Design of a 32 b Monolithic Microprocessor Based on GaAs HMESFET Technology

Chien-Kuo V. Tien, Kelvin Lewis, Hans J. Greub,
Tom Tsen, and John F. McDonald

Abstract—This paper examines the design of a 32-b GaAs Fast RISC microprocessor (F-RISC/I). F-RISC/I is a single chip GaAs HMESFET processor targeted for implementation on a multichip module (MCM) together with cache memories. The CPU architecture, circuit design, implementation, and testing are optimized for a seven-stage instruction pipeline implemented with GaAs super-buffered FET logic (SBFL). We have been able to verify novel GaAs SBFL standard cells and compare measured CPU performance with performance estimates based on circuit and device models. The prototype 32-b microprocessor has been implemented using an automated standard cell approach because of time constraints and fabricated using an experimental process by Rockwell International. The CPU chip integrates 92 340 transistors on a $7 \times 7 \text{ mm}^2$ die and dissipates 6.13 W at 180 MHz. Test results from a prototype fabrication run have demonstrated the operation of the ALU, the program counter, and the register file with delays below 6, 5, and 3.4 ns, respectively. The successful modeling and verification indicate that a 0.5 μm HMESFET implementation of F-RISC/I could achieve a peak performance of 350 MHz. The wiring delays account for 42% of the critical path delay.

Index Terms—GaAs HMESFET, instruction pipeline, microprocessor design, multichip module (MCM), reduced instruction set computer (RISC), super-buffered FET logic (SBFL).

I. INTRODUCTION

Recent advances in GaAs Heterojunction MESFET (HMESFET) technology have led to gate delays below 100 ps [1] and higher integration levels, reaching VLSI complexity and, thereby, allowing the implementation of a 32 b GaAs RISC on a single chip [2]. However, integration levels are still very low compared to CMOS and do not allow the inclusion of sufficiently large caches on the chip. The cache memories must be implemented with high speed SRAM chips which need to be placed close to the CPU chip on an MCM to keep the interconnect delays low. The processor design, therefore, must consider the interactions between architecture, circuit technology, and MCM packaging. The main issues in GaAs microprocessor design are the processor versus memory speed mismatch and the limited off-chip communication bandwidth.

To overcome the difficulties of limited yield and low I/O bandwidth in GaAs, the high speed processing node, consisting of the processor and cache memory hierarchy, must be densely implemented on an MCM [3], [4]. F-RISC/I employs further a pipelined cache memory access [5] to "hide" some of the chip-to-chip delays in pipeline stages since, even on an MCM, the address and data transfer times between chips are of the same order as the processor delays.

Manuscript received March 16, 1996; revised July 29, 1996. This work was supported in part by the IBM T. J. Watson Research Center and Rockwell International and also in part by the companion F-RISC/G Research under ARPA/ARO Contract DAAL03-90-G-0817.

C.-K. V. Tien, H. J. Greub, and J. F. McDonald are with the Center for Integrated Electronics, Rensselaer Polytechnic Institute, Troy, NY 12180 USA.

K. Lewis is with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA.

T. Tsen is with the Microelectronics Technology Center, Rockwell International Corporation, Newbury Park, CA 91320 USA.

Publisher Item Identifier S 1063-8210(97)00741-5.

Fig. 1. FRISC/I MCM system.

The primary goals of the Fast RISC/I (F-RISC/I) project were to verify the novel GaAs SBFL standard cells, to verify that HMESFET yields have reached adequate levels, and to correlate measured CPU performance with simulations based on circuit and device models to check the modeling capabilities of our CAD tools. F-RISC/I is a companion project to the ARPA sponsored heterojunction bipolar transistor (HBT) F-RISC project.

II. GAAS MICROPROCESSOR DESIGN

Fig. 1 shows the F-RISC/I MCM system. High memory bandwidth is achieved using separate instruction and data caches with their own data buses. This allows one 32 b instruction and one 32 b data word to be supplied by the cache memories in each cycle. A shared address bus is used to communicate with both the instruction and data cache memories to reduce CPU pin count and interconnections. This requires a remote program counter (RPC) on the instruction cache controller. Using the RPC the instruction cache can access consecutive instructions without an address transfer from the CPU. The instruction memory needs an address from the CPU only if a branch or an exception is taken. The shared address bus never causes contention in this scalar architecture since load/store and branch instructions are designed to use the address bus in the same pipeline stage.

The relative performance figure of a processor implementation is usually expressed in MIPS (millions of instructions per second), and is inversely proportional to the cycle time T_{cycle} and the average number of Cycles per Instruction (CPI). The principal parameters affecting T_{cycle} are the GaAs circuit technology and the pipeline. The CPI for a scalar RISC processor is basically one instruction cycle plus the average number of wasted cycles due to pipeline hazards, such as branch and load penalties, and stall cycles after cache misses. The CPU/system performance must be optimized by considering both the architecture (especially the pipeline scheme) and

Fig. 2. Potential instruction pipeline schemes for F-RISC/I.

technology parameters (GaAs MESFET and MCM technology) since a change in the design parameters can affect cycle time, pipeline dependencies, and/or cache miss rate in opposite ways. Since F-RISC/I is a processor without pipeline interlocks the load and branch latencies are visible at the architecture level and thus the pipeline depth is not just an implementation issue. In order to select the most advantageous pipeline for the given circuit, memory, and packaging technology, a first order performance evaluation of different pipeline schemes is necessary [5].

The evaluation starts by prototyping the critical circuits, such as the ALU and register file for the HMESFET circuit technology. Then chip placement, wiring rules, I/O pad, and layer assignments as well as propagation delay models are formulated. The MCM characteristics are: delay = 10.4 ps/mm, thermal resistance = 14°C/W, maximum power density = 8 W/cm². The cache memory design is based on a 1.5 ns 4 k × 16 b BiCMOS memory with a power dissipation of 4.5 W described in [6].

The potential instruction pipelines, shown in Fig. 2, were considered and evaluated based on cycle time and CPI. The nine-stage pipeline allocates one full cycle for address/data transfer and one CPU cycle for cache memory access to allow a larger cache size and to minimize cache miss penalties. The seven-stage pipeline can also achieve a cycle time set by the GaAs technology and also provides a pipelined cache memory access, but it only provides half a CPU cycle for address/data transfers. The cycle time of the five-stage and four-stage pipelines are limited to the same cycle time, set by the cache memory access in the Data I/O (D) stage. Since the four-stage pipeline has a lower branch latency, the five-stage pipeline can not be optimal and does not need to be considered further. Although the four-stage pipeline has a longer cycle time with address/data transfer and cache memory access performed in one cycle, it has lower branch and load latencies. To get a first order estimate of the pipeline efficiency of each pipeline scheme, compiler branch and load delay slot fill-in probabilities [7] for a six-stage RISC pipeline machine are used. Table I shows the CPI contributions of branch and load latencies for a dynamic instruction mix derived from set of typical UNIX programs [8].

A longer cycle time or more cycles for address/data transfer allows the implementation of larger caches on the MCM because more

TABLE I
CPI CONTRIBUTIONS DUE TO BRANCH AND LOAD PENALTIES

Instruction	Instruction frequency ¹	Cost 4-stage	Cost 7-stage	Cost 9-stage
ALU	60%	1	1	1
BRANCH (not taken)	5%	1	1	1
BRANCH (taken) ²	15%	1.368	2.673	3.673
LOAD ³	15%	1.1	1.5	2.3
STORE	5%	1	1	1
CPI intrinsic		1.07	1.326	1.596

¹ Average dynamic instruction mix [8].

² Cost of a taken branch depends upon the branch latency slot fill-in probability of the compiler. The compiler branch fill-in probabilities are taken from [7]. (4-fill = 0, 3-fills = 36.8%, 2-fills = 20.2%, 1-fill = 16.5% and 0-fill = 26.5%).

³ Cost of a load instruction depends upon the load latency slot fill-in probability of the compiler. The compiler load latency slot fill-in probabilities are taken from [5].

Fig. 3. Chip placement on MCM for seven-stage pipeline F-RISC/I.

SRAM chips can be reached within the available address/data transfer time resulting in lower cache miss penalties. However, a longer cycle time (four-stage pipeline) results in a slower peak instruction execution rate. Allowing more cycles/stages (seven- and nine-stage pipeline) for address/data transfer increases load and branch penalties. The tradeoff between cycle time, load/branch penalties, and cache miss penalties can be evaluated by comparing the relative performance among these three pipeline candidates in a spread sheet.

Based on the address/data transfer time for each pipeline candidate, a preliminary placement of the cache memory (SRAM) chips can be determined. The chip-placement is then handled in the package design phase which includes the considerations of net routability, noise tolerance, and thermal management [9], [10]. The final package design allows the system designer to evaluate the cache miss penalties primarily based on the cache organization and the size of cache memory.

The size of the first-level cache for the seven- and nine-stage pipeline primarily depends on three factors: thermal management, net routability/topology, and allocated address/data transfer time. Close placement of a large number of chips makes heat removal more challenging and more expensive. We place the chips in a two-dimensional array with a chip pitch of 8 mm on the MCM. The junction-to-ambient resistance for each chip is estimated to be 14°C/W [9], [10] and results in a junction temperature of 64°C above ambient temperature. Since the rise time of the signals on the MCM are in the range of 200–300 ps and the MCM interconnects are 50–60 Ω transmission lines, each long net must be routed in a chaining tree (no forks) to avoid reflections. For example, the address bus originating from the cache controller chip needs to be routed as a chained net across all cache memory chips. Fig. 3 shows an example of a chip placement for the seven-stage pipeline scheme.

TABLE II
PERFORMANCE VERSUS PIPELINE DEPTH

Pipeline Depth	T_{cycle} (Relative)	$CPI_{intrinsic}$	CPI_{cache_miss}	Performance (Relative)
4-stage	1	1.07	0.188	1
7-stage	0.625	1.326	0.202	1.317
9-stage	0.625	1.596	0.140	1.159

Cache miss ratios are based on SPEC92 benchmark trace data [11].

In order to evaluate the average number of stall cycles due to cache misses, assumptions about the second-level cache and the main memory and memory bus bandwidth must be made. Considering implementation cost and switching noise, the bus sizes between the first-level and the second-level as well as the second-level and the main memory are fixed at 16 and 32 bytes, respectively. The second-level cache is direct-mapped, and unified. It has a size of 1 Mbyte and a block size of 64 bytes. Main memory is assumed to be infinite and two-way interleaved. The primary data cache uses write-through to keep the cache and memory coherent. The ratio between the memory cycle time and the CPU cycle time (seven-stage) for the first-level, second-level, and main memory are 1, 4, and 16. The instruction and data cache sizes that yield optimal performance given the MCM and the BiCMOS memory characteristics are (32k, 32k) for the four-stage, (64k, 64k) for the seven-stage, and (128k, 128k) for the nine-stage pipeline. We calculated the cache miss penalties using cache miss ratios from the SPEC92 benchmark suite [11] and used the published statistics [5], [7], [8] for pipeline dependencies. We used SPEC92 benchmark data for a MIPS architecture [11] which has an instruction set similar to that of the F-RISC architecture.

Table II compares the relative performance of three pipeline schemes. Clearly the seven-stage pipeline performs better than the four-stage and nine-stage pipeline. The nine-stage pipeline has the lowest cache miss penalties, but it suffers from large branch/load penalties. The four-stage pipeline has the lowest branch/load penalties and reasonably low cache miss penalties, but its longer cycle time overshadows its higher pipeline efficiency.

III. CIRCUIT DESIGN AND PROTOTYPE PERFORMANCE

Conventional MESFET devices use Schottky barriers to provide gate isolation. The logic swing of a gate is typically between 0.6–0.7 V, limited by the turn-on voltage of the gate diode. This limited logic swing places stringent requirements on the control of the threshold voltage, power rail voltage drops, temperature effects, and fan-in effects for large GaAs circuits. The HMFET process developed at Rockwell (Fig. 4) [1] uses a thin AlGaAs layer under the gate. The Schottky barrier at the surface has the same built-in voltage as a conventional MESFET, but the forward-bias gate current is limited by tunneling through the AlGaAs barrier. The AlGaAs barrier provides a larger turn-on voltage (1.25 V), lower leakage currents, and hence a larger logic swing. The advantages gained from HMFET logic include higher noise margin, improved performance, and lower temperature sensitivity. Most importantly, it reduces yield losses due to random threshold voltage variation.

Direct coupled FET logic (DCFL) is popular for realization of digital circuits because of its high-speed performance and low complexity. However, DCFL has a limited fan-in and fan-out capability and a high sensitivity toward capacitive loading. Therefore, DCFL usually requires more logic levels per function than CMOS [13]. In addition the nonzero voltage low (VOL) is very sensitive to E-mode threshold voltage shifts. This is aggravated when attempting to size the devices in a gate for high drive capability.

Fig. 4. Cross section of Rockwell's HMFET device.

Fig. 5. Schematics of DCFL and SBFL NOR-2 gates.

Super-buffered FET logic (SBFL) cascades a quasicomplementary output buffer stage after the DCFL input stage. The output buffer stage improves noise margin because of the zero-voltage low ($VOL = 0$). Although the input capacitances are approximately doubled in SBFL, the higher current-drive capability still yields lower delays than DCFL. Fig. 5 shows DCFL and SBFL NOR-2 gates. The comparison between DCFL and SBFL NOR-3 gate delays as function of wire length (with fan-out = 3) and fan-out (with wire length = 0.05 mm) are shown in Fig. 6(a) and (b). An SBFL gate compared with an DCFL gate at an equivalent power level has a least twice the drive capability. SBFL has a lower delay than DCFL for a fan-out greater than four and/or interconnect wires longer than 0.1 mm. The power supply voltages for the DCFL input stage and output buffer stage are 1.6 V (V_{dd1}) and 1.2 V (V_{dd2}). The combination of V_{dd1} and V_{dd2} is chosen to make the pull-up device (Q_2) at the output buffer stage operate with a saturation current if Q_2 is turned on. The large logic swing between the DCFL and the buffer stage provided by V_{dd1} promotes the current-drive capability even further since Q_2 is in saturation with a current quadratic in $(V_{gs} - V_{th})$. Keeping V_{dd2} below the clamping voltage of 1.25 V also reduces the static power dissipation of SBFL.

Fig. 7 shows the datapath of F-RISC/I. The potential critical paths are the PC increment in I1 stage, register file reads in DE stage, ALU execution and result feed-forward in the EX stage. The longest path starts at the outputs of the result register (RES_EX), goes through the multiplexers and the ALU, and ends at the input of RES_EX. This critical path is exercised when an ADD instruction needs the result of a previous instruction.

Level sensitive scan design (LSSD) techniques [14] are used in F-RISC/I to test each submodule at-speed. The comparison between

Fig. 6. (a) Interconnect load sensitivity of DCFL and SBFL NOR-3 gates. (b) Fanout sensitivity of DCFL and SBFL NOR-3 gates.

Fig. 7. FRISC/I datapath.

simulated results and measurements is shown in Table III. Table IV shows the delay distribution on the most critical ALU path. The 32 b ALU is implemented using a two-level carry look-ahead adder with 4 b blocks at level 1. Despite the high drive capability of SBFL the interconnect delays account for 42% of the critical path delay.

The miniaturization of FET dimensions has been and continues to be the main driving force to improve circuit speed and packing density. Hence, it is desirable to predict the system performance growth with a scaled process. Based on the same layout one can evaluate the next generation F-RISC/I performance by simulating the critical paths. Using the experimental process as a benchmark, Rockwell's baseline 0.7 and 0.5 μm HMESFET process improves the K' value by a factor of 1.36 and 1.85, respectively, while the interconnect capacitance per unit length remains the same and the wire lengths scale according to published design rules [2].

TABLE III
COMPARISONS OF CRITICAL PATH DELAYS

Critical Path	Simulated Delay	Measured Delay
PC increments in I1 stage	4.21 ns	5.0 ns
Register File Read in DE stage	3.12 ns	3.36 ns
ALU Execution in EX stage	5.17 ns	6.0 ns
LSSD latch Delay	1.42 ns	1.48 ns

The critical path simulations were performed with scaled interconnect capacitance to predict an upper bound for the performance of F-RISC/I implementations. The interconnect capacitances of the automated standard cell implementation have a scale factor of one.

TABLE IV
DISTRIBUTION OF GATE AND INTERCONNECT DELAYS ON CRITICAL ALU PATH

<i>Gate (Power Level)</i>	<i>Fanout</i>	<i>Intrinsic Delay</i>	<i>Interconnect Delay</i>	<i>Total Delay</i>
2-Way MUX(L)	F=1	0.170ns	0.018ns	0.188ns
LSSD	F=2	0.878ns	0.538ns	1.416ns
4-way MUX(H)	F=2	0.228ns	0.174ns	0.402ns
3-Way MUX(H)	F=2	0.187ns	0.072ns	0.259ns
2-Way MUX(L)	F=1	0.170ns	0.046ns	0.216ns
3-Input NOR(H)	F=5	0.102ns	0.128ns	0.230ns
4-Input NOR(H)	F=1	0.104ns	0.053ns	0.157ns
4-Input NOR(H)	F=5	0.126ns	0.198ns	0.324ns
3-Input NOR(M)	F=3	0.102ns	0.093ns	0.195ns
4-Input NOR(L)	F=1	0.146ns	0.120ns	0.266ns
4-Input OR(L)	F=1	0.138ns	0.102ns	0.240ns
5-Input NOR(M)	F=4	0.173ns	0.361ns	0.534ns
4-Input NOR(L)	F=1	0.146ns	0.042ns	0.188ns
5-Input NOR(L)	F=1	0.135ns	0.047ns	0.182ns
2-Input XOR(M)	F=2	0.196ns	0.176ns	0.372ns
Sum		3.001ns	2.168ns	5.169ns
Percent of Total		58%	42%

Scaled Interconnect Capacitances

Fig. 8. System Performance as a function of scaled interconnect capacitances.

Custom circuit design and optimized layout can reduce interconnect capacitances, and yield a scale factor below one. Fig. 8 shows the system performance as a function of scaled interconnect capacitances. The predicted maximum operating frequency for Rockwell's baseline 0.7 and 0.5 μm HMFET processes are 260–485 and 350–660 MHz, respectively. It is notable that interconnect capacitance induced delays are very significant in determining system performance. For example, HMFET gate length scaling from 0.7 to 0.5 μm yields a 35% performance improvement for the automated standard cell implementation. A 50% reduction of interconnect capacitances in the critical paths achieved through custom layout could improve performance by 25–30%. The interconnect delay accounts for 42% of the critical path delay.

Fig. 9. Probing the 212 pin F-RISC/I test chip fabricated with an experimental 0.7 μm HMFET process from Rockwell.

IV. CONCLUSIONS

We were able to verify a novel SBFL cell library including device/circuit models and have measured critical path delays of a prototype 32 b GaAs processor implemented with an experimental HMFET process from Rockwell. The 212 pin test chip shown in Fig. 9 contains 92,340 transistors on a $7 \times 7 \text{ mm}^2$ die and dissipates 6.13 W at 180 MHz. The measured delays of critical paths could be matched within 16% by simulations with HMFET SPICE models and interconnect capacitances from a three-dimensional (3-D) capacitance extraction tool [15].

Reducing interconnect capacitances would be almost as effective for improving system performance as reducing intrinsic gate delays through device scaling. A F-RISC implementation using Rockwell's baseline 0.5 μm HMFET and additional metal layers would operate between 350–660 MHz, depending on the compactness of

the layout. In order to be competitive with state-of-the-art CMOS processors an HMESFET processor would have to be implemented with at least a 0.5 μm process using full custom layout of all critical circuits and/or yields would have to improve by a factor of 4–6 to allow at least the implementation of a dual-issue superscalar RISC.

ACKNOWLEDGMENT

The authors would like to acknowledge the assistance of Cadence, Inc., of the New Jersey office proved invaluable. Special thanks are due to Dr. C. Anderson, P. Vernes, A. Cappon, and J. Toole, whose appreciation of this work made its completion possible. Testing of F-RISC/I was made possible through the use of equipment at IBM. Finally, the authors acknowledge the contributions of R. Sherburne whose work on Berkeley RISC II provided an inspiration for this project while he was a teacher at Rensselaer Polytechnic Institute, Troy, NY, in 1985.

REFERENCES

- [1] K. R. Elliott, "Heterostructure MESFET devices for VLSI applications," in *IEEE GaAs IC Symp.*, 1991, pp. 67–70.
- [2] T. Tsen *et al.*, "0.5 μm AlGaAs/GaAs MESFET technology for digital VLSI products," in *IEEE GaAs IC Symp.*, 1993, pp. 193–196.
- [3] C. K. Tien *et al.*, "F-RISC/I: Fast reduced instruction set computer with GaAs (H)MESFET implementation," in *Proc. IEEE ICCD*, 1991, pp. 293–296.
- [4] —, "F-RISC/I: System design analysis, implementation, and performance evaluation of a 32-bit microprocessor based on GaAs HMESFET technology," Ph.D. dissertation, Rensselaer Polytech. Inst., Troy, NY, 1994.
- [5] V. M. Milutinovic *et al.*, "Pipelined design tradeoffs in a 32-bit gallium arsenide microprocessor," *IEEE Trans. Comput.*, vol. 40, pp. 1214–1224, Nov. 1991.
- [6] K. Yamagushi *et al.*, "A 1.5 ns access time, 78- μm^2 memory-cell size, 64-kb ECL-CMOS SRAM," *IEEE J. Solid-State Circuits*, vol. 27, pp. 167–174, Feb. 1992.
- [7] V. Milutinovic *et al.*, "Architecture/compiler synergism in GaAs computer systems," *IEEE Trans. Commun.*, vol. COM-35, pp. 72–93, May 1987.
- [8] M. Namjoo and A. Agrawal, "Implementing SPARC: A high performance 32-bit RISC microprocessor," *SUN Technol.*, pp. 42–48, Winter 1988.
- [9] R. R. Tummala and E. J. Rymaszewski, *Microelectronics Packaging Handbook*. Van Nostrand Reinhold, 1989.
- [10] M. Kohara *et al.*, "High thermal conduction package technology for flip chip devices," *IEEE Trans. Comp., Hybrids, Manufact. Technol.*, vol. 6, no. 3, pp. 267–271, Sept. 1983.
- [11] J. D. Gee *et al.*, "Cache performance of the SPEC92 benchmark suite," *IEEE Micro*, vol. 13, pp. 17–27, Aug. 1993.
- [12] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*. San Mateo, CA: Morgan Kaufmann, 1990.
- [13] R. Brown *et al.*, "Gallium-arsenide process evaluation based on a RISC microprocessor example," *IEEE J. Solid-State Circuits*, vol. 28, pp. 1030–1037, Oct. 1993.
- [14] E. Eichelberger and T. Williams, "A logic design structure for LSI testability," in *Proc. IEEE Design Automat. Conf.*, 1977, pp. 462–468.
- [15] Y. L. Le Coz *et al.*, "Application of a floating-random-walk algorithm for extracting capacitances in a realistic HBT fast-RISC RAM cell," in *Proc. 11th Int. IEEE VLSI Multilevel Interconnection Conf.*, June 1994, pp. 542–544.

Correction to "Control-Flow Versus Data-Flow-Based Scheduling: Combining Both Approaches in an Adaptive Scheduling System"

Reinaldo A. Bergamaschi, Salil Raje,
Indira Nair, and Louise Trevillyan

In the above paper,¹ the first three words of the title were missing from the table of contents on the front cover.

Manuscript received March 6, 1997.

The authors are with IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA.

Publisher Item Identifier S 1063-8210(97)05143-3.

¹R. A. Bergamaschi, S. Raje, I. Nair, and L. Trevillyan, *IEEE Trans. VLSI Syst.*, vol. 5, pp. 82–100, Mar. 1997.