

The S6000 Family of Processors

Today's Design Challenges

The advent of software configurable processors

In recent years, the widespread adoption of digital technologies has revolutionized the way in which people communicate, work, and seek entertainment. With this adoption comes an insatiable need for bandwidth and computing power for digital processing of audio, video, and information streams. Increasingly powerful and complex systems must be delivered on shorter schedules to keep pace with market demand and competition. Conventional design approaches are increasingly falling short of the requirements on cost, time to market, and compute performance. Consider the options available to a system designer today:

Technology	Cost	Time To Market	Performance	Flexibility
ASIC	High NRE then Low	Long	High	Low
ASSP	Low	Short	High	Low
DSP	Medium	Short	Medium	High
DSP with FPGA	High	Long	High	High

Table 1: Relative merits of existing solutions

The S6000 family from Stretch® Inc. addresses the issues of price/performance, time to market and flexibility with its software configurable processor. A software configurable processor takes software "hot spots" (sequences of operations that are executed repeatedly and consume the majority of the compute resources) and optimizes them into exceptionally fast custom instructions. On Stretch processors, an entire hot spot, expressed only in C/C++, is reduced to a single instruction within the Instruction Set Extension Fabric (ISEF). The fabric is configured through software in real time to adapt the capabilities of the processor to the required algorithm.

At the heart of the S6000 devices is a second generation ISEF that is three times faster than that of previous generations of Stretch software configurable processors and contains 64KB of embedded ISEF RAM (IRAM). A Programmable Accelerator core has also been added to the S6000 family of devices with dedicated acceleration elements specifically targeted at the most common algorithms found in multimedia processing. The result is an enormously powerful and efficient processing engine that is highly optimized for video and audio applications—a single S6000

family device can encode one H.264 High Definition video stream or four concurrent Standard Definition streams.

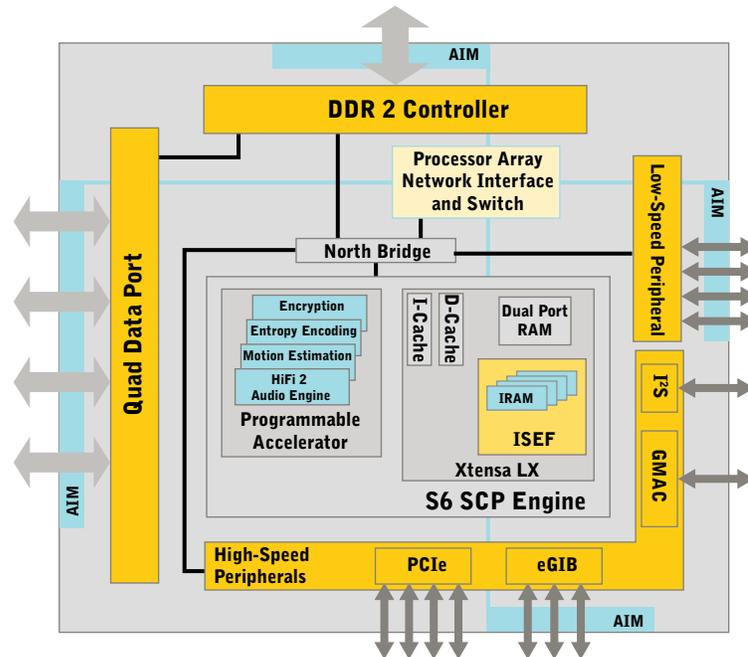


Figure 1: The S6000 family architecture

The S6 Software Configurable Processor Engine

Raising the performance bar

The S6SCP Engine within the S6000 family of devices contains a Tensilica® Xtensa® LX VLIW core and the second generation Stretch ISEF. It is the ISEF that provides the dramatic application acceleration by allowing user algorithms to be instantiated in hardware and called by the processor as single instructions. This use model also has tremendous advantages for silicon efficiency and, hence, platform cost and performance.

Traditional FPGA accelerators are required to host gate level or netlist descriptions of their hardware functions. To accommodate all potential configurations, the FPGA fabric is dominated by programmable interconnects. By contrast, the Stretch ISEF, being tightly coupled to the processor, needs only to host compute-based and logic functions. Since the problem is constrained, the ratio of compute elements to interconnects within the ISEF can be made much higher. The result is a more efficient programmable fabric that occupies less silicon area.

The S6000 ISEF contains 4096 Arithmetic Logic Units that, in addition to traditional ALU functions, can be configured to perform 2x4

multiplies and grouped for larger bit widths. In addition, there are 64 dedicated multipliers capable of 8x16 operations that, again, can be grouped to increase bit widths. Distributed state registers provide local storage for intermediate values and coefficients. Connectivity of the processing elements is enhanced with distributed multiplexers, priority encoders and shifters.

The entire design process for Extension Instructions within the ISEF is completely automated by the Stretch tool flow. The designer is free to explore parallelism within the algorithm in C and, through compilation and cycle-accurate simulation, view the performance improvement achieved by the hardware optimization.

The section of code tagged for compilation to the ISEF is examined by the compiler for parallelism at design time. Inner loops are automatically unrolled, place and route is performed, and connectivity is established to implement the desired function. The compiler generates a resource usage report with which the designer can ascertain whether additional functionality could be added to the fabric by exploiting further parallelism within the application. The Extension Instruction timing is then exported to the application compiler, which automatically schedules Extension Instructions along with regular instructions to optimize execution of application code.

Furthermore, the second generation ISEF can be completely reconfigured in 27 microseconds. This means that the ISEF can be reused in real time by applications whose code can take advantage not only of the Extension Instructions appropriate for that application, but also the Extension Instructions appropriate for that particular portion of the algorithm within the application. The performance improvement potential of this approach is enormous.

For applications that are compute intensive, the S6 ISEF is fed by the same 32 128-bit wide registers carried over from previous generations of Stretch devices. These registers are used for loading data into the ISEF, and their presence in the S6000 ensures maximum compatibility and code reuse from previous software configurable processor designs. The S6000 ISEF also contains 64KB of embedded ISEF RAM (IRAM) distributed throughout the fabric in 32 banks of 2KB each. This IRAM can be used as storage for data, coefficients, look-up tables, or intermediate results for operations. The IRAM is memory mapped into the S6SCP address space, so can be loaded directly by the processor. The IRAM also has a dedicated DMA channel so it can be loaded without processor intervention, dramatically increasing the throughput of data within the ISEF.

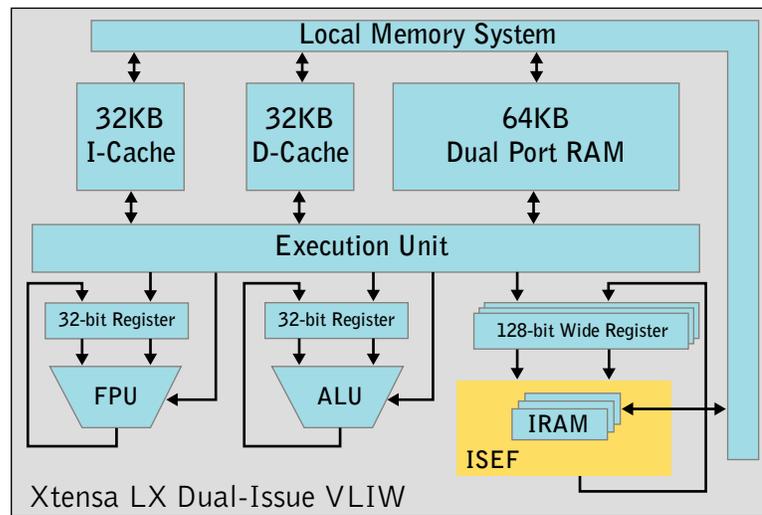


Figure 2: The S6SCPEngine

The Programmable Accelerator

Highly optimized domain-specific acceleration

When considering the range of market segments that would benefit from the software configurable processor concept, several applications appear repeatedly. They are:

- Video and image processing
- Software defined wireless protocols
- Audio processing

To assist in these applications, a dedicated Programmable Accelerator has been added to the S6000 family of devices. The Programmable Accelerator consists of a series of highly optimized functions implemented in hardware. The hardware functions are grouped and exposed to the system programmer as APIs that provide a highly flexible and configurable way to access them at an application level. The library of available APIs includes:

- Motion Estimation for video encoding
- Entropy Encoding for H.264 video (CABAC/CAVLC)
- Encryption/Decryption (AES, DES, 3DES)
- Audio CODECs (AAC, AC3, MP3, etc. 19 in total)

As an example of the degree of application acceleration that is possible using an S6000 device, consider the case of the Motion Estimation accelerator. Within the Motion Estimation accelerator function, an entire 16x16 pixel macroblock Sum-of-Absolute Differences (SAD) operation has been optimized into a single cycle. The pipelined nature of the hardware accelerator means that on each clock cycle, 256 pixel SAD calculations are made. Furthermore, on each cycle, the Motion Estimation accelerator returns all 41 possible H.264 sub-macroblock combinations with their corresponding SAD values.

This operation alone traditionally consumes a large portion of a processor performing motion estimation and, in fact, takes so much processing resource that the majority of H.264 CODECs today are unable to perform the operation on all potential macroblock combinations over large search ranges. In our example, the S6000 device performed the macroblock SAD operation in a single cycle and presented the results for analysis by the CODEC algorithm.

Multiprocessor Architectures

Seamless board level scalability

In applications where extreme performance is required, multiple processors may be needed to achieve the required compute bandwidth. With conventional microprocessors and DSPs, this poses a challenge of partitioning the application as well as connecting the devices and arbitrating among them. The S6000 family of devices overcomes this challenge with its Processor Array. The goal of the Processor Array interface is to abstract away the inter-chip communication and allow multiple devices to collaborate and be designed in a single cohesive integrated development environment.

The S6000 family Processor Array technology consists of a set of BIOS calls (PA-BIOS), a physical interface called the Array Interface Module (AIM), and a Network Interface and Switch. Each processor in the S6000 family has four AIM ports that are designed to communicate at high speed and to connect together without glue logic. Each port can move data between devices at a rate of over 2.4GB/s (1.2GB/s in each direction).

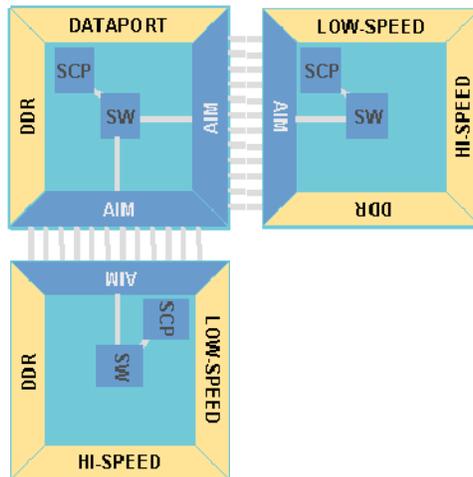


Figure 3: Device connection using the Processor Array

The Network Interface and Switch within S6000 devices provides routing between AIM ports or to the internal processor itself. The hardware implementation of the switch ensures that Physical, Network, and Transport layers of the network model are handled without software intervention. In this way, S6000 family devices can be formed into arrays of arbitrary topology without regard for the underlying communication infrastructure. Session layers are handled by Stretch PA-BIOS calls that ensure efficient and high bandwidth communication.

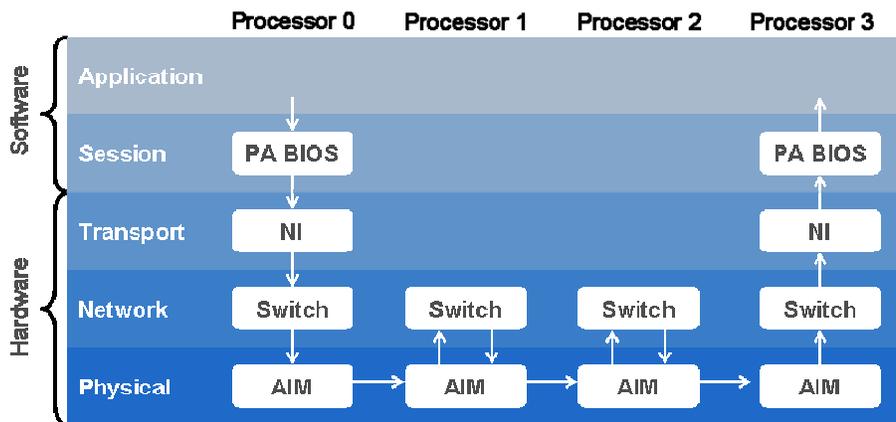


Figure 4: The Processor Array network Implementation

In the S6000 family of devices, the AIM and I/O features for any particular side of the device are multiplexed onto a common set of pins. In this way, device package size is reduced to a minimum and the flexibility to select between inter-device connectivity or I/O system connectivity is

retained. A simple configuration option in the development environment is used to select the appropriate interface and to describe the system topology. In this way, the optimum mix of I/O and processing bandwidth can be selected for any desired system.

The S6000 I/O Infrastructure

Rich I/O interfaces minimize system costs and complexity

The highly accelerated processing capabilities of the S6000 family of devices make it ideally suited for single-chip video and audio processing applications. To minimize system BOM costs, Stretch has also integrated a comprehensive set of I/O standards directly into the devices to give the S6000 family the capability of interfacing directly to a wide variety of audio/video devices and system busses without glue logic. Among the rich set of I/O interfaces are:

Quad Data Port

Four 10-bit data ports are included. Each is designed to interface directly to common video devices as well as sensors and video encoders or decoders from industry leading manufacturers. These ports can be used in native data mode where they are backed by their own independent FIFOs or they can be used in a variety of video modes. In video modes they handle BT656 and BT1120 data directly, or they can handle raw video with associated sync pulses in a variety of color spaces or chroma sampling ratios.

GMAC

A triple speed 10/100/1000 Ethernet MAC is included for seamless integration with high speed data networks.

Serial Interfaces

A variety of serial ports ensures compatibility with external interface standards and provides for configuration and control. Two-wire Interface (TWI) and SPI are used for control and configuration of other devices. I²S provides for audio I/O, and two general purpose UARTs can be configured for many interfaces, including IR. Test and configuration can be achieved through an IEEE1149.1 JTAG interface.

DDR Interface

A 32-bit DDR2 667 SDRAM interface is included to ensure the memory bandwidth meets the demands of high performance video applications.

eGIB/GPIO

The Enhanced Generic Interface Bus (eGIB) is designed for communication with memory-mapped devices, and has specific modes for

communication with removable memory devices such as Compact Flash. GPIO can be used to provide user-defined I/O requirements and system signals such as interrupts, displays, and key pads.

Conclusion

Unlocking the true potential of software configurable processing

The Stretch S6000 devices are the most powerful software configurable processors on the market today. They offer unparalleled performance in the areas of audio/video and wireless data processing. A rich I/O infrastructure and glueless connectivity to system elements opens up performance price points that, until now, have been beyond the reach of system designers. The Processor Array interface, a standard feature on all S6000 family devices, allows for arbitrarily complex networks of devices to be constructed without affecting performance. With its single development environment and transparent board-level scalability, the S6000 family is a natural choice for designers of systems of all sizes.