Processors with On-Die Cryptography Accelerators

Joseph Lee Greathouse

April 13, 2007

Contents

1 Introduction

With the continuing rise of transistor counts in modern microprocessors, computer architects are finding it difficult to effectively utilize the vast resources presented to them. Monolithic processors, the common design paradigm for many years, run far too hot and are far too large to meet performance demands.

Even with multiple processors on a single chip there are applications whose performance demands cannot be met. In these cases, system architects often include accelerators in their design. Accelerator circuits are built to speed up common operations. Because of the abundance of transistors on modern chips, it is easier than ever to justify the inclusion of accelerators like floating-point, signal-processing, and vector-processing units.

One popular accelerator is the cryptography core. These units are found in processors that often deal with encryption and decryption. While these are sometimes external chips that require communication with off-processor resources, quite a few modern processors include cryptography cores on the processor die itself.

Intel's IXP2850 is a processor used to control high-bandwidth network applications such as OC-192 routers. Its design is the same as the IXP2800, except for a pair of onboard cryptography units that allow the IXP2850 to encrypt the massive amount of data it routes.

Sun Microsystems' UltraSPARC T1 and upcoming UltraSPARC T2 are microprocessors aimed at network, web, and application servers. These chips contain multiple Stream Processing Units, each specifically designed to quickly and asynchronously complete a variety of cryptographic operations.

The latest processor in the IBM z/Architecture line, which is used in the System z9 series of mainframes, includes a set of features called Central Processor Assist for Cryptographic Function. CPACF is a series of on-chip cryptography accelerators used for communication-based encryption.

The speed and versatility of the cryptography units on these chips vary due to these processors' diverse application domains. What follows is a case study of these microprocessor designs. This study gives an overview of each processor's architecture and its cryptographic accelerator. It then compares these cores and the design decisions behind them.

2 Hardware Cryptography

System designers have used hardware cryptography engines in designs for many years. For instance, peripheral cards with custom-designed encryption chips are offered by companies such as Sun Microsystems [Sun06b]. Similarly, IBM offers coprocessors for their mainframes that accelerate numerous cryptographic algorithms [AD04]. There are companies that make true random number generators in hardware to supply computers with good sources of random bits [Dav02]. There are even tape backup drives that include embedded cryptography processors which are used to encrypt backup data on the fly [Olt06].

The plurality of cryptography chips is indicative of the sheer amount of information that computer users must keep secret. Financial institutions, government agencies, and private businesses deal with a large amount of encrypted data every day. These chips are used to increase the processing power available to end-users who must deal with this data.

Hardware cryptography engines offer power and performance advantages over general-purpose processors running software cryptography algorithms. For symmetric-key algorithms, one study shows performance gains of 4000% over a modern CPU at similar clockspeeds. At the same time, the cryptography processor in the study used $1/16^{th}$ the power of the general-purpose processor [Bir98].

There are disadvantages to using dedicated hardware for cryptography, however. The added cost of peripherals specifically for data encryption can drive many users away from buying these units.

Some systems, such as embedded devices or small servers, may also have little room for added cards or boxes meant for only one task.

Even with a peripheral accelerator, the CPU must go off-chip to access the cryptography functions; this can take an extremely long time. The overhead of sending cryptography requests off-chip can destroy the performance benefit of the peripheral when working on many small pieces of data.

It is possible to ease these problems by putting the cryptography functions on the central processor itself. This amortizes the cost of cryptography accelerators across every manufactured processor while only slightly increasing the area of the chip. The penalty of going off-chip is removed, though the cryptography accelerator can become bottlenecked by sharing datapath resources with the processor.

Some modern processors have taken this path and include, at one level or another, cryptography accelerators within their core. The functionality of these accelerators can range from arithmetic units optimized for modular operations to full-blown circuits that do entire encryption sequences directly from system memory. The power of the design is dictated by both the expected software load of the processor and the ability to use the processor in conjunction with off-chip cryptography units. The following sections detail four processor designs as well as their on-chip cryptography accelerators.

3 Intel IXP2850

The Intel IXP2850 is a network processor based on Intel's Internet Exchange Architecture. It is an evolution of the IXP2800, and the microarchitectures of these chips are almost identical; the IXP2850 simply adds a pair of on-chip cryptography accelerators. The IXP28x0 line is targeted at a range of common network processing tasks, and these chips are often used in situations such as TCP routing [Int05].

The IXP2850 includes integrated cryptography units to accelerate the real-time encryption and decryption of data in secure communication channels. Examples of these channels include virtual private networks, applications that use IP Security protocols, and SSL communications. It is possible for the IXP2850 to maintain 10 gigabits per second of network throughput even while operating on secure channels. With the help of the specialized cryptography cores, an IXP2850 can meet these bandwidth requirements at a power budget of only 32 watts [Int02a].

3.1 IXP 28x0 Architecture

All IXP28x0 network processors use the same basic microarchitectural layout; the primary difference between the IXP2800 and the IXP2850 is the addition of two cryptographic acceleration units into the IXP2850.

At the center of the processor is an Intel XScale microprocessor core. This is connected through a bus to memory and a set of 16 microengine coprocessors. The IXP2850 also includes two cryptographic processing units on this bus [Int02a].

The XScale core, which is a RISC core based off the ARMv5 instruction set architecture, is the central control unit of the IXP28x0 processor. In order to save area and power, this XScale core contains no dedicated floating-point hardware, though it contains a dedicated multiply-accumulate pipeline. It operates at a frequency of 700MHz and contains a single-issue pipeline [Int02b]. The XScale processor communicates with the system backplane and routes data to the individual microengines.

The 16 microengines in the IXP28x0 are minimal execution cores designed for high throughput. These units do the vast majority of packet processing in this chip, and their designs are optimized for this task. Each microengine is multithreaded and operates at a frequency of 1.4GHz. Each microengine is connected directly to its nearest neighbors in order to facilitate parallel processing of large data sets [Int05]. Every microengine also contains an embedded CRC calculation unit to streamline data hashes for TCP lookup tasks. The independent processors are useful in a network processing environment due to the large amount of data that is functionally independent from all other data (e.g. TCP/IP packets) [MRK+03].

While the microengines have an enormous amount of throughput in regular networking applications, they are not equipped to perform full-speed encryption or decryption of network data. The IXP2850 includes two identical cryptography acceleration units for situations where a network router will deal with large amounts of secure data. The cryptography units include acceleration hardware for 3DES and AES encryption schemes and the SHA-1 hashing algorithm [Int05].

3.2 Cryptography Accelerators

The on-chip cryptography accelerator is designed to encrypt and decrypt DES, 3DES, and AES data streams. Additionally, any data going through a cryptography block can be hashed using a hardware SHA-1 engine.

Each IXP2850 contains two cryptography blocks, and each block has two DES/3DES units, a single AES unit, and a SHA-1 unit. The DES units and AES unit cannot be run simultaneously, though both DES units can be run in parallel. After an encryption units finish its computation, its output can be hashed through the SHA-1 unit. It is also possible to bypass the DES and AES units and simply hash an arbitrary piece of system data [Int02a].

Each cryptography block contains 1KB of RAM to store values received from the system bus and a pair of FIFO queues used to pipeline incoming data while the units are calculating. Like the microengines, each cryptographic block contains a checksum calculation block to prepare the encrypted data for hashing in other parts of the processor [Har05].

3.2.1 DES/3DES Unit

Each cryptography block in the IXP2850 contains two 3DES units. The units themselves can store three 192-bit keys and optional 64-bit initialization vectors (IVs) at a time. After the 3DES accelerator finishes calculating a block of data, it is possible to calculate (and compare) a CRC value and SHA-1 hash for the data.

While Intel has not released the exact details of the cryptography units, it is mentioned that they work in both electronic codebook (ECB) and cipher-block chaining (CBC) modes of operation. The initialization vector would obviously not be used in the ECB.

The data rate that each 3DES unit can reach is not available in public documentation, nor is the low-level design of the hardware accelerators. Available numbers imply that it is possible for all four 3DES units running in parallel to do 24 million encryptions per second [Int02a]. Based on the operating frequency of 700MHz and the fact that there are four units, this yields a minimum of 116 clock cycles per encryption/decryption. While this may seem slow, it is possible for the IXP2850 to saturate an entire 10Gbps line with data and HMAC information; putting more power into the cryptography unit would waste power and chip area.

The same unit that does 3DES calculations also does the single DES calculations. The 64 bit DES key is copied into the 192-bit 3DES key register 3 times. This results in an encryption/decryption/encryption (or vice-versa) sequence that results in a single DES run.

3.2.2 AES Unit

Each cryptography block also contains a single AES unit. This unit has access to the key and IV registers for both 3DES units, and can thus have 6 pairs loaded at any time. Because it shares its key and IV registers with the 3DES units, the two engines may not be used at the same time. The keys can be 128, 192, or 256 bits in length, while the initialization vectors are all 128 bits [Har05].

Because the AES unit takes the place of the two 3DES units when it is activated, it is also possible to find the CRC and SHA-1 of any values computed by the AES unit. The AES unit also has the option of using either ECB or CBC modes of operation [Int05].

The public documentation for the AES core is even sparser than that of the 3DES cores. At its maximum, it may also move the same amount of data as both 3DES cores. This means that it too can possibly complete one encryption/decryption in 116 clock cycles.

3.2.3 SHA-1 Unit

The SHA-1 unit can take its source from either of the 3DES cores, the AES core, or input directly from the system bus. The data from these units is placed into an input buffer and a SHA-1 hash is computed by a hardware block. Data for this block is also not available publicly, but it is known that it is possible to compute 10.6 million HMAC-SHA-1 checksums every second. This means that it takes upwards of 132 cycles to perform a single SHA-1 computation [Har05].

3.3 Design Choices

The cryptography accelerators in the IXP2850 were designed for a single specific type of data workload. Network communication is a simple task at heart: fetch a packet, decode it, and forward the packet to the correct location. However, in high-bandwidth situations, these simple operations must be done an enormous number of times. The IXP2850 was designed to deal with high-bandwidth, massively-parallel workloads. Because each of the 16 microengines can offload its cryptography requirements onto the cryptography units, the units themselves were designed with high throughput in mind.

The cryptography units have the ability to buffer data coming from both memory and the microengines. They keep this data in FIFO queues until there are engines available to process it. This frees the microengines to do other tasks while they wait for the cryptography units to complete their work.

The IXP2850 is not the first network chip to utilize hardware cryptography accelerators. However, the accelerators on older network processors were contained on peripheral cards. This resulted in much longer delays while waiting for data to move to and from the card. The cryptography units in the IXP2850 were moved into the chip itself to take advantage of the high bandwidth available on a single piece of silicon.

The somewhat limited performance of the cryptography units in the IXP2850 is a constraint of the application domain of the IXP2850. IXP28x0 processors are designed to run in tightly space-constrained environments such as server blades in datacenter racks. One common design of an IXP28x0 server blade includes two IXP28x0 processors: one for outgoing data and one for incoming data. This means that building a large, power-hungry processor is infeasible. Much of the limitations of this chip stem from the power and area constraints placed upon it.

The cryptographic functions supported are also a factor of the application domain. It was important to only include the most useful algorithms on the chip because the amount of space and power is critically constrained. Rather than supporting a large number of cryptographic operations, the accelerators on the IXP2850 are designed to perform 3DES and AES encryption and SHA-1 hashing. These functions are the core of most IP Security and SSL applications.

4 Sun Microsystems UltraSPARC T1 and UltraSPARC T2

The UltraSPARC T1 and UltraSPARC T2 are two enterprise server processors designed by Sun Microsystems. These processors are designed to take advantage of the high thread-level parallelism available in server applications in order to yield high application throughput at low power levels. Both chips have eight individual cores and a number of other interesting features to increase performance in applications such as web, mail, and application serving.

Many network serving tasks require secure communication channels, so one of the added features on both the T1 and T2 is onboard cryptography accelerators. The T1 includes a modular arithmetic unit that can be used to accelerate public-key cryptography operations, while the T2 includes a more robust general-purpose cryptography engine.

4.1 UltraSPARC T1 Architecture

The Sun Microsystems UltraSPARC T1 is an enterprise server processor designed for excellent performance per unit power. It is designed to work well with highly-threaded applications and accomplishes this by putting eight in-order cores on each chip.

Each core has a simple six-stage in-order pipeline and is designed to work primarily on integer operations. The T1 has a single floating-point unit that is shared between all eight processing cores; this can become a bottleneck in general-purpose computing. The processor has a 3MB L2 cache on-chip and an on-chip memory controller.

As is evidenced by the limited floating-point capability and the large number of simple processing cores on the chip, the UltraSPARC T1 was space-constrained. Reports of the T1 mention that it is a scaled-back version of the originally-proposed design. The reductions were primarily to fit the T1 onto a single chip at a 90nm manufacturing capabilities [McG06].

4.1.1 UltraSPARC T1 Stream Processing Units

The designers of the T1 wanted to put on-chip cryptography engines into the processor but seemed to be held back by the limited space allocated to each core. Each of the eight cores in the T1 includes a cryptography accelerator called the Stream Processing Unit, or SPU, that runs at the processor's core clock speed.

Due to the space constraints, the SPU can only be used for RSA (and the similar DSA) workloads with keys of up to 2048 bits. Each SPU has its own modular-arithmetic unit that is used to speed up the modular exponentiation used in the RSA algorithm. The multiplier for each SPU is shared with the integer pipeline. A thread in the pipeline sends a request to the SPU in order to initiate the cryptography engine. The SPU then operates asynchronously from the pipeline, and it can either wait for the calling thread to poll and retrieve its data, or it can send out an intterupt to the thread to tell it that the data is ready [McG06].

Public-key encryption is a computationally intensive task, so the addition of a small accelerator may have met Sun's needs for higher performance at lower power. A modular arithmetic unit is smaller on a chip than the relatively large DES or AES accelerators, so the addition of only the RSA accelerator may have resulted in the largest performance gain for the least amount of chip space increase [Ara91].

4.1.2 Design Decision and Performance

The UltraSPARC T1 is first and foremost a server processor. Workloads such as web serving are often highly threaded, and Sun's benchmarks tout this fact [Sun06a]. The T1 is often described in

Sun literature as a server on a chip: it includes large amounts of on-chip memory, memory controllers, and eight processing cores.

Sun's benchmarks show middling performance numbers, but excellent performance/watt ratings. Because the cores on the T1 are simple in-order pipelines, the support circuitry around each core is reduced. This means that more power on each chip goes directly towards actual computations [Sun06a]. The T1 consumes 72 watts of power during its normal operation [McG06].

When benchmarking the power of the T1 at cryptographic tasks, Sun primarily relies on the speed of SSL session starts. SSL communication begins each session by trading public-key encrypted handshakes, and then communicating the secret key between the server and the client, once again encrypted using a public-key system. After this, the communication channel is encrypted using the secret key and a symmetric-key encryption system.

Because the T1 does not have accelerators for symmetric-key algorithms, all of the cryptography benchmarks that Sun publishes rely on showing how many handshakes and key-trades the T1 can work through. Sun's published benchmarks show that it easily outperforms its contemporaries at public-key cryptography for SSL connections. However, there are no published numbers to show the T1's performance after the SSL channel has been established [Sun06a].

One interesting fact about the cryptography performance of the T1 is that it is quite a bit slower at RSA encryption than it is at decryption (proportional to other systems). The encryption scheme used in the RSA algorithm requires only a single modular exponentiation operation, whereas decryption requires two. The performance gained from the modular arithmetic unit for RSA encryption is overshadowed by the time spent setting up the SPU [Lin05].

The T1 is best suited for enterprise server tasks where it must deal with many small requests simultaneously. The architectures of both the processor and its cryptography units are best suited for a large number of parallel tasks. Its performance is weak in individual single-threaded tasks, floating point calculations, and perhaps any communication-based cryptography that is not public-key handshaking.

4.2 UltraSPARC T2 Architecture

The UltraSPARC T2 is the upcoming successor to the T1, and it is slated to improve on many of the lower-performing features of the earlier processor. Sun has moved to a 65nm manufacturing process, which leaves them more room for acceleration hardware. While the L2 cache has only increased from 3MB on the T1 to 4MB on the T2, the actual size of the chip after the process shrink has not changed much at all [McG06].

There are a number of additions to the T2 that are used to increase its overall performance in general tasks. The T2 still has eight cores per chip, though each core now has two execution pipelines. Due to both the poor performance of the T1 in floating point tasks and the additional space afforded by the die shrink, the T2 also includes a floating point unit for each individual core. Additionally, the T2 has a pair of onboard network interface adapters. These will allow the chip to move data directly to the network without sending it to an off-chip resource.

The cryptography accelerator on the T2 is also much more advanced than the one found on the T1. Each core still has access to a single cryptography unit, but each cryptography unit now supports more encryption algorithms. The cryptography unit shares some of its resources with the FPU in order to limit the amount of die space taken by this single-purpose accelerator [Gro06].

4.2.1 UltraSPARC T2 Stream Processing Units

While the UltraSPARC T1 only supported a limited set of public-key encryption algorithms, the UltraSPARC T2 adds acceleration for both symmetric-key algorithms and a new public-key algorithm.

The SPU in the T2 has hardware support for RC4, 3DES, and AES-128/192/256 symmetric-key algorithms. It also supports MD5, SHA-1, and SHA-256 hashing schemes. It maintains support for the RSA and DSA public-key algorithms, and adds in support for binary and integer polynomial elliptical curve cryptography.

Because of these new supported protocols, there are more acceleration circuits included in the T2 SPU. Like the T1 SPU, there is a modular arithmetic unit used for public-key cryptography systems. This unit is used for both RSA and ECC cryptosystems. This time the multiplier for the MAU is shared with the floating-point unit. This resource sharing is undoubtedly due to space constraints. This could possibly lead to a resource bottleneck if one thread in the core is doing floating point calculations while another is trying to do public key cryptography.

The other accelerator in the SPU is a cipher unit used to calculate the symmetric key ciphers and hash functions. The speeds of this unit are not published, but Sun claims that it should be possible to run one of the onboard network adapters at full line speed (10 gigabits per second) on a secure channel.

To enable the cryptography engine to work on large chunks of data, the SPU has a direct memory access (DMA) engine that allows it to access the L2 cache without having to go through the regular pipeline [Gol06]. The benefit of this configuration is the ability of the SPU to autonomously carry out cryptographic actions while the host thread remains stalled.

The final large addition to the SPU is a true random number generator. In order to break away from the determinism of digital systems, this tRNG amplifies Johnson noise detected over a series of resistors. The amplifiers power a series of voltage-controlled oscillators, and these VCOs are sampled, XORed together, and saved in a shift register. After a set period of time the register is cleared and the process begins again [NHW+07]. Because the Johnson noise is a physical process, the system gives an unbiased random sequence of bits for use in operations such as secret-key generation.

4.2.2 Design Decisions

The UltraSPARC T2 has not been released to market yet, and there are no official benchmark results available. It is aimed at the same market segment as the T1, and it has made some major improvements in many of the areas where the T1 fell short.

The additional hardware in the cryptography unit is due to the increased area available to each core in the T2. Even with this extra room the SPU shares some of its hardware with the floating-point unit to reduce overhead. This could possibly be to ameliorate the cost of the eight floating-point units.

With the inclusion of network interface devices on the chip itself, it is important that the T2 be able to provide enough data to put on the network. If the network operations are on a secure channel and there were no cryptography accelerators, performance would suffer greatly. One of the biggest reasons for increasing the number of supported algorithms was to enable the cores to quickly move data onto the network even in symmetric-key secure situations.

5 IBM System z9

System z9 is a class of mainframe computers designed and manufactured by IBM. These systems are used in a number of compute- and I/O-intensive tasks such as financial-transaction processing, warehouse database operations, and insurance computations. System z9 is based on IBM's z/Architecture and is the successor to the IBM zSeries 990.

System z9 mainframes utilize between 8 and 54 proprietary central processors and can be configured with a wide range of additional peripheral cards tailored to the workload of the system. A typical set of tasks for a mainframe such as the z9 is extremely parallel because of the sheer amount of individual, unrelated tasks that must be performed [IBM06c].

In the z9, each central processor contains a group of hardware cryptographic accelerators, collectively known as Central Processor Assist for Cryptographic Function, or CPACF. These accelerators were originally added to the zSeries 890 and were extended for the System z9 to include more cryptographic functions. The functionality of the on-chip cryptography units is limited, but IBM offers a wide range of peripheral cards meant for much more intensive cryptography applications [KDBH05].

5.1 z/Architecture

The architecture of zSeries processors, known as z/Architecture, is a closely-guarded IBM secret. Besides being designed and manufactured entirely within IBM, the processors are only available in expensive mainframe systems. This limits the ability to study these processors closely. It is known that the processors in the zSeries are the first in the IBM mainframe line to support 64-bit addressing and 64-bit primitive data types [IBM05].

The instructions in zSeries processors have 16-bit opcodes, and many point to a number of control registers that contain additional information about running the particular instruction. Simple instructions (such as loads, stores, and basic integer operations) are performed atomically, while more complex instructions (such as cryptographic functions) are performed by millicode. Millicode breaks apart these complex instructions into much smaller machine-level instructions, much like what is done for microcode on modern x86 processors [HF04].

Central processors in the IBM zSeries, from the zSeries 890 up to the new z9, contain on-chip acceleration for cryptographic functions. While the hardware layout for these units is not publicly available, it is known that they can work on large sets of data. The cryptographic functions are called using a series of instructions called Message-Security Assist (MSA) instructions. These pass the cryptography units both the starting address of the data in memory and the number of successive bytes to encrypt. This allows the units to encrypt large chunks of data with little communication and setup overhead [KDBH05].

5.2 Cryptography Accelerators

The IBM System z9, like the zSeries 990 and zSeries 890 before it, has hardware support for a number of different cryptography operations. The accelerators in the central processors support DES, 2/3DES, and AES encryptions schemes. There is support for SHA-1 and SHA-256 hashing schemes as well as a DES-based pseudo-random number generator. The symmetric-key encryption algorithms can be run in ECB or CBC modes of operation.

There are optional peripheral cards that support encryption schemes ranging from RSA to custom algorithms submitted by clients. The central processors, however, only have hardware support for the above-listed schemes. The cryptography accelerator on each central processor runs at full processor speed.

The encryption accelerators are called by one of two z/Architecture assembly instructions. The first, KM, runs the encryption using ECB. The other, KMC, runs the encryption using CBC.

In both KM and KMC, general purpose register (GPR) 0 tells the cryptography accelerator which algorithm to use. GPR 1 points to the area in memory that contains the key for the encryption engine (and in the case of KMC, this area in memory also contains the initialization value). The instruction itself points to a set of registers that point to the first byte of the source data, the number of bytes to encode, and the first byte in memory to begin placing the encoded results. It is possible for one instruction to be used to encode an extremely large chunk of data in this manner. Because these are symmetric cryptography systems, the same commands used to run the encryption scheme are also used for decryption.

Pseudo-random number generation is also performed using the KMC instruction. This PRNG takes a set of keys and data from memory as its input, and it performs a series of DES encryptions on them. At the end of these transformations, a series of pseudo-random numbers are available at a specified destination in memory [IBM05]. The exact methods behind this PRNG scheme are detailed in ANSI X9.17.

The z9 central processors also have assembly instructions (KIMDB and KLMD) that work in a similar manner for computing SHA-1 and SHA-256 digests of chunks of data. The previous is used when the data size is a multiple of 64 bytes, while the latter is used when it is not. The final MSA instruction is KMAC, which is used for keyed-hashed message authentication [IBM05].

This performance of an individual z9 processor at cryptographic tasks depends on the speed of the processor. IBM releases different speeds for its varying levels of mainframes. At the low end are the z9 Business Class mainframes, while the z9-109 mainframes are much more expensive and higher-performing systems. Because the encryptions are done on continuous chunks of data, the throughput of a particular encryption scheme will also increase on large data sets; there is less overhead required, so more work gets done.

On data that is 64 bytes in length, a single z9 Business Class processor can perform almost four million DES encryptions per second. This is a throughput of over 240MB/s. A z9-109 operating on the same data performs almost five million DES encryptions, yielding a throughput of over 290MB/s. When working on data sets that are 1 megabyte in length, the z9-BC performs only 536.9 operations per second. However, that means the throughput is 536.9MB/s. The z9-109 sees similar performance gains, performing 652.1 operations per second.

The z9-109 can encrypt up to 230MB/s using 3DES and 308MB/s using AES. Over 575MB/s of data can be hashed using the SHA-1 hardware, while SHA-256 hashes can be calculated at over 375MB/s. A single processor can also authenticate nearly 700MB/s of data using hardware-powered DES-based HMAC algorithms.

IBM has reported that because the typical workload of a z9 mainframe has an extremely large amount of individual tasks running in parallel, adding more processors typically increases the total cryptographic throughput of a system nearly linearly. That means that a 54-processor z9-109 could encrypt upwards of 16GB/s using AES-128 [IBM06a] [IBM06b].

5.3 Design Decisions

While 16GB/s may seem like a large amount of encryption power, it takes 54 processors to reach this level. A 54-processor z9-109 costs many millions of dollars. The performance is quite low for the cost of a whole system. This is because the on-chip cryptography accelerators in the z9 are not the only available source of cryptography acceleration. The z9 series has room for many peripheral cards that include a number of different cryptographic hardware designs. These designs are much more robust, offering features such as secure key encryption (where a secret key is encrypted with a master key so that it can be securely transmitted or stored along with the data), different encryption schemes, and much higher data throughputs.

The on-chip cryptography accelerators in the z9 central processors are primarily for runtime encryption of data in communication settings. Mainframes, for instance, may run database servers that need to send data through SSL tunnels. Similarly, the mainframe may sometimes serve data across a virtual private network. In both of these cases, the encryption is not the primary job of the process. The cryptography accelerator is used, in this case, to speed up computation for these situations. While these accelerators are not the most powerful units that could be built, they do perform much better than software encryption engines.

IBM has produced a large amount of documentation about cryptography on their mainframes for the industries they target. Financial and government institutions are nearly universally required to protect their data with extremely strong encryption. The on-chip cryptography units in the z9 central processors are the first line of defense in this cryptography scheme. Peripheral cards with special-purpose cryptography chips are the next line, and the z9 system has well-integrated support for this method of cryptography. There are special programming calls that are integrated into the OS specifically for cryptography in this domain.

6 Comparison of Cryptography Cores

Even between very different processor designs, the types of accelerators are quite similar. They are nearly all aimed at speeding up communication-based encryption. This, of course, makes sense for a processor like the IXP2850: its entire workload deals with communication-based data. The surprising thing is that processors aimed at server and mainframe markets also primarily accelerate only communication-based cryptography.

One possible explanation for this is the ubiquity of the schemes such as SSL and secure virtual private networking. Many users may not explicitly encrypt their stored data or their programs, but secure communication is used by nearly every network user. These schemes are used by default and are nearly invisible to the user, so performance when using these methods is important.

The primary difference between the processors is the manner in which they access their data. The IXP2850, for instance, deals with streams of network data. This means that the cryptography units are tuned for high throughput on small chunks of data. The z9, on the other hand, can deal with massive pieces of data. Its ability to automate the encryption of long segments of memory is indicative of this type of workload. The UltraSPARC T2 is a more general-purpose server; if it is a web server on the internet, it may have to use many different encryption schemes for communication with a variety of clients. The more general-purpose cryptography unit in this processor is built for this type of workload.

The UltraSPARC T1 seems to be an outlier in this set of processors. While its workload is the same as that for the UltraSPARC T2, its cryptography engine is much weaker. This, according to many sources, was primarily due to design trade-offs. The T1 was stripped of many of its original features in an effort to get it to market and fit it onto a reasonably-sized chip using a 90nm manufacturing process. The next generation processor, the T2, fixed these flaws when space became available on the chip.

The more generalized processors also include the ability to expand the on-core cryptography accelerators. If the user on these systems is worried about cryptography, they can purchase cards like the Sun Crypto Accelerator 6000 PCI-E Adapter or IBM PCIX Cryptographic Coprocessors. These are designed for more general cryptography operation, and offer high performance as well.

7 Conclusions

This paper has given an overview of three modern microprocessors that have onboard cryptography accelerators. The systems come from a range of domains, from a small network router processor to the processors in giant mainframe systems.

Even though the data sets of the systems powered by these processors vary greatly, they all require the ability to securely communicate data over networks. To facilitate this, each system implements a method for increasing the speed of cryptography for networked data. Although the systems all deal with many different kinds of data, their cryptography needs are all quite similar.

The processors all include these cryptography accelerators in the core of the processor itself. This accelerates the encryption of data due to the proximity to the processing elements. Putting these accelerators on the core demonstrates just how important fast, secure network communication is for these processors.

As processor designs become more parallel, and processor designers look for more ways to speed up common operations, we will see more accelerators find their way into CPU designs. Cryptography accelerators are a step in this direction and are an indicator of things to come.

References

[AD04] Todd W. Arnold and Leendert P. Van Doorn. The IBM PCIXCC: A new cryptographic coprocessor for the IBM eServer. *IBM Journal of Research and Development*, 48(3/4), 2004.

[Ara91] Bernard Arambepola. Common VLSI Architecture for a Practically Useful Residue Number System. In *IEEE International Symposium on Circuits and Systems*, volume 5, 1991.

[Bir98] Mark Birman. Accelerating Cryptography in Hardware. In *HOT CHIPS Symposium on High Performance Chips*, 1998.

[Dav02] Robert B. Davies. True random number generators. `http://www.robertnz.net/true_rng.html`, 2002.

[Gol06] Robert Golla. Niagara2: A Highly Threaded Server-on-a-Chip. In *Fall Microprocessor Forum*, 2006.

[Gro06] Greg Grohoski. Niagara2: A Highly Threaded Server-on-a-Chip. In *HOT CHIPS Symposium on High Performance Chips*, 2006.

[Har05] Luddy Harrison. Intel IXP 2850 & IXA Architecture. `http://www.cs.uiuc.edu/homes/luddy/PROCESSORS/IXP2850.pdf`, 2005.

[HF04] Lisa Cranton Heller and Mark S. Farrell. Millicode in an IBM zSeries Processor. *IBM Journal of Research and Developement*, 48(3/4), 2004.

[IBM05] IBM Corporation. *z/Architecture Principles of Operation*, fifth edition, 2005.

[IBM06a] IBM Corporation. *IBM System z9-109 Performance of Cryptographic Operations*, Jan 2006.

[IBM06b] IBM Corporation. *IBM System z9 Business Class Performance of Cryptographic Operations*, May 2006.

[IBM06c] IBM Corporation. *IBM System z9 Enterprise Class*, 2006.

[Int02a] Intel Corporation. *Intel® IXP2850 Network Processor: High-speed, secure content processing on a single chip*, 2002.

[Int02b] Intel Corporation. *Intel® XScale^{TM}Michroarchitecture: Programmers Reference Manaual*, 2002.

[Int05] Intel Corporation. *Intel® IXP2800 and IXP2850 Network Processors Datasheet*, 2005.

[KDBH05] Patrick Kappeler, Lennie Dymoke-Bradshaw, and Pekka Hanninen. *z9-109 Crypto and TKE V5 Update*, 2005.

[Lin05] Chi-Chang Lin. RSA Performance of Sun Fire T2000. `http://blogs.sun.com/chichang1/entry/rsa_performance_of_sun_fire#commen%t-1134700370000`, 2005.

[McG06] Harlan McGhan. Niagara 2 Opens the Floodgates. *Microprocessor Report*, Nov 2006.

[MRK$^+$03] Dave Minturn, Greg Regnier, Jon Krueger, Ravishankar Iyer, and Srihari Makineni. Addressing TCP/IP Processing Challenges Using the IA and IXP Processors. *Intel Technology Journal*, 7(4), 2003.

[NHW$^+$07] Umesh Gajanan Nawathe, Mahmudul Hassan, Lynn Warriner, King Yen, Bharat Upputuri, David Greenhill, Ashok Kumar, and Heechoul Park. An 8-core, 64-thread, 64-bit, power efficient SPARC SoC: (Niagara 2). In *International Solid State Circuits Conference*, 2007.

[Olt06] John Oltsik. Enterprise Tape Encryption Requirements for the Banking Industry. Enterprise Strategy Group White Paper, 2006.

[Sun06a] Sun Microsystems, Inc. Sun Fire T1000 and T2000 Servers Benchmarks. `http://www.sun.com/servers/coolthreads/t1000/benchmarks.jsp`, 2006.

[Sun06b] Sun Microsystems, Inc. *Sun^{TM}Crypto Accelerator 6000 PCI-E Adapter: Security and economy*, 2006.