# HARDWARE SOFTWARE TRI-DESIGN OF ENCRYPTION FOR MOBILE COMMUNICATION UNITS

*Oskar Mencer, Martin Morf, Michael J. Flynn*

Computer Systems Laboratory, Department of Electrical Engineering
Stanford, CA 94305, USA
email: oskar@umunhum.stanford.edu

## ABSTRACT

We explore the design space of Field Programmable Gate Arrays (FPGAs), Processors and ASICs – Hardware-Software Tri-design – in the framework of encryption for hand-held communication units.

IDEA (International Data Encryption Algorithm) is used to show the tradeoffs for the suggested technologies. The measures for comparing different options are: Performance, Programmability and Power ($P^3$). More specifically we use the Performance to Power, or Operations to Energy ratio MOPS/Watt and Mbits/s/Watt to compare processors, FPGAs and ASICs.

We compare the latest Digital Signal Processor (DSP) from Texas Instruments to Xilinx XC4000 series FPGAs. Many DSP-like applications perform very well on FPGAs. We show the benefits and limitations of FPGA technology for IDEA.

## 1. INTRODUCTION

Present research explores the various tradeoffs in applying Field Programmable Gate Arrays (FPGAs), Digital Signal Processors (DSPs), and Application Specific Integrated Circuits (ASICs) to the design of the digital stage of a mobile communication unit. While this case study focuses on encryption for mobile communication, we believe that the resulting methodology gives some insight into the strengths and weaknesses of Processors and FPGAs.

Using FPGAs for computation is a relatively new field. The most popular terms for computation with FPGAs are “Adaptive Computing”, “Configurable Computing”[6], and “Custom Computing Machines”[2]. The most widely used FPGA technologies for Custom Computing Machines are Xilinx XC4000 and XC6200. We are currently using Xilinx XC4000 FPGAs which consist of simple 4-bit lookup tables on a 2D mesh. This allows the programmer to exploit parallelism on the bit and nibble levels.

A Generic Radio

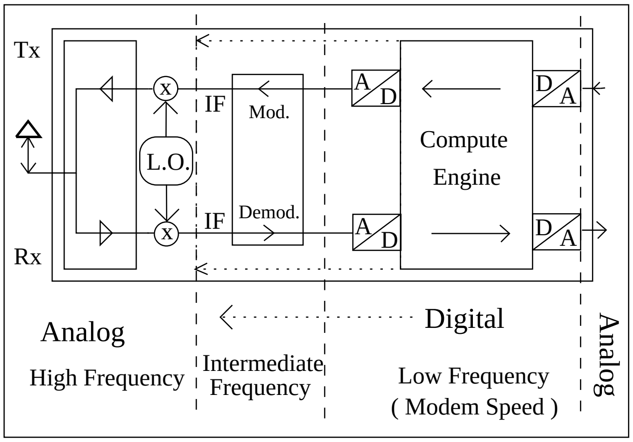

Figure 1: Sound from the microphone (right) goes through an A/D converter, into the digital stage of the pipeline, digital modulation and back to analog. In the IF stage, the bitrate corresponds to around 10 Mbits per second. In the opposite direction demodulation and any additional functionality can be implemented in the digital domain. The dotted lines show how the digital stage is being expanded into higher frequencies.

Performance is the major advantage of FPGAs over conventional processors.It has been shown that for specific applications FPGAs can achieve speedups over processors of 10 to 100 times[1,2,7,8]. The major advantage of FPGAs over ASICs is programmability, which of course has a performance penalty. However, creating a new configuration on FPGAs means designing a new hardware architecture. Therefore, programming FPGA based coprocessors is an order of magnitude more complicated than programming any conventional processor.

We chose IDEA (International Data Encryption Algorithm), a well known encryption algorithm, as the benchmark for this study. The major advantage of choosing a well known application is that there are published designs

This research is supported by DARPA Grant Nr. DABT63-96-C-0106

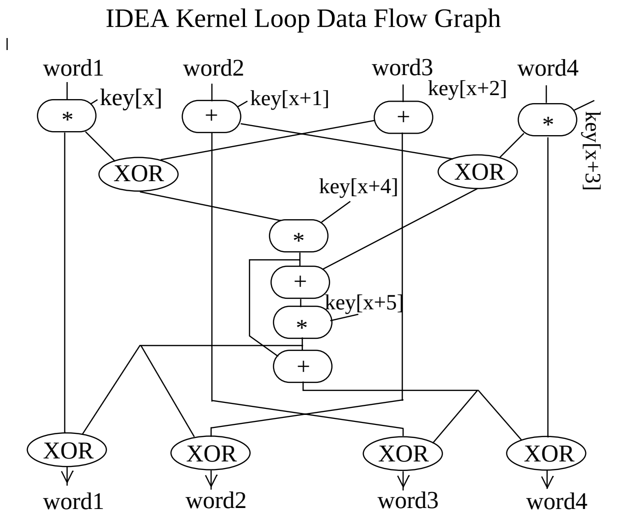

Figure 2: Four 16 bit words of data start in *word1-4*. *key* is a pointer to the array of 52 sub-keys, 16 bits each. The encoded block is returned in word1 to word4 after 8 rounds.

in various technologies which serve as points of reference. IDEA was developed by Xuejia Lai and James Massey at the Swiss Federal Institute of Technology. It was first introduced at EUROCRYPT in 1991 [5]. IDEA encrypts or decrypts 64-bit data blocks, using symmetric 128-bit keys. The 128-bit keys are expanded further to 52 sub-keys, 16 bits each.

Section 2 describes the communication unit as a soft / firmware defined radio. Section 3 introduces our methodology for hardware-software tri-design. Choosing the right technology for a specific application, e.g. IDEA. Section 4 presents an analysis of the results, and section 5 shows our current conclusions about hardware-software tri-design in general and IDEA in detail.

## 2. THE SOFTWARE DEFINED RADIO

Modern radios usually consist of a high frequency analog stage close to the antenna and a low frequency analog stage at the user end. In the midsection the data-stream is handled exclusively in digital form. Figure 1 shows a detailed block diagram of a pipeline implementing a generic cellular phone.

With increasing clock frequencies for digital circuits, the trend is to expand the digital stage more and more into the high frequency domain. The goal of current research in hand-held radios is to increase the functionality of the digital phase of the pipeline to modulation, demodulation, and encryption.

Performance and Power

| DSPs | TI TMX320C6x | DEC SA-110 |
|---|---|---|
| Technology | 0.25 $\mu$m | 0.35 $\mu$m |
| Mbits/s | 53.1 | 32 |
| MOPS | 93 | 56 |
| Clock [MHz] | 200 | 200 |
| Watt | 6 | 1 |

| Designs | XC4000 XL | ASIC “VINCI” |
|---|---|---|
| Area | 3200 CLBs | $107.8mm^2$ |
| Technology | 0.35 $\mu$m | 1.2 $\mu$m |
| Mbits/s | 528 | 180 |
| MOPS | 924 | 315 |
| Clock [MHz] | 33 [1] | 25 |
| Watt | 3.15 | 1.5 |

Figure 3: The table shows the maximal bitrate, Mega Operations per Second MOPS for 4 different technologies. One operation corresponds to one circle in Figure 2. Processors, ASICs and FPGAs use CMOS technology. Power for processors are based on published peak power consumption. Power estimates for FPGAs are based on [13] with pessimistic choice of parameters. While estimating power as proposed in [13] might not be very accurate, it is enough in order to get a sense for the order of magnitude of the result.

Current designs of the digital part of a cellular phone consist of Digital Signal Processors (DSP) and Application Specific Integrated Circuits (ASICs) – both optimized for power. Hardware-software codesign methodology leads to the partition of the workload into DSP code and an ASIC implementation that can meet the real-time and power requirements of the design.

The major drawback of this approach is that the functionality of the ASIC part can not be changed, unless additional functionality is anticipated during the design phase of the ASIC. Adding FPGAs to the design space transforms the design process from hardware-software codesign to tri-design.

## 3. ANALYSIS OF AN ALGORITHM - IDEA

We present a methodology for hardware-software tri-design i.e. selecting the right technology for a specific algorithm.

While the tradeoffs between processors and ASICs are already well understood, using FPGAs for computing is more an art than a science [6].

We use the ratio of Performance to Power, or Operations to Energy as the basis for comparison. More specifically, the measures for evaluating each design option are Operations per Second per Watt or MOPS per Watt, and Mbits/s per Watt.

[1] 33 MHz is a frequency at which we can guarantee that the paper design will work. Power calculations are based on 33MHz.

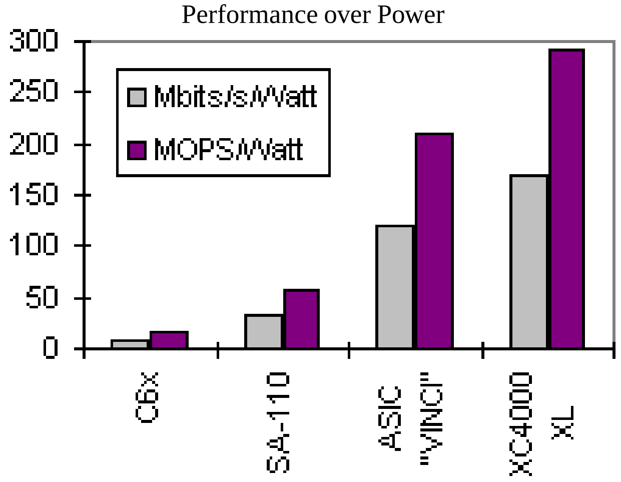

Figure 4: MOPS/Watt determines the power consumption of the technology for a fixed data rate, e.g. 56 Kbits/s modem speed.

First we take a look at the implementation of IDEA on a DSP. We compare two recent DSPs: TMX320C6x from Texas Instruments and StrongARM[12] SA-110 from Digital. The 'C6x [9] DSP is a high performance DSP with 2 multipliers, 4 ALUs and a 4 instructions wide VLIW architecture, requiring 6 Watt at 200 MHz. The StrongARM [12] has only 1 three-stage multiplier and in-order execution, requiring 1 Watt at 200 MHz. Figure 2 shows the kernel loop for one of the eight iterations of IDEA.

Given the available resources on each DSP, the 'C6x from TI takes 30 clock cycles to compute one round of IDEA, compared to 50 clock cycles on the StrongARM.

Figure 3 shows the values we use for comparison of the various technologies.

Next we create a high-throughput paper design for the PCI Pamette[11], an FPGA board developed by DEC. The PCI Pamette consists of 4 Xilinx XC4020 FPGAs. Maximum pipelining and a custom designed konstant coefficient multiplier (KCM in Figure 5) with minimal area requirements, lead to a high-performance and low-power FPGA design. The high performance is achieved by complete loop unrolling of the kernel loop. This was made possible by the fact that all the multiplications in IDEA are multiplying a data word with a word from the key. Maximum pipelining leads to a 56 stage pipeline with a latency of 4 clock cycles per stage, corresponding to the delay of the multiplier in Figure 5. The eight iterations of the IDEA kernel loop fill four Xilinx XC4020 FPGAs (3200 Configurable Logic Blocks).

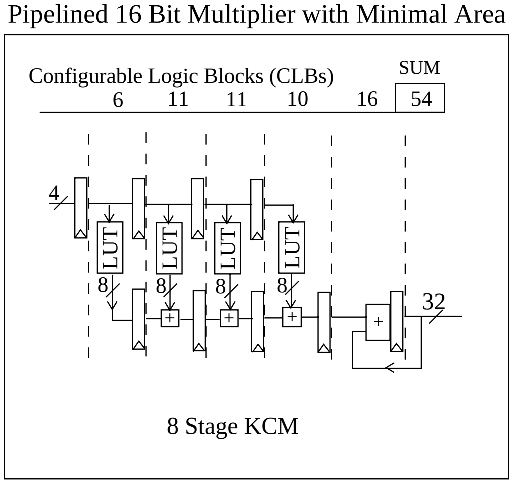
Pipelined 16 Bit Multiplier with Minimal Area
Configurable Logic Blocks (CLBs)
SUM
6
11
11
10
16
54
4
LUT
LUT
LUT
LUT
8
8
8
8
+
+
+
+
32
8 Stage KCM

Figure 5: The figure shows a lookup-table based konstant coefficient multiplier. A 16 by 16 bit multiplication takes 8 clock cycles with a throughput of one 32 bit result every four clock cycles. The number of Configurable Logic Blocks (CLBs) for the datapath described above assumes Xilinx XC4000 cells.

Power estimation was done according to the approach suggested in [13]. In order to improve the fairness of the comparison of power to the peak values used for processors, we used slightly more pessimistic parameters than suggested in [13].

In Figure 3 performance with respect to enciphering with IDEA is given in MBits/s, while the performance with respect to computation of IDEA is given in Mega Operations per Second.

The third step is to look at available ASIC implementations and compare the parameters of the three technologies as presented in Figure 3.

## 4. RESULTS AND CONCLUSIONS

Power consumption is directly proportional to the frequency of the circuit. Therefore the technology with the highest MOPS/Watt and Mbits/s/Watt rating yields the lowest power consumption for a given bitrate.

In our case, the high throughput implementation of IDEA on FPGAs outperforms the ASIC VINCI. The reasons why the FPGAs perform better in our comparison is that we traded latency for throughput, and use a 0.35 $\mu$m CMOS processes compared to 1.2 $\mu$m CMOS which was used for VINCI in 1993.

Figure 4 shows the final comparison of performance over power. Trading latency for throughput results in a very efficient design for FPGAs. The limitation of this design is that we have to load the key into the lookup table prior to enciphering. The latency of loading 128 lookup tables with 16 bytes each, is limited by the available bandwidth to the design. We assume a relatively infrequent change of the encryption key.

The advantages of our methodology are that the values in Figure 3 can be obtained relatively easy. Therefore the methodology can be applied very early in the design cycle to compare the various options for the design.

Due to the heavy use of multiplications, IDEA turned out to be a challenging example to demonstrate the advantages of FPGAs for high throughput and latency tolerant applications.

## 5. FUTURE WORK

Future work will investigate encryption algorithms such as SAFER and Blowfish. During this process we will refine the methodology presented in this paper and be able to compare the different approaches to encryption.

In addition we will focus on optimal multiplier design for FPGAs. Especially we will investigate latency versus throughput tradeoffs, i.e. exploring the benefits of high throughput architectures for various applications.

## 6. ACKNOWLEDGMENTS

The authors acknowledge the support from DEC Systems Research Center, specifically Mark Shand and Stefan Ludwig for help with the PCI Pamette board.

## 7. REFERENCES

[1] N. W. Bergmann, J. C. Mudge, *Comparing the performance of FPGA-based custom computers with general-purpose computers for DSP applications*, Proceedings of IEEE Workshop on FPGAs for Custom Computing Machines, Napa, CA, April 1994.

[2] P. Bertin, D. Roncin, J. Vuillemin, *Programmable Active Memories: A Performance Assessment*, ACM FPGA, February 1992.

[3] A. Curiger, H. Bonnenberg, H. Kaeslin, *Regular VLSI-architectures for multiplication modulo ($2^n + 1$)*, IEEE Journal of Solid-State Circuits, July 1991.

[4] A. Curiger, H. Bonnenberg, R. Zimmermann, N. Felber, H. Kaeslin, W. Fichtner, *VINCI: VLSI Implementation of the New Secret-Key Block Cipher IDEA*, IEEE Custom Integrated Circuits Conference, 1993.

[5] X. Lai, J.L. Massey, S. Murphy, *Markov Ciphers and Differential Cryptanalysis*, EUROCRYPT '91, Lecture Notes in Computer Science 547, Springer-Verlag, 1991.

[6] W.H. Mangione-Smith, B. Hutchings, D. Andrews, A. DeHon, C. Ebeling, R. Hartenstein, O. Mencer, J. Morris, K. Palem, V. Prasanna, H. Spaanenburg, *Configurable Computing*, IEEE Computer Magazine, December 1997.

[7] R. J. Petersen, B. L. Hutchings, *An Assessment of the Suitability of FPGA-Based Systems for use in Digital Signal Processing*, 5th International Workshop on Field-Programmable Logic and Applications, Oxford, England, Aug. 1995.

[8] M. Shand, P. Bertin, J. Vuillemin, *Hardware Speedups in Long Integer Multiplication*, Computer Architecture News, 1991.

[9] J. Turley, H. Hakkarainen *TI's New 'C6x DSP Screams at 1600 MIPS* Microprocessor Report, Vol 11, Num 2, Feb. 17, 1997

[10] S. Wolter, H. Matz, A. Schubert, *On the VLSI Implementation of the International Data Encryption Algorithm IDEA*, IEEE International Symposium on Circuits and Systems, April 1995.

[11] WWW: The PCI Pamette FPGA board at DEC Systems Research Center *http://www.research.digital.com/SRC/pamette/*

[12] WWW: StrongARM SA-110 at Digital Semiconductor *http://www.digital.com/ semiconductor/strongarm/strongar.htm*

[13] Xilinx Application Brief *A Simple Method of Estimating Power in XC4000XL/EX/E FPGAs* XBRF 014, June 30, 1997