*By Nick Marshall, Gary Silcott and Thierry Bailleul,
Wireless Subscriber Systems Group,
Motorola*

Baseband IC Solutions Address the Fast Changing Subscriber Handset Market

In today's fast-paced, highly competitive world of wireless communications, handset manufacturers continually seek to gain an edge on the competition. There are a number of ways to do this – make phones that are cheaper, smaller, run longer on a single battery charge, contain timesaving features and offer secure communications. These are a few of the traditional roads to product differentiation.

INTRODUCTION

As the industry and service providers migrate to third generation systems (3G), with its higher performance requirements, the choice of feature sets will expand the product mix even more. Implementing features – such as high-speed data, video, and position location, will require leading edge performance from the baseband IC. Adding handset "information management" functionality – much like a PDA (Personal Digital Assistant) – will further push the baseband performance envelope. This rising performance trend for the DSP and microcontroller in the handset is illustrated in the table below.

Addressing these requirements in the baseband IC requires new architecture core engines that can master the delicate balance between cost, performance and power consumption.

New Cores For New System Demands

In June of 1998, Motorola jointly announced with Lucent Microelectronics their joint effort, Star*Core, to develop DSP cores for future signal processing demands. The first fruit of this labour is the SC140, a modern DSP architecture deploying a super-scalar architecture with maximum parallelism and multiple Data and Address ALU's for multiple simultaneous operations in a single clock cycle. The Star*Core family of cores is one of the industry's first "compiler-driven" DSP designs in which the C-code compiler and hardware are precisely tuned to each other. The core features a scalable computational model and instruction set. This "post-VLIW" (Very Long Instruction Word) architecture uses a variable length execution set (VLES) which can expand or shrink with each hardware implementation of Star*Core. The variable length execution sets (VLES) are actually groupings of basic instructions intended for specific execution units.

In a handset the SC140 uses these scalable resources to very efficiently handle the voice and data channel coding, equalisation, modulation & demodulation, encryption and decryption. By executing 1.2 billion DSP MAC's per second, with 4 MAC's per tick of a 300MHz clock, the SC140 provides a lot of processing power, but will consume less than 0.1mA per DSP MAC at 1.5V, making it well suited for low-power handsets.

THE VARIABLE LENGTH EXECUTION SET MODEL (VLES)

The Variable Length Execution Set (VLES) model forms the key to understanding how the SC140 core addresses the requirements of DSP kernels while maintaining a very high code density for the whole application, and a very orthogonal, compiler friendly instruction set. Instruction words are always 16-bits wide and each instruction encodes an atomic operation. For example, a multiply-accumulate (MAC) instruction is encoded in 16 bits, a store (MOVE) instruction is encoded in 16 bits etc. Atomic operations need fewer bits to encode; therefore the 16-bit instruction set becomes fully orthogonal and very rich in the function-

PROTOCOL	DSP MIPS (LAYER 1)	MCU MIPS (LAYER 2/3)
GSM Phase 1	40	2
GSM Phase 2	60	4
GSM Phase 2+	100	10
UMTS/W-CDMA	200	30

Table 1. Baseband Processor MIPS Requirements

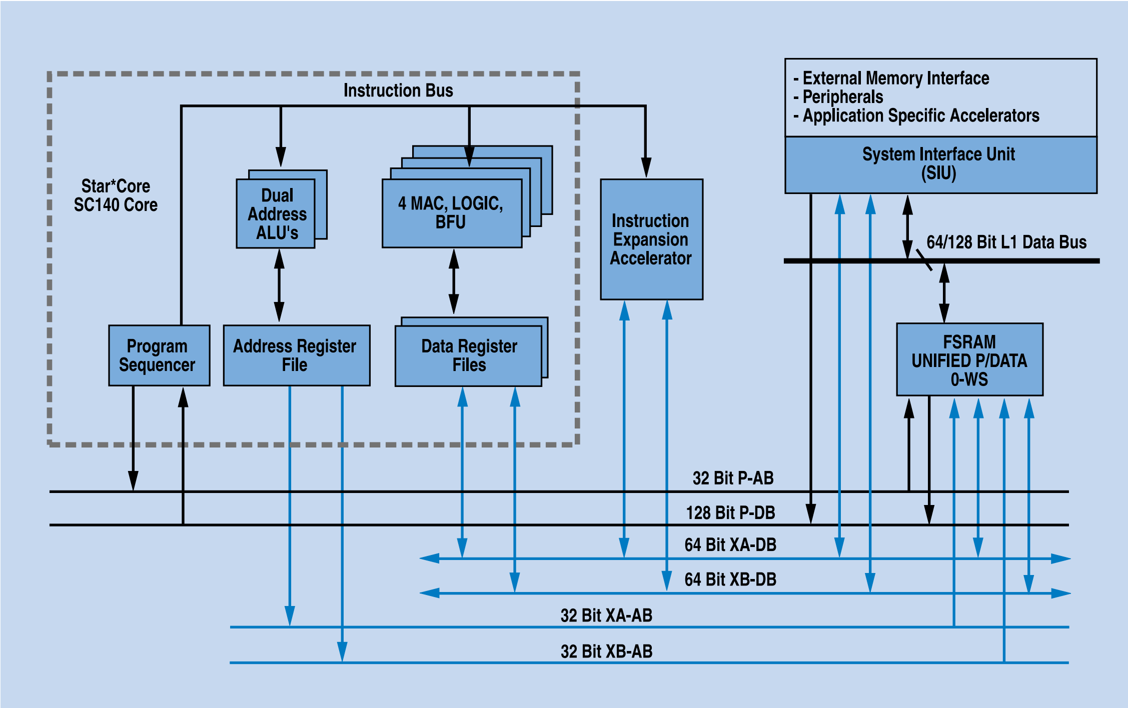

*Figure 1. Star*Core Architecture*

ality it supports. In order to execute Signal Processing kernels, a set of atomic operations needs to be executed in parallel. In order to accomplish this, a Fetch Set in the length determined by the quad-ALU SC140 is fetched every clock cycle from the program memory. During execution time, the SC140's Program Sequencer Unit automatically detects that a portion of the Fetch Set, called the Execution Set, should be executed in parallel. After detecting the execution set, each and every atomic operation that forms the Execution Set is dispatched to the appropriate functional blocks for execution.

To manage and control the system Motorola calls upon its popular M•Core™ architecture and its new M340 microRISC core. M•Core, was conceived specifically for the explosively growing digital portable and battery-powered markets, where extremely compact products live or die, based on their ability to get the user's application performed using the fewest bytes, clock cycles, bus cycles, and transistors as possible. Unlike all other architectures available in the industry today from which one can build portable products, the M•Core programmer's model was developed with the idea that power consumption, total system cost, and memory efficiency are of the utmost priority alongside runtime performance. Portable product designers cannot risk trading-off these objectives in favour of runtime performance. The M•Core architecture's code density, native 16-bit length instruction set, and real-time I/O handling capabilities are the keys to its achieving the industry's best MIPS per Milliwatt rating while at the same time offering the system designer and real-time software developer a clean, uncluttered programmers model with room to grow unencumbered into the future.

It features a 32-bit datapath and fixed length 16-bit instructions: 9% less code than the closest competing commercial architectures – typically 46% less. Three software controlled low power modes – DOZE, STOP and WAIT dynamic power management. Performance benchmarks for M•Core show that it offers 0.56mW/MHz at 1.8V when implemented in 0.22 micron CMOS.

The M•Core real time performance offers flexibility intelligent vectored and auto-vectored interrupts via a re-locatable vector table. A 16-entry alternate register file, machine state shadow registers and scratch registers and specialised bit and byte instructions ("Find First One") and save/restore instructions to eliminate software delays and loops in I/O routines all combine to give a real time performance among the best in the industry.

The M•Core architecture was designed with 11.2% unimplemented opcode space providing for future versions of the core and its instruction set. Bus "bridges" (AMBA, IMB3, M•Core, S-BUS) implemented in the core offers the capability of reusing 68300, Coldfire and ARM peripherals as well as native peripherals.

The M340 brings the addition of a single-precision 32-bit floating point unit and support for fast integer multiply with 16 x 16 executed in a single clock cycle and 32 x 32 in two cycles. Instruction buffering and branch penalty reduction through branch folding has also been added. With its on-chip cache & MMU, little endian support and paged MMU, the M340 will support Java, EPOC32 and WinCE.

These 21st century architectures represent the lowest power, highest performance cores ever made available and have been designed to accommodate the everincreasing complexity of cellular and mobile communications for many years to come. But DSP and MCU core performance alone will not be enough to meet the requirements of customers when designing future wireless systems. Beyond the traditional parameters of cost, performance, and power consumption will be flexibility; flexibility offered by open architectures not tightly coupled to one given supplier, flexibility offered by intrinsic architecture scalability, and the flexibility for customer to introduce their own IP in the baseband chip and to differentiate themselves from the competition.

Motorola addresses this need for flexibility in a number of ways; adding up to a flexible semi-custom business model that can be tailored to each specific customer request. Since both Star*Core and M•core are available from Lucent Microelectronics and Motorola, customers can access 2 viable independent sources and development partners. Further flexibility is available in the architecture itself: the Star*Core architecture is highly scaleable and a strong focus has been put on the compiler technology, making much of the software development possible in C.

Motorola can support a variety of development models, ranging from full Motorola custom design, to co-design or foundry type of relationship. Specialised peripheral IP blocks can be created by Motorola or customers, to be re-used and placed around these cores. In this way each manufacturer can preserve their intellectual property and offer a competitive advantage over the competition. Possible building blocks include Viterbi, encryption, serial codec interfaces, keypad interface, LCD control, external memory interface, SIM, GPS, GPRS or Bluetooth support. ■

Nick Marshall and Gary Silcott work for Motorola's Wireless Subscriber Systems Group in Austin, TX. Thierry Bailleul works for the same organization in Toulouse, France.