MODELING AND IMPLEMENTATION OF AN INTEGRATED PIXEL PROCESSING TILE FOR FOCAL PLANE SYSTEMS

A Dissertation
Presented to
The Academic Faculty

By

William Hugh Robinson, III

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in Electrical and Computer Engineering

Georgia Institute of Technology

August 2003

MODELING AND IMPLEMENTATION OF AN INTEGRATED PIXEL PROCESSING TILE FOR FOCAL PLANE SYSTEMS

Approved by:

Dr. D. Scott Wills, Advisor

Dr. Jeffery A. Davis

Dr. Sung Kyu Lim

Date Approved 8-13-2003

Dedicated in loving memory to
my great-grandmother, Mrs. Sallie Robinson,
who instilled the value of education
into her descendants,
William Sr., William Jr., and William III.

ACKNOWLEDGEMENT

I once wrote "knowledge is the ever-flowing river that quenches the thirst of mankind's inquisitive spirit." At the time, I only referred to academic knowledge in that metaphor. However, I have learned through revelation that God's gift of salvation is the true knowledge that mankind should seek (Eph. 2:8-9). This knowledge is the living water that quenches my inquisitive spirit (John 4:10-14). Therefore, I first acknowledge my Lord and Savior Jesus Christ, who is the author and finisher of my faith (Heb. 12:2). This journey towards a Ph.D. would not have been possible without my Lord's divine providence. Let Your will be done in my life.

As my academic advisor, Dr. D. Scott Wills has directed my development as a scholar and researcher. I am truly grateful for his wisdom, enthusiasm, and support over the years. Our shared passion for education has provided a solid foundation for our friendship. I look forward to a new dynamic in our relationship as I pursue my own academic career.

I would like to thank Dr. Jeffery A. Davis and Dr. Sung Kyu Lim for serving on my dissertation reading committee. I would also like to thank Dr. Gary S. May and Dr. Umakishore Ramachandran for serving on my dissertation defense committee. Their valuable insight has considerably improved the quality of this research.

I would like to thank Dr. Donna C. Llewellyn and Dr. Mark J. T. Smith for their valuable mentorship during my time at Georgia Tech. Each has significantly contributed

towards fostering my career in academia. I am thankful for the professional development you provided.

I have also benefited from a rich research environment during my graduate years at Georgia Tech, thanks to the former and current members of the PICA Research Group. When I first joined, observing other students in the group provided an example of what was required for a Ph.D. I am proud of the tremendous research accomplishments our group has made in computer architecture design. In particular, I would like to acknowledge Dr. Sek M. Chai and Dr. Antonio Gentile for their research and collaboration in the area of focal plane processing. I also thank you for setting a high mark for both presentation skills and PowerPoint creativity.

To my immediate family, my Mom, Mrs. Mae M. Robinson, my Dad, Mr. William H. Robinson, Jr., and my sister, Ms. Elisa C. Robinson, I want to express my sincere gratitude for your loving-kindness. I am truly blessed by your influence in my life. I thank my parents for being outstanding and passionate educators. They instilled within me the desire to pursue this advanced degree. I also thank my sister for the strong friendship we have developed over the years. I look forward to celebrating your educational goals in the near future.

Although I do not have an older brother, I regard my cousin, Mr. Jonathan Lofton, in that capacity. I have always looked to him as a role model to emulate. Because of his achievements, a thirteen-year-old boy sitting in the Alexander Memorial Coliseum was inspired to attend Georgia Tech.

I would like to acknowledge Mr. Gregory E. Triplett for his friendship and research collaboration during our quest for a Ph.D. at Georgia Tech. I never would have

imagined such a fruitful relationship when we first met during freshman week at Florida Agricultural & Mechanical University (FAMU) in 1991. Our talents complement each other, which helps us both to improve.

I also would like to acknowledge Mr. Phillip A. Harp, whose friendship also began at FAMU in 1991. The transition to graduate school at Georgia Tech in 1996 was eased by starting with a close friend. I appreciate your words of encouragement during this journey towards a Ph.D.

I am grateful to the Georgia Tech Black Graduate Students Association (BGSA) for its infrastructure of unity, support, and academic guidance. I am appreciative of all the past and current members who have touched my life. In particular, I acknowledge Mr. Derrick S. Brown for his mentorship when I started graduate school. I have carried his advice until this day, and have done my best to pass along that wisdom to others.

I acknowledge "scholarship, manly deeds, and love for all mankind," which are the three aims of my fraternity, Alpha Phi Alpha Fraternity, Incorporated. This dissertation is a testament to upholding the light.

I would like to thank Rev. Craig L. Oliver, Sr., and the Elizabeth Baptist Church family for providing a spiritual home during my time in Atlanta.

Finally, I want to acknowledge my remaining family and friends too numerous to name for love, support, and prayer. I appreciate you all!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS OR ABBREVIATIONS

ADC	Analog-to-Digital Conversion
APS	Active Pixel Sensor
CMOS	Complementary Metal Oxide Semiconductor
DCT	Discrete Cosine Transform
DSP	Digital Signal Processor
NSIP	Near Sensor Image Processing
PDA	Portable Data Assistant
PE	Processing Element
PPE	Pixels per Processing Element
PVLSAR	Programmable and Versatile Large Size Artificial Retina
S^3PE	Simple and Smart Sensory Processing Elements
SED	Statistical Experimental Design
SIMD	Single-Instruction Multiple-Data
SIMPil	SIMD Pixel Processor
SNR	Signal-to-Noise Ratio
TeSA	Technology Scenario Analyzer
TI	Texas Instruments
VLSI	Very Large Scale Integration

SUMMARY

Technology improvements to monolithically integrate CMOS sensors, analog-to-digital conversion circuitry, and digital processing offer the potential for a highly efficient imaging system for multimedia processing. However, this requires an understanding of the connections among applications, architecture, and technology. This dissertation presents a study of system-level design issues to develop monolithic focal plane architectures. Research contributions include workload characterization of key front-end imaging applications to determine hardware design constraints for the architecture, development of models to predict performance and efficiency of system components (photodetectors, analog-to-digital converters, data storage, and digital processing), and evaluation of design tradeoffs to improve overall system performance. The additional hardware in the focal plane architecture expands application versatility compared to current pixel-level processors. Analysis shows the focal plane architecture can deliver up to 130x the performance of a traditional DSP architecture for key front-end applications. In addition, implementations of the focal plane architecture achieve up to 81x higher area efficiency and up to 11x higher energy efficiency. These benefits are significant because front-end applications can consume over half of the required processing time, allowing that workload to be offloaded to the more efficient focal plane architecture. Further developments in computer architecture, microelectronics, and signal processing will enable low-cost, portable imaging systems.

CHAPTER I

INTRODUCTION

1.1 Motivation

The demand for mobile productivity with image and video applications has sparked the development of highly integrated processing systems. Low-cost, embedded image processing chips appear in numerous portable products, including cellular phones, video and still cameras, and portable data assistants. These systems must deliver high performance with efficient use of resources such as area and energy. The development of these systems can leverage the abundant data parallelism in the application domain as well as improved fabrication techniques for both image sensors and integrated circuits. Key issues for future portable imaging devices include:

- Designing a suitable architecture for an imaging system
- Integrating the system components monolithically
- Utilizing the limited system resources to maintain portability

1.1.1 Architecture design

Numerous architectures have been developed for image processing systems using both analog and digital techniques [1-3]. Efficient handling of the two-dimensional image data is a common issue among these designs. Because of the high spatial locality of image processing applications, the Single-Instruction Multiple-Data (SIMD) [4] construct

provides a method to increase performance through data parallelism in both general-purpose processors [5] and fine-grain parallel processing arrays [6, 7]. However, new architectures must account for the increased impact of interconnect as the limiting factor in the design [8].

1.1.2 System integration

The monolithic integration of CMOS sensors with additional functionality can provide a system-on-a-chip solution for imaging systems [9]. The recent trend is away from the historical use of analog processing circuitry and towards digital pixels [10]. With smaller technology feature sizes, more digital circuitry can be incorporated at the sensor site for analog-to-digital conversion, data storage, and digital processing. However, technology scaling presents a challenge to CMOS imagers, and enhancements to the standard CMOS process are required to maintain image sensing [11, 12]. These modifications hold the promise to increase CMOS image sensor performance [13].

1.1.3 Resource utilization

Higher efficiency ratings (performance per resource) are desired for portable image processing systems because of technology limitations such as interconnect wiring density and heat extraction. Portability limits system resources such as area and power, and overall system performance is affected by design tradeoffs. Silicon area allocation is a significant issue because in single-level VLSI, the photodiode, the analog signal conditioning, the analog-to-digital converter, the memory, and the digital processing core compete for silicon area. Power consumption is also vital because the architecture must

provide a meaningful battery life. Area and energy (power) efficiency are important figures of merit in this design space [14].

1.2 Problem Statement

Incorporating analog and digital components to process pixels at the detection site requires an understanding of the connections among applications, architecture, and technology. The imaging application suite is a rich opportunity to design a targeted architecture that leverages the abundant data parallelism. Improvements in technology dramatically impact architecture design and implementation. The challenge is not maximum performance, but sufficiency with minimal resource cost.

1.3 Related work

An imaging system includes both data acquisition and data processing. Traditional architectures focus on the data processing functionality. Focal plane architectures integrate data acquisition and data processing functionality to utilize the data parallelism inherent within the image applications.

1.3.1 Traditional architectures

The abundant data parallelism inherent in image applications has motivated the development of multimedia extensions for general-purpose processors to improve performance on these applications. Using the Single-Instruction Multiple-Data (SIMD) processing model described by Flynn [4], these extensions supply the processor's functional units with subword data in parallel. General-purpose architectures utilizing

multimedia extensions include Intel's MMX™ [15] and SSE™ [16], Hewlett-Packard's MAX2™ for the PA-RISC™ architecture [17], Sun Microsystems' VIS™ for the SPARC [18], MIPS's MDMX™ [19], the Alpha's MVI™ [20], and Motorola's ALTIVEC™ for the PowerPC™ [21]. Alternatively, digital signal processors (DSPs) such as the TI TMS320C80 or TMS320C6000 families follow a more specialized approach [22]. However, both microprocessors and DSPs require high overhead in both area and power to execute image processing applications. In addition, the most severe limits of these architectures will not be imposed by transistors performing computing functions but by interconnection networks that perform signal communication, clock distribution, and power distribution functions [23].

1.3.2 Focal plane architectures

A logical approach for focal plane processing incorporates analog computational ability into the sensor device. Analog design methods, such as the Silicon Retina, have been presented to implement focal plane arrays [24]. This has led to the development of neuromorphic vision sensors for early image processing [25]. However, the advantages of analog techniques, such as low power and small area, break down as CMOS technology scales [26]. In addition, analog architectures are typically non-programmable, requiring multiple designs to implement different functionality.

Several digital architectures, such as the Near Sensor Image Processing (NSIP) [27], the Programmable and Versatile Large Size Artificial Retina (PVLSAR) [28], and the Simple and Smart Sensory Processing Elements (S^3PE) [29], follow the pixel-level model to perform early image processing in the focal plane. However, these architectures

utilize bit-serial processing techniques with limited memory, which can either restrict processing to binary images or require multiple cycles to perform a single instruction on data words. This may prevent the implementation of some early image processing algorithms.

1.4 Research approach

The research presented in this dissertation addresses system-level design issues to monolithically integrate photodetectors, analog-to-digital converters, data storage, and digital processing into focal plane architectures. These components are combined into a single processing element that is tiled to form a SIMD focal plane processor array with nearest-neighbor communication, capable of executing front-end image applications. The outline of the approach is as follows:

- Characterize and evaluate front-end image processing applications
- Develop component models relating performance and resource usage
- Assess technology scaling for focal plane architecture implementation
- Quantify design tradeoffs in focal plane architectures

An important characteristic of the architecture is the processing granularity, also known as the number of pixels per processing element (PPE). Key performance and cost metrics for focal plane architectures include execution time, system throughput, chip area, power consumption, area-time efficiency, area efficiency, and energy efficiency.

The main task for the system level study of integrated focal plane architectures is developing the framework for interrelationships among components. Although design

expertise exists for data acquisition [30], analog-to-digital conversion [31], data storage [32], and digital architecture [33], the impact of design choices on other components is not as clear. Modeling and simulation can project the behavior of focal plane architecture implementations across technology generations.

Results will show the performance and efficiency benefits gained by integrating digital processing with data acquisition and will identify feasible design configurations and technology for implementation.

1.4.1 Performance analysis of front-end image processing applications

Pixel-level image processing architectures can leverage the abundant data parallelism to provide high performance embedded systems. However, sufficient hardware complexity and data storage must be available to broaden the suite of image applications.

A focal plane architectural simulator is used to determine performance metrics for convolution, discrete cosine transform (DCT), edge detection, and median filtering. Sustained throughput, measured in billion operations per second, is determined for the target system while varying the number of pixels per processing element (PPE).

The applications also have a direct impact on the physical implementation. The hardware must satisfy both the data storage and the data precision constraints to execute the applications. Also, the hardware must support the required instruction set to execute each application.

Results show the sustained throughput of the focal plane architecture exceeds the reported specification of the TI DSP chips up to 130x for 1 PPE. The execution time is

also reduced. The 1 PPE implementation operating at 10MHz executes an imaging sequence 4x faster than the TI TMS320C6411 chip operating at 300MHz. In addition, execution times from TI DSP benchmarks are dependent on the image size. Therefore they increase with larger resolutions, while the execution time of the focal plane architecture is independent of the image size. This makes the focal plane architecture an excellent candidate for an embedded processor for front-end imaging tasks.

1.4.2 Efficiency analysis of focal plane architectures

Future portable imaging products will benefit from the monolithic integration of photodetectors, analog-to-digital converters, digital processing, and data storage. However, the goal is not strictly building the system with the highest performance, but delivering the system with the required performance at the lowest cost.

Software tools provide a means to evaluate potential architectural configurations to determine bounds for system feasibility as well as good candidates for implementation. A focal plane architectural simulator [34] is used to determine performance metrics for median filtering, convolution, and inside edge detection, which corresponds to an imaging sequence of: (1) noise removal, (2) smoothing, and (3) segmentation. Different architecture implementations vary the number of pixels per processing element (PPE). Component area models based upon physical layout are developed to project silicon area for the architecture using different fabrication technologies. A Technology Scenario Analyzer (TeSA) [35] projects power consumption using different fabrication technologies, incorporating parameters from The International Technology Roadmap for Semiconductors [36, 37].

Results show that, despite a significant difference in clock frequency, implementations of the focal plane architecture perform well compared to a traditional DSP architecture while demonstrating higher ratings in area and energy efficiency across fabrication technologies. Performance is increased by 130x when using a focal plane architecture with 1 PPE. In addition, this implementation achieves 81x higher area efficiency and 11x higher energy efficiency when compared to traditional TI DSP chips. However, more aggressive technology shows diminishing returns for area and power usage, indicating that less expensive technologies can be used to implement the system. Because data acquisition and data storage have the highest cost in terms of silicon area, the architecture must address those components to be effective.

1.4.3 Statistical experimental design for photodetector modeling

Imaging chips are being developed that convert the image to the digital domain and process the spatially parallel data within the image plane. However, with monolithic integration, the photodetector, the analog-to-digital converter (ADC), the digital processing core, and the memory compete for silicon area. Modeling the integrated optoelectronics can provide insight for design choices.

A regression model was developed from the theoretical physical implementation of a photodiode. Key input parameters selected for designing the photodiode included the area, integration time, acceptor density, donor density, temperature, and reverse bias. A 2^6 full-factorial experimental design was used to explore the broad design space.

Results show that the regression model accurately depicts the behavior of the SNR response as a function of the input parameters. The two most significant factors,

photodiode area and integration time, provide a helpful design tradeoff in the context of the digital pixel. A 1ms increase in the integration time reduces the photodiode area by $212\mu m^2$ while maintaining SNR. This relationship significantly impacts the design of a focal plane processor by trading time for limited silicon resources.

1.5 Contribution Summary

The contributions of this dissertation relate to the study of system-level design issues to monolithically integrate photodetectors, analog-to-digital converters, data storage, and digital processing into focal plane architectures. The contributions are outlined in three categories.

Performance analysis of front-end image processing applications

- Workload characterization of image processing sequence
  - Implemented front-end image processing application suite
  - Determined data storage requirements to execute applications
  - Evaluated functional units for digital pixel
- Key results from analysis
  - Determined required number of register to execute selected applications equals [(2 • PPE) + 7] with a 12-bit datapath
  - Data parallelism in the selected applications enables utilization exceeding 78%
  - Communication cost for nearest-neighbor applications (3 x 3 window) is less than 8% while communication cost for larger windows (8 x 8) exceeds 12%
  - 1 PPE implementation operating at 10MHz executes an imaging sequence 4x faster than the TI TMS320C6411 chip operating at 300MHz.

Efficiency analysis of focal plane architectures

- Evaluation of focal plane architectures with 1 PPE, 4 PPE, and 16 PPE
  - Developed component models for area projections
  - Developed framework for system analysis of component models
  - Projected power consumption for different technology generations
  - Evaluated area-time efficiency, area efficiency, and energy efficiency of focal plane architectures
- Key results from analysis
  - Sustained throughput is increased up to 130x versus TI Benchmarks using a focal plane architecture
  - Focal plane architectures have up to 81x higher area efficiency and up to 11x higher energy efficiency compared to TI DSP chips
  - Area and power constraints for portability are feasible using 180nm fabrication technologies and beyond to implement a focal plane architecture

Statistical experimental design for photodetector modeling

- Analysis of CMOS photodiode signal-to-noise ratio (SNR)
  - Developed photodiode SNR performance model based upon physical implementation
  - Performed statistical experimental design to create regression model
  - Validated regression model using analysis of variance (ANOVA)

- Key results from analysis
  - Regression model reduces complexity for determining photodiode SNR
  - Significant parameters, in order of importance, are diode area, integration time, donor density, reverse bias, and then the combination of reverse bias and donor density
  - Photodiode area and integration time are the most dominant parameters, with a correlation to SNR of 0.794 and 0.587 respectively
  - Each 1ms increase in integration time reduces photodiode area by 212μm^2 while maintaining constant SNR

1.6 Dissertation Outline

Chapter II presents a characterization of the image application suite for focal plane architectures. A background of image processing applications and architectures is presented. The targeted application suite and the evaluation methodology are described. Performance results are provided with a comparison versus TI DSP benchmarks.

Chapter III presents the analysis of efficiency metrics for focal plane architectures. A background of image processing architectures is presented. Performance, area, and power are projected for focal plane architecture implementations. Efficiency comparisons versus TI DSP chips are provided.

Chapter IV presents the statistical experimental design for photodetector modeling to use within a mixed-signal processing element. A background of the digital pixel and image acquisition is presented. The development and validation of the model is

described. The model is analyzed and applied to the design of a photodiode to meet system constraints

Chapter V presents a summary of the dissertation with a list of contributions and key results. Future research directions are also provided.

Appendix A provides a detailed description of the focal plane architectural simulator used to evaluate front-end image applications, including software features and available metrics.

Appendix B provides a detailed description of the TI DSP chips used for comparison, including reported chip specifications and application benchmarks.

CHAPTER II

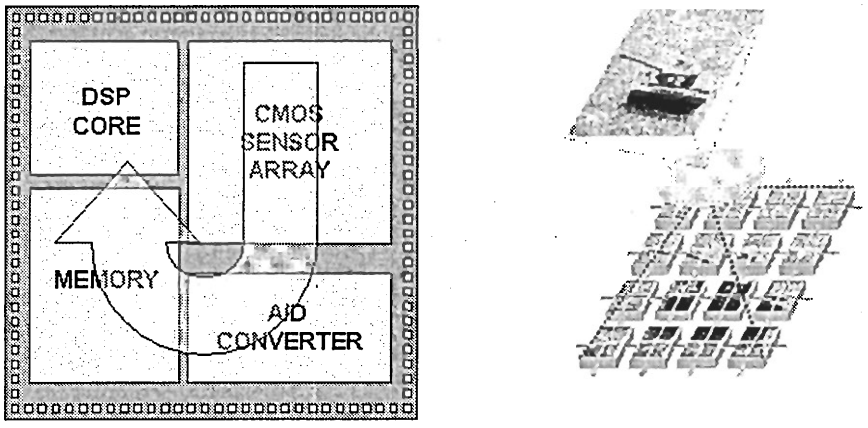
PERFORMANCE ANALYSIS OF FRONT-END IMAGE PROCESSING APPLICATIONS

Summary

Pixel-level image processing architectures can leverage the abundant data parallelism to provide high performance embedded systems. Previous pixel-level focal plane processors were limited to a small set of applications for binary images. However, the addition of sufficient data storage and key functional units, made possible with current technological improvements, expand the application functionality to operate on image formats with higher resolution. This chapter presents performance analysis of an integrated focal plane architecture for common front-end imaging applications. The focal plane architecture combines data acquisition, analog-to-digital conversion, and image processing. Using fine-grain processing of 16 pixels per processing element (PPE) or less, this system achieves performance that exceeds comparable DSP architectures on key front-end imaging applications such as convolution, DCT, edge detection, and median filtering. The 1 PPE implementation operating at 10MHz executes an imaging sequence 4x faster than the TI TMS320C6411 chip operating at 300MHz. These basic tasks can consume over half of the execution time of a typical imaging sequence. That workload can be partitioned to the focal plane architecture, leaving more complex tasks for the DSP.

2.1 Introduction

Recent emphasis has been placed upon multimedia processing to create a ubiquitous computing environment. In addition, our mobile society demands this multimedia processing be incorporated in portable devices, sparking the further development of embedded systems for real-time applications. A wide range of products, such as video and still cameras, laptop computers, and portable data assistants (PDAs), deliver multimedia processing by including inexpensive imaging chips. However, this processing creates a challenging design problem [38] and requires a change in paradigm to accommodate processing requirements [39]. Next-generation portable imaging products will benefit from the monolithic integration of photodetectors, analog-to-digital converters, digital processing, and data storage to improve their performance, efficiency, and cost. A typical system-on-a-chip digital signal processing (DSP) architecture, shown in Figure 13(a), assigns an entire image to a single processing core. This architecture is designed to span all stages of image processing. However, the DSP architecture uses a significant amount of processing to perform basic image enhancement and image analysis applications. For example, the digital image signal multiprocessor [40] reports 53% of its execution time for preprocessing and 47% for feature extraction. Preprocessing includes common tasks, such as noise reduction, smoothing, and segmentation, which are characterized by high spatial locality. Processing in the focal plane can more efficiently handle these basic tasks, leaving the more complex applications to the DSP architecture. A focal plane architecture, shown in Figure 13(b), can be built by integrating analog-to-digital conversion (ADC) and digital processing at each detector site in the focal plane array.



(a) Modular DSP **(b) Focal Plane Processing**

Figure 1: Architectural paradigms for image processing

Incorporating analog and digital components to process pixels at the detection site create new design challenges [9, 41, 42], including the choice of image tasks to perform on the focal plane. Each application requires various computation, communication, and storage costs when mapped to a focal plane processor. These costs are determined by the image processing granularity, where a finer grain size translates to more pixel values external to the processing element (PE). Also, the hardware design of the processing element directly impacts the image workload. Using simple binary processing elements at each pixel severely limit the scope of the application suite and do not provide the required front-end processing capability. However, supporting an extensive Instruction Set Architecture (ISA) with infrequently used hardware may prove costly in terms of silicon area. Understanding the targeted application suite can aid architectural design choices.

This chapter presents a characterization of the image application suite for use in embedded focal plane processing systems. A Single-Instruction Multiple-Data (SIMD) focal plane architectural simulator provides dynamic workload statistics for the image application suite. A key design parameter is the number of pixels assigned to each processing element. The number of Pixels-per-Processing Element (PPE) is adjusted to determine the effect of processing granularity on the computation, communication, and storage costs for these applications.

The workload for an application suite of median filtering, convolution, inside edge detection, and the discrete cosine transform has been characterized. This application suite requires a 12-bit datapath to retain computational precision. In addition, the application suite requires at least [(2 • PPE) + 7] words of data storage to account for the original image, the final image, and processing overhead. Utilization of processing elements within the focal plane architecture ranges from 78% for the discrete cosine transform to 100% for convolution.

For the sequence of median filtering, convolution, and inside edge detection, the execution time with the focal plane architecture is reduced when compared to the conventional DSP architecture despite a 30x difference in clock frequency (10MHz for focal plane architecture, 300MHz for TMS320C6411 DSP). This benefit is for front-end applications that can consume over half of the processing time of an imaging sequence. For a Quad-CIF resolution, the focal plane architecture requires 52.5μs using 1 PPE and 195.2μs using 4 PPE. The DSP architecture requires 210.9μs. However, the DSP execution time is dependent on the image size and would increase for larger resolutions. The execution time of the focal plane architecture is independent of the image resolution.

Because of its specialized nature, this architecture would not replace a conventional DSP chip but could be used in combination as an embedded preprocessing system to perform early image applications faster and more efficiently. This would enable the DSP chip to focus on subsequent, more complex tasks of the image processing sequence. Future work includes translating the workload statistics into hardware projections using models for the analog and digital components of the pixel. These component models will quantify the silicon area tradeoffs to make the best use of available resources for overall system performance.

The organization of this chapter is as follows. Section 2.2 provides the background for applications and architectures used by image processing systems. Section 2.3 describes the application suite targeted for implementation with a focal plane processor. Section 2.4 describes the technique to characterize image application performance on a focal plane architecture. Section 2.5 presents analysis for integrating image processing in the focal plane. Section 2.6 presents performance results and compares the performance to a traditional DSP architecture. Finally, Section 2.7 concludes the discussion on workload characterization for embedded image processing applications.

2.2 Background

An imaging system includes both data acquisition and data processing. Image data enters the system where processing techniques convert the data into a useful format. Processing functions include image enhancement, image analysis, and image transformation. Image enhancement improves the quality of the image data. Image

analysis provides interpretation of image data. Image transformations are conversion operations applied to the image data. Each algorithm within a category requires different computation, communication, and storage costs to process each pixel of data.

Numerous architectures have been developed for image processing systems [2, 3]. Efficient handling of the two-dimensional image data is a common issue among these designs. A natural solution to this issue is to process the spatially parallel data within the image plane [43]. Both analog and digital techniques have been investigated to implement image processing functions. A monolithic system-on-a-chip with pixel-level processing is a potential solution to next-generation portable image products.

This section discusses the image application suite and presents both the traditional architectural approaches used to implement the required functionality as well as the focal plane processing approach. As an alternative, processing images in the focal plane can utilize the data parallelism inherent within the image application domain.

2.2.1 Characterization of the image application suite

Image processing applications are categorized into (1) point operations, (2) local operations, and (3) global operations [44]. A point operation such as thresholding occurs at the individual pixel level. A local operation such as smoothing requires knowledge of an individual pixel and its immediate neighbors. A global operation such as histogramming uses all pixel data from the image. These operations form the basis of applications found in imaging systems. To illustrate the processing sequence required for a typical imaging system of (N x N) pixels, Figure 17(a) describes the processing tasks, Figure 17(b) gives example applications at each stage, and Figure 17(c) quantifies the

amount of data required for processing [45]. In Figure 17(c), the variable *b* represents the number of bits per pixel and *N* represents the image resolution on side for a square image. Point operations are the first stage, and local operations include the first three stages. The fourth and fifth stages are global feature-measuring operations [46]. The first three stages represent a significant proportion of the computational workload. For example, the digital image signal multiprocessor [40] reports 53% of its execution time for preprocessing, which includes tasks from the first three stages such as noise reduction, smoothing, and segmentation.

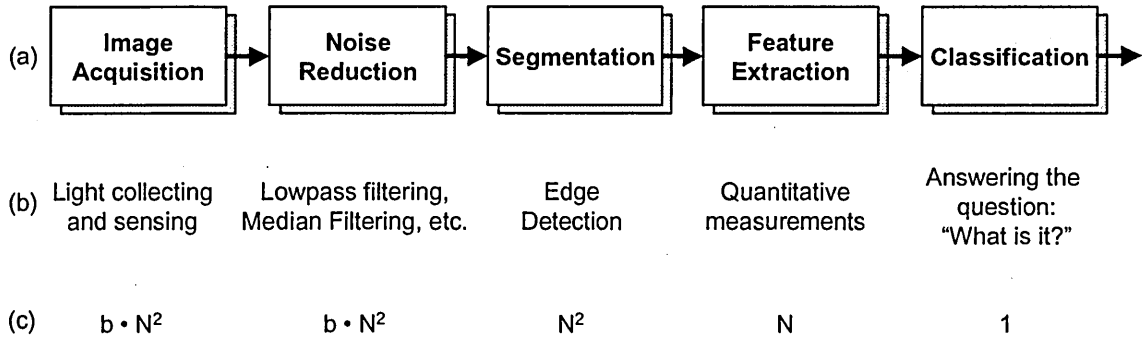


Figure 2: Typical image processing sequence (a) processing tasks (b) examples and (c) amount of data

2.2.2 Traditional architectural approaches

Image applications within the first three stages of the processing sequence exhibit high spatial locality because of the similarity in data volume. The abundant data parallelism inherent in these applications has motivated the development of multimedia extensions for general-purpose processors to improve performance on image applications. Using the Single-Instruction Multiple-Data (SIMD) processing model described by Flynn

[4], these extensions supply the processor's functional units with subword data in parallel. General-purpose architectures utilizing multimedia extensions include Intel's MMX™ [15] and SSE™ [16], Hewlett-Packard's MAX2™ for the PA-RISC™ architecture [17], Sun Microsystems' VIS™ for the SPARC [18], MIPS's MDMX™ [19], the Alpha's MVI™ [20], and Motorola's ALTIVEC™ for the PowerPC™ [21]. Alternatively, digital signal processors (DSPs) such as the TI TMS320C80 or TMS320C6000 families follow a more specialized approach [22]. However, both microprocessors and DSPs lack the computational power required to execute most media applications in real-time. In addition, the most severe limits of these architectures will not be imposed by transistors performing computing functions but by interconnection networks that perform signal communication, clock distribution, and power distribution functions [23]. Processing the image data in the focal plane can potentially address the issues of computational throughput and signal (data) communication.

2.2.3 Focal plane processing approach

Data acquisition occurs through an analog process using techniques such as charge-coupled devices (CCD) or active pixel sensors (APS) [47]. A logical approach for focal plane processing incorporates analog computational ability into the sensor device. Analog design methods, such as the Silicon Retina, have been presented to implement focal plane arrays [24]. This has led to the development of neuromorphic vision sensors for early image processing [25]. However, the advantages of analog techniques, such as low power and small area, break down as CMOS technology scales [26]. In addition, analog architectures are typically non-programmable, requiring multiple designs to

implement different functionality. Although analog architectures remain a viable candidate for certain focal plane processing applications, research efforts are also exploring the digital architecture design space. These programmable systems can utilize the advantages of future improvements in device fabrication.

The Single-Instruction Multiple-Data (SIMD) processing model described by Flynn [4] forms the foundation of the data parallel approach to programmable focal plane processing. Early architectures such as the Massively Parallel Processor (MPP) [48] and the Cellular Logic array Image Processor (CLIP) [49] applied this paradigm to the tasks of image processing and pattern recognition. More recently, SIMD architectures like the Connection Machine models CM-1 [50] and CM-2 [51], the MasPar [52] and the GAPP [53] have been successfully used for image processing applications. However, these designs were targeted to a more general set of applications and achieve performance with high cost, poor data bandwidth, and lack of portability. MGAP [54] and ABACUS [7] are examples of fine grain parallel processing architectures that address portability issues. However, the I/O to individual processing elements limits conventional SIMD arrays. To alleviate this problem, processing elements within the SIMD array are mapped directly to subsets of the image detector array. The granularity of data mapping ranges from column-level [42] to pixel-level [7].

Advances in device fabrication [36] and image sensors [47] enable the development of a system-on-a-chip with pixel-level early image processing. A programmable digital pixel is formed by monolithically incorporating the sensor device, analog-to-digital conversion (ADC) circuitry, digital processing circuitry, and data storage within a processing element (PE). A block diagram of an integrated pixel-

processing tile [26] is shown in Figure 14. This tile element is replicated to form a focal plane array imager with integrated analog-to-digital conversion and SIMD processing. Research at Stanford [55, 56] has demonstrated the feasibility of pixel level ADC. Several architectures, such as the Near Sensor Image Processing (NSIP) [27], the Programmable and Versatile Large Size Artificial Retina (PVLSAR) [28], and the Simple and Smart Sensory Processing Elements (S^3PE) [29], follow the pixel-level model to perform early image processing in the focal plane. However, these architectures utilize bit-serial processing techniques with limited memory, which can either restrict processing to binary images or require multiple cycles to perform a single instruction on data words. This may prevent the implementation of some early image processing algorithms. However, the integration of more data storage and functional units at the pixel level enables the versatility to execute a broader set of applications.

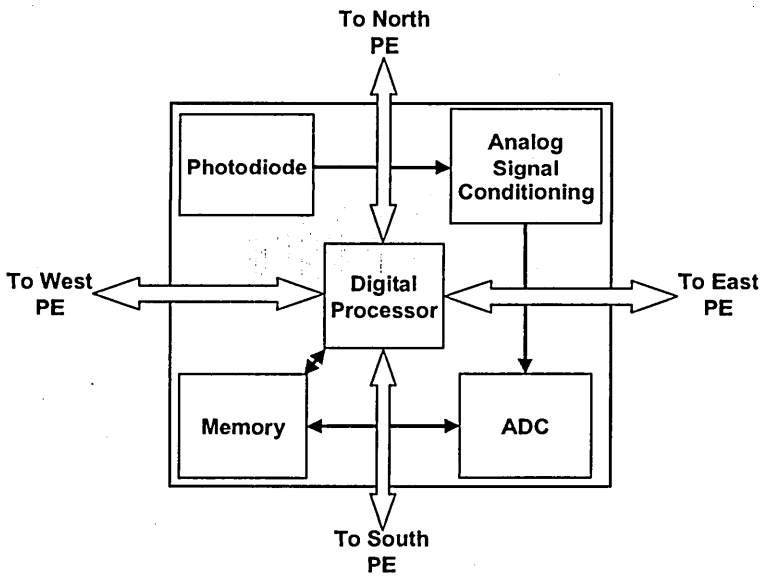


Figure 3: Programmable digital pixel

2.3 Application suite

The grayscale (8-bit) front-end applications, shown in Table 12, provide workload characteristics for image enhancement, image analysis, and image transformation in order to design the focal plane processor. The enhancement and analysis applications of median filtering, convolution, and morphological processing represent the typical early image processing sequence of: (1) noise removal, (2) smoothing, and (3) segmentation. The discrete cosine transform is a fundamental component of image compression standards such as JPEG. The workload characteristics for these applications are used to make efficient architectural choices for processing in the focal plane. This section briefly discusses each application and the algorithm implementation for a focal plane processor.

Table 1: Selected early image processing applications

Applications	Description
Image Enhancement	
Median Filtering	• Removes impulse noise from an image while preserving spatial resolution.
Convolution	• Performs different filtering operations, such as shadowing, smoothing, and edge-detection.
Image Analysis	
Morphological Processing	• Performs feature extraction and segmentation of binary or grayscale images such as inside edge detection.
Image Transformation	
Discrete Cosine Transform	• Exploits the spatial redundancy inherent in image data and is a fundamental component of image compression standards

2.3.1 Median filtering

Median filtering (MED) is useful to remove impulse noise from an image while preserving spatial resolution. The algorithm is a rank-order filter [57] that replaces each pixel in the image with the median value in the window. Generally, a window size is selected to generate a rank-order filter with odd length. A larger window size increases the severity of the median filtering effect [58]. The implemented algorithm performs a 2-D nonseparable rank and selects the median value for a 3 x 3 window.

2.3.2 Convolution

Convolution-based filtering (CONV) has been implemented to perform different filtering operations, such as shadowing, smoothing, and edge detection [58]. The filter mask elements are broadcast one at a time to every processing element. All calculations requiring the mask element are performed before the next element is broadcast. Each PE multiplies the mask element by the corresponding pixel value from the original image and accumulates the result. Values are accumulated in a spiral pattern that places the final result in the center pixel of the filter mask. The implemented algorithm uses a 3 x 3 filter mask for the smoothing operation.

2.3.3 Morphological processing

Morphological image processing refers to the study of the topology or structure of objects from their 2D spatial representation [59]. Binary or grayscale images are morphologically transformed by passing a structuring element over the image in a

process similar to convolution. At each pixel position, a specified logical function is performed between the structuring element and the underlying image. For grayscale images, erosion is the minimum pixel value in the structuring element, and dilation is the maximum pixel value in the structuring element [58]. Depending upon the size and content of the structuring element, different effects such as inside edge detection (IED) can be produced from erosion and dilation operations. The implemented algorithm uses a 3×3 structuring element to perform the morphological operations.

2.3.4 Discrete Cosine Transform

The Discrete Cosine Transform (DCT) is a transform based on the cosine kernel with resulting values mapped into the real number domain [58]. The importance of this application is particularly evident in real-time video compression and decompression, where DCT operations account for 25%-50% of CPU time without dedicated hardware support. An 8 x 8 2D-DCT has been implemented using the row-column method, in which a one-dimensional DCT is applied to the rows and then columns. The implemented algorithm maps to an integer architecture using the lifting scheme for a multiplierless transform [60].

2.3.5 Algorithm implementation

Fine-grain processing with a SIMD focal plane array requires each algorithm to be implemented in a parallel manner. Convolution and inside edge detection both compute a value for a pixel using a 3 x 3 window. A spiral communication pattern was utilized to efficiently implement these algorithms using a single pixel per processing

element (Figure 4). For convolution, the spiral pattern represented the multiply-accumulate chain for computing the center pixel value. Corresponding weights were broadcast for each multiply operation, and the accumulated value was transferred to the next pixel in the spiral. For inside edge detection, the spiral pattern represented the computation of the maximum or minimum value within the window. Each processor compares its pixel value to the transferred pixel value. The minimum is transferred for erosion, or the maximum value is transferred for dilation. For multiple pixels per processing element, the spiral communication was used only for pixels external to the processing element.

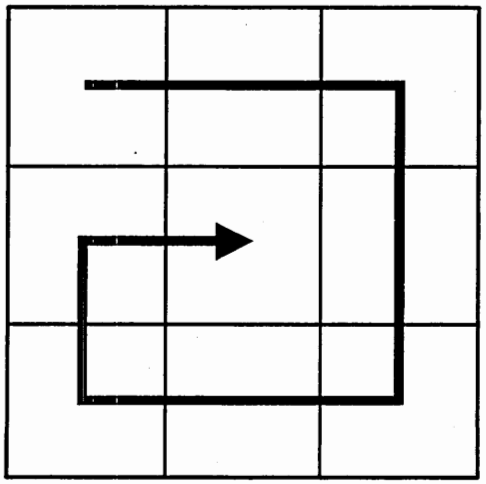


Figure 4: Spiral communication pattern for applications using 3 x 3 window

For the 3 x 3 median filtering algorithm, a novel strategy is employed. Instead of a single-pass sorting algorithm for the nine pixel values in the window, the algorithm uses a two-pass sorting routine with seven values. Any group of seven values can be sorted from minimum to maximum. Next, the remaining two values selectively replace

the second and sixth values from the sorted pixels. After the second sorting pass, the median value from the new group of seven is also the median value for the original 3 x 3 window. This implementation requires two fewer registers than the single pass sort at the cost of increased processing time. However, in the focal plane architecture design, data storage is a significant cost, while processing throughput is in abundance due to the inherent data parallelism of the application. This method also provides the minimum and maximum pixel values of the original 3 x 3 window.

The 8 x 8 discrete cosine transform is a challenge for fine-grain processing because most of the data is external to the processing element. However, the SIMD processing array can be mapped to a forward transform structure, like the multiplierless DCT approximation shown in Figure 5 [61]. Processing elements are grouped to form 8 x 8 pixel blocks. The transform is first performed for the rows. The data is then properly reordered before performing the transform on the columns.

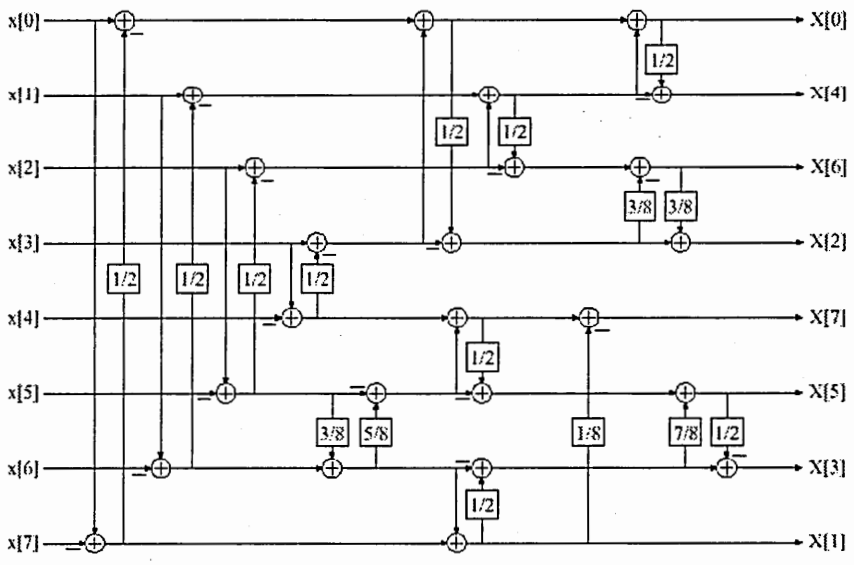


Figure 5: Forward transform of the all-lifting binDCT-C

2.4 Analysis

This section describes the technique to characterize image application performance. A focal plane architectural simulator is used to implement the application suite. A methodology is described that provides dynamic workload statistics from each application. The system performance is calculated using these statistics.

2.4.1 Focal Plane Architectural Simulator

Applications for focal plane architectures can be programmed using the SIMD Pixel Processor (SIMPil) Simulator [34]. This software tool is a windows-based instruction level simulator, running on a PC platform. The SIMPil Simulator allows editing, assembling, executing, and debugging parallel image applications in a single integrated workbench. This tool has been used extensively to evaluate focal plane processing of multimedia applications [62]. The current version of the SIMPil Simulator is available on the download page. An extended description of the focal plane architectural simulator can be found in Appendix A.

2.4.2 Methodology

The block diagram in Figure 6 illustrates the methodology for characterizing the image processing application suite. Image data is sampled by the focal plane architectural simulator based upon the preset number of pixels per processing element (PPE). The PPE is set to 1, 4 or 16. In the 4 PPE case, the pixels are arranged in a 2 x 2 pattern. In the 16 PPE case, the pixels are arranged in a 4 x 4 pattern. Each application is then executed to

determine the dynamic workload statistics. Key measurements include the cycle count, the dynamic instruction count, and the utilization. The cycle count is based upon a single instruction issued per cycle. The dynamic instruction count is the total number of parallel instructions issued to the PE array. The utilization is the average number of active processing elements. The dynamic instructions are classified by functional units to determine the relative usage of each unit within the application. The storage requirement is based upon the number of register words and the operand resolution used during execution.

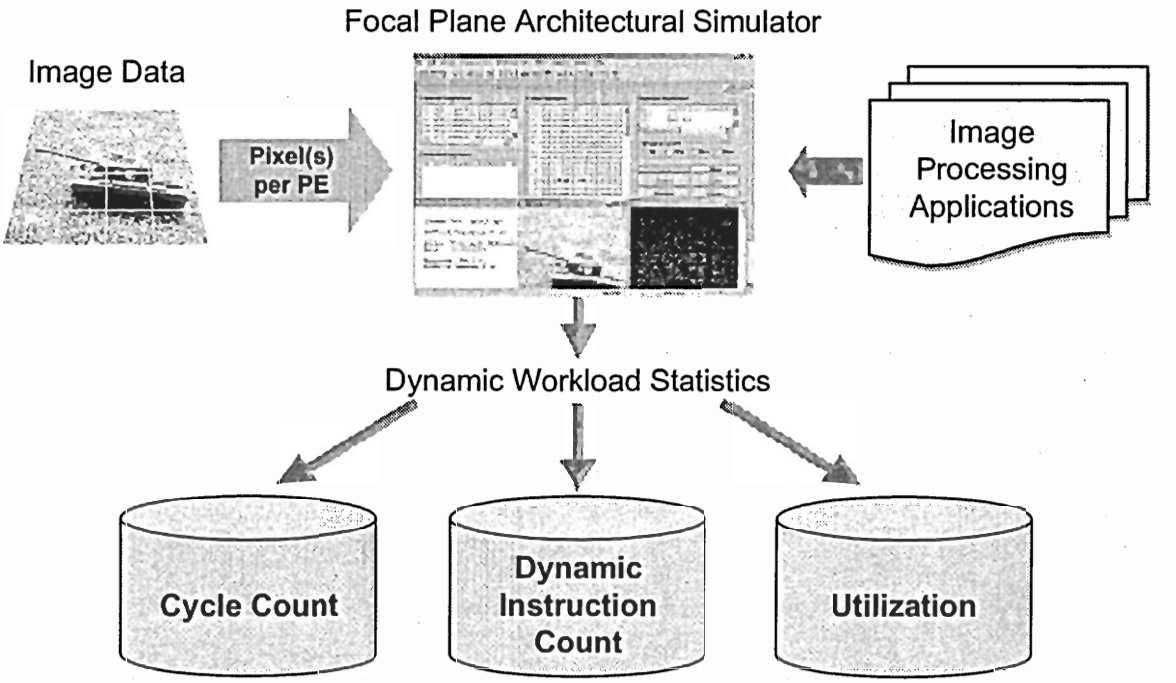


Figure 6: Methodology for application characterization

2.4.3 System performance calculation

Performance is calculated for a target system using grayscale (8-bit) Quad-CIF (QCIF = 176 pixels x 144 pixels). The Quad-CIF resolution is one specification of the H.261 and H.263 video codec standards of the International Telecommunications Union (ITU) [63]. Using the simulator for the target system, execution time t_{exec} is computed as follows:

$$t_{exec} = \frac{C}{f_{ck}} \qquad [s] \qquad (1),$$

where C is the cycle count for a given application and f_{ck} is the clock frequency. The execution time is determined with reference to a 10 MHz target platform This clock frequency addresses both the speed of analog components in each processing element and power density limitations for the high utilization factor of PEs in the array. Using the execution time, the sustained throughput $Throughput_{sust}$, measured in billion operations per second, is determined for the target system as follows:

$$Throughput_{sust} = \frac{IC \cdot U \cdot N_{PE}}{t_{exec}} \qquad \left[\frac{Gops}{s}\right] \qquad (2)$$

where IC is number of parallel instructions issued to the PE array during the application (i.e., the dynamic instruction count which includes both computation and communication). The system utilization U is calculated as the average number of active processing elements determined from the simulator's concurrency meter. N_{PE} is the number of processing elements in the PE array and is determined from the following formula:

$$N_{PE} = \frac{\text{system\_resolution}}{PPE} \qquad (3)$$

where *PPE* is the number of pixels in each processing element (1, 4, or 16). For *PPE* > 1, pixels are arranged in a square (i.e. 2 x 2 or 4 x 4). The target system resolution is QCIF (176 x 144 pixels). The values for N_{PE} are shown in Table 14.

Table 2: Focal plane processor array characteristics for QCIF resolution

PPE	PE Array Dimensions		Total Number of PEs
	X	Y	
1	176	144	25344
4	88	72	6336
16	44	36	1584

2.5 Focal plane processing integration

Processing in the focal plane represents a challenging design problem to integrate the data acquisition circuitry with digital processing functionality. Characterizing the targeted application workload leads to reasonable design choices for register file size, datapath width, and functional units to efficiently utilize silicon area. The hardware must satisfy both the data storage and the data precision constraints to execute the algorithms. Functional unit usage depends upon the neighborhood window size for selected applications (e.g. 3 x 3, 8 x 8).

2.5.1 Application constraints for hardware implementation

Table 16 shows the architectural design parameters for each processing element determined by the application suite. Register requirements are derived from the code for the implemented algorithm and include twice the PPE (to store the input and output

images) plus overhead for intermediate calculations. Datapath requirements are determined from the measured operand resolutions for each instruction during simulation. With an overhead of 2 registers, convolution required the fewest total registers, [(2 • PPE) + 2], but the widest datapath (12 bits) to accommodate the successive multiply-accumulates of pixel values with the corresponding 3 x 3 mask values. The discrete cosine transform also required a 12-bit datapath because of its lifting scheme with arithmetic shifts. The other applications only required the original grayscale resolution of 8 bits. Median filtering required [(2 • PPE) + 7] registers to store and sort the pixels in the 3 x 3 window. A sufficient architecture capable of executing all four applications would require [(2 • PPE) + 7] registers with a 12-bit datapath. These values are used to select the sizes of both the ALU datapath and the register file in each processing element.

Table 3: Application constraints for focal plane processor implementation

Application	Registers required	Datapath required
Median Filtering	(2 • PPE) + 7	8 bits
Convolution	(2 • PPE) + 2	12 bits
Inside Edge Detection	(2 • PPE) + 3	8 bits
Discrete Cosine Transform	(2 • PPE) + 5	12 bits
Full Application Suite	**(2 • PPE) + 7**	**12 bits**

The relationship for required registers impacts the feasibility of processing integration in the focal plane. The data storage competes with other components for silicon area within the processing tile (Figure 14). In addition, the PPE should allow grouping into standard 8 x 8 blocks for processing (i.e. PPE = 2^{2n} where n = 0, 1, 2, etc).

For PPE > 16, the data storage requirement becomes a prohibitive cost in terms of silicon area within the pixel tile.

2.5.2 Functional unit usage percentages

The usage percentages are key for determining which functional units are cost-effective to include in the hardware implementation. The dynamic instruction count for the applications in Table 12 has been categorized based upon the required functional units: arithmetic logic unit (ALU), multiplier (MULT), shifter (SHIFT), PE activity control unit (MASK), communication (COMM), and image loading (PIXEL). The ALU, MULT, and SHIFT units are responsible for the computation for an application. The MASK and COMM units are required for synchronization and data distribution among the processing elements of the SIMD array. The processing granularity (PPE = 1, 4, or 16) combined with the application window size (3 x 3, 8 x 8) affects the distribution of workload among the functional units.

2.5.2.1 Processing for 3 x 3 window

The utilization of the functional units for 1 PPE, 4 PPE, and 16 PPE implementations are shown in Figure 7, Figure 8, and Figure 9 respectively for the applications that use a 3 x 3 window (MED, CONV, IED). As the PPE increases, the communication requirement decreases because more neighborhood image data is already contained within the processing element. However, each processing element is required to compute the final values for multiple pixels, thereby increasing the percentage of ALU

instructions issued. The MASK instructions represent a significant percentage (30% - 35%) to handle the control flow in the SIMD array.

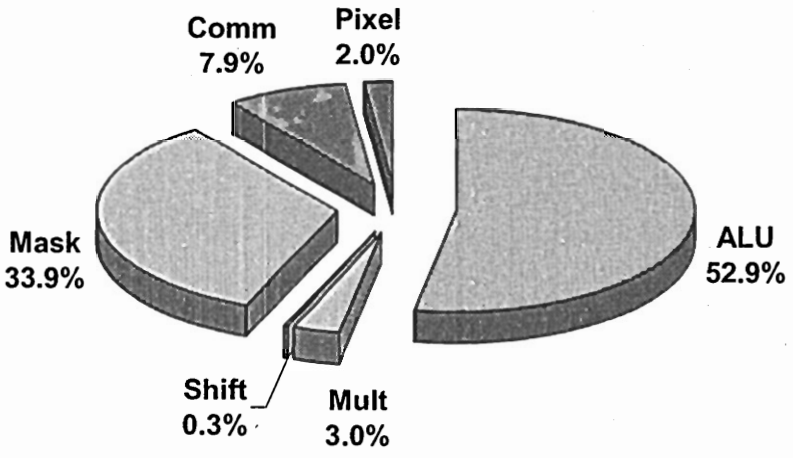


Figure 7: Utilization of functional units for 3 x 3 window applications using 1 PPE

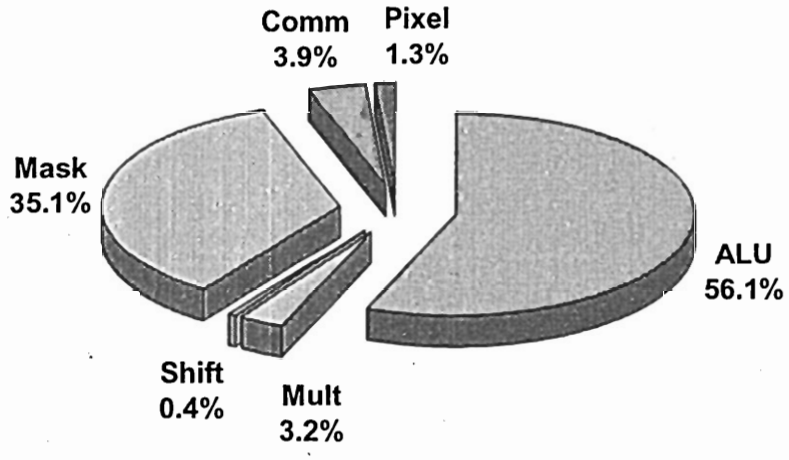


Figure 8: Utilization of functional units for 3 x 3 window applications using 4 PPE

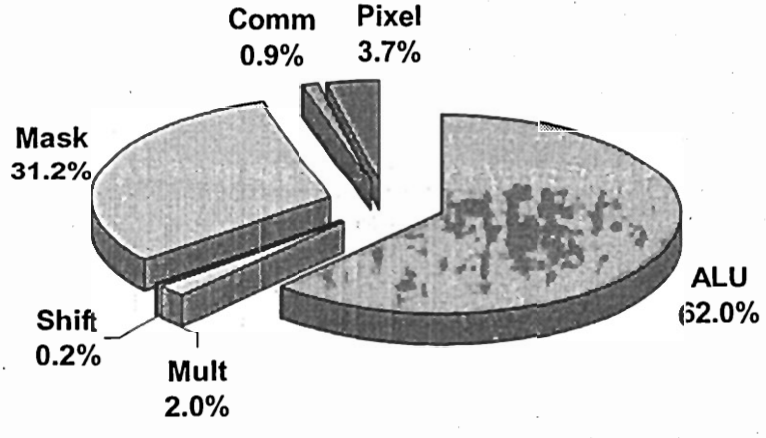


Figure 9: Utilization of functional units for 3 x 3 window applications using 16 PPE

2.5.2.2 Processing for 8 x 8 window

Applications that extend beyond neighborhood processing will challenge a fine-grain processing architecture. For an 8 x 8 2-D discrete cosine transform, using 1 PPE represents that extreme case. The MASK and COMM units dominate the execution workload utilization (Figure 10). Because the algorithm operates on a single row in an 8 x 8 block, the MASK unit is utilized often to deactivate and activate PEs in the SIMD array. In addition, data values must be transferred along each row or column to implement the separable transform. The 4 PPE case reduces some of the COMM usage, but has similar MASK proportions as the 1 PPE case (Figure 11). However, the 16 PPE case, which uses a 4 x 4 pattern, maps well to the DCT algorithm in Figure 5. Because the processing occurs mostly in two 4 x 1 blocks, both the COMM and the MASK proportions are reduced (Figure 12).

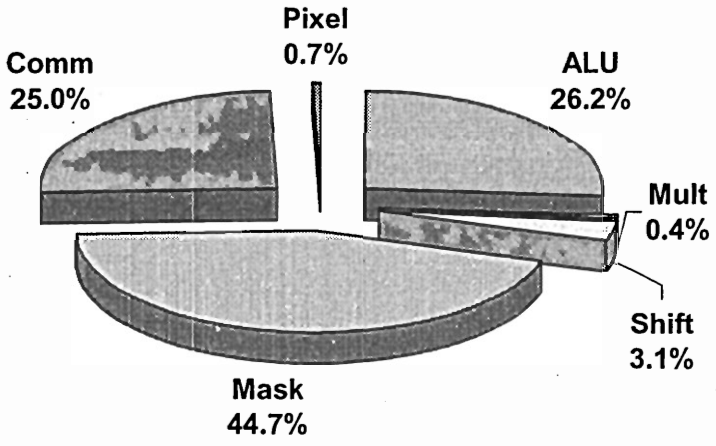


Figure 10: Utilization of functional units for 8 x 8 window application using 1 PPE

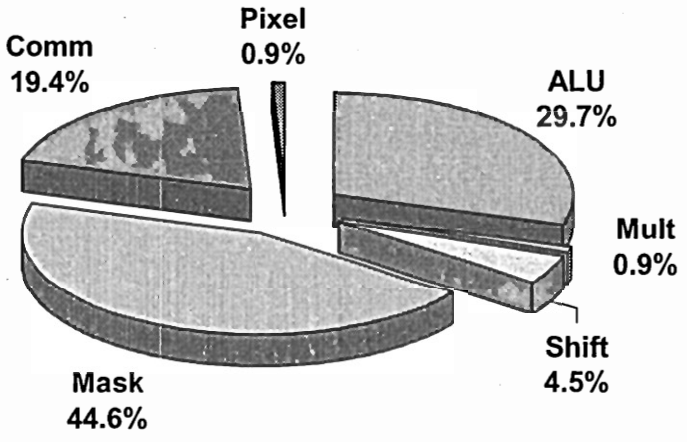


Figure 11: Utilization of functional units for 8 x 8 window application using 4 PPE

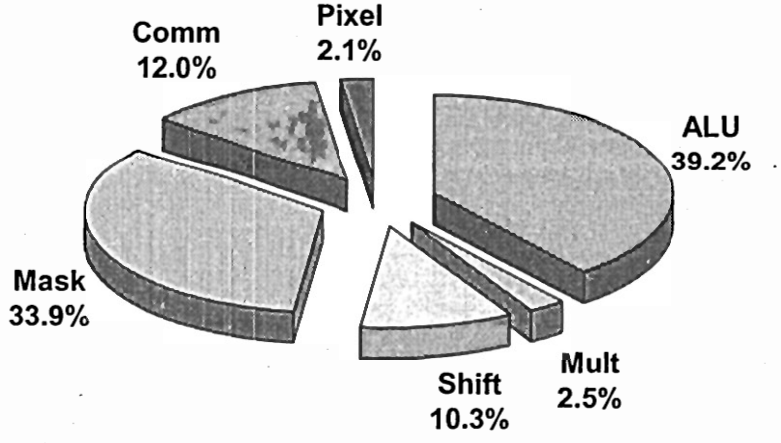


Figure 12: Utilization of functional units for 8 x 8 window application using 16 PPE

2.6 Performance analysis

Previous pixel-level processing architectures, such as the Programmable and Versatile Large Size Artificial Retina (PVLSAR) [28], and the Simple and Smart Sensory Processing Elements (S^3PE) [29], have restricted application suites operating on binary images. However, the focal plane architecture provides a programmable environment that enables high performance on common imaging applications. The workload characterization shows that the number of pixels per processing element impacts the sustained throughput of the architecture. Yet, the architecture exceeds the reported throughput of comparable TI DSP chips. A comparison with TI DSP chips is also presented using the execution time of an imaging sequence.

2.6.1 Workload characterization

Using the focal plane architectural simulator, the dynamic workload has been determined for median filtering (Table 15), convolution (Table 5), inside edge detection (Table 6), and the discrete cosine transform (Table 7) for 8-bit Quad-CIF images. The maximum number of PE instructions executed equals ($IC \bullet N_{PE}$). Utilizations above 90% for both convolution and inside edge detection result from local area calculations at each processing pixel. The algorithms for those two applications require little or no deactivation of PE's during processing. In contrast, median filtering and the discrete cosine transform have utilizations of approximately 80%. The median filtering algorithm involves data-dependent sorting. The discrete cosine transform algorithm requires a sequential calculation of butterfly operations within the 8 x 8 block.

The sustained throughput has been calculated using equations (1), (2), and (3) from Section 0. The sustained throughput achieved by the focal plane architecture implementations (1 PPE, 4 PPE, 16 PPE) exceeds the values achieved by a traditional DSP architecture despite a significant difference in clock frequency. For example, the throughput specification of the TI TMS320C6211B DSP chip running at 150 MHz is 1.2 Gops/s, and the throughput specification of the TI TMS320C6411 DSP chip running at 300 MHz is 2.4 Gops/s [22].

Table 4: Application characterization for median filtering using a 10 MHz clock with Quad-CIF resolution

PPE	Execution Time (μs)	Utilization	Sustained Throughput (Gops/s)
1	38.5	81.50%	**146**
4	153.1	81.47%	**37**
16	1353.1	86.73%	**11**

Table 5: Application characterization for convolution using a 10 MHz clock with Quad-CIF resolution

PPE	Execution Time (μs)	Utilization	Sustained Throughput (Gops/s)
1	5.6	100.00%	**204**
4	18.3	100.00%	**50**
16	82.6	100.00%	**13**

Table 6: Application characterization for inside edge detection using a 10 MHz clock with Quad-CIF resolution

PPE	Execution Time (μs)	Utilization	Sustained Throughput (Gops/s)
1	8.4	93.26%	**166**
4	23.8	94.41%	**42**
16	97.9	94.54%	**11**

Table 7: Application characterization for discrete cosine transform using a 10 MHz clock with Quad-CIF resolution

PPE	Execution Time (μs)	Utilization	Sustained Throughput (Gops/s)
1	39.6	78.16%	**198**
4	72.1	78.62%	**50**
16	112.1	78.59%	**12**

2.6.2 Comparison with traditional DSP architecture

Execution time on imaging applications is an important constraint for developing portable, real-time devices. The execution time for an imaging sequence of: (1) noise removal, (2) smoothing, and (3) segmentation was used to compare the focal plane architecture implementations (Table 8) to TI DSP chips (Table 10). The focal plane architecture implementations have the capability to acquire the image while operating at a slower clock frequency. The execution time comparison uses C62x™ DSP Benchmarks [22] and C64x™ DSP Benchmarks [22] for: (1) 3 x 3 Median Filter (2) 3 x 3 Convolution, and (3) Sobel Edge Detection. An extensive description of these benchmarks can be found in Appendix B.

The 1 PPE and the 4 PPE implementations compare favorably in execution time to the TMS320C6411 considering the 30x difference in clock frequency. The implementation using 16 PPE compares favorably to the TMS320C6211B despite the 15x difference in clock frequency. However, the execution time from the DSP benchmarks depend upon the image resolution. For larger images (e.g. CIF), the execution time increases as a function of both row width and column width. The focal plane architecture is independent of the image dimension due to the parallel data processing. This does not suggest that the architecture replaces a DSP but can handle certain common image enhancement and image analysis tasks more efficiently through specialization. These operations can represent a significant portion of the total operations within an image processing sequence. A combined system could partition the front-end processing to the embedded focal plane architecture, leaving more complex tasks for the DSP.

Table 8: Execution time for imaging sequence using focal plane architectures

	Focal Plane Architectures		
Data Acquisition	1 PPE	4 PPE	16 PPE
Clock Frequency	10 MHz	10 MHz	10 MHz
Resolution	QCIF	QCIF	QCIF
Total Execution Time	**52.5μs**	**195.2μs**	**1533.6μs**

Table 9: Execution time for imaging sequence using DSP architectures

	TMS320C6211B	TMS320C6411
Data Acquisition	N/A	N/A
Clock Frequency	150 MHz	300 MHz
Resolution	QCIF	QCIF
Total Execution Time	**1102.0μs**	**210.9μs**

2.7 Conclusion

The demand for portable image products will continue to saturate the available computation, communication, and storage capabilities of conventional imaging systems. Technological advances in device fabrication and integration are enabling the development of focal plane architectures to meet both the bandwidth and the computational requirements of image processing systems. Processing on the focal plane addresses the potential architectural constraints by exploiting data-parallel processing naturally found in image applications. Focal plane architectures have enormous potential in performance and efficiency for a monolithically integrated system embedded within portable devices.

This chapter presents a characterization of the image application suite for use in embedded focal plane processing systems. Understanding the targeted application suite can aid architectural design choices. The algorithms chosen for the application suite of median filtering, convolution, inside edge detection, and the discrete cosine transform require a 12-bit datapath with at least [(2 • PPE) + 7] words of data storage. High utilization of processing elements (greater than 78%) is achieved for the SIMD focal

plane architecture. An embedded focal plane architecture delivers high performance with high resource efficiency for tasks requiring local neighborhood processing. Using a combination of an embedded focal plane architecture with a traditional DSP architecture, the image processing workload can be partitioned to allow the focal plane architecture to execute front-end applications while the DSP architecture handles complex, global tasks.

CHAPTER III

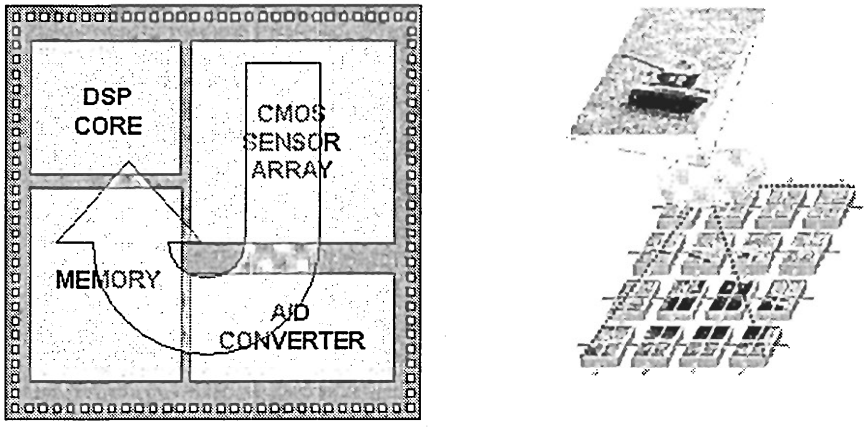
EFFICIENCY ANALYSIS OF FOCAL PLANE ARCHITECTURES

Summary

Monolithic integration of photodetectors, analog-to-digital converters, data storage, and digital processing can improve both the performance and the efficiency of future portable image products. However, digitizing and processing a pixel at the detection site presents the design challenge to deliver a system with the required performance at the lowest cost, not just a system with the highest performance. This chapter analyzes the area-time efficiency, the area efficiency, and the energy efficiency of a mixed-signal, SIMD focal plane processing architecture that executes front-end image applications with neighborhood processing. Implementations of the focal plane architecture achieve up to 81x higher area efficiency and up to 11x higher energy efficiency when compared to traditional TI DSP chips. Higher efficiency ratings are required to maintain portability while addressing technology limitations such as interconnect wiring density, heat extraction, and battery life. Systems can be implemented with a less expensive fabrication technology by increasing the number of pixels per processing element (PPE).

3.1 Introduction

The demand for mobile productivity has led to incorporating embedded systems into handheld devices for real-time applications. A wide range of products, such as cellular phones, video and still cameras, and portable data assistants (PDAs), deliver multimedia processing by including inexpensive imaging chips. However, this processing creates a challenging design problem [38] and requires a change in paradigm to accommodate processing requirements [39]. Future portable imaging products will benefit from the monolithic integration of photodetectors, analog-to-digital converters, digital processing, and data storage to improve their performance, efficiency, and cost. A typical system-on-a-chip digital signal processing (DSP) architecture, shown in Figure 13(a), assigns an entire image to a single processing core. This architecture is designed to span all stages of image processing. However, the DSP architecture uses a significant amount of area and energy resources to perform basic image enhancement and image analysis applications. For example, the digital image signal multiprocessor [40] reports 53% of its execution time for preprocessing and 47% for feature extraction. Preprocessing includes common tasks, such as noise reduction, smoothing, and segmentation, which are characterized by high spatial locality. Processing in the focal plane can more efficiently handle these basic tasks, leaving the more complex applications to the DSP architecture. A focal plane architecture, shown in Figure 13(b), can be built by integrating analog-to-digital conversion (ADC) and digital processing at each detector site in the focal plane array.



(a) Modular DSP **(b) Focal Plane Processing**

Figure 13: Architectural paradigms for image processing

Incorporating analog and digital components to process pixels at the detection site creates a challenging design problem [9, 41, 42]. The goal is not strictly building the system with the highest performance, but delivering the system with the required performance at the lowest cost. The architecture must consider key performance and cost metrics, such as execution time, system throughput, chip area, power consumption, and area-time efficiency. However, in the system-on-a-chip design, two figures of merit are also important, namely area efficiency and energy efficiency [14]. Characterizing the targeted application workload leads to reasonable design choices for register file size, datapath width, and functional units to efficiently utilize silicon area and power consumption. In addition, higher ratings in area efficiency (system throughput per unit area) and energy efficiency (system throughput per unit power) are required because of technology limitations such as interconnect wiring density and heat extraction. The

choice of both fabrication technology and pixels per processing element impact the system design.

This chapter presents the area-time efficiency, the area efficiency, and the energy efficiency from analyzing a mixed-signal focal plane processing architecture. Different implementations are used that vary the number of pixels per processing element (PPE). A focal plane architectural simulator provides the execution time and sustained throughput for the application suite. Component area models project the chip size for the design. A Technology Scenario Analyzer (TeSA) projects power consumption using different fabrication technologies. Because excessive die sizes would limit the realization of these systems, guidelines for affordable manufacturing are used from ITRS specifications [36, 37]. Battery life is another vital concern. Therefore, maximum power for each technology is constrained to ITRS specifications for portable battery operation, which is generally less than 3 Watts [36, 37]. Area-time efficiency, area efficiency, and energy efficiency are calculated using the appropriate projected values.

Despite a significant difference in clock frequency, implementations of the focal plane architecture perform well compared to a traditional DSP architecture. The sustained throughput achieved by the focal plane architecture implementations operating at 10 MHz exceeds the values reported in TI DSP chip specifications (Table 10). For area-time efficiency, the 16 PPE implementation is comparable to the TI TMS320C6211B chip, while the 1 PPE and the 4 PPE implementations show some improvement versus the TI TMS320C6411 chip.

Table 10: Increase in sustained throughput using focal plane architecture implementations

	TMS320C6211B (150 MHz)	TMS320C6411 (300 MHz)
1 PPE	130x	65x
4 PPE	32x	16x
16 PPE	10x	5x

Implementations of the focal plane architecture have higher energy efficiency and area efficiency when compared to traditional TI DSP chips. Energy efficiency increases by an average factor of 11x compared to the TMS320C6211B and by an average factor of 2.9x compared to the TMS320C6411. The energy efficiency demonstrates the potential for extended battery life in a portable device. The area efficiency is dramatically improved as well using the focal plane architecture (Table 11). The area efficiency for the focal plane architecture implementations includes the data acquisition and analog-to-digital conversion circuitry, which is not available with the DSP chips. Although the 1 PPE implementation provides the most efficient architecture, it generally exceeds the constraints for chip size and power consumed. A 4 PPE or 16 PPE with a less aggressive technology may provide a more cost-effective solution. In addition, detector area dominates as feature size shrinks. Therefore, denser detector technology could substantially reduce chip area.

Table 11: Increase in area efficiency using focal plane architecture implementations

	TMS320C6211B (180nm technology)	TMS320C6411 (120nm technology)
1 PPE	81x	56x
4 PPE	47x	27x
16 PPE	20x	10x

The organization of this chapter is as follows. Section 3.2 provides the background for architectures used by image processing systems. Section 3.3 describes the technique to analyze the performance of the focal plane processor array on the targeted application suite. Section 3.4 discusses the area models for the processing element components and presents the projected area for the focal plane processor array. Section 3.5 presents the projected power consumption for the focal plane processor array. Section 3.6 presents the analysis of resource efficiency for design implementations and provides a comparison to a traditional DSP architecture. Finally, Section 3.7 concludes the discussion on efficiency analysis for a mixed-signal focal plane architecture.

3.2 Background

Numerous architectures have been developed for image processing systems [2, 3]. Efficient handling of the two-dimensional image data is a common issue among these designs. A natural solution to this issue is to process the spatially parallel data within the image plane [43]. Processing capability can be integrated directly into the focal plane.

A programmable digital pixel is formed by monolithically incorporating the data acquisition device, analog-to-digital conversion (ADC) circuitry, data storage, and digital

processing circuitry within a processing element (PE). A block diagram of this processing tile [26] is shown in Figure 14. This tile element is replicated to form a focal plane array imager with integrated analog-to-digital conversion, nearest-neighbor communication, and SIMD processing. This section discusses the functionality of data acquisition, analog-to-digital conversion, data storage, and digital processing.

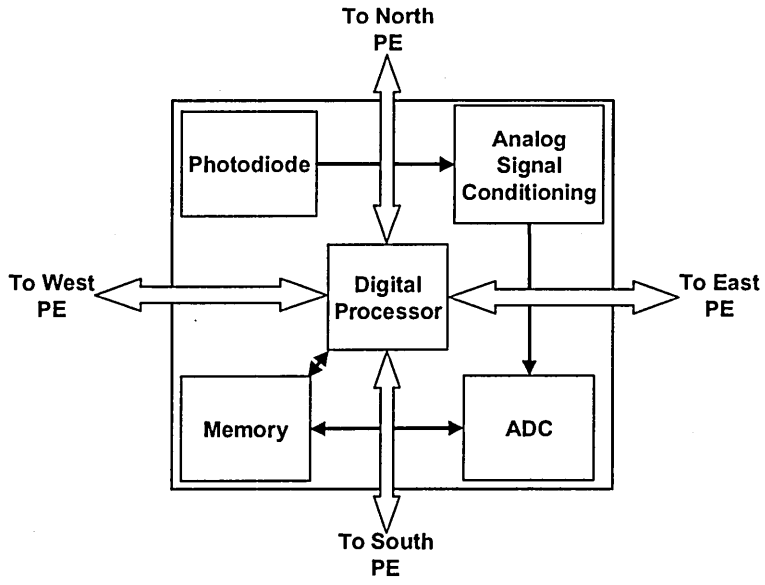


Figure 14: Programmable digital pixel

3.2.1 Data acquisition

The first stage performed by the digital pixel is the acquisition of the analog light intensity of an image. The relative response of the human eye correlates to the major color bands of the visible spectrum, which span from approximately 0.4μm to 0.7μm [64]. Therefore, photodetector devices used in the digital pixel should provide reasonable performance across these wavelengths. Photodetectors absorb photons with energy greater than the material bandgap to generate a current proportional to the number of

optically generated electron-hole pairs [30]. The quantum efficiency of the detector is the number of electron-hole pairs generated per incident photon for a given wavelength [30]. The spectral response, or responsivity, is the quantum efficiency over a range of light wavelengths [65]. Photodetectors made from silicon can detect light wavelengths up to approximately 1.1μm due to its bandgap energy of 1.12eV and provide reasonable performance over the visible spectrum (Figure 15).

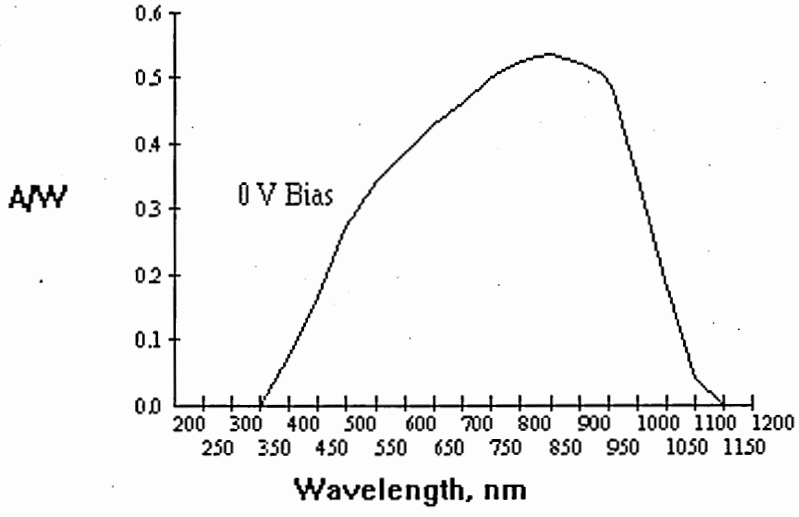


Figure 15: Spectral response of a typical silicon photodetector

Implementing photodetectors in a standard CMOS process enables the integration of other processing components at the detection site. Research efforts have been made to develop and evaluate these CMOS image sensors [12, 66, 67]. A noted development is the active pixel sensor (APS) [68], which has been implemented into a digital camera on a chip with the functionality of analog-to-digital conversion [69].

3.2.2 Analog-to-digital conversion

The next stage is to convert the acquired image signal into a digital value. The interface between analog signals and their digital representation requires three tasks: (1) anti-alias filtering, (2) sampling, and (3) quantization [31]. These tasks can be implemented using an oversampling technique. Oversampling methods use sampling rates far above the Nyquist rate, which is the minimum rate for reconstructing a signal without aliasing [70]. Figure 16 shows a block diagram for analog-to-digital conversion [31]. The oversampled analog signal is passed through one or more integrators, represented by transfer function block H(z), before quantization. Using a feedback loop, the quantization noise is transformed (or shaped) into a high-pass response. The signal is then sent to a decimator, which combines the low-pass filtering operation and rate reduction. This removes the high-pass quantization noise and other signal information above the maximum input frequency of interest.

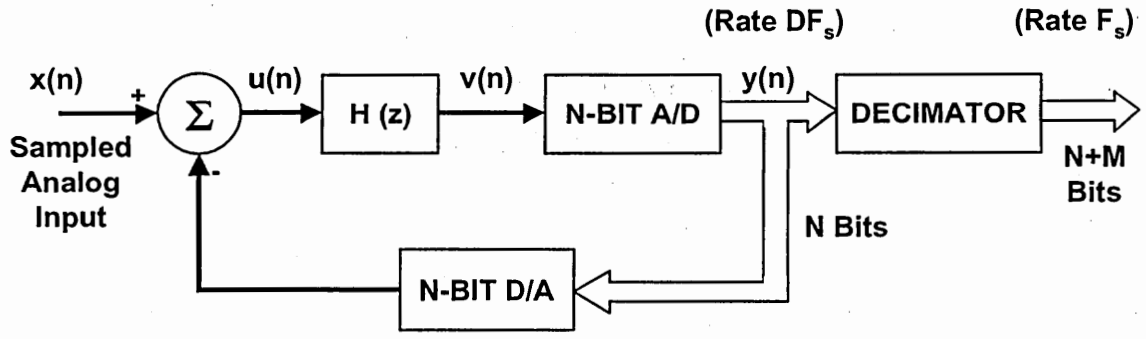


Figure 16: Generic noise-shaping feedback loop

A candidate implementation of analog-to-digital conversion with noise shaping is the delta-sigma (or sigma-delta) converter [71]. These converters have become popular

because they avoid some of the difficulties of conventional A/D conversion, such as the need for high-precision analog circuitry and vulnerability to noise or interference [71]. The performance of these circuits has been studied with standard CMOS implementation [72-74]. Sigma-delta converters have been successfully integrated with CMOS image sensors [75-77].

3.2.3 Data storage

Data storage implementation, the most transistor-consuming function for a pixel-level processing element, affects most other aspects of the design [28]. The architecture must provide the storage capacity that would normally be found in a frame buffer for conventional store-and-process systems. Each application executed on the architecture has a minimum data storage threshold, typically consisting of the input image, the output image, and processing overhead. The data storage is also constrained by the minimum bit-precision required for accuracy. The architecture can utilize the six-transistor SRAM cell as the foundation of memory in the processing element. The instruction set architecture of the focal plane array imager requires a three-ported register file organization [32, 78]. In addition, models have been developed to predict silicon area usage for register file configurations [79, 80].

3.2.4 Digital processing

Once the pixel information is represented and stored digitally, the programmable processing core can implement various image applications. Image processing applications are categorized into (1) point operations, (2) local operations, and (3) global operations

[44]. A point operation such as thresholding occurs at the individual pixel level. A local operation such as smoothing requires knowledge of an individual pixel and its immediate neighbors. A global operation such as histogramming uses all pixel data from the image. These operations form the basis of applications found in imaging systems. To illustrate the processing sequence required for a typical imaging system of (N x N) pixels, Figure 17(a) describes the processing tasks, Figure 17(b) gives example applications at each stage, and Figure 17(c) quantifies the amount of data required for processing [45]. In Figure 17(c), the variable b represents the number of bits per pixel and N represents the image resolution on side for a square image. Point operations are the first stage, and local operations include the first three stages. The fourth and fifth stages are global feature-measuring operations [46]. The first three stages represent a significant proportion of the computational workload. For example, the digital image signal multiprocessor [40] reports 53% of its execution time for preprocessing, which includes tasks from the first three stages such as noise reduction, smoothing, and segmentation. These stages are candidates for processing at the pixel level.

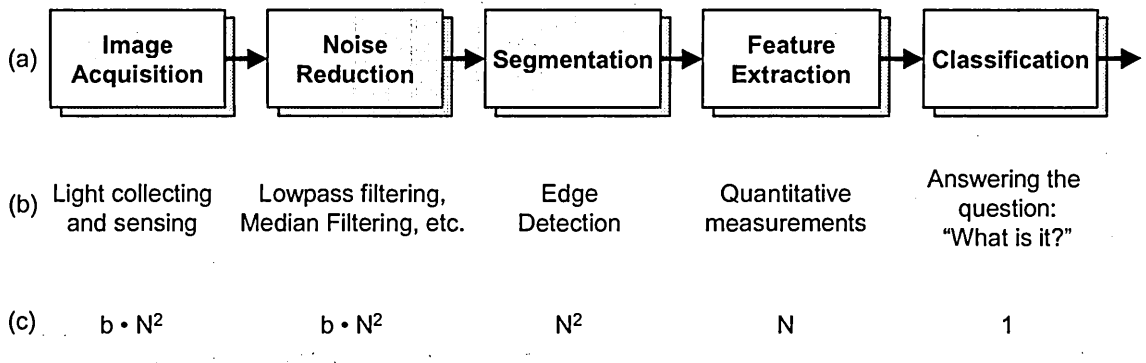


Figure 17: Typical image processing sequence (a) processing tasks (b) examples and (c) amount of data

Several architectures, such as the Near Sensor Image Processing (NSIP) [27], the Programmable and Versatile Large Size Artificial Retina (PVLSAR) [28], and the Simple and Smart Sensory Processing Elements (S^3PE) [29], follow the pixel-level model to perform early image processing in the focal plane. However, these architectures utilize bit-serial processing techniques with limited memory, which can either restrict processing to binary images or require multiple cycles to perform a single instruction on data words. This may prevent the implementation of some early image processing algorithms. However, the integration of additional functional units at the pixel level enables the versatility to execute a broader set of applications.

3.3 Application suite analysis

This section describes the technique to characterize the performance of the focal plane architecture on an image application suite. A focal plane architectural simulator is used to implement the various algorithms. A description is given for each application. The performance is calculated and hardware constraints are determined for the application suite.

3.3.1 Focal plane architectural simulator

Applications for the focal plane architecture can be programmed using the SIMD Pixel Processor (SIMPil) Simulator [34]. This software tool is a windows-based instruction level simulator, running on a PC platform. The SIMPil Simulator allows editing, assembling, executing, and debugging parallel image applications in a single

integrated workbench. The current version of the SIMPil Simulator is available on the download page.

An image processing application is executed on the architectural simulator to provide a dynamic workload. Each application is implemented using an instruction set architecture (ISA) that corresponds to the available functional units within the digital pixel. The number of pixels per processing element is assigned in the application. The simulator is instrumented to measure the operand resolution for each instruction, the distribution for instructions issued in parallel, the concurrency level of the processing elements, and the number of processor cycles required for execution. An extended description of the focal plane architectural simulator can be found in Appendix A.

3.3.2 Description of applications

The grayscale (8-bit) front-end applications, shown in Table 12, provide workload characteristics for image enhancement and image analysis. The enhancement and analysis applications of median filtering, convolution, and morphological processing represent the typical early image processing sequence of: (1) noise removal, (2) smoothing, and (3) segmentation. The workload characteristics for these applications are used to make efficient architectural choices for processing in the focal plane. This section briefly discusses each application.

Table 12: Selected early image processing applications

Applications	Description
Image Enhancement	
Median Filtering	• Removes impulse noise from an image while preserving spatial resolution.
Convolution	• Performs different filtering operations, such as shadowing, smoothing, and edge-detection.
Image Analysis	
Morphological Processing	• Performs feature extraction and segmentation of binary or grayscale images such as inside edge detection.

3.3.2.1 Median filtering

Median filtering (MED) is useful to remove impulse noise from an image while preserving spatial resolution. The algorithm is a rank-order filter [57] that replaces each pixel in the image with the median value in the window. Generally, a window size is selected to generate a rank-order filter with odd length. A larger window size increases the severity of the median filtering effect [58]. This implementation performs a 2-D nonseparable rank and selects the median value for a 3 x 3 window.

3.3.2.2 Convolution

Convolution-based filtering (CONV) has been implemented to perform different filtering operations, such as shadowing, smoothing, and edge detection [58]. The filter mask elements are broadcast one at a time to every PE. All calculations requiring the

mask element are performed before the next element is broadcast. Each PE multiplies the mask element by the corresponding pixel value from the original image and accumulates the result. Values are accumulated in a spiral pattern that places the final result in the center pixel of the filter mask. A 3 x 3 filter mask is used for this implementation.

3.3.2.3 Morphological processing

Morphological image processing refers to the study of the topology or structure of objects from their 2D spatial representation [59]. Binary or grayscale images are morphologically transformed by passing a structuring element over the image in a process similar to convolution. At each pixel position, a specified logical function is performed between the structuring element and the underlying image. For grayscale images, erosion is the minimum pixel value in the structuring element, and dilation is the maximum pixel value in the structuring element [58]. Depending upon the size and content of the structuring element, different effects such as inside edge detection (IED) can be produced from erosion and dilation operations. A 3 × 3 structuring element has been used to implement the morphological operations.

3.3.3 Application performance

The execution time was calculated for application suite described in Section 3.3.2. The execution time is determined with reference to a 10 MHz target platform This clock frequency addresses both the speed of analog components in each processing element and power density limitations for the high utilization factor of PEs in the array. Simulations were run with a system resolution equal to Quad-CIF (176 pixels x 144 pixels). This

resolution is one specification of the H.261 and H.263 video codec standards of the International Telecommunications Union (ITU) [63]. However, the execution time of the focal plane architecture is independent of the image dimension due to the parallel data processing.

In Table 13, the total execution times of the early image processing sequence generally follow the increase of the PPE factor with a baseline of the 1 PPE case. Median filtering is the dominant component of the processing sequence. The median filtering application for 16 PPE is not as efficient as the versions for 1 PPE and 4 PPE; therefore it does not follow the trend of the PPE factor. The total execution time for each PPE is within real-time constraints of 30 frames/sec (33.3ms per frame).

Table 13: Total execution time for Quad-CIF focal plane architecture implementations using a 10 MHz clock

PPE	Application Execution Cycles				Total Execution Time (μs)
	MED	CONV	IED	**Total**	
1	385	56	84	525	**52.5**
4	1531	183	238	1952	**195.2**
16	13531	826	979	15336	**1533.6**

Using the execution time, the sustained throughput $Throughput_{sust}$, measured in billion operations per second, is determined for the focal plane architecture as follows:

$$Throughput_{sust} = \frac{IC \cdot U \cdot N_{PE}}{t_{exec}} \qquad \left[\frac{Gops}{s}\right] \tag{1}$$

where IC is number of parallel instructions issued to the PE array during the application. The system utilization U is calculated as the average number of active processing

elements determined from the simulator's concurrency meter. N_{PE} is the number of processing elements in the PE array and is determined from the following formula:

$$N_{PE} = \frac{\text{system\_resolution}}{PPE} \tag{2}$$

where PPE is the number of pixels in each processing element (1, 4, or 16). For $PPE > 1$, pixels are arranged in a square (i.e. 2 x 2 or 4 x 4). The target system resolution is QCIF (176 x 144 pixels). The values for N_{PE} are shown in Table 14.

Table 14: Focal plane processor array characteristics for QCIF resolution

PPE	PE Array Dimensions		Total Number of PEs
	X	Y	
1	176	144	25344
4	88	72	6336
16	44	36	1584

The sustained throughput has been calculated for focal plane architecture with 1 PPE, 4 PPE, and 16 PPE implementations (Table 15). The maximum number of PE instructions executed equals ($IC \bullet N_{PE}$). The utilization is derived from the weighted average of executing each application. The sustained throughput achieved by the focal plane architecture implementations exceeds the values achieved by a traditional DSP architecture despite a significant difference in clock frequency. For example, the throughput specification of the TI TMS320C6211B DSP chip running at 150 MHz is 1.2 Gops/s, and the throughput specification of the TI TMS320C6411 DSP chip running at 300 MHz is 2.4 Gops/s [22]. Although operating at 10 MHz, the focal plane architecture

implementations for 1 PPE, 4 PPE, and 16 PPE increase sustained throughput by factors of 65x, 16x, and 5x respectively compared to the TI TMS320C6411 chip. These factors are doubled when comparing against the TI TMS320C6211B chip.

Table 15: Application characterization for focal plane architecture implementations using a 10 MHz clock with Quad-CIF resolution

PPE	Execution Time (μs)	Maximum PE Instructions Executed	Utilization	Sustained Throughput (Gops/s)
1	52.5	9,529,344	85.56%	**155**
4	195.2	8,838,720	84.95%	**38**
16	1533.6	19,209,168	87.91%	**11**

3.3.4 Application hardware constraints

The hardware must satisfy both the data storage and the data precision constraints to execute the algorithms. Table 16 shows the architectural design parameters for each processing element determined by the application suite. Register requirements are derived from the code for the implemented algorithm and include twice the PPE (to store the input and output images) plus overhead for intermediate calculations. Datapath requirements are determined from the measured operand resolutions for each instruction during simulation. With an overhead of 2 registers, convolution required the fewest total registers, [(2 • PPE) + 2], but the widest datapath (12 bits) to accommodate the successive multiply-accumulates of pixel values with the corresponding 3 x 3 mask values. The other applications only required the original grayscale resolution of 8 bits. Median filtering required [(2 • PPE) + 7] registers to store and sort the pixels in the 3 x 3 window. A sufficient architecture capable of executing all four applications would require [(2 •

PPE) + 7] registers with a 12-bit datapath. These values are used to select the sizes of both the ALU and the register file in each processing element.

Table 16: Application constraints for focal plane processor implementation

Application	Registers required	Datapath required
Median Filtering	(2 • PPE) + 7	8 bits
Convolution	(2 • PPE) + 2	12 bits
Inside Edge Detection	(2 • PPE) + 3	8 bits
Full Application Suite	**(2 • PPE) + 7**	**12 bits**

3.4 Projected area

Because of limited chip resources in a focal plane architecture, silicon area usage within an integrated digital pixel is a critical design factor. A pixel design tool provides a common context for component area models within an integrated pixel processing array [81]. Area models based upon implementations of detector array circuitry and CMOS functional units are used to project silicon area for the architecture using different fabrication technologies.

3.4.1 Component area models

Area models are developed from fabricated analog and digital components. Area projections of the photodiode and analog-to-digital conversion are based upon the CMOS focal plane array [41]. This 8 x 8 array of Si CMOS detectors, fabricated in 0.8 μm technology, incorporates a current input first-order sigma-delta analog-to-digital converter at each pixel. The transistor circuitry is scaled to feature sizes ranging from

250nm to 100nm. Area projections for memory and digital circuitry are based upon the SIMPil16 focal plane architecture [82] that maps 16 pixels to each processing element. SIMPil16 is a 16-bit implementation fabricated on MOSIS HP 0.8 μm (1.0 μm drawn) CMOS technology. Selected functional units of a SIMPil16 PE used by the pixel-level processing architecture include: (1) ALU, (2) register file, (3) decoder, (4) bus driver, (5) sleep unit, and (6) communication unit. The original areas for these functional units are scaled to feature sizes ranging from 250nm to 100nm for area projections of the focal plane processor. Bit slicing is used to adjust both the ALU area and the register file area for a reduced datapath width. In addition, the register file area is adjusted to correspond to the number of words in the design.

3.4.2 Processing element area

Silicon area allocation is a significant issue because in single-level VLSI, the photodiode, the analog signal conditioning, the analog-to-digital converter, the memory, and the digital processing core compete for silicon area in a small replicated processing element (Figure 14). A P-N photodiode is used in the CMOS focal plane array [41]. The analog signal conditioning is the sampling capacitor. The ADC is a first-order sigma-delta circuit. Memory is a register file with the number or registers based upon the formula required to execute the full application suite (Table 16). The digital processor contains the functional units mentioned in the previous section.

Processing element area has been projected using various fabrication technologies for 1 PPE (Table 17), 4 PPE (Table 18), and 16 PPE (Table 19). By observing the component area trends for 1 PPE (Figure 18), 4 PPE (Figure 19), and 16 PPE (Figure 20),

the digital components (digital processor, memory, ADC) benefit from technology scaling. However, the analog components (photodiode and analog signal conditioning) do not see the same benefit [11, 12]. The photodiode has a fixed area requirement to ensure acquisition of light. The analog signal conditioning (sampling capacitor) also has a fixed area cost. As the PPE increases, a larger percentage of the silicon area within the processing element is used for data acquisition and data storage.

Table 17: Area of a single processing element using 1 PPE

Technology	Components					Total Area (mm^2)
	Photodiode (mm^2)	Analog Signal Conditioning (mm^2)	ADC (mm^2)	Memory (mm^2)	Digital Processor (mm^2)	
250nm	0.0046	0.0024	0.0008	0.0287	0.0453	**0.082**
180nm	0.0046	0.0024	0.0004	0.0149	0.0235	**0.046**
150nm	0.0046	0.0024	0.0003	0.0103	0.0163	**0.034**
130nm	0.0046	0.0024	0.0002	0.0078	0.0122	**0.027**
120nm	0.0046	0.0024	0.0002	0.0066	0.0104	**0.024**
100nm	0.0046	0.0024	0.0001	0.0046	0.0073	**0.019**

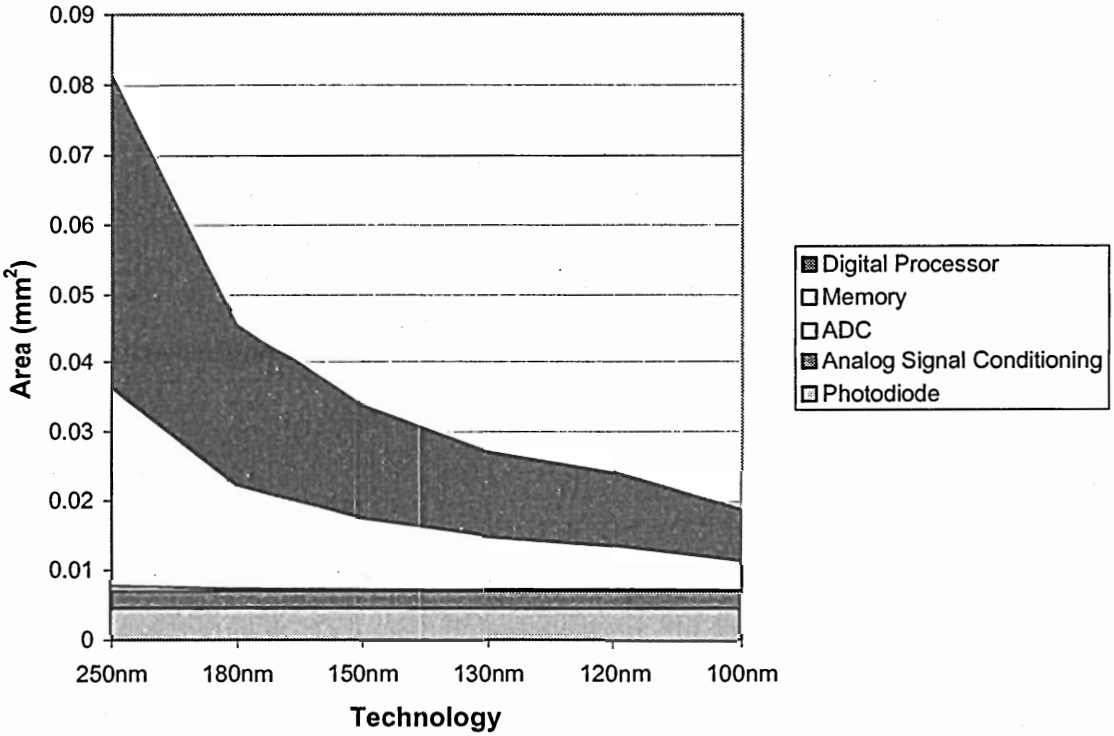


Figure 18: Component area trend for 1 PPE with decreasing feature size

Table 18: Area of a single processing element using 4 PPE

Technology	Components					Total Area (mm^2)
	Photodiode (mm^2)	Analog Signal Conditioning (mm^2)	ADC (mm^2)	Memory (mm^2)	Digital Processor (mm^2)	
250nm	0.0185	0.0096	0.0030	0.0478	0.0453	**0.124**
180nm	0.0185	0.0096	0.0016	0.0248	0.0235	**0.078**
150nm	0.0185	0.0096	0.0011	0.0172	0.0163	**0.063**
130nm	0.0185	0.0096	0.0008	0.0129	0.0122	**0.054**
120nm	0.0185	0.0096	0.0007	0.0110	0.0104	**0.050**
100nm	0.0185	0.0096	0.0005	0.0077	0.0072	**0.043**

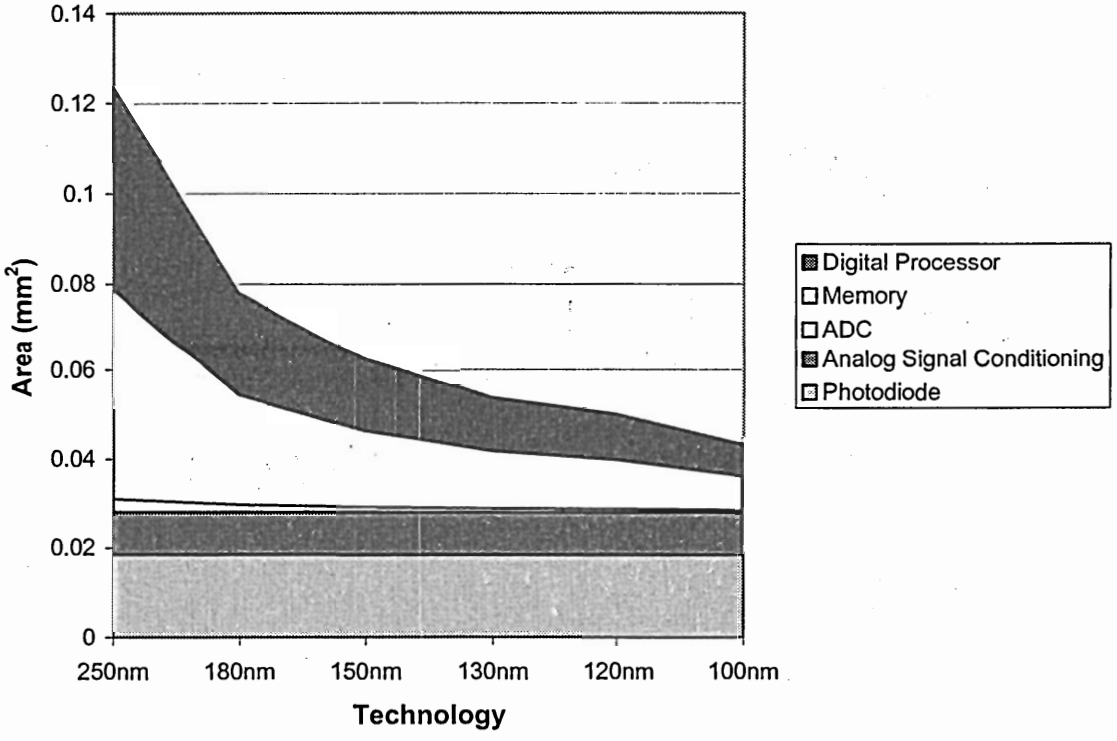


Figure 19: Component area trend for 4 PPE with decreasing feature size

Table 19: Area of a single processing element using 16 PPE

Technology	Components					Total Area (mm^2)
	Photodiode (mm^2)	Analog Signal Conditioning (mm^2)	ADC (mm^2)	Memory (mm^2)	Digital Processor (mm^2)	
250nm	0.0739	0.0384	0.0121	0.1244	0.0453	**0.294**
180nm	0.0739	0.0384	0.0063	0.0645	0.0235	**0.207**
150nm	0.0739	0.0384	0.0044	0.0448	0.0163	**0.178**
130nm	0.0739	0.0384	0.0033	0.0336	0.0122	**0.161**
120nm	0.0739	0.0384	0.0028	0.0287	0.0104	**0.154**
100nm	0.0739	0.0384	0.0019	0.0199	0.0072	**0.141**

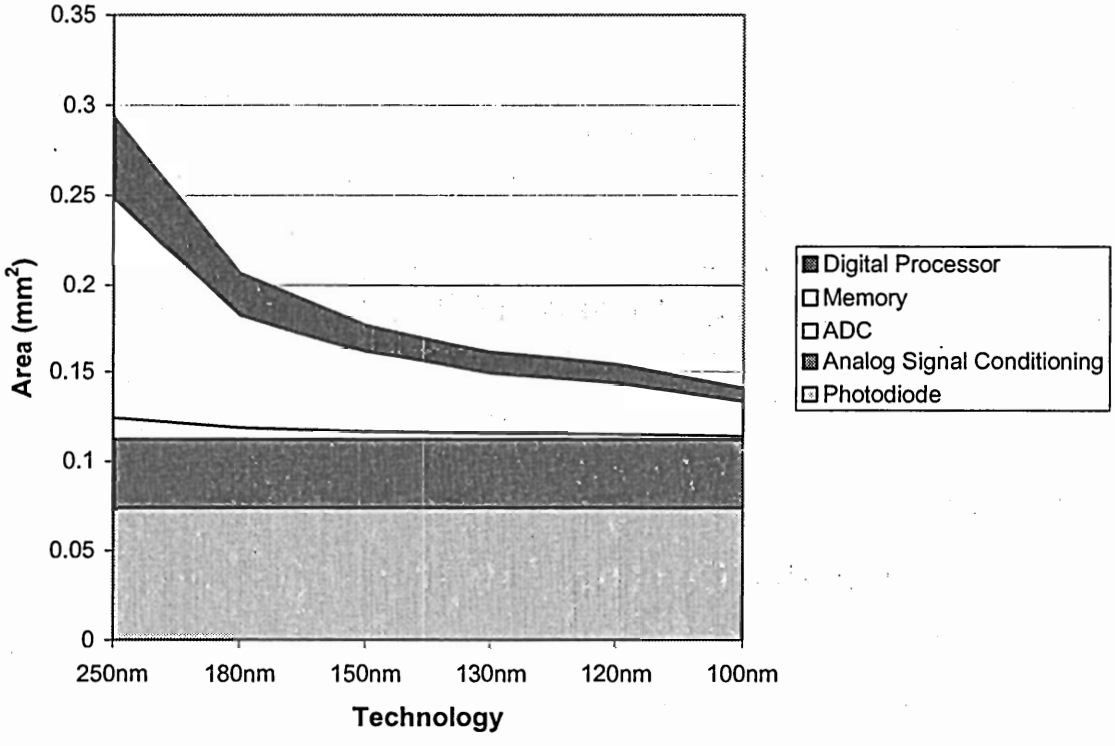


Figure 20: Component area trend for 16 PPE with decreasing feature size

3.4.3 Focal plane processor array area

Portable devices with an embedded focal plane architecture must reduce their silicon area cost for a compact design. Excessive die sizes would limit the realization of these systems. The chip area is tightly coupled to the system resolution because of the integrated functionality within each processing element. For this analysis, the desired image resolution for processing is grayscale (8-bit) Quad-CIF (176 pixels x 144 pixels). The Quad-CIF resolution is one specification of the H.261 and H.263 video codec standards of the International Telecommunications Union (ITU) [63]. The ITRS provides guidelines for affordable chip sizes for each technology [36, 37].

Using the array dimensions for different PPE values (Table 14) and the processing element areas (Table 17, Table 18, Table 19), the focal plane processor array area has been projected for various technologies (Figure 21). Using the ITRS guideline for manufacturing affordability, the 1 PPE implementation does not project to the die size target. The 4 PPE implementation exceeds the guideline by approximately 20% starting with the 150nm technology and reaches the ITRS guideline using 100nm technology. The 16 PPE implementation meets the guideline for all the selected technologies except 250nm, allowing it to be implemented using a less aggressive and less expensive fabrication technology. Because detector area dominates as feature size shrinks, denser detector technology could substantially reduce chip area, making the chip more affordable.

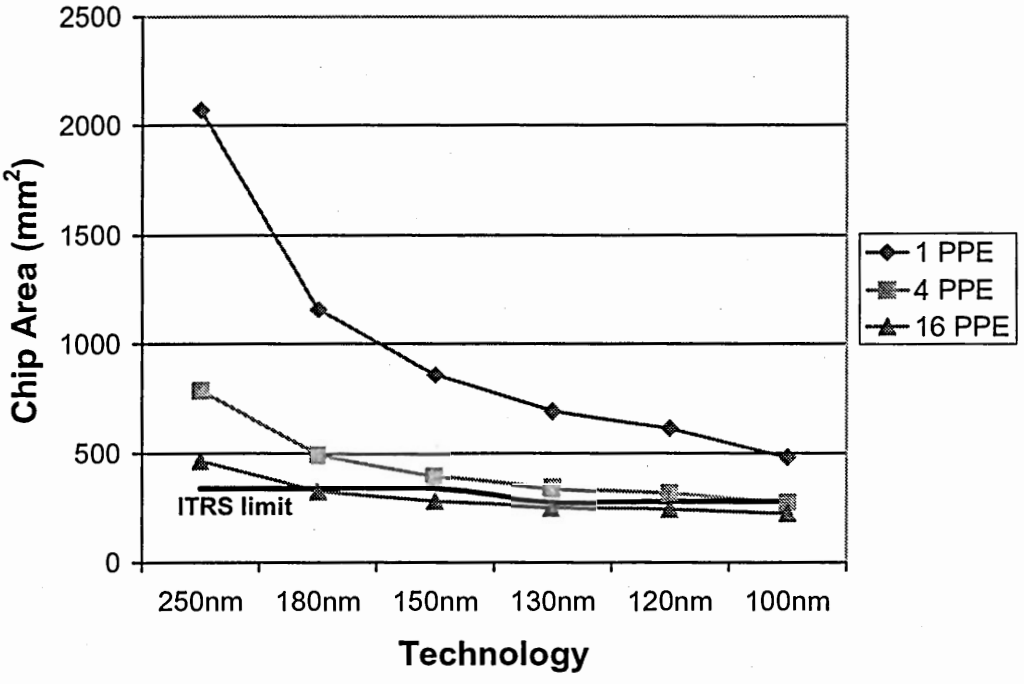


Figure 21: Projected area for a QCIF processor array implementation

3.5 Projected power

Portable devices with an embedded focal plane architecture must provide a meaningful battery life. Setting the minimum time between battery charges (*MTBC*) to a desired value translates into a limit for maximum power consumed during operation for a fixed battery energy. Typical double barrel, NiCd AA sized batteries (3.6 V at 700 mA • hours) have an energy capacity of about 10 Watt • hours. Therefore for $MTBC \geq 3$ hours, the power should not exceed 3 Watts.

A Technology Scenario Analyzer (TeSA) was developed to project the power consumption of the digital components within the focal plane architecture using different fabrication technologies [35]. The power consumption has been projected for fabrication technologies ranging from 250nm to 100nm using a 10 MHz clock (Figure 22). The

maximum power limit for each technology is used from ITRS specifications for portable battery operation [36, 37]. The 1 PPE implementation consumes the most power because of the large number of processing elements required to provide QCIF resolution (176 x 144 = 25344). It requires technology generations of 130nm or smaller to satisfy the ITRS power constraint. However, it may be too close to the limit to account for the analog components. The 4 PPE implementation satisfies the constraint for all technologies except 250nm and 180nm by reducing the required number of processing elements (88 x 72 = 6336). The power consumed is reduced by a factor of 3.9 (i.e. 74%) when compared to the 1 PPE implementation. The 16 PPE implementation, which uses the least number of processing elements (44 x 36 = 1584), consumes the least power and satisfies the constraint for all the chosen technologies. The overall reduction factor is 18 (i.e. 94%) when compared to the 1 PPE implementation.

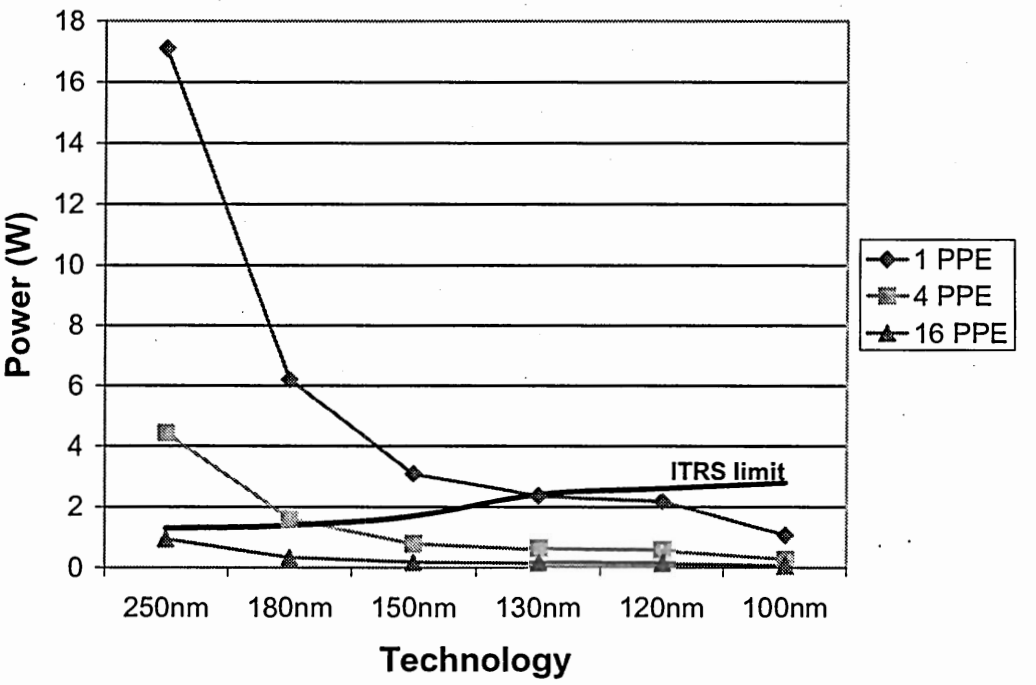


Figure 22: Projected power for a QCIF processor array implementation using a 10 MHz clock

3.6 Resource efficiency analysis

Higher ratings in area-time efficiency, area efficiency, and energy efficiency metrics are desired for image processing systems because of technology limitations such as interconnect wiring density and heat extraction [62]. This section examines the area-time efficiency, area efficiency, and energy efficiency of the focal plane architecture implementations. Comparisons for TI DSP chips are based upon application benchmarks and chip specifications described in Appendix B.

3.6.1 Area-time efficiency

The area-time efficiency is defined as:

$$\eta_{A-T} = \frac{1}{Area \times t_{exec}} \qquad \left[\frac{1}{s \cdot mm^2}\right] \tag{3}$$

where the reciprocal is taken for the product of the area for each system and the execution time for the imaging processing sequence (Table 13). Because a system should execute in the shortest time using the smallest area, the optimal solution is determined by maximizing the reciprocal product.

The $(A \bullet T)^{-1}$ efficiency of focal plane architecture implementations using 1 PPE, 4 PPE, and 16 PPE, has been calculated for different fabrication technologies (Figure 23). The area includes the data acquisition and analog-to-digital conversion circuitry. Two TI DSP chips, the TMS320C6211B and TMS320C6411, are shown for comparison. The 16 PPE implementation is comparable to the TMS320C6211B operating 150 MHz. The 1 PPE and the 4 PPE implementations provide a 3.5x and 1.8x factor increase respectively when compared to the TMS320C6411 operating at 300 MHz. This demonstrates that a focal plane architecture can be utilized as an embedded component to deliver efficient processing for common image applications. These applications can consume over half of the processing for an image.

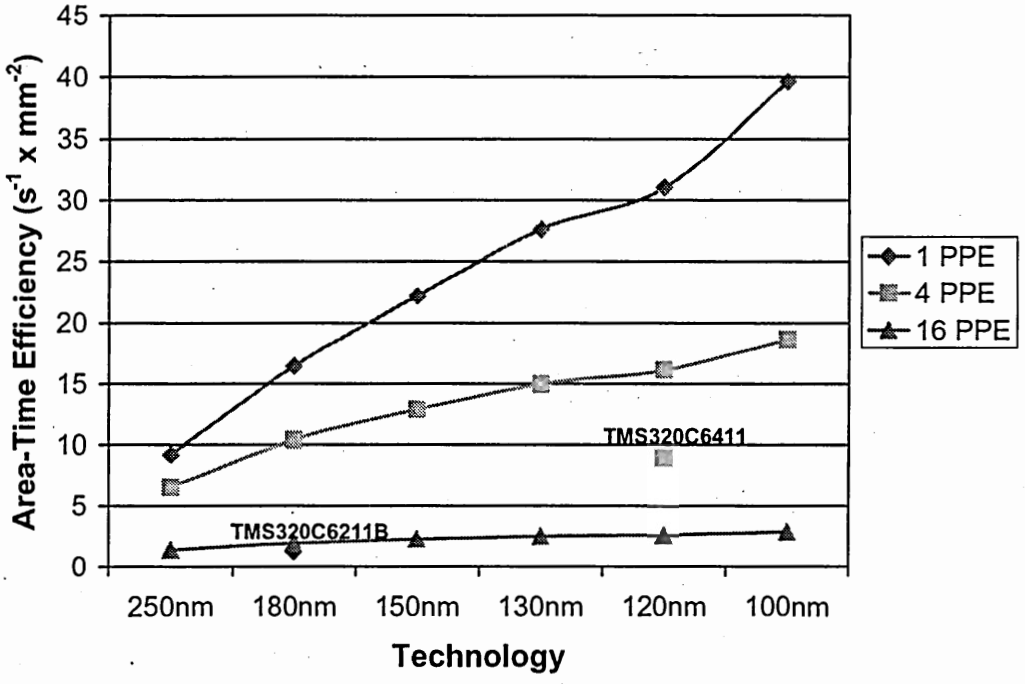


Figure 23: Comparison of (area • time)$^{-1}$ efficiency for image processing architectures

3.6.2 Area efficiency

Area efficiency is defined as the number of operations executed per second per unit area:

$$\eta_A = \frac{Throughput_{sust}}{Area} \qquad \left[\frac{Mops}{s \cdot mm^2}\right] \tag{4}$$

Area efficiency has been established as an important figure of merit for system-on-a-chip architecture design [14]. Because of limited chip resources in a focal plane architecture, silicon area usage within an integrated digital pixel is a critical design factor. Higher area efficiency implies better component utilization within the architecture.

The area efficiency of focal plane architecture for implementations using 1 PPE, 4 PPE, and 16 PPE, has been calculated using different fabrication technologies (Figure

24). Generally, an increase in PPE decreases the area efficiency of the focal plane architecture. However, there is a significant gain when compared to a DSP architecture in the same fabrication technology. For the TMS320C6211B, which uses 180nm technology, the area efficiency increases by factors of 81x, 47x, and 20x for 1 PPE, 4 PPE, and 16 PPE respectively. For the TMS320C6411, which uses 120nm technology, the increase factors are 56x, 27x, and 10x for 1 PPE, 4 PPE, and 16 PPE respectively. This result is significant because the gain in efficiency is for common image enhancement and image analysis tasks that can represent a significant portion of the total operations within an image processing sequence. In addition, the area efficiency for implementations of the focal plane architecture includes the data acquisition and analog-to-digital conversion circuitry, which is not available with the DSP architecture. This does not suggest that the architecture replaces a DSP but can handle certain common image enhancement and image analysis tasks more efficiently through specialization.

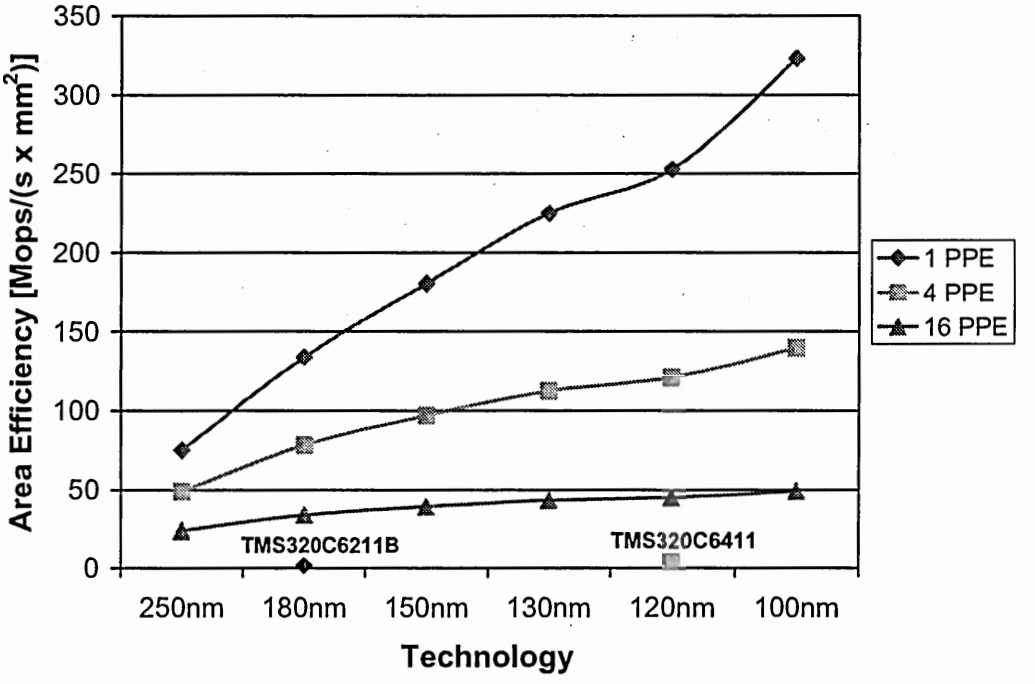


Figure 24: Comparison of area efficiency for image processing architectures

3.6.3 Energy efficiency

Energy efficiency is defined as the number of operations executed per unit energy:

$$\eta_E = \frac{Throughput_{sust}}{Power} \qquad \left[\frac{Mops}{Joule}\right] \tag{5}$$

Previous work [83, 84] has illustrated the validity of energy efficiency for fixed throughput computation in uniprocessor systems. The validity is extended to massively parallel embedded-focal plane architectures by introducing system utilization in (1) for calculating the throughput. Increasing energy efficiency implies enhancing the sustainable battery life in portable devices. Minimizing power dissipation translates into minimizing energy per operation.

The energy efficiency of a focal plane architecture using 1 PPE, 4 PPE, and 16 PPE, has been calculated for the digital components using different fabrication technologies (Figure 25). The 16 PPE implementation delivers the highest energy efficiency. The 4 PPE implementation is slightly lower than the 1 PPE implementation. This may indicate that the application suite optimizes slightly better for using a single pixel in a processing element. Two TI DSP chips, the TMS320C6211B and TMS320C6411, are also shown for comparison. For the TMS320C6211B with 180nm technology, the energy efficiency increases by an average factor of 11x when using implementations of the focal plane architecture. For the TMS320C6411 with 120nm technology, the average increase is 2.9x. This demonstrates the potential to extend battery life in portable devices.

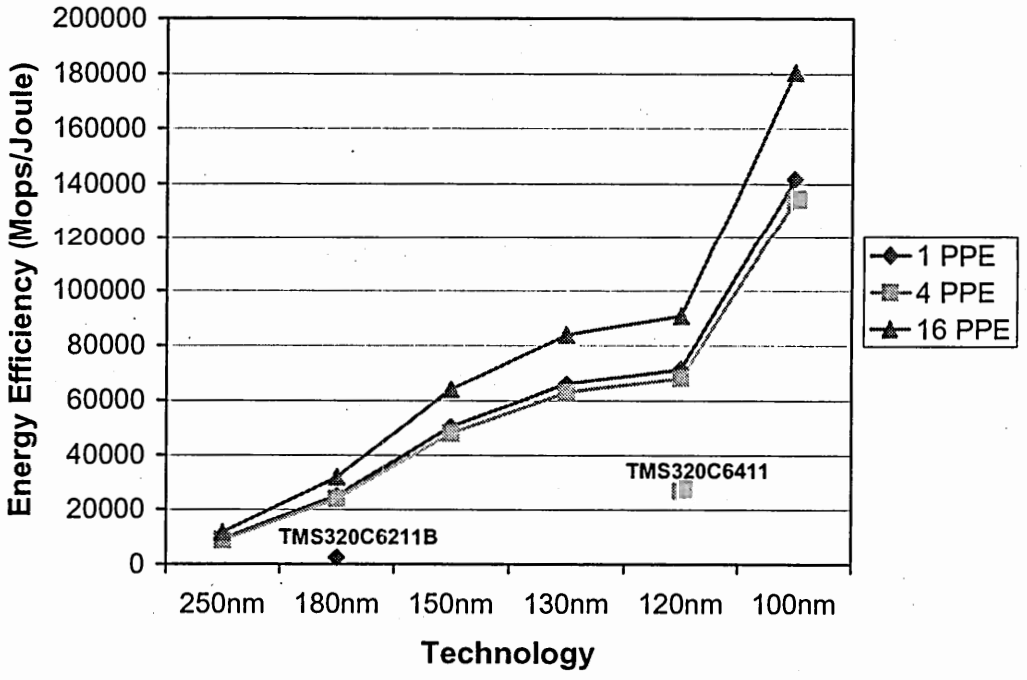


Figure 25: Comparison of energy efficiency for image processing architectures

3.7 Conclusion

Recently, a new dynamic has emerged with users desiring mobile productivity, where image processing functionality is encapsulated within portable devices containing embedded hardware. Portable imaging products will benefit from the monolithic integration of photodetectors, analog-to-digital converters, digital processing, and data storage to perform image acquisition and computation. Processing on the focal plane utilizes data-parallelism naturally found in image applications. Technological advances in device fabrication and integration are enabling the development of these monolithic focal plane architectures with the potential for improved performance, efficiency, and cost versus traditional imaging architectures.

This chapter presents the analysis of a mixed-signal focal plane processing architecture. Key performance and cost metrics for focal plane architectures include execution time, system throughput, chip area, power consumption, area-time efficiency, area efficiency, and energy efficiency. Although the area-time efficiency has been traditionally used, it does not provide guidance for addressing energy consumption. Area efficiency and energy efficiency are important figures of merit when evaluating a focal plane architecture. The choice of both fabrication technology and pixels per processing element impact the system design, particularly for portable devices. Die size for manufacturing cost and power consumption for battery life should follow ITRS guidelines. In addition, higher ratings in area efficiency and energy efficiency are required because of technology limitations such as interconnect wiring density and heat extraction.

The performance, area, and power were projected for focal plane architecture implementations using 1 PPE, 4 PPE, and 16 PPE. This architecture is more area and energy efficient when compared to traditional TI DSP chips. However, the design problem is not strictly building the system with the highest performance, but delivering the required performance with the lowest cost. Therefore, a 4 PPE or 16 PPE with a less aggressive technology may provide a better implementation solution.

CHAPTER IV

STATISTICAL EXPERIMENTAL DESIGN FOR PHOTODETECTOR MODELING

Summary

An embedded imaging system can be created by the monolithic integration of image acquisition with image processing. The most significant issue is silicon area allocation, since in single-level VLSI, the photodetector, the analog-to-digital converter (ADC), the digital processing core, and the memory compete for silicon area in a small replicated tile. This chapter presents the statistical experimental design of a photodetector for use within a mixed-signal processing element. The device performance, measured by the signal-to-noise ratio (SNR), was derived based upon the physical implementation of the photodiode. Key input parameters selected for designing the photodiode included the area, integration time, acceptor density, donor density, temperature, and reverse bias. A regression model was developed using a 2^6 full-factorial experimental design and was validated using analysis of variance (ANOVA). Using the model, a relationship between photodiode area and integration time was determined enabling area reduction of $212\mu m^2$ for every 1ms increase in the integration time while maintaining SNR. This relationship significantly impacts the design of a focal plane processor by trading time for limited silicon resources.

4.1 Introduction

Inexpensive imaging chips are being incorporated into a wide range of devices such as video and still cameras, laptops, portable data assistants (PDAs), and even children's toys. One technique is the use of smart pixels [85] to create an embedded imaging system. A processing element (PE) tile is replicated to form a focal plane array imager with integrated analog to digital conversion and Single-Instruction Multiple-Data (SIMD) processing. This moves the computation closer to the data acquisition to reduce the storage requirements of the system. Data parallel processing on the focal plane offers superior performance and efficiency for front-end image processing applications.

However, digitizing and processing a pixel at the detection site presents new design challenges [9, 41, 42]. The most significant issue is silicon area allocation, since in single-level VLSI, the photodetector, the analog-to-digital converter (ADC), the digital processing core, and the memory compete for silicon area in a small replicated tile. The wide range of design techniques and metrics for these components leads to a complicated design problem, particularly for the integrated optoelectronics. Steady advances in both semiconductor technology [36] and detector design [47] support increasingly complex systems, but offer a moving target to design efforts. The photodetector must retain sufficient area to satisfy system requirements. Narrowing the design space requires an effective measure of merit relating silicon area to performance.

This chapter presents the statistical experimental design (SED) of a photodetector for use within a mixed-signal processing element. The signal-to-noise ratio (SNR) was derived for a photodetector based upon equations that describe the physical implementation. Key input parameters selected for designing the photodiode included

area (A_{diode}), integration time (t_{int}), acceptor density (N_a), donor density (N_d), temperature (T), and reverse bias (V_{bias}). A regression model was developed using a 2^6 full-factorial experimental design. The model was validated using analysis of variance (ANOVA).

The regression model can be used to accurately describe the SNR behavior for the photodiode a function of the input parameters. The model reduces the equation complexity for determining photodiode SNR. The significant parameters, in order of importance, are diode area, integration time, donor density, reverse bias, and then the combination of reverse bias and donor density. By using the two most significant parameters, photodiode area and integration time, a tradeoff is established where increased integration time reduces the required area to meet SNR constraints for bit precision. This relationship significantly impacts the design of a focal plane processor by trading time for limited silicon resources.

The organization of this chapter is as follows. Section 4.2 provides the background for monolithic integration of a mixed-signal processing element. Section 4.3 describes the technique to model the physical response of a photodiode. Section 4.4 analyzes the models to determine candidate photodiodes for a mixed-signal processing element. Finally, Section 4.5 concludes the discussion on statistical experimental design of a mixed-signal processing element.

4.2 Background

Numerous architectures have been developed for image processing systems [2, 3]. Efficient handling of the two-dimensional image data is a common issue among these designs. A natural solution to this issue is to convert the image to the digital domain and

process the spatially parallel data within the image plane [43]. This section describes the programmable digital pixel and the data acquisition component required for an imaging device.

4.2.1 Programmable digital pixel

A programmable digital pixel is formed by monolithically incorporating the data acquisition device, analog-to-digital conversion (ADC) circuitry, data storage, and digital processing circuitry within a processing element (PE). A block diagram of this processing tile [26] is shown in Figure 14. This tile element is replicated to form a focal plane array imager with integrated analog-to-digital conversion, nearest-neighbor communication, and SIMD processing. Several architectures, such as the Near Sensor Image Processing (NSIP) [27], the Programmable and Versatile Large Size Artificial Retina (PVLSAR) [28], and the Simple and Smart Sensory Processing Elements (S^3PE) [29], follow the pixel-level model to perform early image processing in the focal plane.

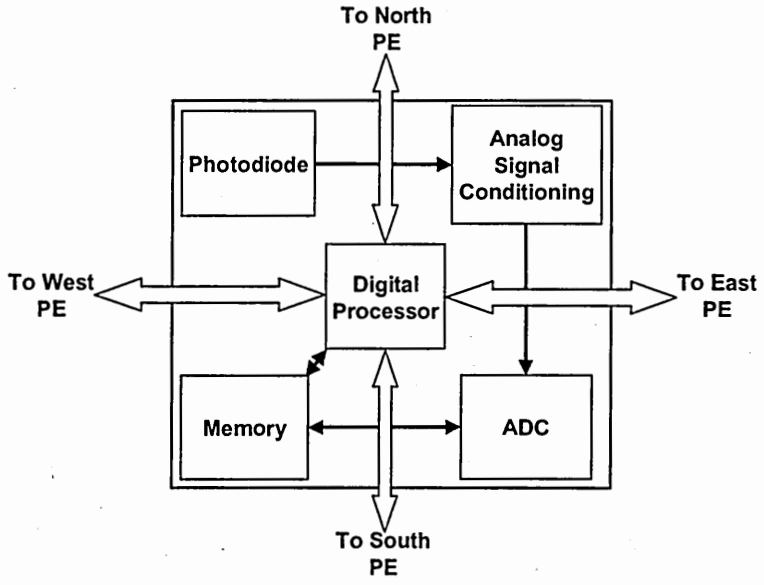


Figure 26: Programmable digital pixel

4.2.2 Data acquisition

The first stage performed by the digital pixel is the acquisition of the analog light intensity of an image. The relative response of the human eye correlates to the major color bands of the visible spectrum, which span from approximately 400nm to 700nm [64] and peaks at 555nm (yellow-green). Therefore, photodetector devices used in the digital pixel should provide reasonable performance across these wavelengths. Photodetectors absorb photons with energy greater than the material bandgap to generate a current proportional to the number of optically generated electron-hole pairs [30]. The quantum efficiency of the detector is the number of electron-hole pairs generated per incident photon for a given wavelength [30]. The spectral response, or responsivity, is the quantum efficiency over a range of light wavelengths [65]. Photodetectors made from silicon can detect light wavelengths up to approximately 1.1μm due to its bandgap energy of 1.12eV and provide reasonable performance over the visible spectrum (Figure 15).

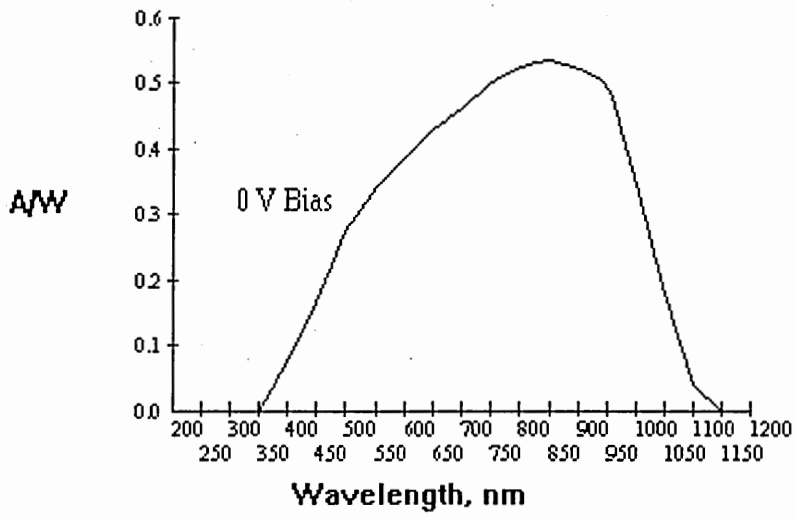


Figure 27: Spectral response of a typical silicon photodetector

Implementing photodetectors in a standard CMOS process enables the integration of other processing components at the detection site. Research efforts have been made to develop and evaluate these CMOS image sensors [12, 66, 67]. A noted development is the active pixel sensor (APS) [68], which has been implemented into a digital camera on a chip with the functionality of analog-to-digital conversion [69].

4.3 Model development

The signal-to-noise ratio for a P-N photodiode was derived using theoretical physical properties of silicon for 555nm wavelength under room light conditions [64]. Key input parameters selected for designing the photodiode included area (A_{diode}), integration time (t_{int}), acceptor density (N_a), donor density (N_d), temperature (T), and reverse bias (V_{bias}). However, the breadth of the design space creates a challenging design

problem to integrate into a digital pixel. Therefore, a 2^6 full-factorial experimental design using the factors in Table 20 was implemented to explore the full design space [86]. The regression model in equation (1) was developed from the *SNR(dB)* responses. The model consists of a constant with weighted values for each of the six inputs.

Table 20: Parameter ranges for statistical experimental design

	Minimum	Maximum
A_{diode} (cm^2)	2.5×10^{-7}	2.5×10^{-5}
t_{int} (ms)	1	30
N_a (cm^{-3})	1×10^{15}	1×10^{18}
N_d (cm^{-3})	1×10^{15}	1×10^{18}
T (K)	255	310
V_{bias} (V)	-3	0

$$
\begin{aligned}
SNR(dB) = 48.188792 &+ 10.018069 \times \left(\frac{A_{diode} - 1.2625 \times 10^{-5}}{1.2375 \times 10^{-5}} \right) \\
&+ 7.406919 \times \left(\frac{t_{int} - 1.55 \times 10^{1}}{1.45 \times 10^{1}} \right) \\
&+ 0.180578 \times \left(\frac{N_a - 5.005 \times 10^{17}}{4.995 \times 10^{17}} \right) \\
&- 1.338731 \times \left(\frac{N_d - 5.005 \times 10^{17}}{4.995 \times 10^{17}} \right) \\
&- 0.012122 \times \left(\frac{T - 2.825 \times 10^{2}}{2.75 \times 10^{1}} \right) \\
&- 0.568397 \times \left(\frac{V_{bias} + 1.5}{1.5} \right)
\end{aligned}
\tag{1}
$$

Analysis of variance (ANOVA) is a mathematical technique used to evaluate the statistical significance of the photodiode input parameters. The regression model

accurately depicts the behavior of the *SNR(dB)* response as a function of the input parameters and reduces the complexity for determining photodiode performance. The p-values from ANOVA (Table 21) show that diode area (A_{diode}), integration time (t_{int}), donor density (N_d), and reverse bias (V_{bias}) are significant factors (values near zero). Also, the combination of reverse bias and donor density ($N_d \cdot V_{bias}$) has a relationship that impacts the SNR. Using the F-ratio, the order of significance is area, integration time, donor density, reverse bias, and then the combination of reverse bias and donor density.

Table 21: Analysis of variance (ANOVA) for photodiode parameters

Source	Significance (p-value)	F-Ratio
A_{diode}	0.0000	2700.00
t_{int}	0.0000	1476.00
N_a	0.3542	0.88
N_d	0.0000	48.22
T	0.9502	0.00
V_{bias}	0.0051	8.69
$A_{diode} \cdot t_{int}$	0.9314	0.01
$A_{diode} \cdot N_a$	0.9927	0.00
$A_{diode} \cdot N_d$	0.9696	0.00
$A_{diode} \cdot T$	0.9919	0.00
$A_{diode} \cdot V_{bias}$	0.9999	0.00
$t_{int} \cdot N_a$	0.9899	0.00
$t_{int} \cdot N_d$	0.9880	0.00
$t_{int} \cdot T$	0.9907	0.00
$t_{int} \cdot V_{bias}$	0.9832	0.00
$N_a \cdot N_d$	0.1332	2.34
$N_a \cdot N_d$	0.9842	0.00
$N_a \cdot T$	0.6496	0.21
$N_d \cdot T$	0.9589	0.00
$N_d \cdot V_{bias}$	0.0099	7.28
$T \cdot V_{bias}$	0.9433	0.01

4.4 Analysis

This section presents the analysis and application of the regression model developed in the previous section. The system requirements provide the constraints for the photodiode. The two most significant factors, photodiode area and integration time,

provide a tradeoff for selecting an appropriate device for the mixed-signal processing element.

4.4.1 Photodiode requirements

Silicon area allocation is a significant issue for a digital pixel because in single-level VLSI, the photodiode, the analog signal conditioning, the analog-to-digital converter, the memory, and the digital processing core compete for silicon area. Each component must satisfy minimum system requirements with the remaining area allocated for optimal performance. The photodiode must provide enough bits of resolution for the analog-to-digital conversion (ADC). Using the following equation:

$$SNR(dB) = 6.02B + 1.76 \quad (2)$$

the required *SNR(dB)* has been determined for different bit precisions (Table 22).

Table 22: SNR requirements for ADC

Number of Bits	SNR(dB)
8	49.92
9	55.94
10	61.96

4.4.2 Diode area versus integration time

Although ANOVA identified several significant parameters, it is helpful to explore the design space from the two most significant parameters, photodiode area and integration time. The correlation matrix demonstrates that *SNR(dB)* increases as either A_{diode} or t_{int} increase, with area as the most dominant factor (Table 23). By selecting

typical values for the remaining parameters (N_a = 1 x 10^{15} cm^{-3}, N_d = 1 x 10^{18} cm^{-3}, T = 300K, V_{bias} = -2V), a tradeoff is shown between photodiode area and integration time (Figure 28). By increasing the integration time, the photodiode area can be reduced. This flexibility helps to reduce the silicon area cost for the digital pixel while satisfying the ADC constraints (Figure 29). The black region represents combinations of photodiode area and integration time that do not satisfy 8-bit precision.

Table 23: Correlation matrix for photodiode parameters

	A_{diode}	t_{int}	N_a	N_d	T	V_{bias}	SNR(dB)
A_{diode}	1.000	0.000	0.000	0.000	0.000	0.000	0.794
t_{int}	0.000	1.000	0.000	0.000	0.000	0.000	0.587
N_a	0.000	0.000	1.000	0.000	0.000	0.000	0.014
N_d	0.000	0.000	0.000	1.000	0.000	0.000	-0.106
T	0.000	0.000	0.000	0.000	1.000	0.000	-0.001
V_{bias}	0.000	0.000	0.000	0.000	0.000	1.000	-0.045
SNR(dB)	0.794	0.587	0.014	-0.106	-0.001	-0.045	1.000

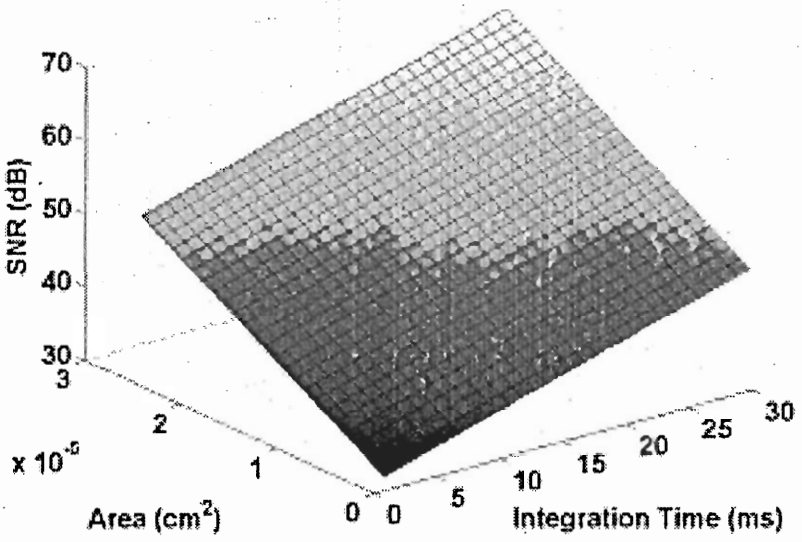


Figure 28: Photodiode response using area and integration time

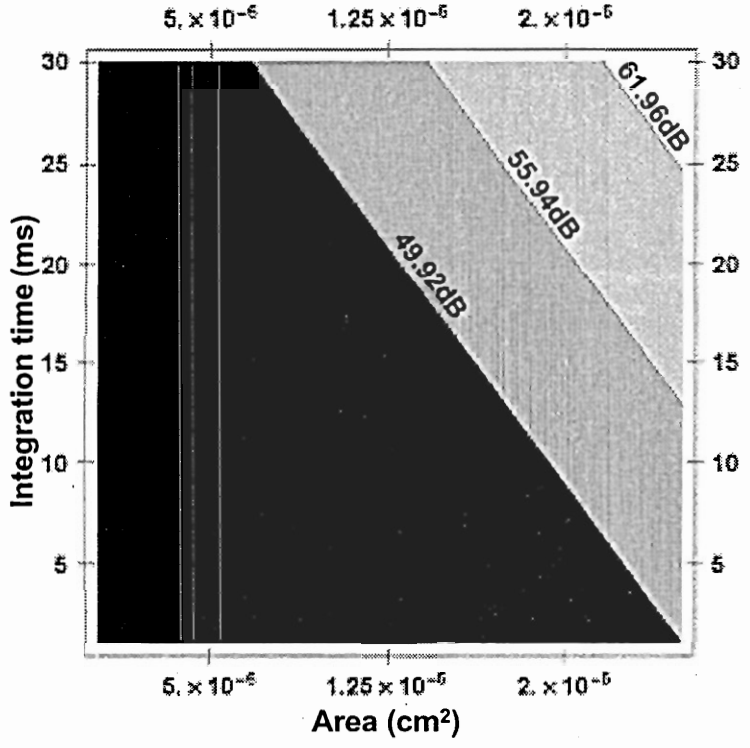


Figure 29: Tradeoff between area and integration time for different bit precisions

The regression model can be solved for (Table 24). Entries with "N/A" mean that the calculated value exceeded the photodiode area range of the statistical experimental design. The regression model has not been statistically validated for values outside the ranges in Table 20 and should not be used to predict those inputs. By selecting an integration time of 30ms, the photodiode area can be reduced to 7.27×10^{-6} cm^2 (27μm x 27μm) while satisfying an 8-bit resolution. This would enable 256-level grayscale digital processing for the embedded imaging system. The smaller photodiode area also allows more devices to be incorporated monolithically, thereby increasing the image resolution.

Table 24: Candidate photodiode areas for different integration times

Number of Bits	SNR(dB)	Photodiode Area Required (cm^2)		
		t_{int} = 5 ms	t_{int} = 15 ms	t_{int} = 30 ms
8	49.92	2.30×10^{-5}	1.67×10^{-5}	7.27×10^{-6}
9	55.94	N/A	2.42×10^{-5}	1.47×10^{-5}
10	61.96	N/A	N/A	2.21×10^{-5}

4.5 Conclusion

A system-on-a-chip, created by a tiled monolithic array of digital pixels, can utilize the anticipated technological improvements in fabrication. However, the limited silicon resources require fundamental tradeoffs among the functional areas within the digital pixel. The large design space of the photodetector creates a complex challenge for determining a sufficient device within the imaging system.

This chapter presents the statistical experimental design (SED) of a photodetector for use within a mixed-signal processing element. A regression model was developed

using a 2^6 full-factorial experimental design and was validated using analysis of variance (ANOVA). The regression model reduces complexity for determining photodiode SNR. Integration time can be increased to reduce photodiode area. This helps reduce the silicon area cost for integrating optoelectronics into an imaging system.

CHAPTER V

CONCLUSION AND FUTURE WORK

This dissertation investigates system-level design issues to integrate photodetectors, analog-to-digital converters, data storage, and digital processing into a monolithic imaging solution. Unique design challenges occur from processing pixels at the detection site. This research addresses those challenges by using both modeling and simulation techniques to determine the impact of both the application workload and the technology implementation on focal plane architecture design. Performance models for each component have been developed based upon physical implementations. These models are combined to characterize key design choices such as processing granularity and fabrication technology. Simulation of imaging applications determines the feasibility and benefit of processing in the focal plane. Statistical analysis of optoelectronics demonstrates the capability to further improve the focal plane architecture design.

5.1 Contributions

The contributions of this dissertation relate to the study of system-level design issues to monolithically integrate photodetectors, analog-to-digital converters, data storage, and digital processing into focal plane architectures. The contributions are outlined in three categories.

Performance analysis of front-end image processing applications

- Workload characterization of image processing sequence
  - Implemented front-end image processing application suite
  - Determined data storage requirements to execute applications
  - Evaluated functional units for digital pixel
- Key results from analysis
  - Determined required number of register to execute selected applications equals [(2 • PPE) + 7] with a 12-bit datapath
  - Data parallelism in the selected applications enables utilization exceeding 78%
  - Communication cost for nearest-neighbor applications (3 x 3 window) is less than 8% while communication cost for larger windows (8 x 8) exceeds 12%
  - 1 PPE implementation operating at 10MHz executes an imaging sequence 4x faster than the TI TMS320C6411 chip operating at 300MHz.

Efficiency analysis of focal plane architectures

- Evaluation of focal plane architectures with 1 PPE, 4 PPE, and 16 PPE
  - Developed component models for area projections
  - Developed framework for system analysis of component models
  - Projected power consumption for different technology generations
  - Evaluated area-time efficiency, area efficiency, and energy efficiency of focal plane architectures

- Key results from analysis
  - Sustained throughput is increased up to 130x versus TI Benchmarks using a focal plane architecture
  - Focal plane architectures have up to 81x higher area efficiency and up to 11x higher energy efficiency compared to TI DSP chips
  - Area and power constraints for portability are feasible using 180nm fabrication technologies and beyond to implement a focal plane architecture

Statistical experimental design for photodetector modeling

- Analysis of CMOS photodiode signal-to-noise ratio (SNR)
  - Developed photodiode SNR performance model based upon physical implementation
  - Performed statistical experimental design to create regression model
  - Validated regression model using analysis of variance (ANOVA)
- Key results from analysis
  - Regression model reduces complexity for determining photodiode SNR
  - Significant parameters, in order of importance, are diode area, integration time, donor density, reverse bias, and then the combination of reverse bias and donor density
  - Photodiode area and integration time are the most dominant parameters, with a correlation to SNR of 0.794 and 0.587 respectively
  - Each 1ms increase in integration time reduces photodiode area by 212μm^2 while maintaining constant SNR

5.2 Summary of Results

Incorporating analog and digital components to process pixels at the detection site requires an understanding of the connections among applications, architecture, and technology. The results of this dissertation are summarized in the following sections.

5.2.1 Performance analysis of front-end image processing applications

The demand for portable image products will continue to saturate the available computation, communication, and storage capabilities of conventional imaging systems. Processing on the focal plane addresses the potential architectural constraints by exploiting data-parallel processing naturally found in image applications to improve performance on key front-end imaging applications.

Understanding the targeted application suite can aid architectural design choices. The workload has been characterized for an application suite comprised of median filtering, convolution, inside edge detection, and the discrete cosine transform. The algorithms used require an architecture with a 12-bit datapath and a minimum of [(2 • PPE) + 7] words of data storage.

Results show an improvement in the performance metrics of sustained throughput and execution time. The sustained throughput of the focal plane architecture exceeds the reported specification of the TI DSP chips. Also, the execution time for these applications is reduced compared to TI DSP benchmarks. This benefit is for front-end applications that can consume over half of the processing time of an imaging sequence.

5.2.2 Efficiency analysis of focal plane architectures

Portable imaging products will benefit from the monolithic integration of photodetectors, analog-to-digital converters, digital processing, and data storage to perform image acquisition and computation. Technological advances in device fabrication and integration are enabling the development of these monolithic focal plane architectures to improve performance, efficiency, and cost versus traditional imaging architectures.

A focal plane architecture with implementations of 1 PPE, 4 PPE, and 16 PPE has been modeled and evaluated to determine area-time efficiency, area efficiency, and energy efficiency of the design. Area-time efficiency has been traditionally used to evaluate architecture designs. In addition, area efficiency and energy efficiency are important figures of merit when evaluating a focal plane architecture.

Results show that implementations of the focal plane architecture perform well compared to a traditional DSP architecture while demonstrating higher ratings in all three efficiency metrics across fabrication technologies. Although the 1 PPE implementation provides the most efficient architecture, it generally exceeds the constraints for both a cost-effective chip size and power constraints for portability. Selecting a higher PPE or a denser detector technology can improve feasibility.

5.2.3 Statistical experimental design for photodetector modeling

A system-on-a-chip, created by a tiled monolithic array of digital pixels, can utilize the anticipated technological improvements in fabrication. However, the limited silicon resources require fundamental tradeoffs among the functional areas within the

digital pixel. The wide range of design techniques and metrics for these components leads to a complicated design problem, particularly for the integrated optoelectronics.

A regression model was developed for a photodiode using a 2^6 full-factorial experimental design and was validated using analysis of variance (ANOVA). This model characterizes the physical behavior of photodiode response based upon the parameters of area, integration time, acceptor density, donor density, temperature, and reverse bias.

Results show that the regression model accurately depicts the behavior of the SNR response as a function of the input parameters. The two most significant factors, photodiode area and integration time, provide a helpful design tradeoff in the context of the digital pixel. This relationship significantly impacts the design of a focal plane processor by trading time for limited silicon resources.

5.3 Future Research

Future directions for research based on the results presented in this dissertation can be characterized into two categories: architecture modeling and architecture implementation.

There is a wide array of choices available to designers of focal plane architectures. Having a full complement of component models would enable high-level architectural choices within the design space to be made without costly fabrication. To extend focal plane architecture modeling:

- Perform analysis of power consumption of analog circuitry to establish the power budget of a focal plane architecture.

- Develop models for other types of CMOS sensors (APS, p-i-n photodiode, etc.) to evaluate other candidate solutions for focal plane architecture integration
- Develop models for various ADC circuits to evaluate other candidate solutions for focal plane architecture integration

Co-locating analog and digital components on a chip creates a challenge to implement a functional imaging system. The architecture needs versatility for additional application development. In addition, testing must be done to insure proper integration of the various components. To extend focal plane architecture implementation:

- Develop additional applications for the focal plane architecture
- Investigate alternatives to improve analog circuitry
- Investigate alternatives to improve data storage and data communication because both are tightly coupled in a focal plane architecture implementation.
- Create VLSI layout of both analog and digital circuitry for analysis of extracted parameters
- Fabricate and test a prototype chip for the focal plane architecture

APPENDIX A

FOCAL PLANE ARCHITECTURAL SIMULATOR

Applications for focal plane architectures can be programmed using the SIMD Pixel Processor (SIMPil) Simulator [34]. This software tool is a windows-based instruction level simulator, running on a PC platform. The SIMPil Simulator allows editing, assembling, executing, and debugging parallel image applications in a single integrated workbench (Figure 30). The current version of the SIMPil Simulator is available on the download page.

Two instruction sets, one for scalar operations and one for vector operations compose the assembly language. Scalar instructions are executed in the system controller, while vector instructions are broadcasted to each processing element to execute in a lockstep fashion. Constants are used to define some architectural parameters, such as the size of the local memory for each PE, the size of the processor array, and the type of network topology (standard NEWS or Torus-NEWS network). A help menu option is available to access information on instruction formats, system architecture, and simulator functionality.

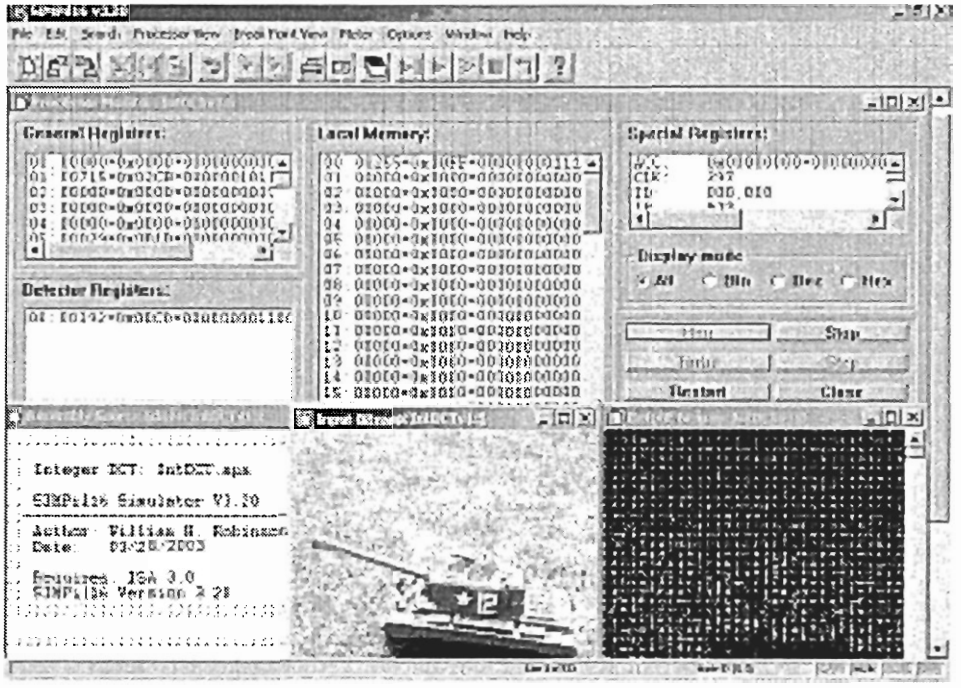


Figure 30: A screenshot of the SIMPil Simulator during execution of the DCT application

The state of each system processor (registers, memory, and detector values) is displayed in the processor monitor window. A similar window is available to inspect the state of the system controller. Three different program execution modes are available to ease the debugging process, and breakpoints can be set to stop execution at any point in the program. The simulator is also instrumented to allow for measuring the concurrency level and the instruction distribution during the execution of the application. In addition, simulator directives can be inserted into the program for tasks such as displaying the output image produced by the system or outputting the content of a specified memory location into a file.

APPENDIX B

TI DSP ARCHITECTURE

Architectural comparisons are based upon the TI TMS320C6211B chip and the TI TMS320C6411 chip (Table 25) [22]. The calculations for cycle count use C62x™ DSP Benchmarks and C64x™ DSP Benchmarks for: (1) 3 x 3 Median Filter (2) 3 x 3 Convolution, and (3) Sobel Edge Detection. This represents the typical early image processing sequence of: (1) noise removal, (2) smoothing, and (3) segmentation. The cycle count (and corresponding execution time) from the DSP benchmarks depends upon the image resolution. The calculated cycle count for each application is provided for the C62x™ DSP architecture (Table 26) and the C64x™ DSP architecture (Table 27). The cycle count is converted to execution time using the appropriate cycle time for the TI DSP chip. For example, the cycle times for the TMS320C6211B and the TMS320C6411 are 6.7ns and 3.3ns respectively. For larger images (e.g. CIF), the cycle count (and corresponding execution time) increases as a function of both row width and column width.

Table 25: TI DSP chip specifications

	TMS320C6211B	TMS320C6411
Technology	0.18μm	0.12μm
Clock frequency	150 MHz	300 MHz
Maximum throughput	1200 MIPS	2400 MIPS
Area	27mm x 27mm	23mm x 23mm
Power	0.49 W	0.09 W

Table 26: C62x™ DSP Benchmarks for cycle count

Application	Description	Benchmark Formula	QCIF cycles rows = 144 cols = 176
3 x 3 Median Filter	Performs 3x3 median filtering. Operates on three lines of input data where each line is cols pixels wide.	(9 • cols + 49) • (rows / 3)	78384
3 x 3 Convolution	Performs a 3x3 convolution on three rows of cols input pixels.	(9 • cols / 8 + 33) • (rows / 3)	11088
Sobel Edge Detection	Performs Sobel edge detection.	3 • cols • (rows - 2) + 34	75010

Table 27: C64x™ DSP Benchmarks for cycle count

Application	Description	Benchmark Formula	QCIF cycles rows = 144 cols = 176
3 x 3 Median Filter	Performs 3x3 median filtering. Operates on three lines of input data where each line is cols pixels wide.	(2 • cols + 32) • (rows / 3)	18432
3 x 3 Convolution	Performs a 3x3 convolution on three rows of cols input pixels.	(9 • cols / 8 + 33) • (rows / 3)	11088
Sobel Edge Detection	Performs Sobel edge detection.	11 • cols • (rows - 2) / 8 + 23	34387

REFERENCES

[1] C. Koch and H. Li, *Vision chips : implementing vision algorithms with analog VLSI circuits*. Los Alamitos, Calif.: IEEE Computer Society Press, 1995.

[2] P. Pirsch and H.-J. Stolberg, "VLSI implementations of image and video multimedia processing systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, pp. 878-891, 1998.

[3] A. Moini, *Vision chips*. Boston: Kluwer Academic, 2000.

[4] M. J. Flynn, "Very High Speed Computing Systems," *Proceedings of the IEEE*, vol. 54, pp. 1901-1909, 1966.

[5] A. Peleg and U. Weiser, "MMX Technology Extension to the Intel Architecture," *IEEE Micro*, vol. 16, pp. 42-50, 1996.

[6] K. Acken, E. Gayles, T. Kelliher, R. M. Owens, and M. J. Irwin, "The MGAP family of processor arrays," in *Seventh Great Lakes Symposium on VLSI*, 1997, pp. 105-110.

[7] M. Bolotski, T. Simon, C. Vieri, R. Amirtharajah, and T. F. Knight, Jr., "Abacus: a 1024 processor 8 ns SIMD array," in *Conference on Advanced Research in VLSI*, W. J. Dally, J. W. Poulton, and A. T. Ishii, Eds. Chapel Hill, NC, USA: IEEE Computer Society, 1995, pp. 28-40.

[8] M. T. Bohr, "Interconnect scaling-the real limiter to high performance ULSI," *Solid State Technology*, vol. 39, pp. 105-106, 109, 111, 1996.

[9] E. R. Fossum, "CMOS image sensors: electronic camera-on-a-chip," *IEEE Transactions on Electron Devices*, vol. 44, pp. 1689-1698, 1997.

[10] J. G. Harris, "The changing roles of analog and digital signal processing in CMOS image sensors," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4. Orlando, FL, USA: IEEE, 2002, pp. IV3976-3979.

[11] H.-S. Wong, "Technology and device scaling considerations for CMOS imagers," *IEEE Transactions on Electron Devices*, vol. 43, pp. 2131-2142, 1996.

[12] T. Lule, S. Benthien, H. Keller, F. Mutze, P. Rieve, K. Seibel, and M. Sommer, "Sensitivity of CMOS based imagers and scaling perspectives," *IEEE Transactions on Electron Devices*, vol. 47, pp. 2110-2122, 2000.

[13] A. El Gamal, "Trends in CMOS image sensor technology and design," in *International Electron Devices Meeting*, 2002, pp. 805-808.

[14] R. W. Brodersen, "Why We Need a Custom Chip-in-a-Day Design Methodology," http://www.mseconference.org/Talks/mse01_brodersen.pdf, International Conference on Microelectronic Systems Education, Mixed Signal & Fabrication Services Keynote Address, 2001.

[15] A. Peleg, S. Wilkie, and U. Weiser, "Intel MMX for multimedia PCs," *Communications of the ACM*, vol. 40, pp. 25-38, 1997.

[16] S. K. Raman, V. Pentkovski, and J. Keshava, "Implementing Streaming SIMD Extensions on the Pentium III Processor," *IEEE Micro*, vol. 20, pp. 28-39, 2000.

[17] R. B. Lee, "Subword parallelism with MAX-2," *IEEE Micro*, vol. 16, pp. 51-59, 1996.

[18] M. Tremblay, J. M. O'Connor, V. Narayanan, and L. He, "VIS speeds new media processing," *IEEE Micro*, vol. 16, pp. 10-20, 1996.

[19] "MIPS extension for digital media with 3D," http://www.mips.com/, Technical Report, MIPS Technologies, Inc., 1997.

[20] R. L. Sites and Alpha Architecture Committee, *Alpha architecture reference manual*, 3rd / ed. Boston: Digital Press, 1998.

[21] H. Nguyen and L. K. John, "Exploiting SIMD parallelism in DSP and multimedia algorithms using the AltiVec™ technology," in *13th Association for Computer Machinery International Conference on Supercomputing*. Rhodes, Greece: ACM, 1999, pp. 11-20.

[22] Texas Instruments Incorporated, http://dspvillage.ti.com/.

[23] J. A. Davis, R. Venkatesan, A. Kaloyeros, M. Beylansky, S. J. Souri, K. Banerjee, K. C. Saraswat, A. Rahman, R. Reif, and J. D. Meindl, "Interconnect Limits on Gigascale Integration (GSI) in the 21st Century," *Proceedings of the IEEE*, vol. 89, pp. 305-324, 2001.

[24] C. Mead, *Analog VLSI and neural systems*. Reading, Mass.: Addison-Wesley, 1989.

[25] J. Kramer and G. Indivèri, "Neuromorphic vision sensors and preprocessors in system applications," in *Advanced Focal Plane Arrays and Electronic Cameras II*, vol. 3410. Zurich, Switzerland: SPIE - The International Society for Optical Engineering, 1998, pp. 134-146.

[26] A. El Gamal, D. Yang, and B. Fowler, "Pixel level processing - why, what, and how?" in *Sensors, Cameras, and Applications for Digital Photography*, vol. 3650. San Jose, CA, USA: SPIE - The International Society for Optical Engineering, 1999, pp. 2-13.

[27] J.-E. Eklund, C. Svensson, and A. Astrom, "VLSI implementation of a focal plane image processor-a realization of the near-sensor image processing concept," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 4, pp. 322-335, 1996.

[28] F. Paillet, D. Mercier, and T. M. Bernard, "Making the most of 15 k lambda^2 silicon area for a digital retina PE," in *Advanced Focal Plane Arrays and Electronic Cameras II*, vol. 3410. Zurich, Switzerland: SPIE - The International Society for Optical Engineering, 1998, pp. 158-167.

[29] T. Komuro, I. Ishii, and M. Ishikawa, "General-purpose vision chip architecture for real-time machine vision," *Advanced Robotics*, vol. 12, pp. 619-627, 1999.

[30] M. L. Riaziat, *Introduction to high-speed electronics and optoelectronics*. New York: Wiley, 1996.

[31] M. W. Hauser, "Principles of oversampling A/D conversion," *Journal of the Audio Engineering Society*, vol. 39, pp. 3-26, 1991.

[32] R. L. Franch, J. Ji, and C. L. Chen, "A 640-ps, 0.25- mu m CMOS, 16*64-b three-port register file," *IEEE Journal of Solid-State Circuits*, vol. 32, pp. 1288-1292, 1997.

[33] D. A. Patterson, J. L. Hennessy, and D. Goldberg, *Computer architecture : a quantitative approach*, 2nd ed. San Francisco: Morgan Kaufmann Publishers, 1996.

[34] SIMPil Programming Environment, http://www.ece.gatech.edu/research/pica/simpil/sim/simulator.html.

[35] S. M. Chai, A. Gentile, and D. S. Wills, "Impact Of Power Density Limitation In Gigascale Integration For The SIMD Pixel Processor," in *20th Anniversary Conference on Advanced Research in VLSI*. Atlanta, GA, USA: IEEE Computer Society, 1999, pp. 57-71.

[36] Semiconductor Industry Association, *The International Technology Roadmap for Semiconductors 1999 Edition*, 1999.

[37] Semiconductor Industry Association, *The International Technology Roadmap for Semiconductors 2002 Update*, 2002.

[38] R. B. Lee and M. D. Smith, "Media processing: a new design target," *IEEE Micro*, vol. 16, pp. 6-9, 1996.

[39] K. Diefendorff and P. K. Dubey, "How multimedia workloads will change processor design," *Computer*, vol. 30, pp. 43-45, 1997.

[40] M. Maruyama, H. Nakahira, T. Araki, S. Sakiyama, Y. Kitao, K. Aono, and H. Yamada, "An image signal multiprocessor on a single chip," *IEEE Journal of Solid-State Circuits*, vol. 25, pp. 1476-1483, 1990.

[41] Y. Joo, J. Park, M. Thomas, K. S. Chung, M. A. Brooke, N. M. Jokerst, and D. S. Wills, "Smart CMOS focal plane arrays: a Si CMOS detector array and sigma-delta analog-to-digital converter imaging system," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 5, pp. 296-305, 1999.

[42] T. Morris, E. Fletcher, C. Afghahi, S. Issa, K. Connolly, and J.-C. Korta, "A column-based processing array for high-speed digital image processing," in *Conference on Advanced Research in VLSI*. Atlanta, GA, USA: IEEE Computer Society, 1999, pp. 42-56.

[43] E.-S. Eid and E. R. Fossum, "Real-time focal-plane array image processor," in *Automated Inspection and High-Speed Vision Architectures III*, vol. 1197. Philadelphia, PA, USA: SPIE - The International Society for Optical Engineering, 1989, pp. 2-12.

[44] R. S. Bajwa, R. M. Owens, and M. J. Irwin, "Image processing with the MGAP: a cost effective solution," in *International Parallel Processing Symposium*. Newport, CA, USA, 1993, pp. 439-443.

[45] R. Forchheimer and A. Astrom, "Near-sensor image processing: a new paradigm," *IEEE Transactions on Image Processing*, vol. 3, pp. 736-746, 1994.

[46] A. Astrom, J.-E. Eklund, and R. Forchheimer, "Near-sensor image processing - theory and practice," in *Advanced Focal Plane Arrays and Electronic Cameras*, vol. 2950. Berlin, Germany: SPIE - The International Society for Optical Engineering, 1996, pp. 242-253.

[47] E. R. Fossum, "Active pixel sensors: are CCD's dinosaurs?" in *Charged-Coupled Devices and Solid State Optical Sensors III*, vol. 1900. San Jose, CA, USA: SPIE - The International Society for Optical Engineering, 1993, pp. 2-14.

[48] K. E. Batcher, "Design of a massively parallel processor," *IEEE Transactions on Computers*, vol. c-29, pp. 836-840, 1980.

[49] M. J. B. Duff and D. M. Watson, "The cellular logic array image processor," *Computer Journal*, vol. 20, pp. 68-72, 1977.

[50] L. W. Tucker and G. G. Robertson, "Architecture and applications of the connection machine," *IEEE Computer*, vol. 21, pp. 26-38, 1988.

[51] "Connection machine model CM-2 technical summary," Technical Report, Thinking Machines Corporation, 1989.

[52] "MasPar (MP-2) System Data Sheet," MasPar Corporation, 1993.

[53] W. F. Wong and K. T. Lua, "A preliminary evaluation of a massively parallel processor: GAPP," *Microprocessing & Microprogramming*, vol. 29, pp. 53-62, 1990.

[54] M. J. Irwin and R. M. Owens, "A two-dimensional, distributed logic processor for machine vision," *IEEE Transactions on Computers*, vol. 40, pp. 1094-1101, 1991.

[55] B. Fowler, "CMOS Area Image Sensors with Pixel Level A/D Conversion," in *Electrical Engineering*. Palo Alto, CA: Stanford University, 1995, pp. 161.

[56] D. Yang, H. Tian, B. Fowler, X. Liu, and A. El Gamal, "Characterization of CMOS image sensors with Nyquist rate pixel level ADC," in *Sensors, Cameras, and Applications for Digital Photography*, vol. 3650. San Jose, CA, USA: SPIE - The International Society for Optical Engineering, 1999, pp. 52-62.

[57] G. Heygster, "Rank filters in digital image processing," *Computer Graphics and Image Processing*, vol. 19, pp. 148-164, 1982.

[58] M. J. T. Smith and A. Docef, *A study guide for digital image processing*, Early release ed. Riverdale, GA: Scientific Publishers, 1997.

[59] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image Analysis Using Mathematical Morphology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, pp. 532-550, 1987.

[60] J. Liang and T. D. Tran, "Fast multiplierless approximations of the DCT with the lifting scheme," *IEEE Transactions on Signal Processing*, vol. 49, pp. 3032-3044, 2001.

[61] T. D. Tran, "Fast multiplierless approximation of the DCT," in *33rd Annual Conference on Information Science and Systems*. Baltimore, MD, USA, 1999, pp. 933-938.

[62] A. Gentile, "Portable multimedia supercomputers: system architecture design and evaluation," in *Electrical and Computer Engineering*. Atlanta, GA: Georgia Institute of Technology, 2000, pp. 170.

[63] International Telecommunications Union, http://www.itu.int/home/index.html.

[64] R. F. Pierret, *Semiconductor device fundamentals*. Reading, Mass.: Addison-Wesley, 1996.

[65] P. Bhattacharya, *Semiconductor optoelectronic devices*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1997.

[66] A. Dupret, E. Belhaire, and J.-C. Rodier, "A high current large bandwidth photosensor on standard CMOS processes," in *Advanced Focal Plane Arrays and Electronic Cameras*, vol. 2950. Berlin, Germany: SPIE, 1996, pp. 36-44.

[67] M. L. Simpson, M. N. Ericson, G. E. Jellison, Jr., W. B. Dress, A. L. Wintenberg, and M. Bobrek, "Application specific spectral response with CMOS compatible photodiodes," *IEEE Transactions on Electron Devices*, vol. 46, pp. 905-913, 1999.

[68] S. K. Mendis, S. E. Kemeny, R. C. Gee, B. Pain, C. O. Staller, Q. Kim, and E. R. Fossum, "CMOS active pixel image sensors for highly integrated imaging systems," *IEEE Journal of Solid-State Circuits*, vol. 32, pp. 187-197, 1997.

[69] E. R. Fossum, "Digital camera system on a chip," *IEEE Micro*, vol. 18, pp. 8-15, 1998.

[70] J. G. Proakis, *Digital communications*, 3rd ed. New York: McGraw-Hill, 1995.

[71] J. C. Candy, G. C. Temes, Institute of Electrical and Electronics Engineers, and IEEE Circuits and Systems Society, *Oversampling delta-sigma data converters: theory, design, and simulation*. Piscataway, NJ: IEEE Press, 1992.

[72] M. W. Hauser and R. W. Brodersen, "Circuit and technology considerations for MOS delta-sigma A/D converters," in *International Symposium on Circuits and Systems*, vol. 3. San Jose, CA, USA: IEEE, 1986, pp. 1310-1315.

[73] M. W. Hauser and R. W. Brodersen, "Monolithic decimation filtering for custom delta-sigma A/D converters," in *International Conference on Acoustics, Speech, and Signal Processing*, vol. 4. New York, NY, USA: ICASSP, 1988, pp. 2005-2008.

[74] M. W. Hauser, "Technology scaling and performance limitations in delta-sigma analog-digital converters," in *International Symposium on Circuits and Systems*, vol. 1. New Orleans, LA, USA: IEEE, 1990, pp. 356-359.

[75] J. Nakamura, B. Pain, T. Nomoto, T. Nakamura, and E. R. Fossum, "On-focal-plane signal processing for current-mode active pixel sensors," *IEEE Transactions on Electron Devices*, vol. 44, pp. 1747-1758, 1997.

[76] L. G. McIlrath, "A low-power low-noise ultrawide-dynamic-range CMOS imager with pixel-parallel A/D conversion," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 846 - 853, 2001.

[77] Y. Perelman and R. Ginosar, "A low-light-level sensor for medical diagnostic applications," *IEEE Journal of Solid-State Circuits*, vol. 36, pp. 1553-1558, 2001.

[78] M. Nomura, M. Yamashina, K. Suzuki, M. Izumikawa, H. Igura, H. Abiko, K. Okabe, A. Ono, T. Nakayama, and H. Yamada, "A 500-MHz, 0.4- mu m CMOS, 32-word by 32-bit 3-port register file," in *IEEE 1995 Custom Integrated Circuits Conference*. Santa Clara, CA, USA: IEEE, 1995, pp. 151-154.

[79] J. M. Mulder, N. T. Quach, and M. J. Flynn, "An area model for on-chip memories and its application," *IEEE Journal of Solid-State Circuits*, vol. 26, pp. 98-106, 1991.

[80] S. Rixner, W. J. Dally, B. Khailany, P. Mattson, U. J. Kapasi, and J. D. Owens, "Register organization for media processing," in *Sixth International Symposium on High-Performance Computer Architecture*. Touluse, France, 2000, pp. 375-386.

[81] W. H. Robinson, G. E. Triplett, and D. S. Wills, "Component modeling for an integrated digital pixel," in *15th Annual Meeting IEEE Lasers and Electro-Optics Society (LEOS)*, vol. 1. Glasgow, Scotland: IEEE, 2002, pp. 37-38.

[82] A. Gentile, A. López-Lagunas, S. M. Chai, H. H. Cat, K. S. Chung, L. Codrescu, M. Deb, S. J. Ryu, M. F. Wolff, J. C. Eble, W. H. Robinson, T. Taha, D. S. Wills, M. Viera-Vera, W. E. Lugo-Beauchamp, L. Bustelo, R. Olivieri, J. Figueroa, and J. L. Cruz-Rivera, "Portable Multimedia Supercomputing," *submitted to IEEE Transactions on Computers*, vol. TC 114531, pp. 43 pages, July 2001.

[83] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS Digital Design," *IEEE Journal of Solid-State Circuits*, vol. 27, pp. 473-484, 1992.

[84] T. D. Burd and R. W. Brodersen, "Energy Efficient CMOS Microprocessor Design," in *Twenty-Eighth Annual Hawaii International Conference on System Sciences*, vol. 1. Wailea, HI, USA, 1995, pp. 288-297.

[85] P. Seitz, "Smart pixels," in *2001 International Symposium on Electron Devices for Microwave and Optoelectronic Applications*. Vienna, Austria, 2001, pp. 229-234.

[86] G. Box, W. Hunter, and J. Hunter, *Statistics for Experimenters*. New York: Wiley, 1978.

VITA

William Hugh Robinson, III was born on July 13, 1973, in Wauchula, Florida. In 1996, he received his Bachelor of Science in electrical engineering from Florida Agricultural and Mechanical University in Tallahassee, Florida. He subsequently entered the Ph.D. program in Electrical and Computer Engineering at the Georgia Institute of Technology in Atlanta, Georgia, where he joined the Portable Image Computation Architectures (PICA) Research Group led by Dr. D. Scott Wills. His research explores the system-level integration of computer architecture to understand the impact of technology on architecture design. Topics of interest include VLSI design, parallel computer architectures, and image processing. In 1998, he received a Master of Science in electrical engineering. Under the direction of Dr. Wills, William received his Ph.D. degree in Electrical and Computer Engineering in August 2003. During his time at Georgia Tech, William was awarded several graduate fellowships, including a National Science Foundation Minority Graduate Fellowship in 1996, a Facilitating Academic Careers in Engineering and Science (FACES) Fellowship in 2001, and a Ford Foundation Dissertation Fellowship for Minorities in 2002. In IEEE, he is a member of both the Computer Society and the Lasers and Electro-Optics Society. Other professional memberships include the Association for Computing Machinery (ACM), the American Society of Engineering Educators (ASEE), the National Society of Black Engineers (NSBE), and SPIE – The International Society for Optical Engineering.