MxSICO: A SILICON COMPILER FOR MIXED ANALOG DIGITAL CIRCUITS

E. Berkcan

General Electric Company, Corporate Research and Development

P.O. Box 8, Schenectady, New York 12301

ABSTRACT

This paper presents a silicon compiler for mixed analog digital circuits and the results and examples of compilation. The tool addresses the compilation of sampled data analog circuits and bit/nibble serial digital circuits. The class of designs targeted by the compiler are high performance medium speed sampled data analog circuits including oversampled A/D converters and bit-nibble serial digital circuits. The compiler is targeted for the compilation of noise shaping oversampled A/D converters including anti-aliasing filtering, modulation, and decimation. The details of the approach and the algorithms within a general framework are presented; the approach includes a mixed analog digital design description language and its compiler, automatic partitioning of analog and digital subcircuits, the digital compilation, analog synthesis, module compilation, and physical assembly.

1 INTRODUCTION

A large class of ASIC's involve a combination of analog and digital circuits. Moreover the analog subsystem increasingly involves intricate and high level functions such as high order modulators, A/D converters, etc. These aspects are further complicated by the presence of digital subsystems on the same ASIC's; the mixed signal circuits present major problems for their design as well as simulation. In addition fast and efficient investigation and evaluation of the algorithms and design are required prior to complete layout. Lastly a stringent set of analog as well as technological requirements must still be met. The above considerations represent an odd parity between the time and pressures on the system design cycles of ASIC's, and the current state of mixed analog-digital design automation and silicon compilation (Cf. [1] and the references therein.).

There has been some recent developments in analog synthesis or compilation[2][3][4][5][6][7][8]. Mixed analog digital CAD, however, was initially based on extending existing tools to admit the analog or digital subsystems as a single block . Automated layout and routing of VLSI chips has been obtained by using analog standard cells and design strategies for the automatic routing of sensitive nets between them[9]; most sensitive interconnect and critical device matches are hidden inside the manual layouts of these cells. Mixed signal standard cells have also been used with switched capacitor circuitry [10]. An alternative approach uses analog layout tools and the digital portion of the circuit is taken as a block created separately [11]. These tools are based on an extension of the classical standard cell approaches and suffer from the same limitation and drawbacks. Channel routing of mixed signals has been obtained by extending digital routers to account for the coupling of signals between nets, the effects of parasitics couplings and wire resistance [12]. Differential circuit architectures and balanced routing of these signals are not addressed in this approach. There has been attempts at synthesizing oversampled A/D converters[13]; this tool however admits only few fixed (closed) design styles, does not incorporate high level simulation, and can not address high order, multi-bit, or fully differential topologies. None of the approaches mentioned heretofore address the ability to describe a mixed analog digital design at high level, to evaluate this design, and to compile the description into layout.

This paper describes an approach to the silicon compilation of mixed analog and digital circuits; the results from a tool that implements these algorithms are presented. This tool includes a mixed analog digital design description language and its compiler, automatic partitioning of analog and digital subcircuits, the digital compilation, analog synthesis, module compilation, and physical assembly. The compiler has been developed to provide the basic capability to combine the compilation of A/D converters with DSP compilation for sensor interfacing and signal processing. The goal is to aid the designer in assembling ASP and DSP sub-systems from compilation of a high level description of the design using parametrized leaf cells. In particular the compiler is being used for the compilation of oversampled noise shaping A/D converters. The results from this application of the compiler are presented in detail. The paper is organized as follows. The details of the framework and the approach are presented in Section 2. Examples and results are given in Section 3, and finally Section 4 presents our conclusions, discussion, and a brief comparison to the literature.

2 MxSICO FRAMEWORK

Within this report, we limit ourselves to single-bit encoding generated by Σ-Δ modulators and noise shaping oversampled (*NSOS*) A/D converters; this aids in the compilation of highly linear converters. In particular integral non-linearity of the in-loop *DAC* can limit the harmonic distortion performance; one-bit *DAC*, with its only two discrete values, defines a linear transformation between the analog and digital portions of the design. Single bit-encoding is also advantageous for the digital signal processing part of the design and its compilation. For analog synthesis, a fully differential architecture is allowed in order to achieve high power supply rejection ratio, improved linearity, increased dynamic range, as well as reduced clock feedthrough and switched charge injection errors. For decimation, the digital part of the compiler is used to provide a low-pass filter function to remove the quantization noise outside the baseband. Thus an accurate, high resolution representation of the input is achieved at the output of the digital processor, Figure 1.

The objective of the mixed analog digital silicon compilation reported herein is to provide a uniform framework in which a limited set of mixed signal designs can be compiled from their high level description. Currently the approach is targeted at compiling the mixed signal front-end, including NSOS converters, together with the rest of the ASIC, Figure 1

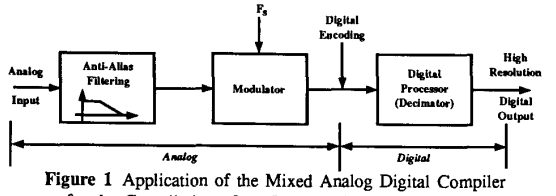

Figure 1 Application of the Mixed Analog Digital Compiler for the Compilation of an Oversampled A/D Converter

A salient feature of the present approach is that it is not based on the extension of digital tools or digital silicon compilers. The high level description is obtained by extending an analog design description that we reported earlier [7]. Similarly the other aspects of the compiler such as floorplanning and routing are developed specifically for analog and mixed signal designs. These points are further detailed later.

To achieve the mixed analog digital synthesis, the general system architecture illustrated in Figure 2 has been implemented. The input is the high level mixed analog digital design description, **MxADD**. This description is automatically partitioned into the analog and digital parts. The resulting analog description is run through a behavioral to structural transformation to obtain the structural representation with operator instances, bias requirements, and their interconnection, Figure 3. A partially designed library of leaf cells is used to provide basic analog and digital operators; module compilation parametrizes the analog set and extends it to more complicated functions. The analog structural information is then passed to the analog physical assembly for floorplanning, placement and the balanced routing of combined differential and single ended analog signals. For the digital compilation, the approach extends existing silicon compiler technology; these extensions are processed to obtain the standard digital behavioral language of an existing digit-serial compiler [14][15]; this is a compiler that allows digit-serial architectures with a digit width that can be any divisor of the word size. This last description is processed through the digital compiler to obtain the digital netlist and layout. The resulting digital and analog layouts are combined by macro level physical assembly to yield the final compiled circuit.

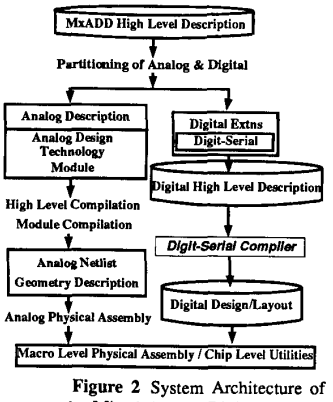

Figure 2 System Architecture of the Mixed Analog Digital Compiler

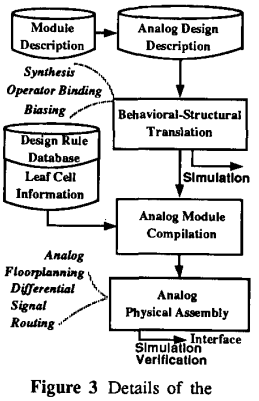

Figure 3 Details of the Analog Subsystem Architecture

The high level description **MxADD** is used to specify the circuit at an algorithmic level and provides enough flexibility to address a wide range of high performance, medium speed analog sample data systems and digit-serial digital circuits. Binding of language operators to cells can be given explicitly; in addition macromodels can be bound to these language operators to be used in the high level simulation. Currently the operators supported in the **MxADD** include mathematical (addition, subtraction, multiplication, integration) and relational operators, general operators such as parametrized subcircuits, analog multiplexer, programmable gain amplifiers, comparator, and 1 bit *ADC* or *DAC*'s. The digital operators include the standard arithmetic and logical operators [16]. At the highest level of hierarchy, there exists a mixed signal processor described in terms of analog and digital processors. Each one of these latter processors can be either *symbolic* or *composite*. A symbolic analog description is given in terms of the aforementioned operators and internal variables; differential and single ended analog as well as digital signals are allowed. The composite description can contain lower level constructs such as explicit instances of leaf cells, other composite cells, and the netlist. Leaf cells consist of an arbitrary combination of predesigned laid-out portion and parametrized module calls. These modules are in turn compiled at run time based on the parametrization, a description of the technology, and the modules [17]. A built-in high level behavioral simulation is used to verify the functionality of the algorithms in the analog design description. This description is later translated into a structural description and a special signal flow graph that has the same circuit schematic as the sampled data circuit is constructed. The maximum fan-in and fan-out requirements are verified and the graph is modified to incorporate biasing. The analog circuit is biased automatically since the biasing is strongly related to operator binding, the power requirements of the leaf cells, and the layout. A generalization of the current shipping scheme proposed in [18] is used for this biasing. Finally the completed structural information is used for the physical assembly.

Analog physical assembly is used to incorporate explicitly the analog layout requirements for floorplanning, placement and routing. A heuristic floorplanning algorithm based on standard cell approach is used to yield the placement [8]. The sensitive channels are used for signal and analog power routing, and the insensitive channels for digital routing [7]. New algorithms have been developed for the balanced channel routing of differential and single ended analog signals [25]. This router can be used for insensitive channels when its analog requirements are turned off. The primary goal of the router is to incorporate analog requirements related to the fully differential signals, (*FDS*), Figure 4; these requirements include the matching of parasitic capacitances and resistances of the $\pm$ components of *FDS*'s. In addition the balancing of the crossovers of these two components to match the cross talk has to be considered. To this end we use a new gridless channel routing algorithm based on two graphs, Figure 4. Similar to [19][12] we use a *vertical* constraint graph to represent the overlapping pin ranges (directed edges) and overlapping net lengths (undirected edges). In addition a horizontal constraint graph is used to represent the relative locations of the pins at the top and bottom of the channel, the relative locations of vertical segments, and the *extensions* to net segments used to match the parasitics of *FDS* pairs. These graphs also provide the data structures to implement variable widths and net couplings in both the vertical and horizontal directions. The horizontal graph incorporates additional vertices to represent cell boundaries and net extensions. The input data is used to classify and prioritize the nets to be balanced. In constructing the vertical graph, the nets are clustered within the channel based on their type and overlaps; two-net *FDS*'s with top pins only are clustered above multiple-net *FDS*'s with top connections only; the subgraphs of these latter are clustered above that of the *FDS*'s with top and bottom pins, etc. The power (AVDD and AVSS) nets are also distinguished

and clustered to the very top and(or) bottom based on whether the sensitive channel is(is not) two sided. To balance the parasitics, the *FDS* nets are extended or assigned a wider width; the extensions are represented using *pseudo-pins*. The two graphs are later used for checking the signal net crossings; the horizontal graph is used for determining the crossings while the vertical graph provides the net positioning for this part of the procedure. Based on this information, an iterative algorithm is used to extend either pair of *FDS*'s to balance the crossovers. An effort is made to balance a *FDS* pair within the range of the cell boundaries that they connect to.

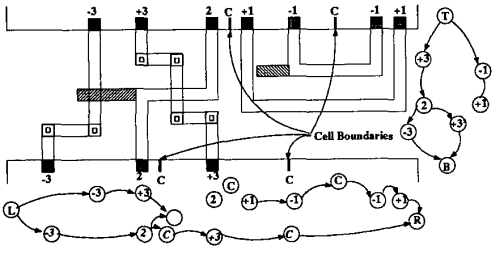

Figure 4 Balanced Differential Signal Routing and the two Constraint Graphs showing a Subset of Their Edges

The constraints represented by the (now) directed graphs are used to solve an optimization problem that determines the variable widths and the location of the nets. This optimization is also used to incorporate general coupling between nets as well as the effect of wiring resistance; additional constraints such as forcing a minimum separation between two nets or coupled constraints between the two directions can also be incorporated. The cost function can be a general quadratic function of the degrees of freedom in the problem based analog requirements rather than only minimizing the width of the routing channel, as it is typically done for digital layouts.

3 RESULTS

This section presents some of the examples and results obtained from the compilation of *NSOS* A/D converters. The compiler is written in the C-language and runs under UNIX™ on a Sun™ workstation. The leaf cell designs and the layout have been obtained from existing designs in BiCMOS [20] that have been modified to satisfy the compiler requirements.

The compilation of a 2nd order Σ-Δ A/D converter is illustrated within the resulting top level layout plan, Figure 5. This converter is designed for a clock rate of 5.12 Mhz, a dynamic range of at least 70 dB, and a conversion rate of 40 kHz. This dynamic range corresponds to better than 11 bits of resolution. The details of the 2nd order modulator Σ-Δ_MOD is shown in Figure 6. A simplified block diagram of this modulator is given in Figure 7; in this Figure, all signals are represented as single ended voltages for simplicity; the actual design is fully differential in order to achieve high power supply rejection ratio, improved linearity, increased dynamic range, as well as reduced clock feedthrough and switched charge injection errors. The in-band quantization noise is shaped by the two integrators INT_1 and INT_2. The noise injected at the modulator input and non-linearity in INT_1 are treated carefully by the choice, design, and parametrization of INT_1. The feedback loop of the modulator affords the advantages of attenuating the errors of the input of INT_2 and the quantizer LATCMP. The design of the comparator LATCMP is less critical since large offset and hysteresis can be tolerated without significant degradation of the converter performance. A simple regenerative latch without preamplification fulfills the comparator requirements; the two integrators INT_1 and

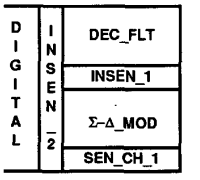

Figure 5 Top Level Layout of a 2nd Order Σ-Δ A/D Converter

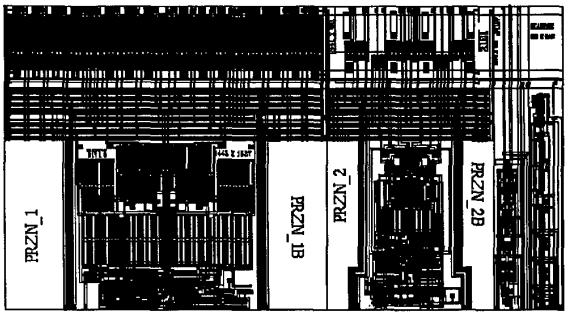

Figure 6 2nd Order Oversampled A/D Converter

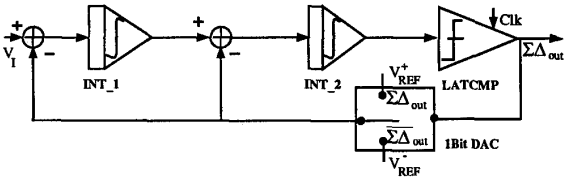

Figure 7 Simplified Block Diagram of a 2nd Order Oversampled A/D Converter

INT_2 provide the preamplification of the signal, and due to the feedback, the comparator offset is stored in INT_2. The last cell BIAS is the bias end cell that is automatically inserted to bias the circuit. The digital signals and the clocks enter the cells from the insensitive side, at the top in Figure 6, and the analog signals and power from the opposite sensitive side at the bottom. The analog ground and the substrate are run internally and are connected by abutment. The control, the switches, and the *DAC* are located at the top of the cells. The active devices are located in the center of the cells facing the sensitive side. Finally the passive devices are located on both sides of the active device area. Due to the fully differential architecture, a symmetry is preserved about the center of the leaf cells. The details of the parametrization of INT_1 obtained by module compilation is shown for PRZN_1 in Figure 8.

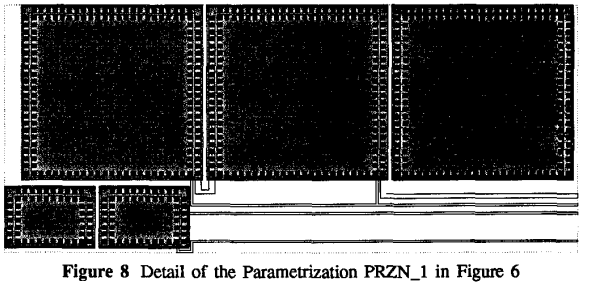

Figure 8 Detail of the Parametrization PRZN_1 in Figure 6

The routing of insensitive and digital signals is illustrated for INSEN_1 in Figure 9. In contrast, the analog balanced routing of the differential signals is shown in Figure 10 for channel SEN_CH_1. The balancing feature of the differential router is detailed further in the blow-up view of SEN_CH_1, Figure 11.

Figure 9 Details of the Routing of the Insensitive Channel

Figure 10 Details of the Routing of the Sensitive Analog Signals

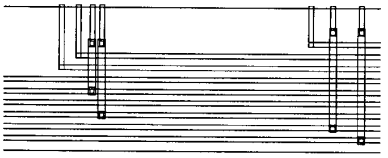

Figure 11 Details of the Balanced Differential Signal Routing

The decimation portion DEC_FLT of the digital processor is used for removing the out-of-band noise, down-sampling to the Nyquist rate, and providing any additional anti-alias rejection. To this end, a triangularly weighted *FIR* filter (or a Bartlett window) proposed in [21] is used. A review of such a digitally compiled filter with a window length of 256 and decimation rate 128 is given in [1]; for further details the reader is referred to this review.

4 CONCLUSION

This paper presented an approach for the silicon compilation of mixed signal designs. In particular the class of designs targeted are sampled data circuits for the analog portion, and digit-serial circuits for the digital counterpart. The results from the compilation of Σ-Δ oversampled A/D converters have also been presented. Our results constitute the first compilation of a high level mixed analog digital design description into layout; in particular automatic partitioning, high level simulation, as well as tailored physical assembly and differential signal routing have been achieved; the method allows design trade-off between digital and analog portions of the circuit.

The approach presented is distinguished from those in the literature in several aspects. The method is not confined to a set of (few) preselected design styles; it is not *closed* by virtue of admitting a high level description of the design. This allows the evaluation of new algorithms and topologies. The binding of macro-models to the language elements and high level simulation significantly facilitate this evaluation; this is extended by the ability to incorporate feedback from lower level parasitics by using the module compilers. One can view interpolative modulation in NSOS A/D converters as providing high pass filtering function to the quantization noise. Increasing the loop order, coupled with changing the topology can provide stronger filtering to further suppress the quantization noise in baseband[22][23]. In some topologies, the design of the modulator is analogous to analog filter design [24]. Moreover the compilation of mixed signal description allows design trade-off between the two subsystems including reduction of the ASP complexity as well as further DSP such as equalization and echo cancellation. These are some of the advantages afforded by MxSICO framework.

Current work includes the comparison, by simulation, of compiled converters to prior manual designs. Also of current interest is to extend the router to include a richer set of constrains and speeding up of the algorithms. The investigation of M-bit *NSOS* converter compilation, (conditional) stability of higher order modulators for robust compilation of converters, and the extension of the present approach to more general analog circuits are very important. [26]

REFERENCES

[1] E. Berkcan and F. Yassa, "Towards Mixed Analog/Digital Design Automation: A Review," Proc. of 1990 International Symposium for Circuits and Systems, 1990.

[2] M.G. DeGrauwe, et al, "IDAC: An Interactive Design Tool for Analog CMOS Circuits," IEEE Journal of Solid State Circuits, vol. SC-22, no. 6, pp. 1106–1116, 1987.

[3] R. Harjani, R. Rutenbar, and L. Carley, "A Prototype Framework for Knowledge-Based Analog Circuit Synthesis," Proc. of the 24th Design Automation Conference, pp. 42–49, 1987.

[4] H.Y. Koh, et al, "Automatic Synthesis of Operational Amplifiers Based on Analytic Circuit Models," Proc. IEEE Int. Conference on Computer-Aided Design, pp. 502–505, 1987.

[5] E. Berkcan, M. d'Abreu, and W. Laughton, "Analog Compilation Based on Successive Decompositions," Proc. of the 25th ACM/IEEE Design Automation Conference, pp. 369–375, 1988.

[6] E. Berkcan and M. d'Abreu, "Physical Assembly for Analog Compilation of High Voltage IC's," Proc. of the Custom Integrated Circuits Conference, pp. 14.3.1–14.3.7, 1988.

[7] E. Berkcan, C. Kim, B. Currin, and M. d'Abreu, "From Analog Design Description to Layout: A New Approach to Analog Silicon Compilation," Proc. of the Custom Integrated Circuits Conference, 1989.

[8] C. Kim, E. Berkcan, B. Currin, and M. d'Abreu, "A new Floorplanning Algorithm for Analog Circuits," Proc. of the Custom Integrated Circuits Conference, 1989.

[9] C.D. Kimble, et al, "Autorouted Analog VLSI," Proc. of the Custom Integrated Circuits Conference, pp. 72–78, 1985.

[10] D.G. Maeding, et al, "Combining Analog and Digital Standard Cells," Proc. of the Custom Integrated Circuits Conference, pp. 491–494, 1985.

[11] T. Pletersek, et al, "Expert System for Automatic Mixed Analog Digital Layout compilation," Proc. of the Custom Integrated Circuits Conference, pp. 165–169, 1987.

[12] R.S. Gyursnik and J-C. Jeen, "A Generalized Approach to Routing Mixed Analog and Digital Nets in a Channel," IEEE Journal of Solid State Circuits, vol. SC-24, no. 2, pp. 436–442, 1989.

[13] O. Nys, et al, "A CAD Tool for Oversampling A/D Converters," Proc. Symp. VLSI Circuits, pp. 113–114, 1987.

[14] R. Hartley and P. Corbett, "A Digit-Serial Silicon Compiler," Proc. of the 25th ACM/IEEE Design Automation Conference, pp. 646–649, 1988.

[15] F. Yassa, J. Jasica, R. Hartley, and S. Noujaim, "A Silicon Compiler for Digital Signal Processing: Methodology, Implementation and Applications," IEEE Proceedings, vol. 75, no. 9, pp. 1272–1282, 1987.

[16] R. Hartley and J. Jasica, "Behavioral to Sructural Translation in a Bit-Serial Compiler," IEEE Transactions on Computer Aided Design, vol. 7, no. 8, pp. 877–886, 1988.

[17] E. Berkcan and B. Currin, "Module Compilation for Analog and Mixed Analog Digital Circuits," Proc. of 1990 International Symposium for Circuits and Systems, 1990.

[18] L. Letham, et al, "A High Performance CMOS 70–Mhz Palette/DAC," IEEE Journal of Solid State Circuits, vol. SC-22, no. 6, pp. 1041–1047, 1987.

[19] H. Chen and E. Kuh, "Glitter: A Gridless Variable-Width Channel Router," IEEE Transactions on Computer Aided Design, vol. CAD-5, no. 4, pp. 459–465, 1988.

[20] D. Ribner and D. Baertsch, "Design of Partial Leaf Cell Library," To appear, 1990.

[21] J. Candy, et al, "Using Triangularly Weighted Interpolation to Get 13-Bit PCM from a Sigma-Delta Modulator," IEEE Transactions on Communications, vol. COM-24, no. 11, pp. 1268–1275, 1976.

[22] Y. Matsuya, et al, "A 16-Bit Oversampling A/D Conversion Technology using Tripe-Integration Noise Shaping," IEEE Journal of Solid State Circuits, vol. SC-22, no. 1, pp. 921–929, 1987.

[23] L. Longo and M. Copeland, "A 13 bit ISDN-band OverSampled ADC Using Two Stage Third Order Noise Shaping," Proc. of the Custom Integrated Circuits Conference, pp. 21.2.1–21.2.4, 1988.

[24] W. L. Lee and C. Sodini, "A Topology for Higher Order Interpolative Coders," Proc. of International Symposium for Circuits and Systems, pp. 459–462, 1987.

[25] E. Berkcan and C.-Y. Wang, "Balanced Analog Router for Single Ended and Differential Signals," technical report, General Electric CRD, 1989.