A Complementary GaAs Microprocessor for Space Applications

Todd D. Basso and Richard B. Brown
Electrical Engineering and Computer Science Dept.
University of Michigan
1301 Beal Avenue
Ann Arbor, MI 48109-2122, USA

Abstract—This paper describes the development of a complementary GaAs PowerPC™ microprocessor suitable for space applications. Motorola's 0.5μm Complementary GaAs (CGaAs™) process was selected as the semiconductor technology because of its unique abilities to meet the requirements of space and satellite systems. Recently published findings indicate that CGaAs has a power-delay product of 0.01 μW/MHz/gate. Additionally, the CGaAs process is resistant to single-event upset (SEU) (10^{-9} to 10^{-10} upsets/bit-day), total dose radiation (10^{8} rads), and latchup. These properties make CGaAs attractive for space and satellite applications. However, CGaAs presents design challenges such as reduced power-supply voltage, gate leakage, subthreshold drain-source leakage, and low integration level, all of which impact digital circuits. The processor architecture described in this paper is driven by the constrained integration level. Architectural simulations reveal that an acceptable level of performance can be reached by efficiently utilizing the resources available. The processor implements a small on-chip primary instruction cache and a larger off-chip primary data cache. The instruction fetch mechanism is guided by a small two-level dynamic branch prediction mechanism. Computation is performed by a small superscalar execution core. The architecture, running at 200MHz, is capable of achieving 153 MIPS, translating to a 27% performance increase over a comparable traditional pipelined microprocessor. Researchers at the University of Michigan have designed a prototype CGaAs 32-bit PowerPC microprocessor based on this architecture, which is now being fabricated. This paper will summarize the advantages and design challenges of developing a radiation-hard CGaAs microprocessor. The constraints and limitations of the technology as they affect system architecture will be discussed. Architectural features that improve the performance of the processor despite the limited integration levels will be presented.

I. Introduction

Our society is becoming highly dependent upon satellites. We rely upon satellites for everything from cellular phone conversations to military defense systems. The failure of the Telstar 401 satellite in January of 1997 underscores how vulnerable microelectronics in space are to the effects of solar disturbances. The number of communication satellite licenses granted by the FCC indicate that over one thousand satellites could be launched into low-earth-orbit over the next five years [1]. Microelectronic components in these systems must be resistant to total dose radiation, single-even upset, and latchup in order to accomplish their missions.

Radiation hardened (rad-hard) electronics have been operating in space since the beginning of the space program decades ago, but the demand for inexpensive, high-volume rad-

hard ICs is expected to increase dramatically as the communication market continues to expand. Traditional CMOS processing can be enhanced to render the IC radiation tolerant, but the cost of rad-hard CMOS ICs can be 10 to 100 times that of commercial CMOS ICs. In light of the increased cost, alternative technologies such as silicon-on-insulator, silicon-on-sapphire, and gallium arsenide must be considered. Complementary gallium arsenide (CGaAs) operates at low voltages, providing a power savings over CMOS, while also being innately rad-hard. CGaAs is also a relatively low-cost alternative; the cost of a CGaAs IC is approximately 5 times that of a similar high-volume commercial CMOS IC [2], but significantly less than that of a rad-hard CMOS IC.

In this paper we explore the use of CGaAs processing elements for space-based applications. Section II summarizes the technological advantages of CGaAs and addresses the disadvantages and design challenges the technology presents. Section III presents issues related to system architecture. Section IV describes architectural trade-offs aimed at improving processor performance while operating under the resource-limited design constraint. A prototype CGaAs microprocessor developed by a team at the University of Michigan is presented in Section V. The final section summarizes our work.

II. Complementary GaAs Technology

The application of CGaAs, a complementary heterostructure-insulated-gate FET technology, for high-speed VLSI circuits has been described recently [2]. A sketch of the device structure is shown in Fig. 1. CGaAs integrates an enhancement-mode P-channel HFET with a high performance N-channel HFET. Historically, the primary interest in GaAs and other III/V materials has been their high electron mobilities. While holes in III/V materials do not enjoy an intrinsic mobility advantage over those in silicon, the pseudomorphic P-channel HFETs in this process have three to five times higher transconductances at given gate dimensions than their silicon counterparts. The current CGaAs process has three levels of Al-Cu interconnect. Standard gate lengths are now 0.7μm and 0.5μm; typical parameters for 0.7μm channel-length devices with ±0.55V thresholds (measured with Vdd = 1.5V) are given in Table 1. As seen in the table, both N and P channel devices have good output conductances and pinch-off characteristics.

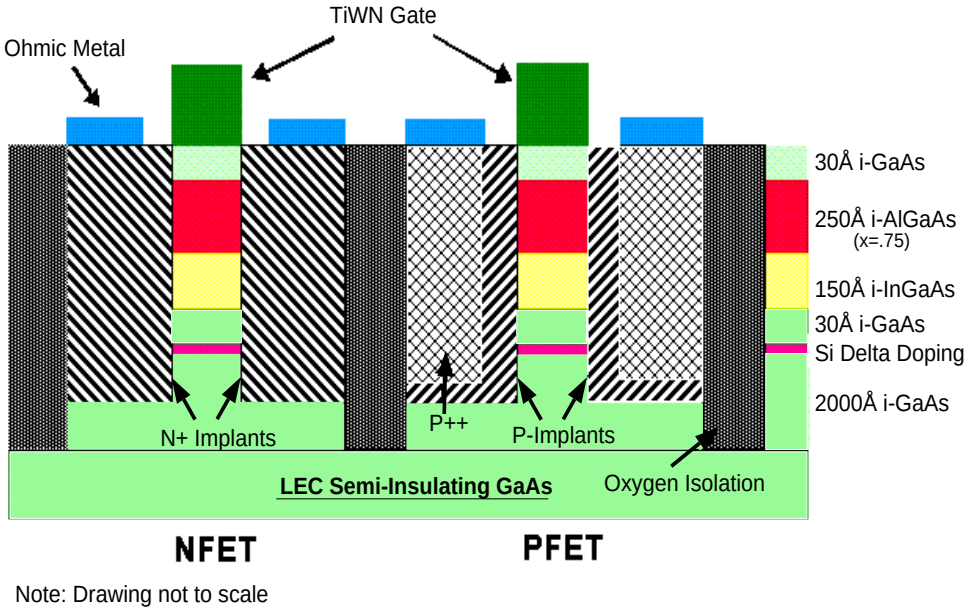


Fig. 1: CGaAs process cross-section [2].

Table 1: CGaAs Device Parameters [2].

Parameter	NFET (0.7x10 µm)	PFET (0.7x10 µm)
V_{th} (V)	+0.55	-0.55
I_{dss} (mA)	1.8	0.5
g_m (mS/mm)	280	60
Beta (mA/V^2-mm)	270	50
Subth slope (mV/dec)	75	90
Subth Current (nA) (V_{gs}=0V)	< 1	< 10

A low temperature GaAs (LTG) layer is used in the HIGFET structure to guard against charge collection from ionizing particle radiation [3]. Its epi structure makes the CGaAs process resistant to single-event upset (SEU) and latchup. Complementary logic in CGaAs is resistant to SEU with fewer than 10^{-10} upsets/bit-day and it will not latchup, even at 10^{12} rads. The absence of a gate oxide and field oxide in CGaAs makes it resistant to total dose radiation up to 10^8 rads.

Fig. 2 shows unloaded ring oscillator gate delays versus supply voltage for several logic families with ±0.55V device thresholds. The delay of 1.0µm CGaAs is less than that of commercial 0.5µm CMOS or thin-film silicon-on-insulator (TFSOI), and 0.5µm CGaAs shows delays below 100ps with a 1.2V power supply. Circuits used for process monitoring, such as a 32-bit shift register, have demonstrated a power-delay performance of 0.01µW/MHz/gate at 0.9V and 0.1µW/MHz/gate at 1.2V.

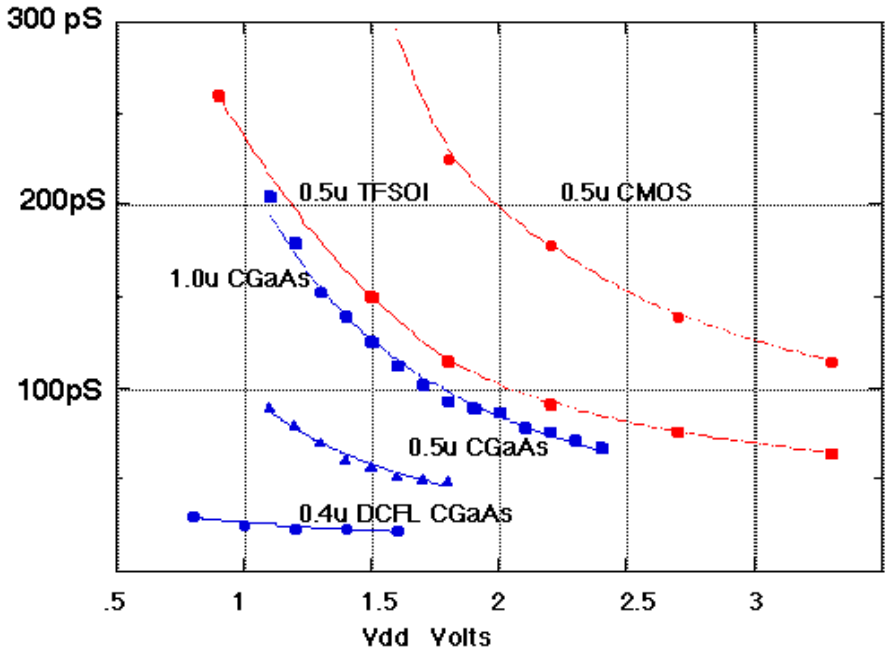


Fig. 2: Propagation delay of CGaAs, CMOS, and TFSOI versus supply voltage [2].

Two key parameters of concern in the CGaAs process are gate leakage and subthreshold drain-source leakage, which determine the stand-by power dissipation of complementary circuits. Drain-induced barrier lowering (DIBL) increases gate current when the drain-to-source voltage is high, as is the case when a logic input changes state.

A further design challenge results from the low power supply voltages. While low supply voltages, particularly when coupled with the comparatively high device thresholds (±0.55V), reduce power consumption, this combination of characteristics reduces drain currents of both N and P devices, requiring more time to charge and discharge circuit nodes. This is especially detrimental for P devices, which, because of the lower mobility of holes than electrons in the InGaAs channel, have transconductances only 1/4 those of comparable N transistors. This transconductance ratio requires P device geometries to be wider, significantly increasing the input capacitance of complementary logic gates. The high input capacitance and small (V_{in} - V_{th}) slow complementary CGaAs circuits. Threshold voltages could be reduced, and in fact, have been reduced on experimental wafers, significantly improving transonductances at the expense of higher leakage currents.

The greatest challenge in designing CGaAs integrated circuits is the low level of integration. Commercial CGaAs circuits of 170K transistors have been demonstrated, and the integration level has been raised to 400K transistors for current designs. Complex ICs, such as microprocessors, require far more transistors than that. The challenge in this CGaAs project was to design the architecture to optimize performance given a limited integration level, and to partition the design most effectively across multiple chips.

III. System Architecture

The High-Performance Microprocessor Project at The University of Michigan is devel-

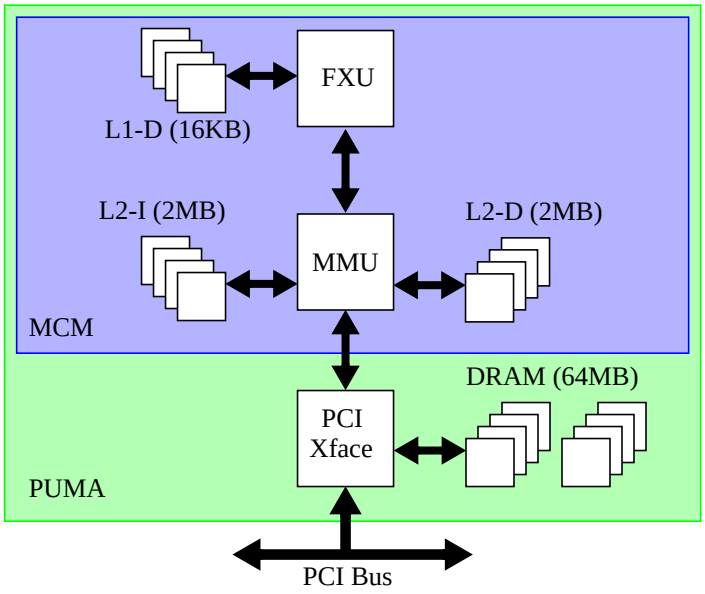


Fig. 3: PUMA system architecture.

oping a prototype PowerPC system based on CGaAs process technology [4]. A block diagram of the system, referred to as PUMA, is shown in Fig. 3. The limited integration levels of CGaAs force a partitioning of the processor across several ICs. The MCM serves as a high-bandwidth interconnect for the elements of the microprocessor [5]. The proposed system is comprised of an integer processor (FXU), primary data cache (L1-D), and a memory management unit (MMU). Perhaps the most critical aspect of the processor is the FXU to primary data cache interface. The following section presents a detailed study of how the microarchitecture was optimized.

IV. Microprocessor Architecture

The limited integration level of the CGaAs technology has a profound impact upon the microprocessor architecture. In this section we describe a method for optimizing a superscalar architecture for the CGaAs process technology. We investigate the performance impact of stream buffers, data cache and branch prediction configurations, and out-of-order superscalar execution in the design of a resource-limited microprocessor.

A cycle-level architectural simulator based on ssim [6] was used to evaluate architectural trade-offs. The simulator processes a dynamic instruction trace and performs a detailed performance analysis on various aspects of the architecture. The Spec95 integer benchmark suite was used as the basis for evaluating performance. The baseline microprocessor configuration used in the following studies was a dual-issue superscalar microprocessor with on-chip 4KB instruction and data caches. The baseline model has five execution units (floating-point, integer, branch, load, and store) and the out-of-order execution core is maintained by a 16 entry integer reorder buffer (and an 8-entry floating point reorder buffer) along with 4 reservation stations per functional unit. The baseline model does not utilize instruction prefetching or speculative execution. The following studies evaluate enhancements and trade-offs relative to this baseline configuration. The metric for comparison is instructions per cycle (IPC) for the processor. Overall IPC is the simplest measure of the effects of small modifications relative to processor performance.

An instruction cache improves performance by taking advantage of both temporal and spatial locality [7]. The limited-resource nature of the CGaAs technology restricts the size of the instruction cache. As an alternative to simply increasing the size of the cache, other means of reducing the instruction miss rate were investigated. Instruction prefetching [8] can improve the effective miss rate and also hide some of the miss penalty associated with instruction cache misses. A simple prefetch mechanism is the stream buffer, which fetches consecutive lines of data from the location following a cache miss. When the fetch mechanism requests another data item that is not present in the cache, the stream buffer may be able to provide it. If the access hits in the stream buffer, then the line is supplied to the fetch engine and also copied into the instruction cache.

Fig. 4 graphs the machine performance versus instruction cache size for a variety of stream buffer configurations. A single-entry stream buffer provides a large improvement over the baseline machine. Stream buffers consisting of two and four entries provide additional performance improvements, but beyond four entries no noticeable performance gain is seen. The simulations show that, in general, a single-entry stream buffer can approximately double the effective size of the instruction cache.

Fig. 5 shows that a stream buffer also improves performance by hiding some of the latency incurred on a primary instruction cache miss. This figure shows processor performance for various stream buffer configurations and memory subsystem latencies. A single-entry stream buffer compensates for an additional eight cycles of secondary cache latency. Also note the slope of the performance curves corresponding to the various stream buffer configurations. The slope of the baseline processor with no stream buffer is greater than that of the other configurations, indicating that the stream buffer reduces the sensitiv-

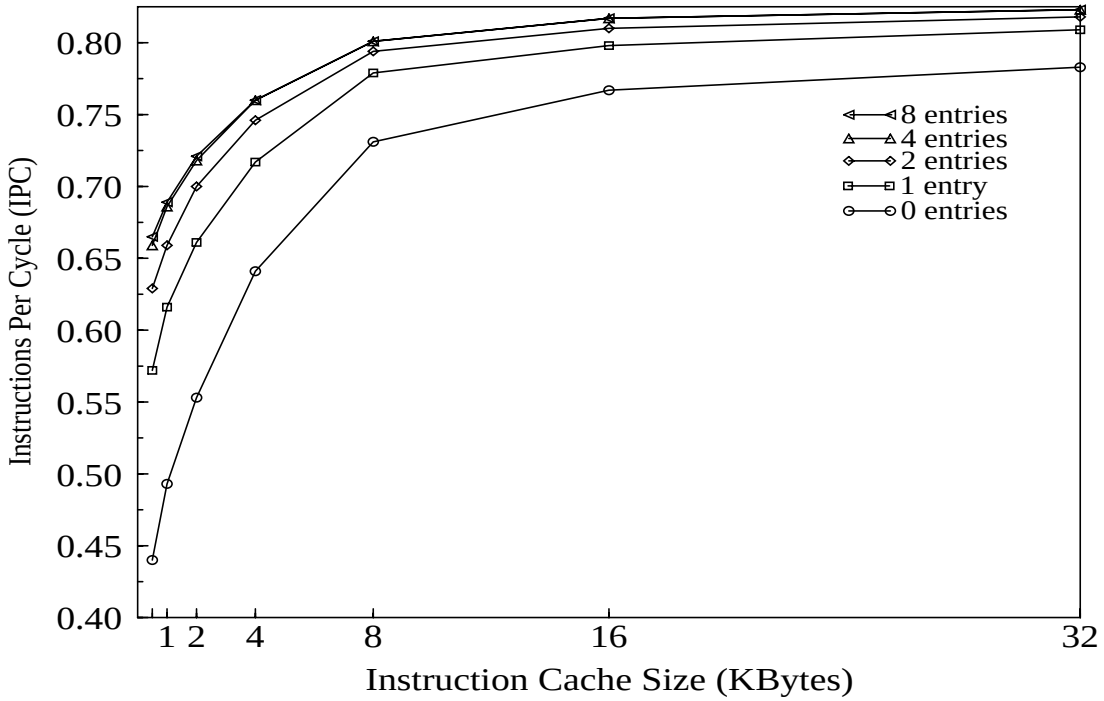


Fig. 4: Stream buffer compensating for reduced instruction cache size.

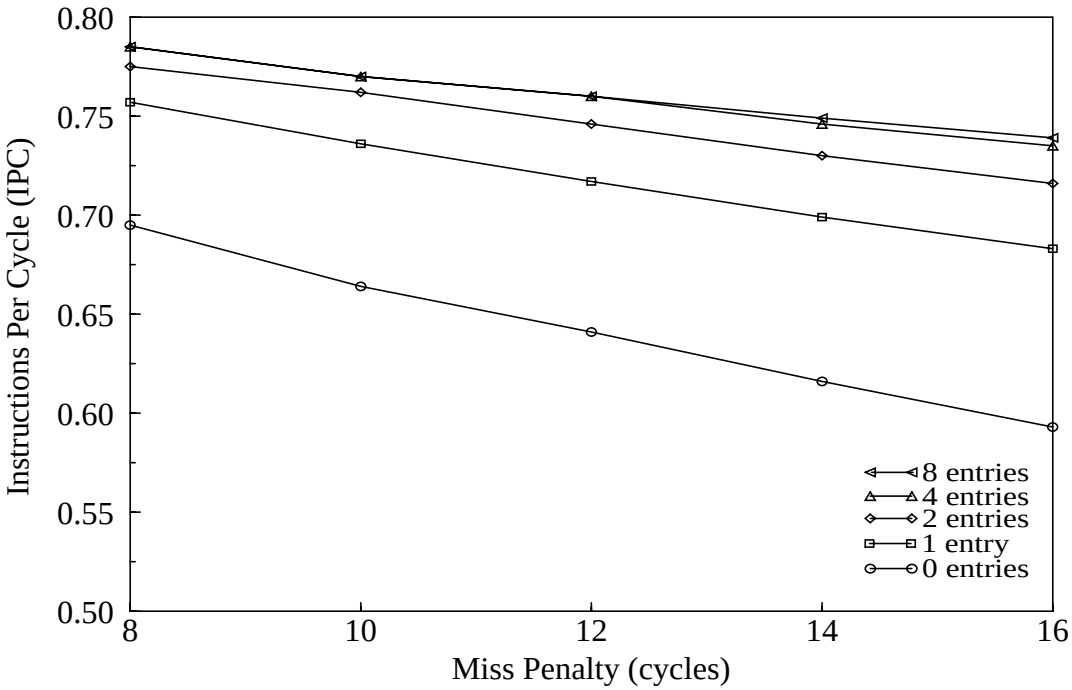


Fig. 5: Stream buffer compensating for increased memory latency.

ity of the processor to increased memory latency.

Our limited-resource design constraint dictates that the primary data cache be either a small on-chip cache, or a larger off-chip cache. The off-chip cache, by virtue of its larger size, achieves a lower miss rate than the smaller on-chip cache. This can translate into improved performance provided the additional off-chip latency does not negate the benefits of the lower miss rate. Fig. 6 plots the performance of the baseline machine with various data cache sizes and access latencies. An on-chip configuration would have a single-cycle latency. An off-chip cache may have a three-cycle latency, one cycle each for transmitting the index, accessing the array, and receiving the data. From these simulations, the general rule is that to maintain a given level of performance, the data cache must double in size for every additional cycle of latency incurred. An off-chip (three-cycle latency) data cache must be 16KB to provide the same performance as a 4KB on-chip (single-cycle) data cache. Processors that are unable to incorporate a data cache on the main die can place the cache off-chip provided that the interconnection is relatively high-speed and the cache is made large enough to compensate for the additional latency.

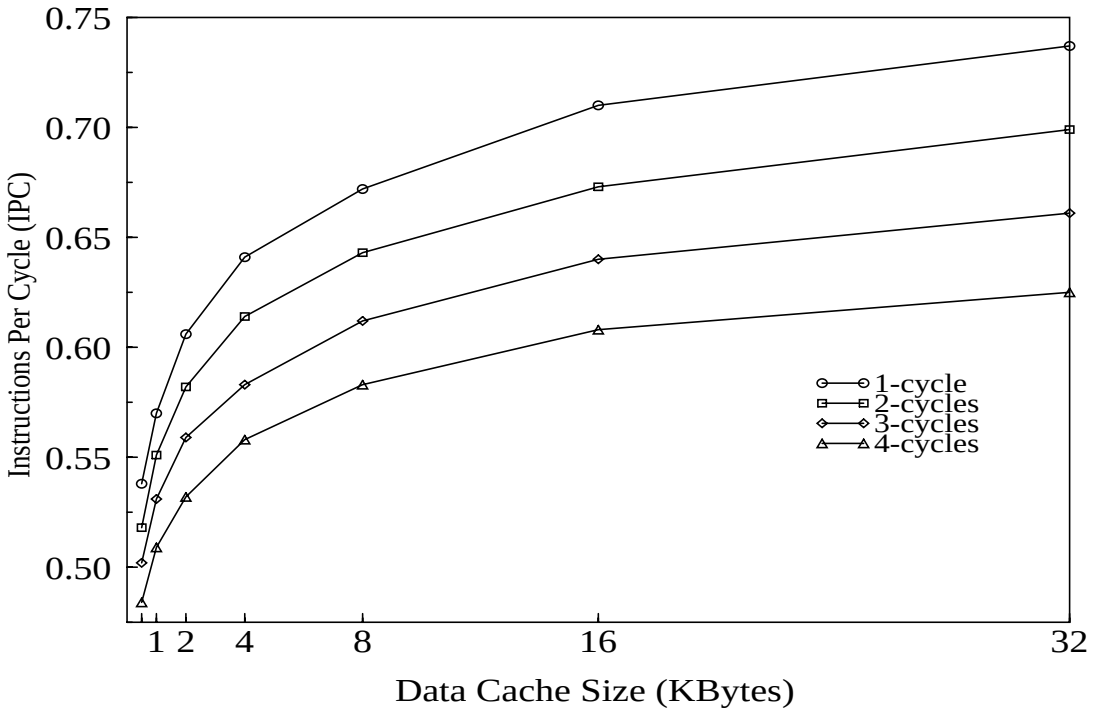


Fig. 6: Effects of increased primary data cache latency on overall performance.

Perhaps one of the most powerful techniques for achieving high performance is speculative execution, allowing the microprocessor to execute past unresolved branch instructions to achieve a higher degree of instruction level parallelism. Predicting the outcomes of branch instructions with only 50% accuracy would increase performance by 70%, while perfect prediction would provide a 117% improvement. Modern commercial microprocessors, such as the Alpha 21264 [9], use approximately 36K bits of branch prediction hardware to attain a prediction accuracy of 99%. Such a hardware intensive solution becomes impractical in a resource-limited microprocessor design. Fig. 7 shows the relationship between prediction accuracy (in terms of misprediction rate) and predictor cost, based on our simulations. The configuration labeled array is a simple array of two-bit counters [10]. The other schemes, GAg, GAs, SAg, SAs, PAg, PAs, and gshare, are two-level adaptive schemes [11]. These schemes are variations of the two-level branch predictor in which a branch history register (BHR) is used to index into a table of two-bit branch predictors (PHT). The schemes in the study differ on the number of BHRs and PHTs used. The simple counter scheme outperforms all two-level predictors at very low costs (below 1K-bits). However, above 1Kb, the two-level schemes are generally superior. A 1KB gshare predictor can achieve 90% accuracy; this is a simple and efficient enhancement to a resource-limited microprocessor.

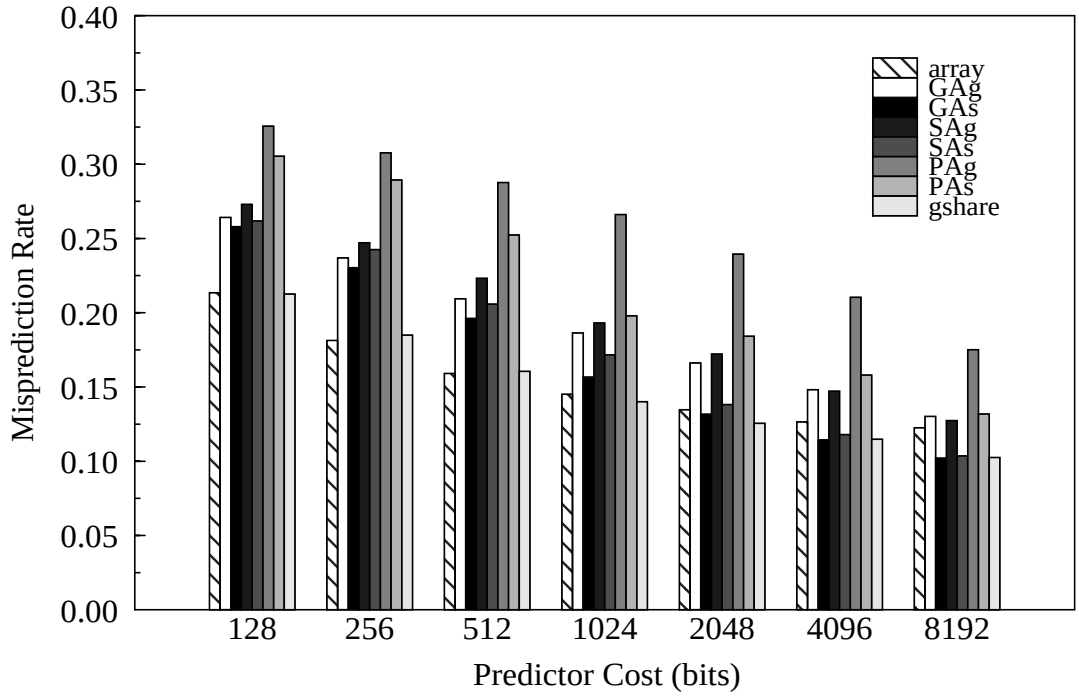


Fig. 7: Effects of predictor cost on misprediction rate.

The results of the previous studies, along with an additional investigation into the parameters that govern superscalar execution, were used to propose two microprocessor architectures requiring fewer than 1M transistors each. The configurations for the two proposed architectures, extra-small and small, are summarized in Table 2. The two processors take advantage of both instruction prefetching and speculative execution. The extra-small architecture uses a two-entry stream buffer along with 256B of branch prediction hardware (86% prediction accuracy). The small architecture uses a four-entry stream buffer and 1KB of branch prediction hardware (90% accuracy). The extra-small machine has an off-chip 16KB data cache with a 3 cycle access latency while the small processor has an on-chip 4KB data cache. The table also summarizes the performance relative to a comparable sequential-pipelined microprocessor and the baseline machine. The three architectures improve performance over a sequential pipeline by 10 to 40%. It is important to note that the extra-small design, with only half as many transistors, is able to outperform the baseline processor by almost 20%. The small architecture, with a transistor budget nearly equal to the baseline machine, improves performance by 45%.

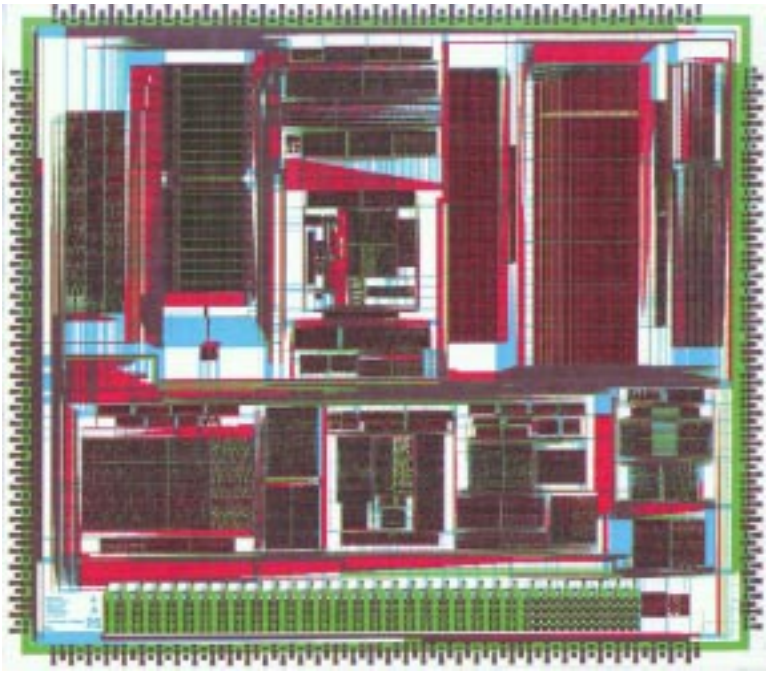
Table 2: Hardware budget of various machine configurations.

parameter	baseline	extra-small	small
superscalar width	2	2	2
branch predictor (KB)	0	0.256	1
reorder buffer (entries)	16	8	12
reservation stations (per functional unit)	4	2	3
branch target buffer (entries)	0	64	128
dcache size (KB)	4	0	4
stream buffer (entries)	0	2	4
icache size (KB)	4	1	2
transistors (K)	1035	503	1003
instructions per cycle (IPC)	0.6410	0.7642	0.9296
MIPS (at 200MHz)	128	153	186
advantage over sequential pipeline	1.1033	1.2749	1.3996
advantage over baseline configuration	1.0000	1.1922	1.4502

V. CGaAs 32-bit Microprocessor

A prototype of the architecture proposed in Section IV has been designed in Motorola's 0.5µm CGaAs process. The goal of the prototype microprocessor was to demonstrate viability of both the architecture and the technology. The processor is a 32-bit PowerPC Fixed-Point Unit (FXU). Because the current CGaAs process technology is able to support integration levels of only 400K transistors, the proposed architecture was simplified by removing instruction prefetching, and speculative, superscalar, and out-of-order execution, architectural elements needed to implement these features are still included in the design. The processor was designed using a standard cell library and a customized version of the EPOCH tool suite [12].

A plot of the physical layout and performance statistics for the FXU are presented in Fig. 8. The die size is large, 13 x 11 mm, owing in large part to the coarse CGaAs design rules, and compounded by the inefficiency associated with constructing a microprocessor entirely of standard cells. As this paper is being written, the design is still in fabrication. Static timing analysis predicts that the processor will operate at 73MHz, while power estimates show a consumption of 2.1 W when operating from a 1.3 V supply. The FXU is the first microprocessor developed in the CGaAs process technology. This design example shows the potential for a commercial microprocessor in the CGaAs technology.



- 32-bit Power PC
- 1KB Icache / off-chip 16KB Dcache
- 0.5μm CGaAs, 3 layer metallization
- 383,027 transistors
- 13.1 x 11.4 mm die
- 73 MHz
- 2.1 W @ 1.3 V
- 288 pins (228 I/O)
- 391-pin PGA
- 250 area I/O pads
- Standard cell design
- Full complementary circuit design

Fig. 8: CGaAs prototype microprocessor design statistics.

VI. Summary

The CGaAs process provides a low-power, radiation-hard, and relatively inexpensive alternative to design systems suitable for space applications. To overcome the low integration levels of the CGaAs process, key elements of the system were partitioned across several ICs.

The architectural simulations show that implementing a limited-transistor superscalar microprocessor in CGaAs is practical. A stream buffer was shown to be a relatively inexpensive means of increasing instruction cache performance. The studies proved that a single-entry stream buffer more than doubled the effective instruction cache size, and compensated for an extra eight cycles of secondary cache latency. Simulations have shown that an off-chip data cache is able to outperform a smaller on-chip cache, despite the additional access latency. As a general rule, the off-chip cache must double in size for every additional cycle of memory latency, as compared to an on-chip single-cycle cache structure. Studies of branch prediction configurations with limited hardware costs demonstrated that 90% prediction accuracy could be achieved with a memory structure of only 1KB. Finally, two proposed microarchitectures were shown to significantly outperform the traditional sequential pipeline, proving that efficient superscalar architectures can be implemented with fewer than 1M transistors. A 500K-transistor design outperformed the sequential pipeline by 27%, while a 1M-transistor design provided a 40% improvement.

The development of a prototype CGaAs microprocessor validates the proposed architecture and shows that digital CGaAs is a viable technology for microprocessors targeted for space applications.

ACKNOWLEDGMENT

This work is funded by the U.S. Defense Advanced Research Projects Agency, under ARO DAAH04-94-G-0327.

REFERENCES

[1] J. Benedetto, “Economy-Class Ion-Defying ICs in Orbit,” IEEE Spectrum, vol. 35, no. 3, pp. 36-41, March 1998.

[2] R. Brown, B. Bernhardt, M. LaMacchia, J. Abrokwah, P. Parakh, T. Basso, S. Gold, S. Stetson, C. Gauthier, D. Foster, B. Crawforth, T. McQuire, K. Sakallah, R. Lomax, and T. Mudge, “Overview of Complementary GaAs Technology for High-Speed VLSI Circuits,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 6, no. 1, pp. 47-51, March 1998.

[3] M. LaMacchia, J. Abrokwah, B. Bernhardt, D. Foster, B. Crawforth, B. Mathes, T. McQuire, and T. Weatherford, “Radiation Hardened Complementary GaAs (CGaAS™),” Proceedings of the 1997 GaAs IC Symposium, pp. 59-61.

[4] R. Brown, T. Basso, P. Parakh, S. Gold, C. Gauthier, R. Lomax, and T. Mudge, “Complementary GaAs Technology for a GHz Microprocessor,” Proceedings of the 1996 GaAs IC Symposium, pp. 313-316.

[5] B. Davis, C. Gauthier, P. Parakh, T. Basso, C. Lefurgy, R. Brown, and T. Mudge, “Impact of MCM’s on High Performance Processors,” INTERpack-97: International, Intersociety Electronic & Photonic Packaging Conference, June 1997.

[6] M. Johnson, *Superscalar Microprocessor Design*, Prentice Hall, Englewood Cliffs, New Jersey, 1990.

[7] D. Patterson, and J. Hennessy, *Computer Architecture: A Quantitative Approach*, 2nd ed., Morgan Kaufmann Publishers, Inc., San Francisco, California, 1996

[8] N. Jouppi, “Improving Direct-Mapped Cache Performance by the Addition of a Small Fully-Associative Cache and Prefetch Buffers,” International Symposium on Computer Architecture, May 1990, pp. 364-373.

[9] D. Leibholz, and R. Razdan, “The Alpha 21264: A 500MHz Out-of-Order Execution Microprocessor,” IEEE COMPCON 1997, pp. 28-36.

[10] J. E. Smith, “A Study of Branch Prediction Strategies,” Proceedings of the 8th International Symposium on Computer Architecture, pp. 135-148, May 1981.

[11] T-Y Yeh, and Y. Patt, “Two-Level Adaptive Training Branch Prediction,” The 24th ACM/IEEE International Symposium and Workshop on Microarchitecture, pp. 51-61, Nov. 1991.

[12] Cascade Design Automation, *EPOCH User’s Manual*, *ver. 3.2*, Bellevue, WA 98006, 1995.