A Fully Pipelined, 700MBytes/s DES Encryption Core

Ihn Kim, Craig S. Steele, Jefferey G. Koller
Unifersity of Southern California/Information Sciences Institute
{ihnkim, steele, koller}@ISI.edu

Abstract

Fully-pipelined, 56-bit DES de/encryption and authentication at memory-bus bandwidths is now feasible. We describe a custom, 7 square mm, 120mW core in 4-metal 0.35μm CMOS. Performance allows on-the-fly encryption of 64-bit, 66MHz PCI traffic, and hence typical network traffic. FPGA, synthesized, and 3-metal versions are compared.

1. Introduction

The Data Encryption Standard (DES) is a widely used encryption algorithm amenable to simple hardware implementation, and several commercial and custom chips are available[1],[2]. The complete algorithm of DES involves 16 similar repetitions of a more primitive scrambling operation called a DES *round*. Existing chips implement one round, plus a controller to cycle data through it. For many applications, this is adequate, and higher throughput can be achieved by placing many such units in parallel. This paper reports on an alternative implementation, a fully-pipelined custom VLSI chip able to perform 56-bit DES en/decryption on 64-bit words at 87.5 MHz, i.e. 700MB/s.

The motivation for the design is a scalable computer being built to address quality of service issues in scalable computing. The basic replicated unit in this machine is a triad of nodes: two semi-independent nodes optimized for numeric computation and device access respectively, with an interposed bridge node comprising a bus bridge and embedded control processor. The bus bridge serves as a hardware security barrier between the computational node, which must operate on unencrypted data, and the device node, which does not need to operate on the content of the data blocks it manages and moves around. The DES chip performs on-the-fly encryption/decryption for data moved between the computational and device nodes. The DES encryption and decryption runs at bus speeds (nominally 66MHz) when transferring data between the 64b-wide node busses.

Encryption provides privacy. Equally desirable is authentication: the ability to detect accidental or intentional modification of data. Widely used cryptographic message authentication codes (MACs) contain cyclic data dependencies that severely limit performance of hardware implementations. However, our DES pipeline core enables implemention of a high-performance version of a newer authentication code, the XOR MAC [5], which is amenable to pipelined and parallel implementations. The DES version of the XOR MAC splits each 64 bit word of a message into two, conatenates each half to a 32-bit counter, DES-encrypts the results, and then forms a cumulative XOR over the entire message. Authentication therefore requires twice as many DES operations as encryption. For simultaneous encryption and authentication code computation, we have the option of either implementing three distinct DES pipelines, or performing a three-way interleave of encryption and authentication computations. To allow such interleaving, our DES pipeline implementation carries the (possibly different) DES keys through the pipeline stages along with the intermediate data.

2. Organization and Design

The main steps in each round involve merging data with a key, permuting 64 bits, and repeatedly substituting overlapping 6-bit fields with 4-bit fields [4]. The inherent routing complexity of DES can make silicon area utilization very poor especially for a high-throughput, fully pipelined implementation. Other known DES chips attempt to reduce the routing complexity by using an equivalent representation of DES implementable with reduced internal/external bus widths at the expense of data throughput[3]. To understand the feasibility of implementing a fully pipelined DES algorithm we experimented with four versions of the basic DES round: FPGA, VLSI synthesis, and two custom hand

Table 1: Comparison of implementation of one DES round

Implementation	Size (mm x mm)	Speed (MHz)	Throughput (Mbit/s)
EPF10K100GC503 (FPGA)	-	24.4	260
XC6216 (FPGA)	-	23	200
Synthesis (Cascade)	3.77 x 2.64	-	-
Custom (0.6μm, 3metal)	1.71 x 1.43	58	3712
Custom (0.4μm, 4metal)	**0.52 x 0.46**	**87.5**	**5600**
VMS110 [VLSI tech.]	-	40	280
VMS007 [VLSI tech.]*	-	32	1600
* Fastest commercial chip			

0-7695-0104-4/99 $10.00 © 1999 IEEE

layouts (Table 1). The FPGA version is implemented using an Altera EPF10K100GC503-3DX device with compilation parameters set at fast global logic synthesis and maximum optimization for speed. Although it showed performance competitive with a commercial XC6216 implementation from Xilinx, the single round consumed 30% of the logic cells of this $900 device and was too slow to meet our 66MHz requirement. The synthesized version used the Cascade Epoch placement and route tool. The same VHDL code as the FPGA version was fed into Epoch for synthesis, placement, and route, but it produces too large a silicon requirement for a reasonable cost, mostly due to the complex routing. For the custom hardware implementations, one version is implemented using the Hewlett Packard HP14b process from MOSIS, which provides 1 poly and 3 metals and minimum feature size of 0.6μm. The other uses the HP10b process (same vendor), with 1 poly and 4 metal layers and minimum feature size of 0.4μm. Both led to fabricable implementations, but the one extra metal layer and advanced device technology of the latter process provided a more than 4-fold savings in silicon area, at a speed exceeding our application requirements.

We sought to implement DES with a minimal number of gates to relieve the enormous area requirements and routing complexity of a fully pipelined DES. Fig. 1 shows the circuit elements for each functional block. Pass transistor logic is used to implement all the circuit elements except the S-BOX to minimize the number of transistors and reduce loading. Key shifting is hardwired by multiplexing left-shifted and right shifted key input (shift is 1 or 2 bits) to exploit bit parallelism and reduce shifting delay. The pipeline scheme is implemented using an array of pass transistors instead of conventional flip-flops to reduce the number of gates and the timing requirements. We were able to keep the number of transistors used for a round below 9000.

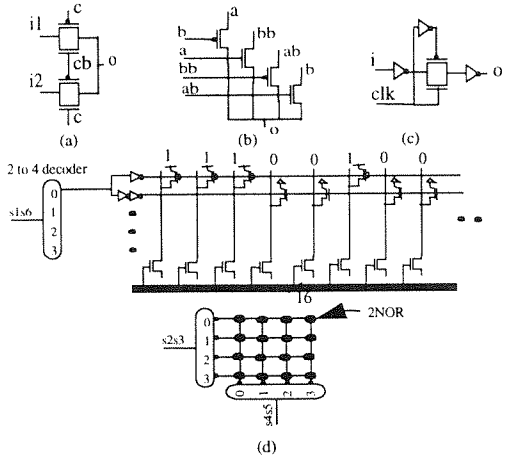

Figure.1 Circuit elements schematic ((a) Key shift, (b) XOR, (c) Pipeline register, (d) S-BOX (16 control signals generated by 4 x 4 array of NOR's)

Pipelining is effected by propagating signals of the same input block over 2 rounds every clock cycle and clocking alternate rounds with different clock phases. The delay budget of 7.5ns per round for 66MHz operation was carefully managed to make this possible. The critical path in a round comprises propagation through the key generation, 48XOR, S-BOX, and 32XOR blocks. The S-BOX is the most critical delay element and also occupies more than 30% of the whole chip area. Figure 1(d) shows our optimization for S-BOX design. Each S-BOX consists of 8 blocks of 256-bit ROM with row and column decoders driven by different inputs in parallel.

3. Simulated Results

The final layout was performed using Berkeley Magic, and MOSIS HP10b design rules. The physical dimension of the layout is 64.2μm x 241.4μm for an S-BOX, 0.52 x 0.46mm for a round and 2.5mm x 2.5mm for the whole 16 rounds. Functionality was checked using Berkeley IRSIM and timing was measured using Epic Powermill. The S-BOX showed less than 1.8ns of delay without output loading and a round showed less than 7ns of delay, leading to a maximum operating frequency of around 85MHz. Power dissipation for the whole core is 115mW.

4. Conclusions and Status

We have shown that a fully pipelined implementation of 56-bit DES, with throughput exceeding 66MHz 64-bit PCI bandwidth, is technically and economically feasible, provided one uses at least a 4-metal process. This is further evidence that 56-bit DES, while adequate for our application, is becoming more and more vulnerable to cracking.

References

[1] Wiener, "Efficient DES Key Search", Practical cryptography for Data Internetworks, W.Stallings ed., IEEE Computer Society Press, pp 31 - 79, 1996.NIST document, "DES V

[2] Validation List", http://csrc.ncsl.nist.gov/cryptval/des/des-val.htm, 1998.

[3] A.G.Broscius and J.M.Smith, "Exploiting Parallelsim in Hardware Implementation of the DES," *Advances in Cryptography-CRYPTO '92 Proceedings* , Springer-Verlag, 1992, pp.367-376.

[4] Bruce Schneier, *Applied Cryptography*, second edition, John Wiley & Sons, 1996.

[5] M. Bellare, R. Guérin, P. Rogaway, "XOR MACs: New Methods for Message Authentication Using Finite Pseudorandom Functions," *Advances in Cryptology – Crypto 95 Proceedings*, Lecture Notes in Computer Science Vol. 963, D. Coppersmith ed., Springer-Verlag, 1995.