

IEEE Princeton Section

Sarnoff Symposium



March 26, 1993

David Sarnoff Research Center

Princeton, New Jersey



IEEE



Sponsored by IEEE/LEOS, the LEOS and MTT-S/ED/AP Chapters of the IEEE Princeton Section and in cooperation with MTT-S

A High Performance GaAs Microprocessor

T. R. Huff, M. Upton, P. Sherhart, P. Barker, R. McVay, T. Stanley, R. Brown, R. Lomax, T. Mudge, K. Sakallah
Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan 48109-2122

ABSTRACT: A 32-bit RISC microprocessor has been fabricated in a 0.6 μ m GaAs DCFL process. It includes 160,000 transistors on a 13.9 x 7.8mm² chip, and dissipates 24W. The chip contains an ALU, 32x32 register file, 4-word write buffer, small on-chip I-cache, and support for off-chip instruction and data caches.

1. INTRODUCTION

Since the introduction of GaAs circuits, high-performance digital systems have been considered a major application area for compound semiconductor technology. Early overoptimistic predictions of the digital GaAs performance advantage over silicon, followed by the fabrication challenges of an immature technology, caused a negative reaction among system designers which hindered acceptance of GaAs. To support VLSI circuits, a process technology must have, in addition to good device switching times: high levels of integration, good yields, reasonable power dissipation, and dense, multilevel interconnect. A digital logic family should have gates with good load-driving characteristics, reasonable noise margins, special features for embedded memory, and support by appropriate design automation tools. Only recently has the promise for high-speed digital VLSI in compound semiconductors begun to be fulfilled.

Every technology has its own unique design qualities and GaAs is no exception. Currently, the GaAs logic family which offers the most potential for large digital designs is direct-coupled FET logic (DCFL) which is characterized by NMOS-like circuit topologies, low noise margins and fast gate speeds. This paper will describe the Aurora II 200MHz RISC microprocessor [1] which is implemented in DCFL and will highlight the lessons learned with regard to circuit issues and the importance of advanced interconnect for high speed GaAs VLSI circuits.

2. DESIGN ENVIRONMENT

Our design methodology and tools provide the needed support for high-performance design in general, and for DCFL in particular. The tools provide design-rule portability, so that a given design can be evaluated in different rule-sets or easily translated into a newer rule-set. The design methodology creates circuits with physical datapaths organized as one would in a handcrafted design, minimizing chip area and total interconnect length compared to standard cell- or array-based methodologies. The routers support multilevel interconnect, variable width signal routing, multiphase clock distribution, ground planes, and automatic power-rail sizing for IR drop and electromigration. The analysis capability includes automated approaches for accurate parasitic extraction, delay calculation, back-annotation to several commercial simulation environments, delay-accurate digital simulation, critical path identification, and

analysis of clock distribution. The clock analysis supports single or two-phase non-overlapping clocks; in the Aurora II processor, we minimized local skew along critical paths to less than 400 ps.

3. MICROPROCESSOR OVERVIEW

The single-chip GaAs microprocessor described includes a Ling adder-based ALU, 32-bit shifter, 32-word register file, 4-word write buffer, 32-word on-chip instruction cache, support for 2 levels of off-chip instruction and data caches and an asynchronous system interface. It integrates 160,000 transistors on a 13.9 by 7.8-mm die. When operating from a 2-V supply, the chip typically dissipates 24 W. A photomicrograph of the integrated circuit is shown in Fig. 1.

The microprocessor implements 39 instructions from the MIPS R3000 instruction set; the chip is restricted to full-word loads and stores with byte instructions emulated in software. Exceptions, system calls and stalling are all supported. Most of the control circuitry was synthesized, but a few critical sections were hand generated. All control signals are pipelined and decoded during the cycle before they are used. The chip was fabricated in the Vitesse HGaAs III E/D MESFET process, which features 1- μm (0.6 μm effective) channel lengths, refractory gate metal (also used for intracell wiring), three levels of aluminum metallization for signal and power interconnection, and a top level aluminum ground plane. Two-phase clocking is used to minimize clock skew problems. The microprocessor was designed in six months with high level synthesis and automated physical layout generation based on Cascade Design Automation tools.

The 32-bit Ling adder, used in several sections of the chip, consists of 3598 transistors, has a density of 2125 transistors/ mm^2 , and operates with a delay of 1.6 ns. The 32-bit by 32-word register file is similar to the on-chip

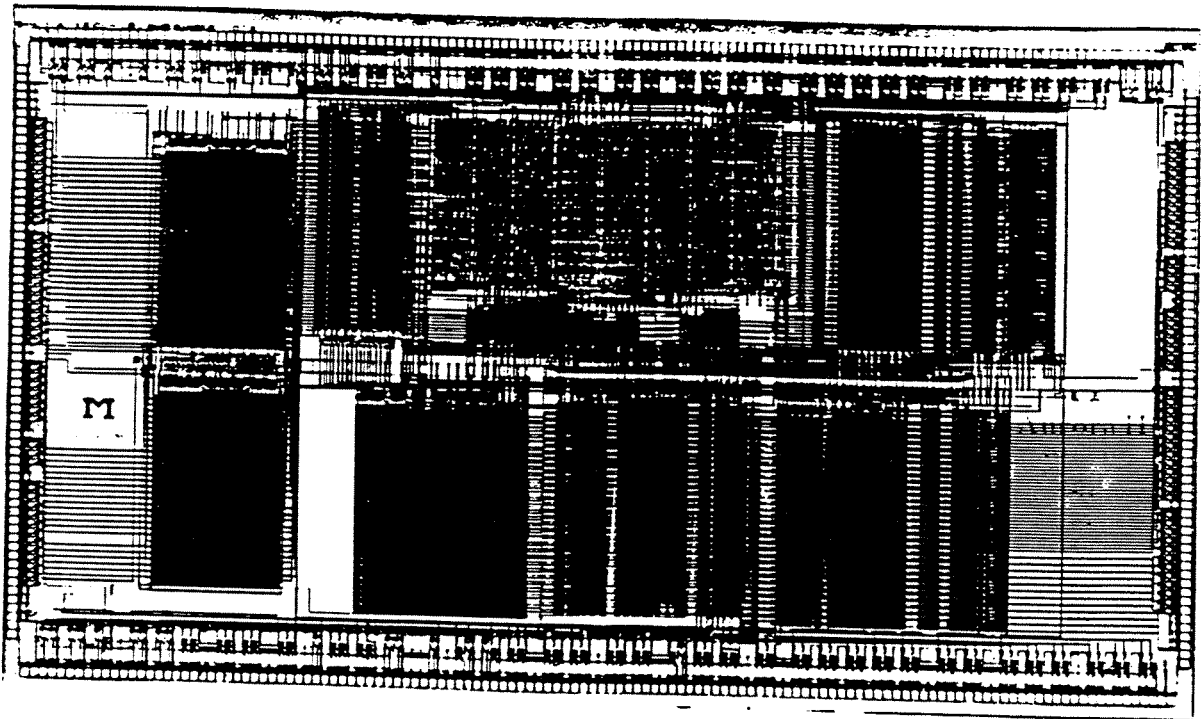


Fig. 1. Aurora II GaAs Microprocessor

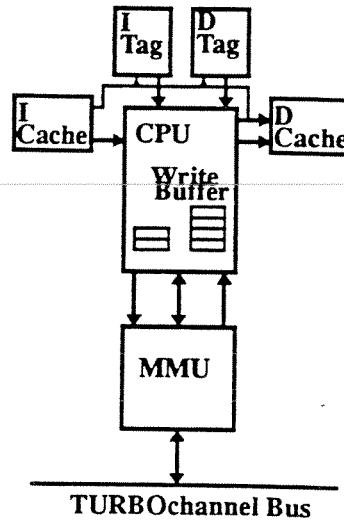


Fig. 2. Aurora II Block Diagram.

instruction cache; it has 23,278 transistors with a density of 4237 transistors/mm², and is accessible in 1.7 ns. Control for the chip is comprised of 14,922 transistors, with a density of 353 transistors/mm². Overall circuit density is 2474 transistors/mm².

The microprocessor and four GaAs static RAM chips make up a high speed processing module (Fig. 2). All information needed for cache miss detection is brought on chip to minimize miss detection delay; instruction and data cache tag comparators are on the CPU chip. The processor communicates with an off-module MMU via a 32-bit bidirectional ECL-level asynchronous bus. The processor handles all cache miss processing using this bus and four 32-bit interface registers. The first-level memory interface can be operated in one of four modes: force-hit (buffer), force-miss, external cache (instruction and data), and internal cache (instruction).

When a cache miss or exception is detected, the address of the offending word is loaded into the instruction or data interface register, as appropriate, and the MMU is signaled that data is ready to be transmitted. The MMU reads the address over the MMU bus and returns the requested data to the appropriate register. The CPU then writes the cache tag and cache data with the updated values and releases the pipeline.

Portions of the chip operate at 200MHz and full functionality has been verified at 100MHz. The speed of the instruction decode in the first prototypes is limited by an incorrect clock-phase assignment. A corrected version of the chip is currently in fabrication

4. DIGITAL CIRCUIT ISSUES IN GAAS

As mentioned, DCFL logic gates are similar in topology to NMOS, with inverters and NOR gates comprising the basic building blocks. Enhancement pulldown and depletion pullup devices are ratioed in such a way as to provide desired output high and low voltages over normal operating conditions. Depletion devices are usually source-gate

connected to provide a constant current source. Gate delay for an unloaded device is on the order of 60ps and loaded gates typically have delays in the range of 100-150ps. The gate of a MESFET is actually a Schottky diode and there is no gate dielectric as is normally found in MOSFET devices. The diode gate introduces several unique issues into the design of VLSI circuits, one example of which is the small voltage swing for these devices. Since the gate acts as a diode, the gate voltage for a logic high will be clamped to a single diode drop, on the order of 0.6 volts. For logic gates to function properly with such low output-high voltages, their enhancement transistors must have small threshold voltages, typically about 0.2 volts. This makes designs sensitive to voltage drops along the ground rail. A top-level aluminum plane is used to provide a clean ground with less than 20mV of noise; each cell connects directly to this plane. IR drops along Vdd are not as critical and gates operate correctly with little loss in speed for a power supply voltage as low as 1.2 volts. Power is routed in Metal 3 and is sized such that no gate sees an IR drop along Vdd of more than 0.5 volts.

The diode gate of a MESFET also results in an unusual inverter transfer characteristic, as shown for the NOR gate of Fig. 3. As the input voltage increases beyond 0.8 volts, the output voltage begins to rise; at high enough input voltages the output erroneously becomes a logic one. This phenomenon occurs when the diode gate current is large and the gate-drain junction becomes forward biased. Large currents are required to change the state of highly capacitive interconnect quickly. The buffer used to drive such a net may provide a current which is appropriate for charging the wire but too large for destination gates. A feedback buffer is used to solve this problem, as shown in Fig. 4 [2]. This buffer provides a large transient current to charge the lines, then reduces its drive, providing a smaller current to maintain a stable logic high voltage.

Another characteristic of the GaAs technology described is a high transistor source resistance, which tends to limit the use of stacked transistor logic, such as NAND gates. The use of only DCFL NOR gates can increase the number of gate levels on critical paths unless special circuit structures are used. One such structure used extensively on the CPU is an Earle latch that combines a 2-input mux with a latch and a high current output buffer. The latch output buffer operates in a feedback mode similar to that just discussed. This circuit accounts for 40% of the chip area.

5. EMPHASIS ON INTERCONNECT

The importance of interconnect in a VLSI process cannot be overstated. The switching delay, τ , for any logic family, is related to the difference in charge between states at the output of a logic gate, and to the current available to effect

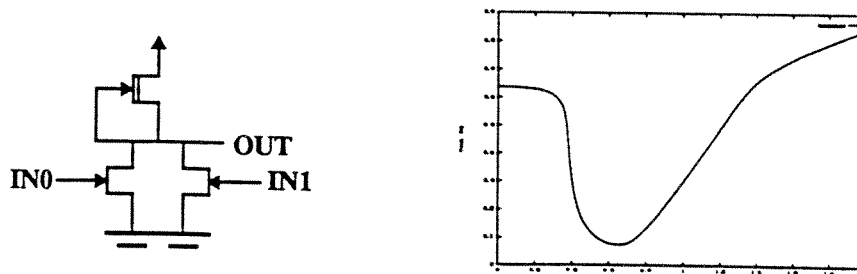


Fig. 3. Two-input NOR and its transfer function.

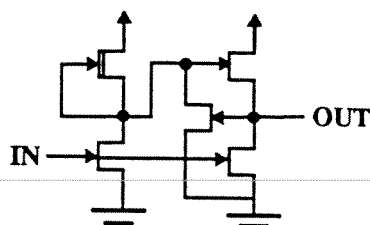


Fig. 4. Feedback Buffer.

a change of state: $\tau \propto \frac{C\Delta V}{I}$. Sensitivity to parasitic loading varies with process and logic family. In FET technologies, this is the dominant delay mechanism; it calls for small logic swings, high transconductance, and low parasitic capacitance.

Most of the parasitic capacitance comes from interconnect. Of primary importance is keeping the circuit area as small as possible to minimize wire length; this reduces both parasitic capacitance and time-of-flight for signals. Routing capacitance is minimized by using enough levels of interconnect, narrower lines, larger separation between interconnect layers, and lower dielectric-constant insulators. The effect of narrowing the separation between lines is not immediately obvious; while it reduces the circuit area, it does increase horizontal line-to-line capacitance. However, the total-routing-area data shown in Table 1 makes a strong case for reducing interconnect spacing to the fabrication limit [3]. Table 1 also demonstrates how smaller transistor dimensions affect overall layout utilization.

The importance of minimizing interconnect capacitance is illustrated in Figs 5 and 6, which show the effects of reducing capacitive load (Fig. 5) and of reducing unloaded gate delay (Fig. 6) on four critical paths in our microprocessor. The logic paths in these plots are from the register file (RF), adder (A1 and A2), and branch logic (BR). (These figures ignore the fact that faster gates would have greater transconductance and therefore drive the capacitive loads more effectively). The plots do show clearly that performance is dominated by interconnect loading, and therefore, reducing interconnect capacitance would be even more effective at increasing circuit speed than would reducing intrinsic gate delay. The sensitivity to these effects vary among the paths simulated. The closest results are for the branch logic, where a 50% reduction in capacitance has a 40% greater effect than a similar reduction in unloaded gate delay. The biggest difference is in the register file, where capacitance has a 248% greater effect. The branch path consists of a large number of lightly loaded paths, whereas the RF path involves a smaller number of heavily loaded gates.

Table 1: Comparison of 8x8 multipliers in three DCFL processes. All parameters are normalized.

	Gate Metal	Metal 1	Metal 2	Metal 3	Total Layout Area	Total Routing Area
Process A	1.00	1.00	1.00	1.00	1.00	1.00
Process B	0.90	0.60	0.50	0.28	0.49	0.21
Process C	0.50	0.97	1.11	1.43	0.97	0.82

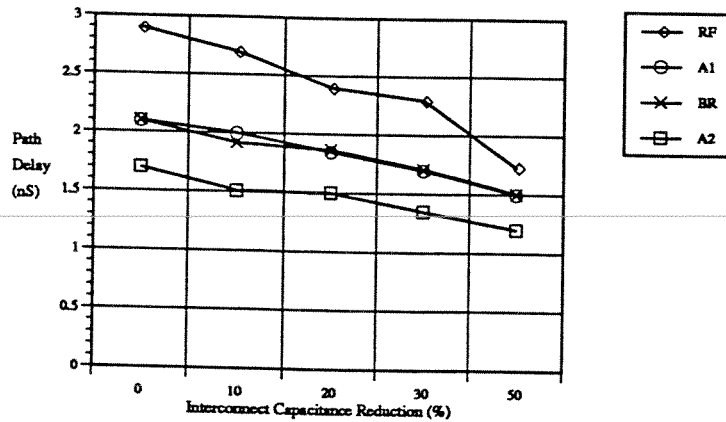


Figure 4

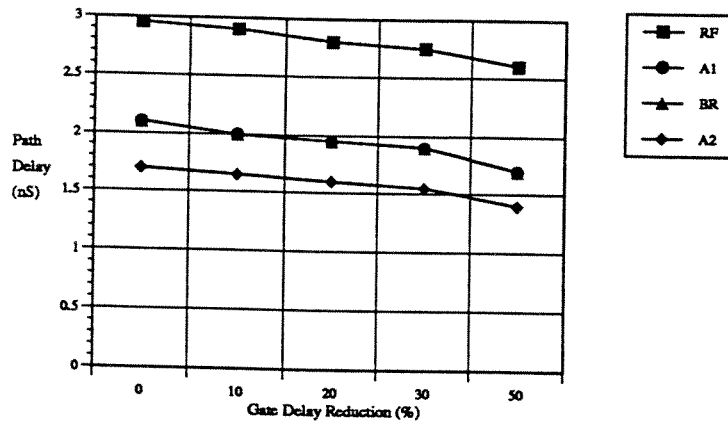


Figure 5

The importance of having enough layers of interconnect merits further illustration. In our designs, we use Gate Metal and Metal 1 for wiring inside of leaf cells, and Metal 1, 2, and 3 for datapath, standard cell, and global routing. Metal 4 is a ground plane, and Vdd is distributed on Metal 3. Table 2 shows the improvement in density which we have achieved in moving from HGaAs II (a 3-metal process) to HGaAs III (a 4-metal process). Of course, geometric design rule changes between the processes and other factors also effect the density. The control blocks are different circuits (bypass logic in HGaAs II and stall logic in HGaAs III), but they are about the same size, and both are implemented in standard cells using the same logic synthesis tool (Finesse, from Cascade Design Automation). The register files in Table 2 are both 32-word x 32-bit, three-port, tree-decoded, pass-gate latch implementations, which differ only in buffering

The density numbers for the CPU's include all of the unoccupied space in the pad frame - there is actually more space in the version with 4-metal interconnect. Some of the increase in density is due to the inclusion of additional memory structures for the small on-chip instruction cache on the 4-metal chip. But aside from this, the HGaAs III version of the CPU is still about 2.4 times denser. In our analysis, half of this improvement is due to the third layer of routing;

Table 2: Density comparison between 3-metal and 4-metal processes.

Circuit	HGaAs II Transistor Count	Density (Trans./mm ²)	HGaAs III Transistor Count	Density (Trans./mm ²)
Largest Control Block	582	1067	516	1364
Register File	21,910	2014	23,278	4253
CPU	60,500	540	160,000	1475

improved circuit structures and layout techniques incorporated into our newer CAD tools account for another 35%; and the remaining 15% of improvement results from smaller line widths in the HGaAs III process.

Adding interconnect layers to a digital process beyond a routeable gate metal, 3 interconnect levels, and a ground plane would result in diminishing returns. Trying to achieve high performance with fewer layers than this or with coarse interconnect pitch or an inefficient design style, though, starts a vicious cycle. A larger layout has more capacitance, therefore requiring larger buffers, which further increase the layout size, parasitic capacitance and power dissipation, requiring larger buffers, yet.

5.0 SUMMARY

It is the performance of semiconductor processes in systems, not ring-oscillator speeds, that will dictate their future in digital systems. Device and process development for digital circuits should be as concerned with integration levels, dictated by yield and power dissipation, as with the inherent speed of the nonlinear devices. Circuit performance can be improved more by improving interconnect than by improving device switching speed. All technologies have characteristics which present unique circuit design challenges and these must not overly constrain the inherent benefits of the technology. And finally, design methodology and CAD tools have a major influence on the competitiveness of a technology.

REFERENCES

- [1] Upton, M., et al., "A 160,000 Transistor GaAs Microprocessor", ISSCC DIGEST OF TECHNICAL PAPERS, pp. 90-91, Feb. 1989.
- [2] Fulkerson, D. E., "Feedback FET Logic: A Robust, High-Speed, Low-Power GaAs Logic Family", IEEE J. Solid-State Circuits, vol. 26, pp. 70-74, January 1991.
- [3] Brown, R. B., "Compound Semiconductor Device Requirements for VLSI", Symposium on GaAs and Related Compounds, Sept. 1992, Karuizawa, Japan.