Efficient hardware and software implementations for the DES

Marc Davio[1,3], Yvo Desmedt[2], Jo Goubert[2], Frank Hoornaert[2] and Jean-Jacques Quisquater[1]

[1] Philips Research Laboratory, Avenue Van Becelaere, 2, B-1170 Brussels, Belgium;

[2] Katholieke Universiteit Leuven, Laboratorium ESAT, Kardinaal Mercierlaan, 94, B-3030 Heverlee, Belgium;

[3] Université Catholique de Louvain, Batiment Maxwell, Place du Levant, 3, B-1348 Louvain-la-Neuve, Belgium.

Abstract

Importance of DES: NBS, ANSI and ISO (in study) have DES as standards.

The available devices or programs have some tedious properties for an extensive use:

- hardware is expensive or slow, and limited,
- software is slow.

We describe methods for obtaining efficient hardware and software implementations for the DES, *i.e.*:

Hardware

- Cheap and fast hardware,
- all standard modes,
- available for IC library;

Software

- Fast *i.e.* 150 kbit/s (VAX 11/780 without accelerator),
- possibility of using small microprocessors (*i.e.* small programs with relative high speeds).

These efficient designs are obtained using, *e.g.*, tables which are distinct from the tables described by the NBS norm. *This leads to new problems for testing and for certification.*

Tools

General

- DES paper presented at CRYPTO–83,
- further simplications,
- analysis of modes;

Hardware

- Taking the routing problems in consideration;

G.R. Blakley and D. Chaum (Eds.): Advances in Cryptology - CRYPTO '84, LNCS 196, pp. 144-146, 1985.

Software

- Precomputations of some tables,
- using *effectively* the size of words of the processor (8, 16, 32, 48) and the available operations.

Common techniques

CRYPTO–83 paper

- Analytical properties: IP, E and PC_1,
- equivalent representations: iterative DES, modification of the table P.

Feedback modes

Idea

- The idea is to put IP as close as possible at the input of the feedback and IP^{-1} as close as possible to the output of the feedback. This simplifies the routing and the clock circuits in a hardware implementation. For the hardware and for the software, if possible, one can then perform IP, IP^{-1} and DES' in parallel, where DES' is a IP–free DES. Another reason is related to the security of the implementation (key confinement).

Key generation

- Precomputation versus parallel computation.

$P \cdot E$: Analytical expression.

Remark. The key remains constant in the four DES modes.

Software

Good software designs for the DES are obtained using algorithm transformations; for instance, function composition, good match with primitives of the used processor, precomputations. Some time-memory tradeoffs are necessary in order to avoid too expensive tables. The term "too expensive" is relative to a given processor: a small microprocessor has only 8 registers of one byte, 100 bytes of internal RAM and 1000 bytes of program while a big computer is composed of about 10 megabytes of data and program (not using the virtual memory which gives bad performances in very repetitive tasks).

Use of the CRYPTO–83 paper:

- Analytical properties: IP, E, PC_1,
- equivalent representations,
- idea of $P \cdot E$,
- iterative DES,
- modification of the table P.

Special technique exists for E. Another technique is the *precomputation of the key scheduling*. This precomputation requires 96 bytes of RAM for storing the 16 intermediate keys. For some microprocessors, this value is prohibitive: other techniques with precomputations exist with only 16 bytes of RAM but using more complicate procedures.

The use of a two-stage iterative DES model simplifies the program, using the fact that DES is sequential in nature.

The precomputations of tables are useful for obtaining fast software. For instance, each S–box realizes four functions of six variables, *i.e.* requires 256 bits of memory (total: 256 bytes for 8 S–boxes). If we realizes these S-boxes as four groups of two S–boxes, we now need 32 kbytes, but the number of accesses to the S-boxes has been halved.

Other technique is to combine the S–boxes with the permutation P. This technique demands 2096 bytes of memory.

The 48 bit model (see CRYPTO–83 paper) is very useful on computers with words of 48 bits.

A software implementation of the DES on a VAX 11/780 has been made. The exspected speed of about 150 kbits has been obtained. Other implementations are studied. A complete paper will appear in the near future.

Hardware

See the relevant paper in these proceedings.