Wireless SDR Solutions:
The Challenge and Promise of Next Generation Handsets

John Glossner, Ph.D., Erdem Hokenek, Ph.D., and Mayan Moudgill, Ph.D.
Sandbridge Technologies, White Plains, NY
glossner@SandbridgeTech.com

ABSTRACT

In this paper, some aspects of implementing convergence devices in software are described. First, the motivation for Software Defined Radio (SDR) solutions. including market drivers, is discussed. Next, a technical introduction to high-complexity wireless applications is presented. The Sandbridge solution to multimode baseband processing is described, including an SDR processor architecture and results. We then present a solution that provides a 768kbps WCDMA compliant transmit in less than 400MHz on the Sandblaster^{TM} multithreaded processor.

INTRODUCTION

The performance requirements for mobile wireless communication devices have expanded dramatically from their inception as mobile telephones. Consumers are demanding full data & voice integration as well as a variety of computationally intense features and applications such as web browsing, MP3 audio, and MPEG4 video. Moreover, consumers want these wireless subscriber services to be accessible anywhere in the world. .

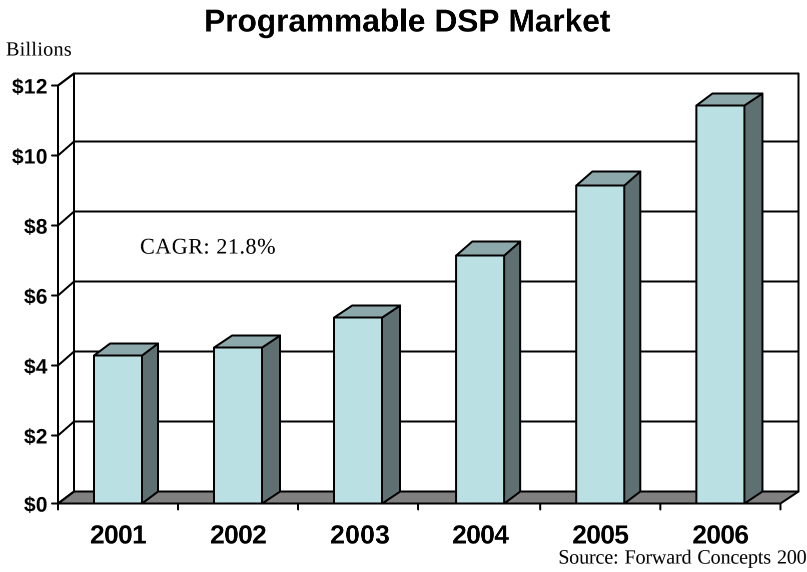
The technologies necessary to realize true broadband wireless handsets and systems present unique design challenges. Wireless handset manufacturers are challenged to deliver products that offer expanded services and operate transparently worldwide. Product designers are challenged to create extremely power efficient yet high-performance, broadband wireless devices.

The design tradeoffs and implementation options inherent in meeting these demands highlight the extremely challenging requirements for next generation baseband processors. Tremendous hardware and software challenges exist to realize convergence devices. Power dissipation constraints are requiring new techniques at every stage of design - architecture, microarchitecture, software, algorithm design, logic design, circuit design, and process design. With performance requirements exploding as bandwidth demand increases, power conscious design becomes more difficult. System-on-a-chip

(SOC) integration and low voltage process technologies will contribute to lower power SOC integrated circuits (ICs) but are insufficient as the only solution for streaming multimedia.

Convergence applications are fundamentally DSP applications. A large number of standards exist or have been proposed for the wireless and wired communication markets. Such a diversity of standards necessitates a programmable platform for their timely implementation. In wireless communications, GSM and IS-54 data rates were limited to less than 15 Kbps. Future third-generation (3G) systems may provide data rates more than 100 times the previous rates. Higher communication rates are accelerating higher DSP processing requirements.

This article describes the first implementation of a software defined radio (SDR) solution that is power efficient and provides a flexible technology path. The derivation of this solution outlines the major challenges in this market, particularly the need to future proof designs against continually evolving standards and air interfaces.



Source: Forward Concepts 2002

Figure 1. Programmable DSP Market

MARKET OVERVIEW

SDR solutions, in particular, are DSP applications due to the large amount of signal processing required for baseband implementation. Programmable DSP represent just fewer than 35% of the total DSP market and about 55% of programmable DSPs are used in wireless communications [1]. Typically, fixed function ASICs have been required to implement wireless baseband processing. The SDR solutions move this function into software programmable platforms. Therefore, the total market for efficient SDR solutions may be anticipated to be large.

Figure 1 shows Programmable DSP market growth. While much of the semiconductor industry is experiencing market growth contraction, DSP's continue to experience growth although at a slower than expected 5% for 2002. Overall, DSP Market growth is expected to grow at a compounded rate of 21.8% driven largely by wireless communications.

A significant market trend for 3G wireless communications is Java execution. Future 3G wireless systems will make significant use of Java. A number of carriers are already providing Java-based services and may require all 3G systems to support Java[2]. Java is a C++ like programming language designed for general-purpose object-oriented programming [3]. An appeal for the usage of such a language is its ``write once, run anywhere" philosophy [4]. This is accomplished by providing a Java Virtual Machine (JVM) interpreter and runtime support for each platform[5].

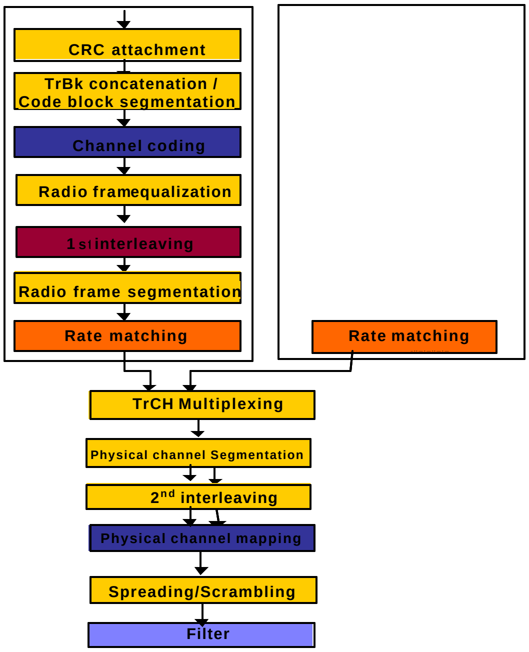


Figure 2. WCDMA 768bps Transmit Processing

WIRELESS COMMUNICATIONS

Figure 2 shows the processing chain for the UMTS 3G WCDMA transmit processing [6]. Previous communications systems have been developed in hardware due to the high computational requirements. DSP's in these systems have been limited to speech coding and orchestrating the custom hardware blocks. In high-performance 3G systems, there may be over 2 million logic gates to implement the system. A complex 3G system may also take many months to implement. After logic design is complete, any errors in the design may cause up to a 9 month delay for correcting the bugs and refabricating the device. This labor intensive process is counter productive to fast handset development cycles. The Sandbridge design takes a completely new approach to communications system design.

Rather than designing custom blocks for every function in the transmission system, Sandbridge has implemented a processor capable of executing operations appropriate to broadband communications. The small and power efficient core is then highly optimized and replicated to provide a platform for broadband communications. This approach scales well with semiconductor generations and allows flexibility in configuring the system for future specifications and any field modifications that may be necessary.

The Sandbridge process is to design the communications system in Matlab thus ensuring the bit and block error rates for the transmission system are achieved. The Matlab system design is then ported to fixed point C code. From that point, no further programmer intervention is required. The Sandbridge highly optimizing compiler extracts the parallelism in DSP operations and optimizes performance on the SandBlaster™ DSP.

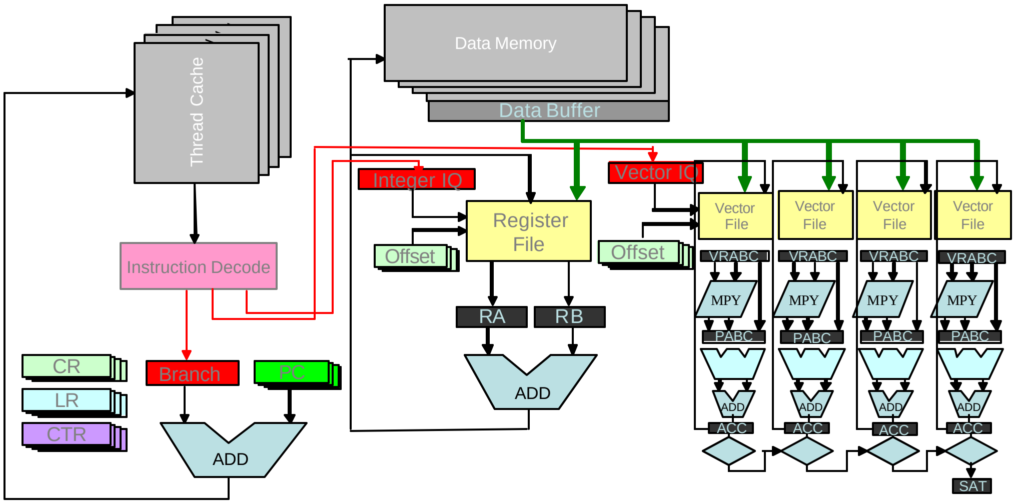


Figure 3. SandblasterTM Multithreaded Processor

SANDBRIDGE SDR PROCESSOR

Sandbridge Technologies has designed a multi-threaded processor capable of executing DSP, Control, and Java code in a single compound instruction set optimized for handset radio applications. The Sandbridge design overcomes the deficiencies of previous approaches by providing substantial parallelism and throughput for high-performance DSP applications while maintaining fast interrupt response, high-level language programmability, and very low power dissipation.

As shown in Figure 3, the design includes a unique combination of modern techniques such as a SIMD Vector/DSP unit, a parallel reduction unit, and RISC-based integer unit. Instruction space is conserved through the use of compounded instructions that are grouped into packets for execution. The resulting combination provides for efficient Control Code, DSP, and Java processing execution.

Java Execution

JVM translation designers have used both software and hardware methods to execute Java bytecode. The advantage of software execution is flexibility. The advantage of hardware execution is performance. The Delft-Java architecture, designed in 1996, introduced the concept of dynamic translation of Java code into a multithreaded RISC-based machine with Vector SIMD DSP operations [7][8]. Another of the authors also explored dynamic translation [9]. The important property of Java bytecode that facilitated this translation is the statically determinable type state [3]. The Sandbridge approach is a unique combination of both hardware and software support for Java execution.

Software Productivity

Programmer productivity is one of the major concerns in complex DSP applications. Because most classical DSPs are programmed in assembly language, it takes a very large software effort to program an application. For modern speech coders, [10] for example, it may take up to nine months or more before the application performance is known. Then, an intensive period of design verification ensues. If efficient compilers for DSPs were available, significant advantages in software productivity could be achieved.

One of the outputs of the Sandbridge tool chain is a traditional ISA compiler. GCC is an example of this type of compiler. With an ISA compiler, a high-level language (HLL) is compiled to the instruction set of the processor. Often, optimizations are performed in translating the HLL code into assembly language. Sandbridge Technologies has developed its own highly optimizing compiler. Software compilation enables the efficient translation of high-level language such as C/C++ into optimized machine language.

A unique aspect of the Sandbridge compiler is that DSP operations are automatically generated. The Sandbridge compiler uses a technique called semantic analysis. In semantic analysis, a sophisticated compiler must search for the meaning of a sequence of C language constructs. A programmer writes C code in an architecture independent manner - such as for a micro controller - focusing primarily on the function to be implemented. If DSP operations are required, the programmer implements them using standard modulo C arithmetic. The Sandbridge compiler analyzes the C code, automatically extracts the DSP operations and synthesizes optimized DSP code without the excess operations required to specify DSP arithmetic in C code. This technique has a significant software productivity gain over intrinsic functions.

Another challenge DSP compiler writers face is parallelism extraction. Early VLIW machines alleviated the burden from the compiler by allowing full orthogonality of instruction selection. Unfortunately this led to code-bloat. General purpose machines have recognized the importance of DSP operations and have provided specialized SIMD instruction set extensions. Unfortunately, compiler technology has not been effective in exploiting these instruction set extensions, and library functions are often the only efficient way to invoke them. Sandbridge architectures make liberal use of these so called multimedia instruction sets because DSP applications are amenable to them. The Sandbridge vectorizing compiler is efficient at extracting this parallelism using Vectorizing optimizations. The Sandbridge compiler also handles the difficult problem of outer loop vectorization which is often a requirement for inner loop optimizations.

SANDBRIDGE WCDMA RESULTS

Sandbridge Technologies has developed complete Matlab and C code for the 2 Mbps UMTS WCDMA standard. Using our internally developed compiler on our own algorithms, Sandbridge has achieved real-time performance on a 768kbps transmit chain which includes all the blocks shown in Figure 2. Having exceeded our estimated target, the entire transmit chain requires less than 400MHz of processor capacity to sustain a 768 kbps transmit capability.

SUMMARY

Sandbridge Technologies has introduced a completely new and scalable design methodology for implementing multiple transmission systems on a single chip. Using a unique multithreaded architecture specifically designed to reduce power consumption, efficient broadband communications operations are executed on a programmable platform.

The processor is combined with a highly optimizing compiler with the ability to analyze programs and to generate DSP instructions. This obviates the need for assembly language programming and significantly accelerates time-to-market for new transmission systems.

To validate our approach, we designed our own 2Mbps WCDMA physical layer. We then implemented the algorithms in both Matlab and fixed point C code. We then compiled them to our platform using our internally developed tools. Our measurements confirm that our WCDMA design will execute within field conformance requirements in real-time completely in software on the Sandblaster processor.

REFERENCES

[1] Will Strauss, Private email communication, July 16th, 2002.

[2] J. Yoshida, **"Java chip vendors set for cellular skirmish"**, *EE Times*, January 30th, 2001.

[3] J. Gosling, **"Java Intermediate Bytecodes"**, *ACM SIGNPLAN Workshop on Intermediate Representation (IR95),* pp. 111-118, January, 1995.

[4] J. Gosling and H. McGilton, **"The Java Language Environment: A White Paper"**, *Sun Microsystems Press*, October, 1995.

[5] T. Lindholm and F. Yellin, **"Inside the Java Virtual Machine"**, *Unix Review*, Vol. 15, No. 1, pp. 31-39, January, 1997.

[6] 3GPP TS 25.212 V4.2.0 (2001-09), Technical Specification, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Multiplexing and channel coding (FDD) (Release 4). Available from http://www.3gpp.org

[7] J. Glossner and S. Vassiliadis, **"The Delft-Java Engine: An Introduction"**, *Lecture Notes in Computer Science*. Third International Euro-Par Conference (Euro-Par '97), pp 776-770, Passau, Germany, August, 1997.

[8] J. Glossner and S. Vassiliadis, "**Delft-Java Dynamic Translation"**, *Proceedings of the 25th EUROMICRO conference (EUROMICRO '99)*, Milan, Italy, September, 1999.

[9] K. Ebcioglu, E. Altman, and E. Hokenek, **"A Java ILP Machine Based on Fast Dynamic Compilation"**, *IEEE MASCOTS International Workshop on Security and Efficiency Aspects of Java*, Eilat, Israel, January, 1997.

[10] European Telecommunications Standards Institute, Digital cellular telecommunications system, **ANSI-C code for the GSM Enhanced Full Rate (EFR) speech codec** (GSM 96.53), March, 1997, ETS 300 724.