FA 7.2: A 1Mbs Energy/Security Scalable Encryption Processor using Adaptive Width and Supply

James Goodman, Anantha P. Chandrakasan

Massachusetts Institute of Technology, Cambridge, MA

In many applications, it is desirable to design digital processors that allow a trade-off between the quality of service (QoS) provided and the energy consumed to process a sample. This allows the user to evaluate the application requirements and set the desired quality while minimizing the energy consumption. This paper presents an energy-scalable encryption processor in which the level of security (i.e., quality) and energy consumed to encrypt a bit can be traded-off dynamically, based on demand. Since transmitted data streams can often be partitioned into different priority levels, an energy-scalable processor ensures that important information is adequately protected, while sacrificing some security for low priority data, to reduce total system energy.

The energy-scalable encryption processor in this work is based on a variable-width quadratic residue generator (QRG). The QRG is a cryptographically-secure pseudo-random bit generator that is based upon the work in Reference 1.. The QRG operates by performing repeated modular squarings. The modular squaring is performed using an algorithm based on Takagi's iterated radix-4 algorithm that requires $(\log_2 Q)/2$ iterations to compute the result $P = X \cdot Y \bmod Q$ [2]. The least-significant $\log_2 \log_2 Q$ bits of each result can be extracted and used as a strong pseudo-random source for applications such as a stream cipher or key generator. Unfortunately, common optimizations found in similar modular multipliers used in RSA-based schemes are not applicable to the QRG as the actual result is required at the end of each iteration. Hence, it is not possible to amortize the overhead costs of techniques such as the Chinese Remainder Theorem and Montgomery Multiplication.

Energy-scalable computing requires dynamically-reconfigurable architectures that allow the energy consumption per input sample to be varied with respect to quality. Ideally the quality (i.e., security) should scale much more rapidly than the energy consumption so that relatively small increases in the energy consumption yield significant gains in quality. In the case of the QRG, the quality scales exponentially with the modulus length, while the energy consumption scales polynomially. A fully scalable QRG architecture is developed where the width ($w=\log_2 Q$) can be reconfigured on the fly to range from 64 to 512b in 64b increments (Figure 1). The scalable nature of this architecture can be used to extend the processor to even larger widths with a minimal amount of effort, making it particularly well-suited to increasing security demands. The design makes extensive use of clock gating to disable unused portions of the QRG both before and during the multiplication. Hence the switched capacitance of the QRG is minimized and energy scalability is achieved.

The energy consumption is minimized by reducing the required operating voltage by minimizing the cycle time of the multiplier in a variety of ways: eliminating the need for time-consuming input/output conversion by using an algorithm whose inputs and outputs use the same redundant representation, minimizing the delay of the quotient estimation by using only the signs of the intermediate results that are generated using fast carry-lookahead circuitry, distribution of control and memory among the bit-slices to minimize global interconnect, and using redundant number representations to eliminate time-consuming carry-propagation chains. With these optimizations, a 512b version requires a 2.5V supply to produce a 1Mb stream using a 29MHz clock. The energy consumed is 134nJ/bit (P=134mW).

The large datapath width requires minimization of spurious glitching, achieved by a self-timed gating approach to partition each iteration into 3 separate phases: R_j computation, C_j computation, and P_j computation. The phases are gated by passing the clock through a delay chain, modeling the critical path and tapping it at various points corresponding to the generation of R_j and C_j (Figure 2). Simulations have shown an energy savings of 20% (including the delay chain overhead) using this technique.

Energy scalable computing is achieved using two approaches. First, when less than the maximum width of the multiplier is used, portions of the multiplier are shut down reducing the switched capacitance. Second, when operating at a reduced width, the number of cycles required per multiplication is reduced and therefore the supply voltage can be reduced for a given throughput. The supply is varied using an embedded custom dc/dc converter. The use of an adaptive supply enables substantial reduction of energy consumption as both the throughput and multiplier width are varied (e.g., Figure 3 and Figure 4).

Figure 5 shows a plot of security (in MIPS-years, the amount of time it will take a 1MIPS processor to attack the generator) as a function of energy using shut-down and variable supply approaches. Table 1 summarizes implementation details and experimental results.

For energy-constrained applications a full-size 512b QRG may be too energy intensive. Figure 6 depicts a hybrid system for such energy-constrained applications. The strong pseudo-random source of the QRG can be used to periodically re-initialize a much more energy-efficient linear feedback shift register-based stream cipher (E = 33 pJ/b). In this configuration it is possible to operate the QRG at 1V and a greatly reduced throughput. The hybrid solution consumes 150µW while encrypting data at 1Mb/s using a 1V supply for both the seed generator (QRG) and the LFSR.

Acknowledgments:

The dc/dc converter is designed by A. Dancy. This work is sponsored by DARPA contract DAAL-01-95-K3526.

References:

[1] Blum, L., M. Blum, M. Shub, "A Simple Unpredictable Pseudo-Random Number Generator," SIAM Journal on Computing, v. 15, no. 2, 1986.

[2] Takagi, N., "A Radix-4 Modular Multiplication Hardware Algorithm for Modular Exponentiation," IEEE Transactions on Computers, Aug., 1992.

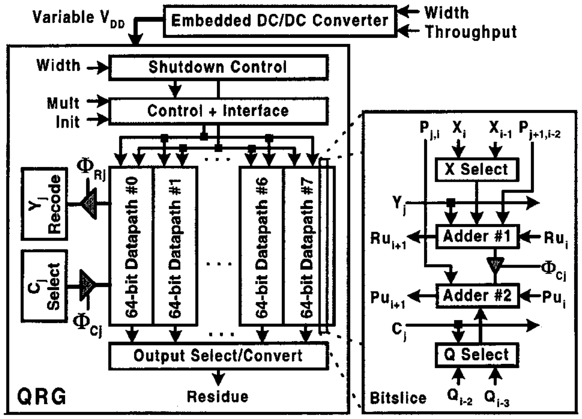

Figure 1: QRG architecture.

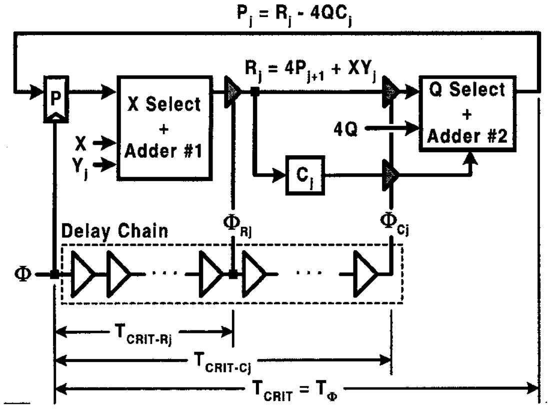

Figure 2: Self-timed approach to minimize glitching.

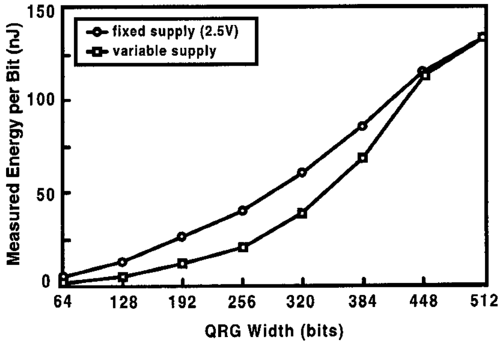

Figure 3: Energy per bit vs. QRG width.

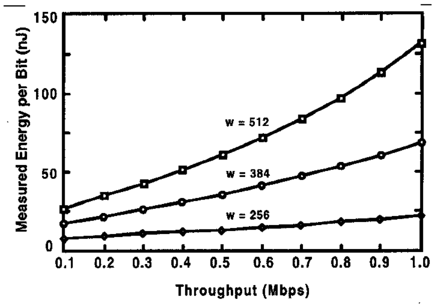

FIgure 4: Energy per bit vs. throughput using shut down and variable supply.

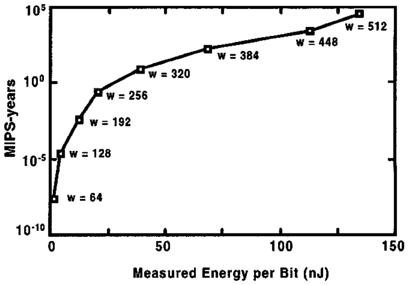

Figure 5: MIPS-years vs. energy per bit at 1Mb/s.

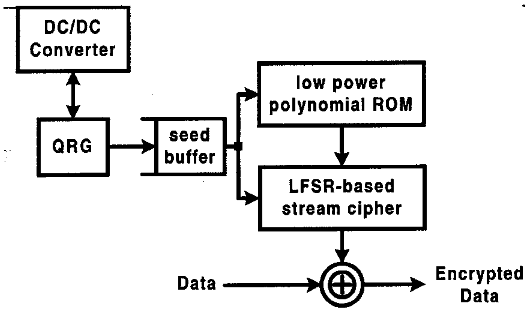

Figure 6: Hybrid system.
Figure 7: See page 422.

Dimensions (QRG only)	6.2mm x 7mm
Device Count (QRG only)	260k
Process	0.6μm DPDM
Threshold Voltages	V_{tP} = -0.88V, V_{tN} = 0.75V
Minimum Operating Voltage	1V (@ 18 kbs, 20 nJ/bit)
$P_{QRG-512}$ @ 1 Mbs (Vdd = 2.5V)	134 mW
P_{Hybrid} @ 1 Mbs (LFSR Seed updated @ 5 kbps)	150 μW

Table 1: Implementation details and experimental results.