE size below 50μm×50μm, so that eventually a 128×128 array could be implemented on a 60mm^2 vision chip. A further design goal was to achieve low power

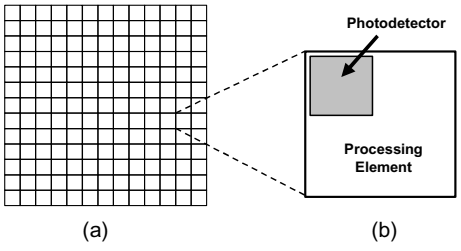

Figure 1: A general structure of the vision chip (a) image sensor/processor array, (b) single array element corresponding to one pixel.

* Department of Electrical Engineering and Electronics, University of Manchester Institute of Science and Technology (UMIST), PO Box 88, Manchester, M60 1QD, United Kingdom; e-mail: p.dudek@umist.ac.uk

consumption, while maintaining acceptable levels of accuracy and speed of operation.

When designing circuits, trade-offs between speed, accuracy and power dissipation have to be resolved. In our view, the operating speed of a single APE is not the most critical issue - the massively parallel operation of the processor array usually guarantees high computational performance and ensures that for most applications the useful image processing algorithms that can be performed on a single image frame are easily executed in real-time. Each APE on the previous version of the SCAMP chip was executing a moderate 2.5 MIPS (millions instructions per second). The chip executed the Sobel Edge Detection algorithm in 11.6μs and the Median Filtering algorithm in 61.6μs [5]. It was decided that the computational performance of the new APE should be equal to 1 MIPS. For most image processing algorithms this speed of operation should easily enable the achievement of processing speeds beyond 1000 frames/second,.

Power dissipation, on the other hand, becomes critical with the increased size of the array. It is also of paramount importance for portable, miniaturised, battery-operated applications. The present APE was designed to consume below 12μW of power at full operating speed. This means 0.2W per 128×128 array, operating at 16 GIPS (excluding I/O) - but it also has to be noted, that the power consumption in a particular application will depend on the algorithm and the frame rate, and so may be typically much lower.

A processing accuracy equivalent to approximately 5-bit or 6-bit integers is usually considered adequate for most early vision applications. When considering accuracy requirements for analogue systems, however, it is not sufficient to consider single-instruction accuracy only. This is due to the fact that each subsequent operation on the analogue processor (including simple data transfers) leads to an error accumulation effect. In the SCAMP chip, for example, a single storage instruction introduces a maximum error of only 0.6% (systematic error and noise), while the average difference between image processing results obtained by executing algorithms on the SCAMP chip, as compared with the "ideal" results computed numerically was approximately 2% [5]. Such accuracy should be acceptable for early vision applications and so it was decided to maintain this level of accuracy for the SCAMP-2 chip.

3. APE DESIGN

The overall architecture of the APE is shown in Fig.2. All of the APE components are connected to the analogue bus node. The data in the system is represented by current values, flowing in and out of the analogue bus (both positive and negative signals are possible). The APE is an analogue, sampled-data system. Its operation is akin to a datapath of a simple microprocessor, where data transfers and transformations are performed according to a "software" program – a sequence of instructions issued from the controller.

3.1 Registers

One of the most obvious trade-offs when designing an SIMD array involves deciding upon the amount of local memory (registers) associated with each processing element. In the present design we have decided to include nine registers (labelled A,B,C,D,H,K,P,Q and O). Each register can store a grey-scale pixel value or a variable, so assuming a 7-bits storage accuracy the local memory of the APE is equivalent to 63-bits of storage, which is more than is usually possible to implement on digital SIMD vision chips [1,2].

As registers, switched-current memory cells are used. The register cells employ the S^2I technique for error cancellation and their design is similar to those previously reported [5]. The register cells are dynamically configured using switch-control signals (represented by the short grey arrows in Fig.2), which

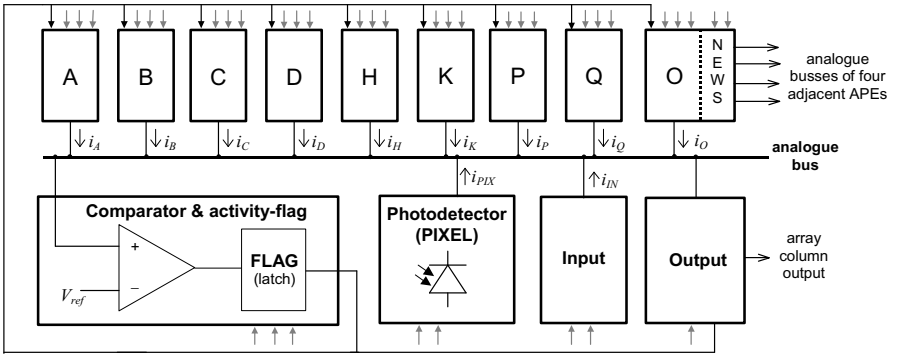

Figure 2: The schematic diagram of the APE.

are broadcast to the APEs in the SIMD array from the controller. These control signals can put a register in one of the three modes: 'idle', 'store' or 'readout'. When the register is 'idle', it is disconnected from the analogue bus (very high resistance) and its biasing current is switched off to reduce the power dissipation in the APE. When the register is configured to 'store', it presents a low resistance port to the analogue bus, while the current flowing into the register is sampled and stored inside the memory cell. When the register is configured to 'readout', it provides the previously stored current to the analogue bus, while presenting a high output resistance port.

It has to be noted that the register operation effectively changes the sign of the stored data value. For example, if we have a current i_B that has been previously stored in register B and then 'readout' this register and 'store' the current i_A in register A (all other registers 'idle', not contributing any current to the analogue bus node), then we get $i_A = -i_B$. This basic transfer operation is denoted as A←–B.

In practice, there will be a small error of the transfer operation, associated with noise, charge injection, clock feedthrough and output conductance effects in switched-current memory cells. The systematic, signal-dependent part of this error is easily cancelled-out algorithmically [4] while the signal-independent part and noise set the limit on the achievable accuracy of the processors.

3.2 Arithmetic operations

The ability to achieve a compact memory cell is one of the features of the switched-current APE. Another feature is the compact implementation of arithmetic operations, which does not require any additional circuitry. As we have seen in the previous section, the inversion operation is performed with each storage. Current summation can also be performed directly on the analogue bus, by reading-out the currents from several registers at the same time. For example, the operation A←–B+C+D is performed in a single clock cycle, in a manner similar to the register transfer, but now three registers (B, C and D) are configured to 'readout' while the register A is configured to 'store'. The currents i_B, i_C and i_D are summed on the analogue bus node and flow into the register A, thus $i_A = -(i_B+i_C+i_D)$

Another basic operation that can be performed directly in this analogue registers/bus system is current division. For example, the 'division-by-two' operation can be denoted as A+B←–C. This means that the current provided to the analogue bus by register C (configured to 'readout') is directed to registers A and B (both configured to 'store'). In effect, the current i_C is split into two equal parts, in the ideal case, so that the currents which are stored are equal to $i_A=i_B=-i_C/2$. In reality, a transistor mismatch in the registers introduces some error into the division operation, and care must be taken to size the transistors so that this error is at an acceptable level. It is also possible to cancel-out this error – an accurate division method for switched-current systems has been presented in [7] and it can be implemented algorithmically on the APE.

The ability to perform current divisions "in registers", as described above, means that the APE does not need to contain an explicit multiplier. The multiplier (which could be based on a programmable, binary-scaled current mirror as in [5]) would need to be quite large in order to achieve matching accuracy comparable with the accuracy of the storage cells. The multiplication coefficients in the APE are instead achieved by repeated current additions and divisions, as required. For typical values of coefficients used in image processing algorithms this scheme does not incur large time overhead, while the area-savings are significant. Consequently, it is possible to increase the number of general-purpose registers that can be used to perform division as well as other operations.

3.3 Activity-flag

An important feature of any SIMD array is the provision for local autonomy of processors. This is typically done by introducing an activity-flag in each processor. The activity-flag can be set/reset by conditional tests performed on local data. The instructions broadcast from the controller are executed only in those processors that have the activity-flag set. To implement these features, the APE contains a current comparator and the activity-flag register, built using a differential amplifier and a D-type latch, as shown in Fig.2.

The conditional tests are performed by enabling the comparator and configuring one or more registers for the 'readout' operation. The registers are designed so that the analogue bus voltage during the 'store' operation is maintained at a reference voltage level V_{REF}. During the conditional instruction, however, the analogue bus has a high resistance to ground (no registers are configured to 'store'). Therefore the voltage at the analogue bus node will rise or fall below V_{REF} depending on the sign of the total current stored in the registers selected to 'readout'. Consequently, comparing the analogue bus voltage with V_{REF} gives us the sign of the sum of currents stored in the selected registers. The logic level corresponding to this sign can be latched in the FLAG register.

The FLAG signal controls data transfers in the APE. Control logic in the registers ensures that the 'store' operation is only performed when the FLAG signal is in the logic high state. From an array programming perspective this means that those APEs for which the sign of the total current from the

registers selected for comparison was negative will become disabled and will not perform subsequent instructions. These APEs will remain disabled until a new comparison is made or until the FLAG register is explicitly set by the controller. This corresponds to executing a conditional branch such as: IF (A+B)>0 THEN … ENDIF

3.4 I/O circuits

The remaining building blocks of the APE, shown in Fig. 2, are associated with input/output functions.

Each APE can communicate with four neighbour APEs. To facilitate this, one of the registers (labelled O) can be directly connected to the analogue busses of the neighbour APEs via the NEWS (North, East, West, South) switch. Careful design is required to ensure that the communication register is matched to the other registers.

The analogue bus of the APE can be connected to the array column output line in response to the row-select signal, so that the current from any register can be read-out off-chip. The state of the FLAG register can also be read-out to the array column output line. The output circuit supports global array operations (analogue SUM and logic OR). The flexible readout architecture, designed for the SCAMP-2 chip to reduce the amount of data that will be transmitted to a next processing level in a computer vision system, has been described in more detail in [6].

An input current circuit (biased by a voltage controlled by an external D/A converter) is also included in each APE. In this way a particular current i_{IN}, set by the controller, can be input to the analogue bus node.

Each APE also contains a photodetector to provide parallel optical image input to the processor array. The photodetector can work in integration mode or in continuous-time mode and the current i_{PIX} corresponding to the pixel value (grey-scale level) can be read-out to the analogue bus.

4. PHYSICAL DESIGN

The APE has been designed in a 0.35μm 3-layer metal, single-poly CMOS process. The APE circuitry contains a total of 111 transistors. The total layout area is equal to 49.35μm × 49.35μm. The layout has been designed with particular attention to minimising capacitive coupling between various signals and sensitive high-impedance nodes at the gates of the storage transistors in the registers. The control signals are routed over the APE area, which is shielded by the power supply planes. This also protects the APE circuitry from the incident light, while the photodetector areas are fully exposed to increase the photosensitivity. The APE design has been verified by post-layout simulations.

The nominal biasing current in registers is equal to 1.7μA while the power supply voltage is equal to 2.5V. The biasing currents in all components of the APE are switched-off when these components do not take part in a particular instruction. Based on simulations, the overall power consumption in the APE operating at 1 MHz, executing a typical mix of instructions, is equal to 10.8μW. Additionally, power required to drive digital control signals is equal to approximately 1μW per APE.

5. CONCLUSIONS

The APE design presented in this paper combines low power dissipation and software-programmability, while being extremely compact. The basic arithmetic instructions are performed by exploiting inherent properties of the switched-current memory cells connected to a common analogue bus, without the need for any additional hardware. The basic parameters of the prototype APE implementation are: total power dissipation of 12μW, computational performance of 1MIPS and silicon area of 2436μm^2. The expected efficiency figures of merit for this design are therefore as follows: performance/area equal to 410 MIPS/mm^2 and performance/power dissipation equal to 83.3 GIPS/W.

The prototype SCAMP-2 chip, which includes a 39×49 array of APEs, together with biasing circuitry and the readout circuitry, has been submitted for fabrication. A subsequent fabrication of a 128×128 array vision chip is planned.

Acknowledgement: This work has been supported by the EPSRC, UK, grant no. GR/R52688/01

References

[1] M.Ishikawa et. al. “A CMOS Vision Chip with SIMD Processing Element…”, Proc. ISSCC’99, TP 12.2, 1999.

[2] F. Paillet, et. al. “Making the most of 15kλ^2 silicon area for a digital retina”, Proc. SPIE, Vol. 3410, AFPAEC 1998

[3] G. Liñán-Cembrano et al. “A Processing Element Architecture for High-Density Focal-Plane Analog Programmable Array Processors”, Proc. ISCAS 2002, May 2002, vol.III, pp.341-344.

[4] P.Dudek and P.J.Hicks, “A CMOS General-Purpose Sampled-Data Analog Processing Element”, IEEE Trans. on Circ. and Syst. II: Analog and Digital Signal Processing, vol. 47, no. 5, May 2000, pp. 467-473

[5] P.Dudek and P.J.Hicks, “An Analogue SIMD Focal Plane Processor Array”, Proc. ISCAS 2001, vol.IV, pp.490-493, May 2001.

[6] P.Dudek, “A Flexible Global Readout Architecture for an Analogue SIMD Vision Chip”, Proc. ISCAS 2003, vol.III, pp. 782-785, May 2003.

[7] J.-S. Wang and C.-L. Wey, “Accurate CMOS Switched-Current Divider Circuits”, Proc. ISCAS’98, vol I, pp.53-56, May 1998.