Towards Fully Reconfigurable Multimedia Platforms*

Marco Antonio Simon Dal Poz Ricardo Bittencourt Vidigal Leitão
Regis Rossi Alves Faria Roseli de Deus Lopes Marcelo Knörich Zuffo
Laboratório de Sistemas Integráveis
Departamento de Engenharia de Sistemas Eletrônicos
Escola Politécnica da Universidade de São Paulo, Brazil
{mdalpoz,ricardo,regis,roseli,mkzuffo}@lsi.usp.br

Abstract

This paper describes a novel approach to implement multimedia access platforms: reconfigurable oriented co-design. Nowadays we observe a rapid and mutant evolution of multimedia standards and its de facto adoption into many consumer electronics, such as audio/video appliances, digital set-top boxes for TV access networks, mobile technologies and portable digital assistants (PDAs). Among the disadvantages of current multimedia access platforms, we can mention the relative high cost, the very short obsolescence cycles, the communication and computational performance limitations, the large sizes of software components, and system architectures that do not match future multimedia standards requirements. To overcome these, we propose the use of a reconfigurable computing architecture to implement a family of multimedia access platforms. As an example, we present an implementation of a multimedia access platform that runs a web browser (HTML 3.2 compatible) over a 32-bit RISC microprocessor, described using less than 256 Kbytes of data to represent software and hardware components on a FPGA (Field Programmable Gate Array).

Keywords

Multimedia platforms, reconfigurable computing, reconfigurable systems, hardware/software co-design, network computers, on demand reconfiguration

1 INTRODUCTION

We observe in the last 10 years a large-scale adoption of multimedia standards into consumer electronics products. Both the need to assure compatibility among manufacturers and the fast growing multimedia consumer market respond as major causes for this.

Consumer electronics industry and mass multimedia content providers – *such as TV networks, Web/Internet, the music and the movie market* – pose hard requirements on storage, processing, delivery and interaction capabilities.

Multimedia appliances more and more have to deal with multiple media and channel formats, and to cope with bandwidth restrictions, stronger interaction requirements, higher quality media display, intellectual property management, and shorter obsolescence cycle.

We propose a novel approach for the fore coming multimedia architectures, based on the paradigm of reconfigurable computing and modular multimedia circuitry, with obvious advantages in exploring algorithm parallelism and enabling a functional upgradability, therefore improving the lifetime for consumer terminals and also reducing the time to market, both for manufacturers and content providers.

As a proof of concept, our proposal includes the deployment of a family of multimedia platforms: a series of devices with increasing complexity and functionalities, based on a modular and incremental reconfigurable architecture.

The first member of this family should be able to offer essential multimedia services, such as web browsing and e-mail facilities, making it possible to alter the status of digital exclusion noticeably in the lower-income classes, specially in the developing countries.

Subsequent members should incorporate more complex multimedia functionalities, such as CATV tuning, video decoding, networking and interactivity with service providers.

This paper addresses the concept of convergence of reconfigurable computing and multimedia platforms to guide a strategic approach towards the design and implementation of fully reconfigurable multimedia platforms, which we foresee as a strong candidate for future consumer electronics multimedia platforms standard.

2 MULTIMEDIA STANDARDS

Multimedia platforms receive, process and display multimedia data (audio, video, text). There are several formats for encoding media streams and produce packets of data, which travels through a network and can be read, decompressed, processed and rendered for proper display.

* Published at the First Ibero-American Symposium in Computer Graphics - SIACG, Guimarães, Portugal, 2002.

Popular formats include JPEG, TIFF, GIF, BMP for image; WAV, AIFF, MP3 for audio; AVI, MOV, MPEG-1, MPEG-2, for video, and more recently DivX, based on MPEG-4 codecs.

MPEG (Motion Picture Experts Group) [MPEG] is an international workgroup of ISO/IEC in charge of developing standards for compressing, processing and coded representation of multimedia (video, audio, metadata and combinations).

By complying with a common format, devices from different manufacturers can communicate and exchange media (interoperability). MPEG objectives mainly address the reduction of the number of parameters/information contained in images/video/audio (without loss of perceived quality) and to describe a bit stream format for coded video/audio, assuring high quality, low cost, "reproducibility" and, of course, interoperability.

MPEG-1 (1992) [MPEG1] describes technologies for compressing, storing and recovering of video/audio in digital networks. The popular MP3 audio format derives from MPEG-1 Audio part. MPEG-2 (1994) [Watkinson99] addressed digital television, and brought several improvements over MPEG-2. Current satellite TV and DVD video coding and transporting schemes are based on MPEG-2 format.

MPEG-4 [MPEG4] came to establish standards for multimedia applications, and introduced better codecs for audio/video. MPEG-4 is still ongoing, and more tools are to be released. So far, few manufacturers have introduced MPEG-4 based products (hardware/software) into the market. The whole standard is supposed to cover a wide range of applications, resolutions (media quality) and bandwidth restrictions, depending on the target market. The DivX video format, which is rating popularity in the Internet, derives from MPEG-4 video format. Such new codecs exhibit very good media quality with relative less storage demands, and their adoption promises to increase in the near future.

More recent initiatives are not oriented anymore to media compression, but to content description, indexing and management (e.g., MPEG-7 and MPEG-21, under development).

We envision a future with extended usage of media browsers, a larger offer of networked multimedia content providers, and high quality audio/video terminal display.

Given this evolving scenario, reconfigurable computing is the most attractive approach to implement multimedia terminal (user) platforms, and respond fast to the introduction of new standards, formats and applications.

3 RECONFIGURABLE COMPUTING

For many digital systems, general-purpose processors do not provide enough processing power for satisfactory operation in real time environments. Specialized computational resources are frequently used for improving the available performance, such as Digital Signal Processors (DSPs) and application specific processors [Lanner95].

One relatively new approach is to improve the available processing power of dedicated systems by implementing application specific circuits in FPGAs (Field Programmable Gate Arrays). Although still more expensive than customized circuits, FPGAs provide a simplified project environment with a lesser cost.

The most attractive feature in operating with FPGAs is its capability of re-synthesize an entire new digital system on the fly. The component cost can be amortized compared with several circuits in operation in the same device. In this way, the reconfigurable computing and, mainly the reconfigurable hardware, can be considered as a method that tries to mix the programmability of systems based on processor with the reconfigurability of the hardware on the FPGA.

Advances in chip integration technologies have made possible the entire *System on a Chip*. This kind of systems integrates programmable processors and dedicated hardware components, allowing entire systems to be integrated into one only chip in such a way that performance requirements for the applications that have multifunctional nature can be achieved.

Projects with such characteristics are extremely complex, because decisions taken in the beginning of the project tend to cause great impact over the final performance, and also because the IC design has a fast market insertion.

As new standards are released and new applications appear, it is important that the device (in which applications should be implemented) exhibit programmability, reconfigurability and scalability. With these three properties, a posterior higher processing power (for running new applications with higher computational cost) can be reached only with a small adjust in the parameters of the architecture and code re-synthesis.

With these alternatives in hand, it becomes easier for the Engineers to create chips and systems based on a kind of architecture that supplies efficient and optimized solutions for a determined application domain, by just tuning and adjusting the reconfigurable elements presents in the architecture and in the system for holding the application being created.

The Reconfigurable Computing field appears as a technological alternative that can customize itself, with easiness for the use of a general-purpose processor while keeping the advantages of dedicated hardware performance.

With this inherent speed and adaptability, the reconfigurable computing has a great potential to be explored specially in applications that need a good performance with rapid prototyping, reducing designing time and cost in project phases (development, implementation and testing), and adjusting actual working parameters of most modern applications.

It is a suitable substrate technology to support our developments towards the desired degree of achievements.

4 RECONFIGURABLE MULTIMEDIA PLATFORMS

The main approach is to conceive and design multimedia platforms based on reconfigurable technology. We are developing a family of microprocessed multimedia terminals based on reconfigurable hardware. All devices in the proposed family derive from a general architecture with four relevant main components:

- application specific microprocessor(s)
- distributed memory
- a memory crossbar
- I/O interface modules

Figure 4.1 depicts these components.

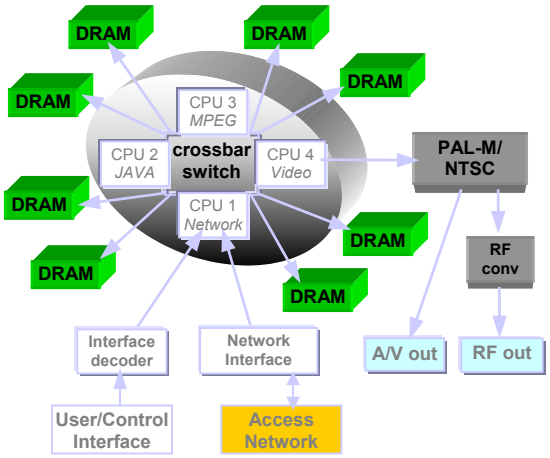

Figure 4-1 General Architecture for the Multimedia Reconfigurable Platforms

Some initiatives are described in the literature proposing reconfigurable circuitry for multimedia systems, but frequently target designs of individual components or co-processors [Mirsky96] [Miyamori98].

Microprocessors are tailored to cover specific sets of functionalities [Sato91], counting on an optimized set of instructions and balance of performance *versus* area.

Four types of processors are under research and development, to cover (1) MPEG audio/video decoding, (2) network computing, (3) JAVA (applications) computing, and (4) audiovisual output rendering.

Multimedia devices may require more than one microprocessor on the same die (up to 4) to provide a desired set of functions.

A typical example of such a device is a set-top box: the subscriber's terminal decoder vastly employed by the cable and satellite TV networks all over the world.

Newer generations of set-top boxes exhibit more intelligent features, requiring an extended degree of processing power to deal with multimedia (audio, video, text) decoding and display. For example: digital video decoding and interactive capability to perform bi-directional transactions with services providers requires at least a MPEG video decoder and an application processor, besides some video rendering and user interface circuitry.

Next generations of set-top boxes are expected to be fully digital, and act as a home gateway for a variety of services through diverse digital networks. The next section introduces a digital set-top box we are developing based on reconfigurable computing.

5 THE RECONFIGURABLE DIGITAL SET-TOP BOX

Currently most available Digital Set-Top Boxes works with MPEG-2 data streams. Some of them have an internal MPEG-2 chip decoder (hardware ASIC[1] implementation), and others use a fast processor running a MPEG-2 software decoder.

The hardware implementation usually has the best performance, which means that the decoder operates with low clock frequencies, low power consumption and small chip area, obtaining a low cost decoder, but has no upgrade capability.

In this way, when cable TV providers adopt the MPEG-4 (or better) video compression standard, all digital set-top boxes based on MPEG-2 hardware decoder will not still be compatible with the new contents being broadcasted through the network. So, a new hardware investment will be necessary to change all subscribers' set-top boxes, which represents a high cost (for the provider or for subscribers).

The software implementation usually has poor performance, which means that the decoder demands a fast processor (higher power consumption) and larger chip area, obtaining a high cost decoder, but has the advantage of being upgradeable.

However, the upgrade capability of the software implementation is limited. Since the complexity of video compression and decompression algorithms is growing, the performance of the software tends to decrease. This effect usually is reduced by the adoption of a very fast processor, which implies in a high cost digital set-top box. To improve the performance of the software, the processor should have specific instructions designed for running the new software decoders. But, since a processor is implemented in an ASIC, further modifications are impossible without hardware exchange, which is also too much expensive.

The solution for this problem of performance *versus* cost can be obtained with the reconfigurable computing. Reconfigurable computers have part of the hardware implemented in FPGAs, which implies that parts of the hardware can be modified after installed in the subscriber's home and while it is working. If the digital set-top box uses at least one FPGA in the hardware design, the internal logical functions can be modified *a posteriori*.

The solution being proposed here is to implement a simple RISC core microprocessor in a fast FPGA and, when the cable TV provider decides to change the video

[1] *Application Specific Integrated Circuit*

compression method, a new core can be sent and written in the FPGA, producing an upgrade in all the set-top-boxes working within that cable TV network. With this reconfigurable capability, new instructions can be implemented in the processor, and so the performance of the software will be much better than with pure software upgrade [DalPoz00].

The implementation of a processor in an FPGA can become practical if the following requirements are satisfied:

- The FPGA should be capable of working in high clock frequencies. This is necessary because the performance of a digital circuit implemented in an FPGA is usually worse than the one implemented in an ASIC;
- The FPGA should have a large number of gates. This is required to allow several future modifications in the digital design;
- The FPGA should not be very expensive. Of course, the FPGA implementation of a processor will be more expensive than an ASIC, since FPGAs will require better microelectronics technology (smaller dimensions) to achieve similar performance.

The problem of the high cost of such a FPGA can be overcome if we consider the long lifetime that this kind of implementation may have, supporting future types of video coding techniques. Another way of cost reduction is through large utilization of FPGAs, which can reduce their price.

First prototypes were designed for *stand-alone* module implementations, focusing on testing of concept, algorithmic performance/efficiency, hardware synthesis and general improvements. Second generations address modular integration, aiming the synthesis of distinct application circuits in the same die, implementing designed connectivity structures, testing, and applying improvements.

The next figure shows the first board, designed for synthesis and testing of individual multimedia components and other peripheral modules, such as video composition and I/O interfaces.

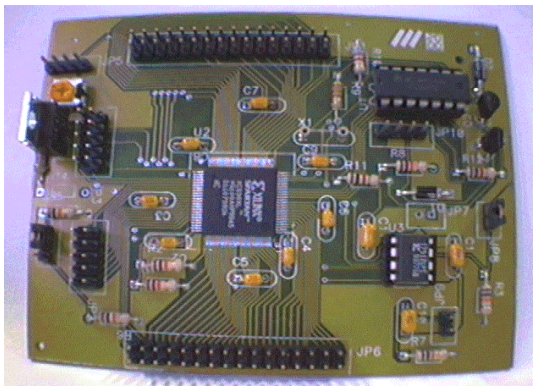

Figure 5-1 Digital SET TOP BOX I – Module Synthesis/Testing board

In the center one can see the FPGA chip. Hardware components are designed, simulated and described in VHDL[2] and programmed into the FPGA. Peripheral circuitry includes access busses, signal conditioning and I/O ports.

Already tested components include MPEG-2/4 decoding algorithms, video signal processing and composition circuits. With performance in mind, several algorithm descriptions and configurations [Feig92] [Loeffer89] [Arai88] were designed, simulated and then field tested to determine best performance/area ratios.

Prior to the development of the full digital set-top box platform, however, we have selected a simpler platform to prototype: the "web box", the first actual operational platform of the proposed family.

The "web box" was conceived to run a small web browser application and offer email facilities. The whole implementation of hardware and software into the FPGA is described using less than 256KBytes.

Next sections introduce the web browser application, named *FlyBrowser*, and the central microprocessor designed for the "web box" and subsequent platforms.

5.1 The FlyBrowser Application

The FlyBrowser is a reduced-core web browser application, specifically developed to run in the proposed platform, and optimized for its architecture and restrictions.

The purpose of this application is the creation of a new system for mobile Internet browsing, with better characteristics than the ones found in the market, providing better graphical quality and the capacity of www navigation without necessity of content reformulation.

To achieve this goal, it is necessary to notice the limitations of the cellular telephony systems, which are typically systems with very limited memory and low processing power. These characteristics are incompatible with the desired goal: the presentation of web pages has always high computational cost, which needs much memory and fast processors.

Besides the above problems, cellular telephones have a very limited bandwidth. Web pages, mainly those with high resolution display, can become too large, and this would become cause an excessive slow down in the transmission of its content for the final user.

To solve above problems, a solution proposed in this work is to decouple rendering and presentation. A server, which acts as intermediate between the viewer device and the Internet, is responsible for all the computational heavy tasks, allowing the cell phone to be implemented with low complexity technology and with low power consumption.

[2] Very High Speed Integrated Circuit Hardware Description Language

In the traditional model, the mobile device is responsible for the connection with the Internet, for downloading the page and the images, for the rendering of the content, for the conversion of the pictures to a format compatible with the display device, and for the final presentation of the content (including the user interface) [W3C].

This does not happen in the model proposed in this work. The server not only makes the download of the web pages web, but also is responsible for the conversion of the parameters of HTML logical formatting in simple instructions of text and image positioning, in such a way that the mobile device has only the function of implementing the user interface. As an extra function, the server has the capacity of data compression of the web page before sending them to the mobile device, reducing the needed bandwidth, and speeding up the connection.

As the rendering task is in charge of the server, we have two extra advantages over the traditional model. The first one is that there is no more need of creation of specific content for mobile Internet, since the server can convert common pages of the web. The second is that new protocols can be added without the need of update in the client (the viewer device). The same hardware can be used to visualize any new version of HTML that can be released, thus reducing the cost for the final user [Fly00].

The figure 8.2 below shows the FlyBrowser "in action", depicting three rendered web pages.

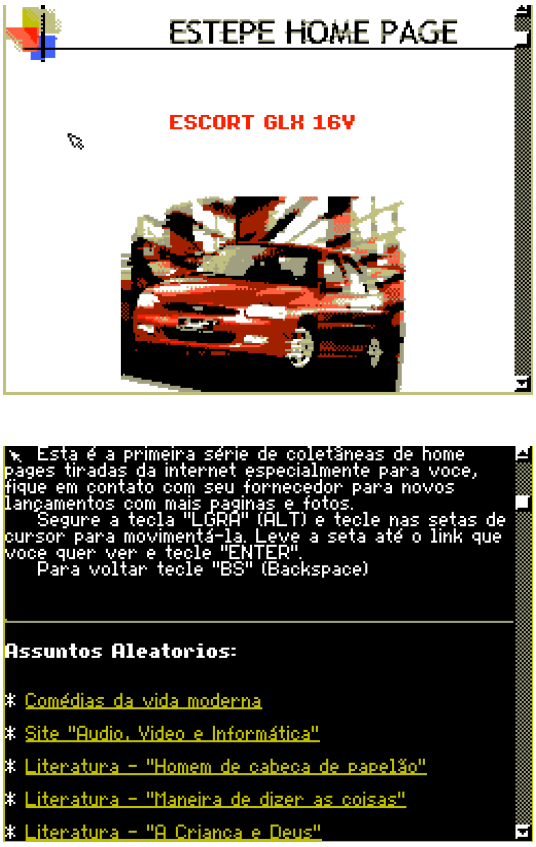

Figure 5-2 – FlyBrowser: an example of three rendered web pages

5.2 A 32-Bit Multimedia RISC Microprocessor

A microprocessor design is needed to implement the desired set of high-end controllers and multimedia processors.

A microprocessor based on a 32-bit RISC core is under development to serve as computer platform for most specific applications. Design requirements include the right selection of the instruction set and internal organization, to promote performance and match applications needs.

Figure 5.2 (below) shows the microprocessor internal organization proposed for the "web box" platform, which will incorporate an extremely reduced core web browser application and email facilities written in Z-80 compatible instruction set [Zilog].

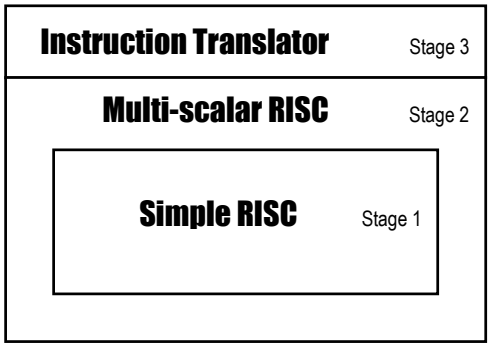

Figure 5-3 – Internal Organization of a 32-Bit Multimedia RISC Microprocessor

In the first stage, a simple RISC core has been implemented based on a DLX compatible architecture [Patterson96].

In the next stage, a multi-scalar RISC processor is in charge of instruction scheduling, control of scoreboards, and of solving hazards occurrences [Fauth95].

An instruction translator (stage 3) is needed to permit other instruction set based applications (as FlyBrowser's Z80 [Zilog] instructions) to run on the machine, converting CISC instructions to RISC ones [RiscCisc].

Stage 1 design was described in VHDL, simulated and synthesized in FPGA using Xilinx development tools[3], and has been successfully completed. Stage 2 and 3 are under development.

The following figure shows the multi-scalar architecture proposed.

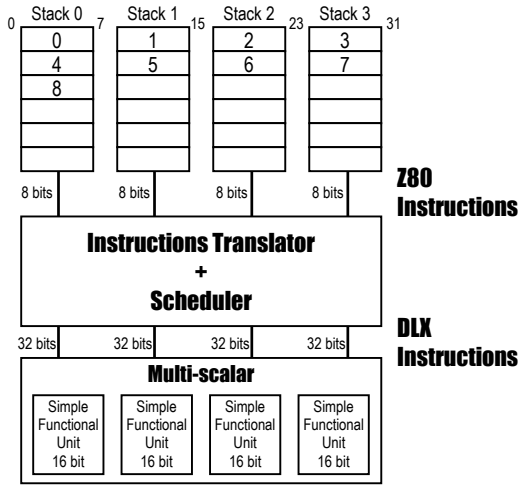

Figure 5-4– Multi-Scalar Architecture

To read external instructions it is used a program memory divided in 4 blocks. In a prefetch stage, the 4 blocks are accessed simultaneously, taking 32-bits at once. Internally, however, one can have 8, 16 or 32-bit instructions [Athanas93].

5.3 Design Methodology and Implementation Issues

Prior to the selection of the current architecture, it was necessary to determine the application software requirements, critical bottlenecks, and to set desired performance specifications.

Statistical analysis of instruction usage by the target application (FlyBrowser) revealed that it is a program with massive demand for Load/Store instructions (over 51% of time execution), which strongly suggests the usage of a RISC core [Huang95].

Figure 5-5 shows a frequency distribution of instructions during the FlyBrowser execution.

To generate a RISC processor to run web browsing, email and other multimedia basic applications efficiently, it is desired to match the following design criteria:

- execution of 1 instruction per clock cycle (for most instructions)
- supporting speeds in the range 20-50 MHz
- having Z-80 compatibility.

With an actual Z-80 architecture it would be very difficult to match these requirements.

[3] Xilinx Foundation, v.4.1 and ISE 4.1i integrated design environments

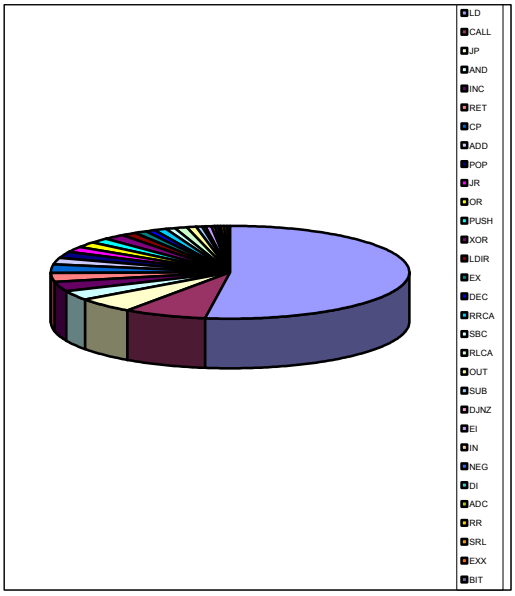

Figure 5-5 – Frequency of execution of instructions in the FlyBrowser

Additionally, the desired processor should possess some features considered necessary, such as 32-bit busses, separate busses for data and instruction, a reduced instruction set and be Z80 compatible.

The architecture of a DLX microprocessor [Patterson96] was then proposed as main core. It permits the implementation of an efficient pipelined RISC processor, match all specification criteria, and in addition, is a public domain processor (open core).

The FPGA design of the DLX microprocessor was successfully accomplished. To overcome delays introduced by FPGAs restrictions, it was implemented a 4 clock phase scheme, permitting to achieve the desired 1 instruction executed per clock cycle [Holmer93].

Performance measurements for the DLX design in simulations are around 20 MIPS.

6 FUTURE WORK

Next multimedia user's devices should be able to incorporate functionalities such as MPEG-2/4/7/21 decoding, and multiple interfacing capabilities (NTSC/Digital TV out, IEEE FireWire I/O, LAN, Modem, 5.1 Audio, etc.).

In this way, we foresee a timeline for the development and deployment of a series of multimedia devices, members of this family.

The series of multimedia devices foreseen in the timeline (figure 8-1) are:

1. Web Box, offering basic web browsing and email (proof of concept)
2. TV Digital Set-Top Box, with video out and MPEG-2 decoding [Watkinson99]

3. interactive Multimedia Box, an improved digital set top box, with MPEG-4 codecs and MPEG-7 basic browsing capabilities
4. 3DiMM Box, aiming a device capable of rendering immersive experience, with audio/video multiprojections.

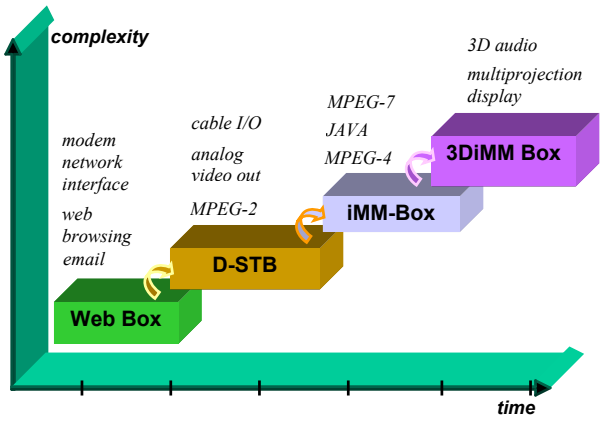

Figure 6-1 Reconfigurable Multimedia Platforms: implementation timeline

The Digital Set-Top Box, under development, is a multimedia computer platform which incorporates current set-top box functionalities, is indicated for home gateway once interaction and multimedia processing is added and available for hosting several services, such as web browsing, home network gate, entertainment, security systems, electro domestic control, on line and real time informational services, etc.

The second generation of DSTB, under design phases, incorporates new functionality, aiming the migration of newer multimedia codecs, communication and network standards, multimedia browsing and synthesis of their respective circuits.

Finally, a fourth generation of devices is under planning, and will require the blending of distinct multimedia technologies under the same computing infrastructure. We envisage the concept of immersive multimedia experience, by making use of virtual reality tools, multiprojection display and spacialized audio.

As with previous implementations, the 3Dmm Box aims the offer of high quality digital services in a near future, for a reasonable cost, spreading over the society all benefits of the digital revolution

Adopting reconfigurable hardware technology will naturally establish a model of automatic upgrading for installing new and reconfigure current applications without user intervention, bringing benefits for digital service networks, and shorting installation times for services such as video/audio on demand, interactive services, on line transactions, real time data consultation, etc.

7 CONCLUSIONS

We are proposing the use of reconfigurable computing as approach for designing and manufacturing of next generations of multimedia platforms, counting on advantages such as easier hardware upgradability, improved performance, transparent and user independent system upgrade, and longer time life.

It was introduced the concept of reconfigurable multimedia platforms, and presented a timeline for development and deployment of a series of such devices, with increasing complexity.

We have developed a reduced core web browser (less than 20KBytes of code) and an implementation of a 32-bit DLX microprocessor in FPGA.

Together, they are the main components of the first proposed reconfigurable multimedia platform: the "web box". A digital set-top box is currently under development.

The mass introduction of low cost *web boxes* and digital set-top boxes are expected to have a great social impact, specially in the low-income classes.

8 ACKNOWLEDGEMENTS

We would like to express our gratitude to José Bolivar and Javier Higuita (undergraduate research assistants) for their valuable participation in the project.

9 REFERENCES

[DalPoz00] M. A. S. Dal Poz, J. E. Aedo Cobo, W. A. M. Van Noije, M. K. Zuffo "A Simple RISC Microprocessor Core Designed for Digital Set-Top Box Applications". Proceedings IEEE ASAP, pages 35-44, July 2000

[Feig92] E. Feig and S. Winograd. "Fast Algorithms for the Discrete Cossine Transform". *IEEE Transactions on Signal Processing, vol. 40, no. 9,* pages 2174-2193, Sep. 1992

[Loeffer89] C. Loeffer, A. Ligtenberg, and G. S. Moschytz "Pratical fast 1D DCT algorithms with 11 multiplications". *Proceedings ICASSP 1989*, pages 988-991, 1989.

[Arai88] Y. Arai, T. Agui, and M. Nakajima "A Fast DCT-SQ scheme for images". *Transactions on IEICE, vol. E-71, no.11,* pages 1095-1097, Nov1988.

[Fauth95] A. Fauth, J. Van Praet and M. Freericks. "Describing Instruction Set Processor using nMLx". *Proceedings European Design and Test Conference,* Paris, Mar 1995.

[Sato91] J. Sato et al "An Integrated Design Environment for Application Specific Instruction Processor". *Proceedings ICCD*, pages 414-417. 1991.

[Holmer93] B. Holmer "Automatic Design of Computer Instruction Sets" *University of California at Berkeley*, Ph.D. dissertation, 1993.

[Huang95] I. Huang, A. M. Despain "Synthesis of Application Instruction Sets". *IEEE Transactions*

on CAD of Integrated Circuits and Systems, vol.14, no.6, pages 663-675, June 1995.

[Athanas93] Peter M. Athanas and Harvey F. Silverman "Processor reconfiguration through instruction-set methamorphosis". *IEEE Computer*. vol. 26, no.3, pages 11-18, Mar 1993.

[Fly00] FlyBrowser: An Alternative Solution for Mobile Internet Implementation <http://www.lsi.usp.br/~ricardo/flybrowser/>

[W3C] W3C: World Wide Web Consortium <http://www.w3.org/>

[Patterson96] D. A. Patterson and J. N. Hennessy "Computer Architecture: a Quantitative Approach". *Morgan Kaufmann Publishers*, second edition, 1996.

[Lanner95] D. Lanner, J. Van Praet, A. Kiflt et al. "CHESS - Retargetable Code Generator for Embedded DSP Processor". *Kluwer Academic Publisher,* pages 85-103, 1995.

[MPEG1] ISO/IEC JTC1/SC29/WG11 MPEG, International Standard ISO 11172, coding of moving pictures and associated audio for digital storage media up to 1.5 Mbits/s, 1992

[Watkinson99] J. Watkinson, MPEG-2, Focal Press 1999

[RiscCisc] Arquitectura RISC vs CISC. <http://ww.azs.uam.mx/publicaciones/enlinea2/1-2.htm>

[Zilog] Z80 CPU User Manual. <http://www.zilog.com>

[MPEG] MPEG Homepage <http://www.mpeg.org/>

[Mirsky96] E. Mirsky; A. DeHon. "MATRIX: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources". FCCM'96 – IEEE Symposium on FPGAs for Custom Computing Machines, April 17-19, Napa, 1996.

[Miyamori98] T. Miyamori; K. Olukotun, "REMARC: Reconfigurable Multimedia Array Coprocessor", FPGA'98 (as poster paper), 1998.

[MPEG4] MPEG-4 Video Coding Standard <http://wwwac.hhi.de/mpeg-video/>