Reconfigurable FPGA's in the 1-20 GHz Band with HBT BiCMOS

John F. McDonald and Bryan S. Goda
Rensselaer Polytechnic Institute
Troy, New York 12181

Abstract -- This paper describes the operation of a field programmable gate array (FPGA), the basics of current mode logic, and examines the idea of creating a SiGe heterojunction bipolar (HBT) version of the Xilinx 6200 FPGA. A new proposed device would be bitwise compatible with the 6200, but would operate in the 1-20 GHz range due to the HBT technology being used for the logic and routing and CMOS for storing the configuration bits. This is possible due to the IBM cointegration process of a HBT with a BiCMOS process. Information in this paper is based on an HBT having a f_T of 50 GHz, but later in 1999 IBM will be unveiling a process that will double the speed.

By replacing and redesigning key parts of the 6200 FPGA, a 100-200X operating speedup is possible. The core logic cell in the 6200 consists of two input multiplexers and flip-flops, which can easily be converted to current mode logic (CML). Routing in a conventional FPGA is done via pass transistors, which can act like a low pass filter for a high-speed signal. A SiGe HBT CML multiplexer can be used for routing, which can pass signals with a 12-14 picoseconds delay. Through the use of a side decoder, memory planes of configuration could be used to store current and future configurations. Interchange could occur between memory planes if the old flip-flop values are stored, new flip-flop values are restored, and then the new configuration plane is activated. Countless applications such as DSP, Ethernet routing, missile control, and artificial intelligence could utilize a SiGe HBT FPGA.

Introduction

A field programmable gate array (FPGA) is a chip that you can configure yourself. It consists of an array of logic blocks surrounded by a programmable interconnect structure. A designer performs design entry and simulation, then uses special software to create a configuration file. The FPGA is programmed according to this configuration file and then the design can be tested, validated, and optimized. Due to this flexibility, a FPGA can be used to create configurations from a common set of digital resources that can be optimized for the instantaneous needs of the system. Unfortunately, only limited use of FPGAs is currently possible due to relatively slow operating speeds (100-200 MHz). The delay in routing is significantly greater that a simple wire because the programmable interconnect contain significant resistance and capacitance. Often this connection delay exceeds the delay of a logic block; thus the flexibility of a FPGA hinders its widespread use due to slow operating speed and poor bandwidth.

Current mode logic (CML) is the fastest logic circuit family. High speed in CML is achieved by operating all transistors out of saturation and keeping the logic signal swings relatively small (250 mV). BJT differential pair logic uses a pair of BJT transistors as a current switch. Using this type of logic produces low noise when switching, a desirable feature when operating at high speeds. Other advantages of differential pair logic is that the complement signal is always available and often-desired logic can be implemented in sometimes half the number of gates compared to contemporary CMOS logic.

Very recently there have been breakthroughs in heterojunction bipolar (HBT) technology in the SiGe materials area. A mature foundry capability has been developed at IBM Burlington that can produce both 0.5 micron CMOS and 0.5 micron (50GHz f_T) HBT devices in one process. This allows the creation of circuits in the 10-20 GHz operating range, with a CML 13ps gate delay and a CMOS 60ps gate delay. By taking advantage of the flexibility of a FPGA, the speed of differential logic, and the IBM HBT BiCMOS process, it is now possible create a high speed FPGA that will be 100 times faster.

The Xilinx 6200 chip was chosen for emulation since the information about the configuration bits are in the public domain. Available CAD tools and VHDL can be utilized to create bit streams to test circuits. The core logic cell in the 6200 chips consists of two input multiplexers and flips flops, which can easily be converted into CML logic. FPGAs use pass transistors for routing, which are high resistive and tend to make the signal path act like low pass filters (bad for high speed signals that we are interested in). If a CML multiplexer is used instead of a pass transistor for routing, speed and flexibility can be maintained. With the co-integrated SiGe/CMOS process, it makes sense to make the logic and routing in SiGe (speed) and the memory in CMOS (low power). The current design of the 6200 chips requires any new configuration to be completely shifted in, even if only a few bits were altered. If a directly addressable memory is

used, then parts of the FPGA can be reconfigured in a few nanoseconds. Through the use of a side decoder, planes of configuration memories could be used to store current and future configurations. Interchange could occur if the old flip-flop values are stored, new flip-flop values are restored, and then the new configuration plane is activated.

Technology Basis and Current Mode Logic

The IBM SiGe HBT BiCMOS technology that makes this all possible is relatively new but has progressed rapidly out of research into mass production over a period of only about 5 years. The SiGe HBT is an advanced form of a bipolar transistor that brings some of the speed of III-V technology into silicon logic circuits. It now being commercially offered as a mature, low cost, low power, high speed, and high yield process. There are still many interesting research questions to answer because this capability has appeared so quickly. One issue is to determine is just how high the yield has become. A FPGA is a good candidate to test yield because once the basic building blocks of an FPGA are created, it is relatively easy to scale up the design. Running such a design experiment might help quantify this issue better to determine just how large these FPGAs might be made and still have decent yield.

In research labs, GaAs/AlGaAs HBT devices with 500 GHz performance have already been demonstrated, indicating that further performance increases are feasible with HBTs without expensive X-ray lithography. In addition, bipolar devices have a much higher gain than CMOS devices permitting very low logic swings, which is essential for GHz operation at low power. Recent HBT device and material improvements have improved yields to the point where the implementation of subnanosecond computers starts to become feasible. A subnanosecond computational engine would be important in a heterogeneous network for tasks that change their computational requirements dynamically. As is the case in many Digital Signal Processing (DSP), Scientific, or GHz networking applications where low latency requirements prohibit using only CMOS circuits. However, while GaAs/AlGaAs HBTs are fast, they don't permit co-integration with existing CMOS memory technology. FPGAs and other reconfigurable circuits will require configuration memories to store personalizations of the fixed resources of these circuits. CMOS, even if it is not scaled indefinitely into the future, makes a very low power storage medium for this memory. So ultimately what one desires is a unified process in which the ultimate version of CMOS can be used for memory, and the fastest device possible for logic is co-integrated with the CMOS. Currently this device is the SiGe HBT. Interfacing fast HBT logic with slower CMOS memory over extremely wide data paths is an excellent combination.

In attempting to visualize why bipolar transistors can play a role in fast computation, observe that bipolar devices achieve their speed through the use of a very thin base region. This is because the upper limit of the ability of the transistor to modulate or control collector current depends on how long it takes for a charge carrier to travel through the base. That path is vertically oriented into the body of the substrate, so that epitaxy, and not lithography, determines the base thickness. Comparing the technical challenge and expense of making a thin base against the challenge of sub 0.1 micron lithography in CMOS, epitaxy or the growth of these thin layers seems less formidable. Heterojunction bipolar transistors (HBTs) involve making the emitter and base out of different semiconductor alloys. The band gap structure of the base and emitter for these alloys makes it possible to dope the base very heavily (even 10^{19} dopant atoms per cubic centimeter is possible in this technology) without aggravating the base capacitance of the device. More importantly as the volume of this base becomes small, the number of dopant atoms is still large, permitting better control over large numbers of devices. Small lithographic features are also helpful for the HBT. Small base area means less base capacitance to overcome, and hence less current to supply the charge needed by these device parasitic capacitances, hence less power. The fundamental speed of the device is set by its base transit time and ultimately by the vertical base dimension. While capacitance can be thought of as a delay inducing element, one can always reduce this delay by lowering the voltage swing or increasing the amount of current to deliver the charge. Smaller lithographic features tend to reduce the power required to operate at the transit time limits of either CMOS or bipolar devices. In CMOS, both the capacitive power dissipation of the device and the transit time are determined by lithographic feature sizes, whereas for the bipolar device these two effects are decoupled into horizontal and vertical scalings. Lithography impacts primarily parasitic capacitance and power levels in bipolar. Make the bipolar device sufficiently small to operate at satisfactory power levels, and you need not make it smaller just to speed the device up. That job is handled by making the base transit time smaller either by thinning the base or improving saturation drift velocity of the carriers.

The speed of CMOS is degraded unless the power supply voltage is at least 5 times larger than the threshold, which implies that FET logic operates with a full power rail voltage swing. HBTs do not suffer from this problem, and it becomes possible to set thresholds extremely accurately for all devices provided temperature is uniform across the circuit. In addition, the threshold is sharp enough that a base-emitter voltage swing around the threshold (0.7 volts for SiGe HBTs) by only 250 mV at room temperature is more than enough to assure adequate noise margin for logic operations. This swing of only 250 mV compared with rail to rail logic swings of CMOS is

what makes it possible to have a much lower dynamic power dissipation for bipolar than CMOS. Dynamic power dissipation is the power lost associated with charging and discharging all the circuit capacitances. Lower voltage swings demand less dynamic power given the same circuit capacitance. In fact the reduction is proportional to the square of the ratio of the voltage swings since $P_{Dynamic}=1/2\ C_{Total}\ (\text{Change in V})^2 f$. Hence for a reduction of the AC voltage swing by a factor of 10 from 2 volts down to .2 volts we see a reduction in dynamic power by a factor of 100. At the same time the bipolar circuit does not slow down. To enjoy the benefit of low dynamic power dissipation one must also reduce the DC component of power dissipation in bipolar circuits. This component does not occur in CMOS until extremely small gate lengths arise. In order to take full advantage of this low voltage swing a special class of circuits is required. Small voltage swings require minimizing or eliminating all simultaneous switching noise, which characterizes single ended logic switching in CMOS.

One class of circuits that accomplishes this noise reduction is a current steering network. In this type of circuit current flows continuously in a circuit, and is merely steered through different paths. In this way the inductance in the power feed to the circuit sees only a constant current and does not generate any ground bounce or power rail noise. To maximize noise margins the inputs to gates can be made differential. This is termed full differential current mode logic. The tail resistor at the bottom of the current steering tree is used to fix the current in this circuit, but it can be replaced by a reference driven transistor source to obtain more temperature immunity. Voltage droop is less of a problem in CML because only the difference in the input signal is important. Hence, this is another form of robustness for this class of circuits, and in fact any disturbance that shows up as common mode noise will be suppressed by the differential circuit inputs. Wire coupling is another one of the serious problems for design at higher frequencies. Often the coupling is difficult to visualize, and frequently the coupling can end up converting a piece of GHz logic into a feedback oscillator generating unwanted oscillatory behavior. This is almost completely suppressed by the differential features of the circuit.

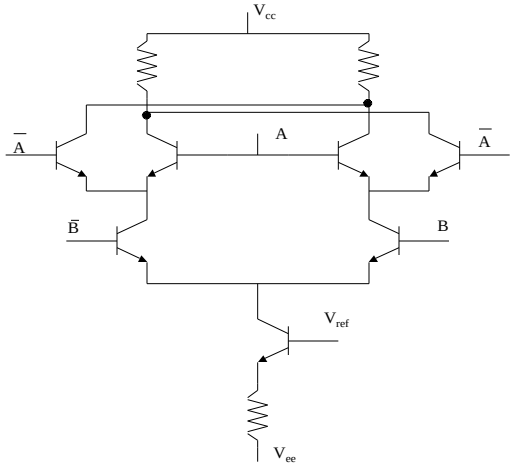

Figure 1. Full Differential Current Mode Logic (CML) XOR

Another advantage of full differential CML is that inverters are replaced by twisting or interchanging the connections for complemented and uncomplemented versions of a signal. In addition, it is possible to create complex functions in what amounts to a one gate macro, with propagation delays comparable to that of a simple gate. This results in faster operation still because one does not have to traverse several levels of logic to obtain such complex functions. Together with the high speed of the devices and the low voltage swing on wire capacitances, CML excels at speed, and does so with attractive transistor counts. Although a simple NAND gate can have 6-7 transistors in it, the above XOR cell has only 7 HBTs and 3 resistors. CML favors thinking in terms of such complex macros to keep the transistor counts low.

CML is not free of problems. One problem that can occur is that the true and complementary inputs need to switch skew free. Otherwise the differential pair will be blocked on either of its two paths which would result in an interruption of current, the very thing we sought to avoid. Preventing this is accomplished by closely tracking the wire resistance and capacitance in every complementary pair of wires. Also these pairs need to be routed together to improve the degree to which common mode noises are in fact common. Another problem of CML is the doubling of the wires in the design. On average, CML uses about half the number of transistors of CMOS designed at the standard cell level. But there can be situations where the wiring is noticeably worse with full differential CML. Related to this is the fact that the voltage swing to turn on or off the opposite members of a given differential pair depends on the level on which the pair sits in the tree. This is to prevent saturation of any of these transistors, which slows down the switching of the transistor due to charge storage in this state. To avoid saturation, each transistor needs to have a bias or centering offset voltage that is different for each level. This requires level translator circuits to move from one level to another or to several others. The drive capability of CML is not large and emitter follower transistor circuits may be required for both the complemented and uncomplemented signal outputs. This can increase power and transistor count. But it also provides a place to incorporate the level translation circuitry, essentially a chain of voltage dropping diodes formed by shorting the collector and base together, which guarantees that the drop is one transistor just at the margin of saturation.

Xilinx 6200 Emulation

Logic can be flexibly implemented with ROMs or SRAMs to store truth tables directly, or MUXs to implement the address selection mechanism of the SRAM. The latter technique is the one used in Xilinx's XC6200 series of FPGAs. CAD tools already exist and extensive information exists in the public domain about the configuration bits. This means that linkages exist to the high level languages such as VHDL, and special coarse grain macros exist for the 6200. The intention is to make the proposed SiGe HBT/CMOS part completely compatible with the CAD tool so that personalities can be created using Xilinx tools but future designs would provide much wider bandwidth, and hence, higher computational throughput. The only tool that needs to be redone is the network extraction tool which back annotates the design to include signal delays between logic elements, and effective gate delays and clock rates for simulation. The core logic cell used in the XC6200 is based on two input multiplexers and flip-flops. Flip-flops and multiplexers are easy to implement in CML. A four way two address input multiplexer in CML as shown in Figure 2. A two way multiplexer would be only two transistor levels high instead of three.

A key assumption in Figure 2 is the notion that the CMOS reconfiguration logic is capable of controlling the lower inputs into the multiplexer current steering tree. Level converters can be used but for static signals on the lower branches one need not worry about saturation. The level converters become necessary only when attempting to scale into the future, when CMOS itself would require lower voltages to turn on a transistor, CMOS would have to deliver a current path to the base at a voltage

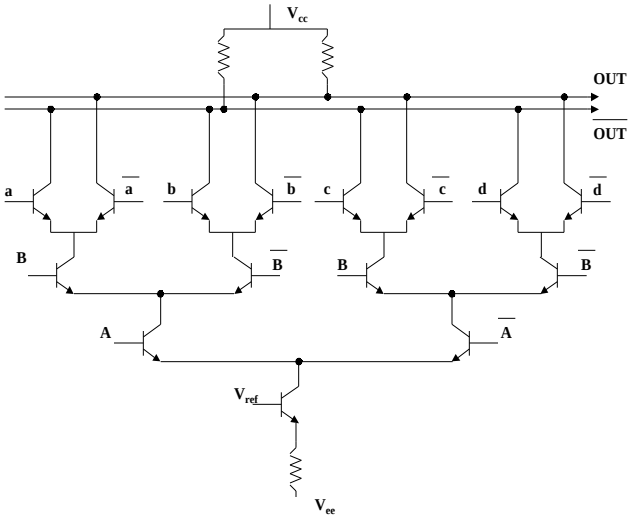

Figure 2. Four Way Multiplexer in CML.

exceeding the threshold for the bipolar device by some small voltage. For SiGe HBTs this means at least 0.7 Volts plus 100-200 mV. Given a threshold voltage for such a generation of CMOS, that would require power supplies of 0.7 V plus about 0.4 V, or 1.1 Volts. In these latter generations of CMOS, the level converters would be needed to climb to the voltage offset of second (or even third) level inputs. The uppermost level of the CML tree is where the fastest switching capability exists, and fortunately this is also where the FPGA visible signals would enter. This means that the nearly static CMOS signals would occupy only the slow lower levels of the current tree.

The most important point is that HBT CML can be used to implement all of the logic components visible to the designer in a part as flexible as the XC6200, as well as all the hidden routing resources**.** CML multiplexers pass signals with delays of about 12-14 picoseconds in the 50 GHz SiGe HBT process. Hence, high GHz signal bandwidths are sustainable in such a class of circuits, provided wire lengths are limited, or repeaters are used to reshape signal rise and fall times. Routing is accomplished with multiplexers in the XC6200 rather than pass transistors**,** which are commonly used in CMOS. Minimal sized pass transistors typically used in most other FPGAs are highly resistive and tend to make the wiring behave like low pass filters with extremely low cutoff frequencies. For flow-through or heavily pipelined architectures where circuit delays are less important than signal bandwidths one can expect that much of this circuit performance can be captured in larger systems of logic circuits. The key to capturing this performance is the technique used to route the flexible signal paths that give personalization to the system. These surround the core logic block, or Component Logic Block (CLB). For the XC6200 all of the programmable signal routing is accomplished with multiplexer circuits, making a direct mapping to CML extremely tractable.

To avoid having the resistance of the wires limit the bandwidth of the lines, only wires on the highest levels of metallization for the integrated circuit process are permissible for use for these signals. In the IBM SiGe/CMOS process, a kind of reverse scaling philosophy is retained on these upper levels of interconnections, which permits these lines to have large enough cross sections to minimize their resistance, and large pitch to minimize capacitance. The two uppermost levels of the IBM interconnection stack will also employ low dielectric constant insulators that are also thick enough to further reduce capacitance.

The existing CMOS-only XC6200 architecture lends itself to rapid reconfiguration if only a small part of the architecture needs to be altered. This is because all the configuration bits are directly addressable through an address port, and many of the bits can be parallel loaded in

one access at a given address. Earlier Xilinx offerings required that any new configuration, however slightly modified, had to be completely shifted in along with all the other reconfiguration bits. This meant reconfiguration times were in the range of 10 ms. Use of directly addressable memory for the reconfiguration bits in the 6200 means that selected areas in the memory can be reconfigured in a much shorter time, possibly only a few nanoseconds. However, truly massive reconfigurations would still require rewriting all the reconfiguration bits. Although not provided by the existing XC6200, it is relatively easy to imagine a simple way to greatly enhance the speed of reconfiguration. This would be by augmenting the memory of the FPGA into several memory planes. By use of a side decoder the signals could be used to select between two reconfiguration bit memories stored on the same chip. Complete interchange between these two configurations can then be accomplished in only a few nanoseconds by simply selecting the appropriate plane, however, an additional nonstandard pair of planes would be needed to save and restore all the flip-flop values so a prior state would be restorable.

With only minor alterations, one of these configuration memories may be selected for personalization of the FPGA while the other is being reloaded with another configuration. The contexts could also be switched back and forth. For computer operations that last for orders of milliseconds, the loading time can be completely hidden. Alternatively configuration plane swapping can be back and forth between two personalities in orders of nanoseconds. Extensions to additional reconfiguration planes are possible, but likely to decrease the density of the CML logic, extending lengths of wiring which the fast CML would have to drive. The impact of having more than two such planes will be explored. Each CML control input coming out of the configuration control store must pass through a plane select MUX, and the relevant memory cells need to be nearby the place where they are used.

Conclusions and Future Work

One of the questions to be answered is how much of the 20 GHz flip-flop toggling frequency in a 50 GHz baseline process and the 12-14 ps gate/macro delay is finally harvested by typical design. This is unknown because interconnections (especially flexible repro-grammable interconnections) can be the source of a great deal of extra delay and bandwidth loss in FPGAs. For example, working with a part with 200 MHz flip-flops one can perhaps see a system implementing 30-60 MHz system clock. System level performance is not just the flip-flop rate and produces a loss of between a factor of 3 and 6. This would put the system clocks for the new HBT based FPGA part at 3-6 GHz.

One of the questions is how to package such a fast part. The high speed and different power requirements testify that it is not meaningful to imagine a simple plug-in replacement for the part that just happens to run at GHz rates. Technically, CML is a full differential technology and needs this kind of interconnection scheme to maintain low system switching noise for its low voltage swings. Even if the exterior pad drivers were made to look like single ended signals, the high frequencies of these signals would dictate a switchover to ECL signals from any current CMOS design. However PGA packages don't support full differential wiring on all signals due to pinout limitations, especially if full compatibility with the IOB and padout of the Xilinx CAD tool are to be maintained. Full differential CML is critical if the part is to be integrated into analog systems. Single ended switching noise is extremely disruptive to analog parts of systems, and at 6 GHz the disruption would be excessive. Full differential CML or current steering logic is much quieter, making integration with analog parts of the system more reliable.

In conclusion, it is now technically possible to combine the best features of an FPGA with SiGe and current steering logic. All that remains now is to design, test, build, and publish the circuit.

References

[1] J.F. McDonald, H.J. Greub, T. Yamaguchi, "Cell Library for Current Mode Logic using an Advanced Bipolar Process," *IEEE Trans. on Solid State Cir*, Vol. JSCC-26(#5), May 1991, pp. 749-762.

[2] J.F. McDonald, K. Nah, R. Philhower, J. S. Van Etten, S. Simmons, V. Tsinker, Maj. J. Loy, and H. Greub, "F-RISC/G: AlGaAs/GaAs HBT Standard Cell Library," *Proc. IEEE Int. Conf on Computer Design*, Boston MA, October 1991, pp. 297-300.

[3] S. Singh, P. Chow, D. Lewis, "The Effect of Logic Block Architecture on FPGA Performance," *IEEE Journal of Solid-State Circuits***,** Vol. 27, No. 3, March 1992, pp. 281-287**.**

[4] J. Rose, R. Francis, D. Lewis, P. Chow, "Architecture of Field-Programmable Gate Arrays: The Effect of Logic Block Functionality on Area Efficiency," IEEE Journal of Solid-State Circuits, Vol. 25, No. 5, Oct 1990, pp. 1217-1225.

[5] D.L. Harame, E.F. Crabbe, J.D. Cressler, J.H. Comfort, "A High Performance Epitaxial SiGe-Base ECL BiCMOS Technology," *IEEE IEDM Tech Digest*, 1992, pp. 2.1.1-2.1.4.

[6] "Xilinx Series 6000 User Guide," Xilinx Inc., San Jose CA., 1997.

[7] M. Smith, *Applications-Specific Integrated Circuits*, Addison-Wesley, 1997.

[8] "IBM SiGe Designer's Handbook**,"** IBM Inc., Burlington VT., 1998.