dispTEC: industry-university collaboration in Costa Rica on the field of embedded systems and digital signal processing

Michael Gruner-Monzón  
Electronics Engineering School  
Tecnológico de Costa Rica  
E-Mail: grunermonzon@gmail.com

Bayron Pérez-Vega  
Electronics Engineering School  
Tecnológico de Costa Rica  
E-Mail: bayron241@yahoo.es

Pablo Alvarado  
Electronics Engineering School  
Tecnológico de Costa Rica  
E-Mail: palvarado@itcr.ac.cr

Abstract—Projections indicate that in 2010 there will be three embedded devices for each human being on Earth, and the market keeps growing. The particular case of embedded applications that use digital signal processing is of special interest for Costa Rica, since the engineers required are being already trained at the local universities and the engineering services market has a high added value.

This work describes the experience of a collaboration project between RidgeRun Engineering, an US-American company installed in Costa Rica and “Tecnológico de Costa Rica” (TEC), a Costa Rican State University, who decided to work together on a plan to improve the skills of the students in the embedded DSP algorithm implementation.

The know-how has been transferred from RidgeRun to TEC, and in response, detailed documentation on all necessary steps has been generated. This documentation allows students without any experience in Unix nor in DSP algorithm design to produce at the end of one semester applications running in an ARM EVM based on Texas Instruments’ OMAP-L138 heterogeneous system-on-chip.

Index Terms—embedded systems, industry-university collaboration, OMAPL138, DSP

I. INTRODUCTION

Embedded computers are today ubiquitous, since microcontroller-based systems are found inside almost any sophisticated device from specialized medical equipments to airborne systems, from single production space satellites to massive production consumer electronics including cellphones, TV-sets, game consoles, PDA, etc. In 2008 about 90% of all computing devices were estimated to be embedded systems [6]. Projections indicated that in 2010 the number of embedded devices reached 16 billions (almost 3 embedded devices per human being on Earth) and will be over 40 billions by 2020 (5-10 embedded device per human being). The areas of Digital Image Processing (DIP) and Digital Signal Processing (DSP) are rapidly progressing and provide the fundamentals for many of the above mentioned systems. The costs involved in embedded DSP product development are high, due in part to the high license fees of the specialized software and the expensive engineering skills which have to be scaled up by the time required to finish a marketable product.

The companies producing microcontrollers and systems-on-chip (SoC) have aggressive strategies to make the use of their products attractive. This involves providing tools and libraries that shorten the time-to-market, as well as adding new appealing technical features. One attractive alternative for these companies is to support Free and Open Source Software (FOSS) that simplifies the implementation of particular applications, improves the software maintainability, and accelerates the debugging, test and validation processes by exposing the code to a large community of software developers. Furthermore, the source code is always available for continuous reviews and as real coding example in the training phases of new developers.

On the other hand, the use of FOSS in education, research and services to the community has several advantages including reduced costs in software licenses, and simplified transfer of research results into teaching and community service activities. Additionally, FOSS serves to position universities in the international community which, at the same time, serves to attract grants to support further research and development. Finally, FOSS offers a common ground for industry and universities to cooperatively integrate software development efforts into a field of shared interest.

Reasons for industry-university collaboration in Costa Rica

It is estimated that over 70% of the signal processor revenue is related to the communications sector, with a major fraction corresponding to mobile phone devices. The market of DSP processors is expected to grow in other segments of consumer electronics, and also in the industrial market, prominently medical electronics, for which increasing demands in design innovation exist [2].

New products for use at home, mobile or in-vehicles are characterized by refined graphics, audio and data capabilities, which demand better hardware support to execute DSP algorithms. In [2] it is estimated that the current market volume of 1.5 billion DSP processors shipped (US$5.3 billion) will increase the next five years at a rate of 11% in average per year.
The previous facts justify why, for instance, initiatives like ARTEMIS [8] in Europe have been created to integrate the development strategies for embedded systems among all related sectors, including universities and industry.

In Costa Rica, sectors related to the development of embedded systems are growing. Companies like HP/Procurve, Avionyx, Teradyne, Softtek, InbandSoft and RidgeRun Engineering, among others, have already on-going development processes of embedded systems, some of which make use of advanced methods of digital signal and image processing. It is also known that Intel makes a clear effort to strategically position their Atom processor into the embedded systems market, effort which undoubtedly sooner or later will reach their Costa Rica site.

This opens the opportunity for Costa Rican universities to establish collaborations with the industry to support academic research that directly finds application in the products being developed. At the same time that research provides the academic environment where future engineers and graduate students develop those technical skills required by the industry. This kind of relationships are worldwide encouraged as a way to effectively attach the knowledge creation in the academia to the product development in the industry; this way, the universities directly contribute to the local economy [7].

Goals of the project

This work presents the results of a one-year project between "Tecnológico de Costa Rica"1 (TEC) and RidgeRun Engineering2 who started a collaboration activity to increase the skills of future engineers in the fields of embedded systems and DSP, and also to search for a common ground that will facilitate future research for industrial innovation.

The development of advanced digital signal and image processing applications for embedded systems is strongly tied to costly hardware and software tools and libraries. The limited budgets available at the local universities limit the acquisition of such tools, and on the other side, the costs associated which them will also elevate the development and final product costs. Closed source libraries additionally make it difficult or even impossible to modify available algorithms to fit specific requirements of academic research endeavors or in product development.

The current project is aimed to build the necessary know-how at the Electronics Engineering School at TEC of the complete development chain for advanced DSP/DIP embedded applications as encountered nowadays in the industry, making it available for research and teaching activities at the university. The deliverables are in form of open documents and FOSS libraries and software for embedded DSP applications. Specifically the BSD License have been chosen for its convenience for all parts.

This has several advantages:
1) TEC improves the qualifications of its alumni, who are trained in the complex issues of the development and implementation of embedded applications and DSP/DIP algorithm design.
2) TEC is enabled to provide actualization courses to alumni and other engineers requiring the skills for embedded DSP/DIP application development.
3) The international visibility of RidgeRun and TEC increases through the availability of documentation an FOSS, on which they hold the copyrights.
4) The industry directly supports research activities in the areas of DSP/DIP, as the response times of TEC is competitive, and the algorithms produced are compatible with industry standards. Students of all levels (bachelor, masters, and doctoral studies) find research opportunities as part of these projects.
5) The industry-university cooperation on advanced DSP/DIP research improves the chances to find technologies with exploitable intellectual property.
6) Particularly of interest for Costa Rican State Universities, the industry provides high-end hardware and software infrastructure.

One of the specific objectives of the project has been to provide an effective way to teach students how to deal with design and implementation of DSP applications in embedded systems. Two candidates with no previous experience in this field were chosen and asked to document every single difficulty they encountered while learning all sides of the development chain of embedded software for DSP. The results of this one-year experiment are invaluable, since they allow to develop teaching methodologies that will considerably accelerate the introduction of students to the embedded-DSP software development processes using state-of-the-art hardware platforms. This results are being incorporated into the course EL-5805 Digital Signal Processing at TEC. The rest of this article describes the experiences of the first year of collaboration between RidgeRun and TEC. These experiences have been grouped into four chronological phases:
1) Hardware and software basic platform.
2) Setting up the embedded system environment.
3) Working with an heterogeneous architecture.
4) A simplified DSP software development approach.

II. First phase: Hardware and Software

In order to develop a teaching strategy for DSP processing in embedded systems, a versatile platform has been chosen: a Zoom evaluation board (EVM) based on Texas Instruments OMAP-L138 System-on-Chip (SoC). This SoC has a heterogeneous architecture combining an ARM9 General Purpose Processor (GPP) and a TI C674x Digital Signal Processor (DSP) [11]. The choice on a Texas Instruments’ processor was influenced by the fact that TI holds approximately a 65% share of the digital signal processor market, making them the leaders of the sector. Additionally RidgeRun’s partnership

---

1Tecnológico de Costa Rica, known as TEC, is a Costa Rican State University devoted to the engineering fields.
2RidgeRun Engineering is an US-American company specialized in embedded software. They own an engineering office in Costa Rica and are strategic partners of Texas Instruments (TI).
with TI helped to provide the initial hardware required for the collaboration.

The Zoom EVM allows the development of teaching projects that expose students to real problems in the implementation of embedded systems applications and additionally the challenges of efficient programming for DSP processors.

Figure 1 depicts the hardware connections chosen for the embedded software development. This is a fairly standard configuration that allows fast application development: on a host computer the code is cross-compiled for the target board (in this case, the Zoom EVM) and the executables are directly stored in a file system remotely mounted via NFS on the embedded board. The board is controlled by the developer using a terminal connected through the serial port.

At the lab, the file system server is shared among several terminals, while at home the students may use one single machine as terminal and file system server.

JTAG and TI’s Code Composer Studio usage will be included in future activities to be executed by the students in the labs of the Electronics Engineering School at TEC.

Since the project goals included the use of FOSS, the operating system chosen to develop the applications is GNU/Linux, for which there is a plethora of free development tools available, and a large supportive community of developers. RidgeRun’s Software Development Kit (SDK) for TI’s OMAP-L138 was chosen to install all required software on the board and on the host computer.

RidgeRun and Texas Instruments partnership provided all hardware and software necessary for the project.

III. SECOND PHASE: SETTING UP THE EMBEDDED SYSTEM ENVIRONMENT

Software Development Kit

A major difficulty towards the use of the selected development platform in teaching activities is the generalized lack of knowledge among students regarding Unix systems in general, and of GNU/Linux in particular. The majority of students are used to, and almost dependent on Graphical User Interfaces to operate their computers in single user operating systems. Therefore, the jump into a terminal-based concept to operate the embedded system and the use of multi-user operating system has represented a non-negligible obstacle. However, most students agree that Unix knowledge is necessary for their future work as engineers and are willing to work hard to develop the necessary technical skills.

It is a challenge to provide an environment that provides students a not so steep learning curve to cope with all details of a –for them new– terminal-based development environment and additionally to deal with the complexities of the digital signal processing theory.

The first task of every student willing to work at home is to install the development environment. Therefore, the first task was to prepare a detailed description for the installation of RidgeRun’s SDK on a GNU/Linux machine. This first experience was time-consuming, but exposed the industry-university collaborative nature in this project: RidgeRun’s support was absolutely necessary to deal with a large set of unknown variables and problems encountered by the assistant students who never worked with similar systems. At the same time the student assistants detected many issues that a common engineering student will face when installing a GNU/Linux system from scratch into the embedded system:

1) security concepts in Unix, (changing file and directory rights)
2) Makefiles usage,
3) seek valuable information in large text outputs of the compilation/installation processes
4) configuration of a TFTP server,
5) configuration of an NFS server,
6) installation of software in the embedded system,
7) access to the embedded system via a terminal,
8) the bootloader concept,
9) eventually, usage of secure shells to remotely access a host computer, when it differs from the terminal computer.

All the accumulated experience helped to prepare a detailed How-to available at [5], and additionally the documentation process provided feedback to RidgeRun used to improve their SDK, by adding missing dependencies, and making adjustments to the default configurations to accelerate the process of a first installation.

This phase of the project took four months. Following the generated documentation, a user with experience in desktop GNU/Linux needs no more than two days to complete the whole process (including the download times of more than 1 GB of installation packages from RidgeRun and Texas Instruments). A student with no previous GNU/Linux experience needs from one to two weeks to get the EVM up and running (after he or she installs a GNU/Linux host machine).

A first “Hello-World” application

The SDK installation is a process that depends on many factors out of the control of the student installing it. It is the goal of the SDK packager to provide the easiest possible way to install all software required to develop new applications for
the embedded platform. Until now the student mainly learns in the first three layers of Bloom’s taxonomy of educational objectives [4].

The next task for the student will be to produce a first “Hello world” application. Again, the lack of previous Unix knowledge represents an obstacle to be reduced by providing detailed descriptions of all the necessary steps:

- how to create directories
- using a text editor to edit code (not an Integrated Development Environment)
- dealing with environment variables (for instance, PATH)
- cross-compilation
- the remote file-system concept
- running in the embedded system a program using the terminal

The creation of this “Hello-World” application took almost two weeks for the student assistants to master, and with the produced documentation it takes no more than two hours for an unexperienced student to configure and run. This is also an activity moving in the knowledge-comprehension-application levels of Bloom’s taxonomy, but it provides the fundamentals to go up into the higher-order thinking skills.

One of RidgeRun’s strengths is their experience in GStreamer application development, which eliminates the need of dealing with low-level device management and provides a large collection of coders and decoders of media streams. However, for future research projects it is desirable to be able of producing low-latency signal processing applications, for which the OMAP-L138’s DSP-Processor is ideal. GStreamer’s design is not compatible with the low-latency concept, since it was devised as a framework for processing multimedia streams where latency is not a critical issue (for instance, audio and video playback) [9]. Therefore, as a next step, a simple framework for capturing, processing and playing audio data using ALSA (Advanced Linux Sound Architecture) was developed. ALSA implements a relatively low-level abstraction layer just above the audio hardware. Hence, the implemented programs are parameterized to the particular hardware platform used. It is therefore necessary to port a low-latency abstraction layer (for instance Jack [3]) to the current Zoom EVM and to any future embedded platform used in this collaboration, since it would allow to isolate hardware details from the DSP application development.

The development of three ALSA-based applications (capture, playback and duplex) took one month. The documentation and code is also freely available at [5], and allow a student without experience to concentrate on the signal processing algorithms.

IV. THIRD PHASE: TI’s eXpressDSP SOFTWARE STACK

Figure 2 shows a simplified diagram of the components involved in the development of embedded applications for Texas Instruments’ heterogeneous architectures. The whole concept aims software interoperability, which is necessary to reduce development costs of DSP applications. Algorithms implemented following this standard (e.g. multimedia encoders) by a producer are easily integrated by clients into their applications, independently of the exact model of SoC used, and without dealing with internal issues of the algorithms or their source code.

This eXpressDSP framework [10] allows the application running on the GPP processor to use the higher computational capabilities of the DSP processor by means of several conceptual layers, each specialized on different needs. Some layers are application-agnostic (like XDAIS), while others are specialized in video, imaging, speech and audio (VISA) algorithms (such as XDM). The complete concept has more components and subcomponents as the ones shown here.

Although the framework advantages are clear, a student with limited knowledge in advanced software engineering concepts and without previous experience on DSP algorithm development needs two to three months until the underlying concepts are understood and the methods to manually create all required source code files, paths, Makefiles, wizards, etc. are under manageable. Even though TI’s Code Composer Studio simplifies these tasks, the initial idea has been to provide a FOSS-based methodology to enable the students to work at home.

Even though a detailed guide on how to create an eXpressDSP compliant algorithm has been created as part of this project (see [5]), the authors believe the whole concept is relatively complex for a student walking his/her first steps in the DSP world, and therefore not exactly appropriate for the first implementations of DSP algorithms in an introductory course.

It is worth mentioning that one of the major difficulties in this phase was to filter out the huge amount of documentation available on the topic in the form of wikis and technical documents. There are many obsolete documents still around, and no way to notice they are obsolete until newer documents are found. Additionally, the software specifications
and implementations of this standard evolve fast, making the documents and guides produced in the current project obsolete in just a few months. How to cope with this real-world fast-paced product evolution in an university program is a difficult problem: the classical paradigm of providing just the theoretical fundamentals seems in this case inappropriate, as practical training is undoubtedly required. Some trade-offs need to be found.

V. FOURTH PHASE: EASY START WITH C6RUN

A new concept was proposed by Texas Instruments in September 2010, less than three months before the conclusion of the project. The new C6Run Software Development Tool [1] is ideal for the first steps in DSP programming. This concept does not ensure interoperability of modules, as it does the eXpressDSP framework, and therefore the software engineering complexities associated with the latter are absent, as well as the implementation issues related to the management of separate codec servers, codec engines and the applications using them.

Additionally, the C6Run framework partially hides the complexities associated with the communication of the GPP and the DSP on the SoC. The students can easily decide which code should be run in each processor structuring their software using functions, and employing the provided tools for the compilation.

The integration of the C6Run tools with RidgeRun’s SDK took about two months: The software versions required by this new tool were not fulfilled by the SDK in use (which by the time was nine months old), and a complete upgrade was necessary.

VI. CONCLUSIONS AND FUTURE WORK

The embedded system market steadily grows and has already reached the industry installed in Costa Rica. Modern embedded systems present in multimedia applications, medical devices, or vehicle stability controls need to process large amounts of data. For Costa Rica, the specialized market of embedded DSP algorithm design and implementation is attractive, but the development of this area will be possible only if the local universities produce engineers with all necessary skills.

The development of DSP algorithms for real applications has two components: the theoretical foundations that support the processing task, and the implementation issues. An introductory course in DSP has to find a balance between both components. Traditionally, rapid prototyping tools have been employed to develop the practical experience, but the students are not confronted to the implementation issues found in real-time DSP applications. Rapid prototyping tools permit to illustrate the theoretical principles, and therefore should be considered as support for teaching the theoretical foundations.

The implementation of DSP algorithms is growing in complexity, and nowadays there is a large spectrum of implementation platforms that include general purpose computers, embedded systems, reconfigurable hardware (e.g. FPGA) and even ASIC design. The latter platforms are complex enough to deserve their own courses at a graduate level; the former platforms are ideal to introduce the pre-graduate students to real-time DSP algorithm design.

The collaboration between RidgeRun and TEC provided in one year the necessary know-how to incorporate in the EL-5805 Digital Signal Processing course the development of DSP algorithms using Texas Instruments’ OMAP-L138 heterogeneous SoC, a state-of-the-art device with a general purpose processor and a digital signal processor. Additionally, as part of the cooperation activities a common working framework has been established that allows the development of advanced DSP algorithms requiring advanced research at the university on the theoretical foundations component, with deliverables provided in an industry accepted standard format.

Further refinements of the SDK for the students is necessary to simplify the first steps. At this time the authors are working on two image processing applications that may serve as platform to also incorporate the usage of embedded systems in the introductory CE-5201 Digital Image Processing and Analysis course at TEC.

ACKNOWLEDGMENTS

This project has been supported by RidgeRun Engineering and Texas Instruments, who provided the necessary hardware and the financial support for the student assistants. At Tecnológico de Costa Rica the project 5402 1360 2801 is supported by the Electronics Engineering School.

The authors want to particularly thank Todd Fischer, Esteban Zúñiga and Marta Peraza for their continuous support in all technical and administratıve tasks during the project.

REFERENCES


