Acronym of the proposal	COACH
Title of the proposal in French	**C**onception d'**A**rchitecture sur FPGA par **C**ompilation et synt**H**èse
Title of the proposal in English	Architecture Design on FPGA by Compilation and Synthesis
Theme	■ 1 ⊡ 2 ⊡ 3 □ 4 ⊡ 5 ●: secondary theme
Type of research	□ Experimental Development ■ Industrial Research □ Basic Research
Type of scientific project	□ Platform

Total requested funding	1195931 €	Project Duration	36 months

Contents

1 Executive summary

The market of digital systems is about 4,600 M$ today and is estimated to 5,600 M$ in 2012. However the ever growing applications complexity involves integration of heterogeneous technologies and requires the design of complex Multi-Processors System on Chip (MPSoC).
During the last decade, the design of ASICs (Application Specific Integrated Circuits) appeared to be more and more reserved to high volume markets, because the design and fabrication costs of such components exploded, due to increasing NRE (Non Recurring-Engineering) costs. Fortunately, FPGA (Field Programmable Gate Array) components, such as the Virtex5 family from XILINX or the Stratix4 family from ALTERA, can nowadays implement a complete MPSoC with multiple processors and several dedicated coprocessors for a few Keuros per device. Many applications are initially

captured algorithmically in High-Level Languages (HLLs) such as C/C++. This has led to growing interest in tools that can provide an implementation path directly from HLLs to hardware. Thus, Electronic System Level (ESL) design methodologies (Virtual Prototyping, Co-design, High-Level Synthesis...) are now mature and allow the automation of a system-level design flow. Unfortunately, ESL tool development to date has primarily focused on the design of hard-wired devices i.e. ASICs and ASSPs (Application Specific Standard Product). However, the increasing sophistication of FPGAs has accelerated the need for FPGA-based ESL design methodologies. ESL methodologies hold the promise of streamlining the design approach by accepting designs written in C/C++ language and implementing the function straight into FPGA. We believe that coupling FPGA technologies and ESL methodologies will allow both SMEs (Small and Medium Enterprise) and major companies to design innovative devices and to enter new, low and medium volume markets.

The objective of COACH is to provide an integrated design flow, based on the SoCLib infrastructure [8], and optimized for the design of multi-processors digital systems targeting FPGA devices. Such digital systems are generally integrated into one or several chips, and there are two types of applications: They can be embedded (autonomous) applications such as personal digital assistants (PDA), ambiant computing components, or wireless sensor networks (WSN). They can also be extension boards connected to a PC to accelerate a specific computation, as in High-Performance Computing (HPC) or High-Speed Signal Processing (HSSP).

The COACH environment will integrate several hardware and software technologies:

Design Space Exploration: The COACH environment will allow to describe an application as a process network i.e. a set of tasks communicating through FIFO channels. COACH will allow to map the application on a shared-memory, MPSoC architecture. It will permit to easily explore the design space to help the system designer to define the proper hardware/software partitioning of the application. For each point in the design space, metrics such as throughput, latency, power consumption, silicon area, memory allocation and data locality will be provided. These criteria will be evaluated by using the SoCLib virtual prototyping infrastructure and high-level estimation methodologies.

Hardware Accelerators Synthesis (HAS): COACH will allow the automatic generation of hardware accelerators when required. Hence, High-Level Synthesis (HLS) tools, Application Specific Instruction Processor (ASIP) design environment and source-level transformation tools (loop transformations and memory optimisation) will be provided. This will allow further exploration of the micro-architectural design space. HLS tools are sensitive to the coding style of the input specification and the domain they target (control vs. data dominated). The HLS tools of COACH will support a common language and coding style to avoid re-engineering by the designer.

Platform based design: COACH will handle both ALTERA and XILINX FPGA devices. COACH will define architectural templates that can be customized by adding dedicated coprocessors and ASIPs and by fixing template parameters such as the number of embedded processors, the number of sizes of embedded memory banks or the embedded the operating system. However, the specification of the application will be independant of both the architectural template and the target FPGA device. Basically, the following three architectural templates will be provided:

1. A Neutral architectural template based on the SoCLib IP core library and the VCI/OCP communication infrastructure.
2. An ALTERA architectural template based on the ALTERA IP core library, the AVALON system bus and the NIOS processor.
3. A XILINX architectural template based on the Xilinx IP core library, the PLB system bus and the Microblaze processor.

Hardware/Software communication middleware: COACH will implement an homogeneous HW/SW communication infrastructure and communication APIs (Application Programming Interface), that will be used for communications between software tasks running on embedded processors and dedicated hardware coprocessors.

The COACH design flow will be dedicated to system designers, and will as much as possible hide the hardware characteristics to the end-user.

To reach this ambitious goal, the project will rely on the experience and the complementariness of partners in the following domains: Operating system and communication middleware (TIMA, LIP6), MPSoC architectures (TIMA, LAB-STICC, LIP6), ASIP architectures (INRIA/CAIRN), High Level Synthesis (TIMA, LAB-STICC, LIP6), and compilation (ENS Lyon/LIP).
The COACH project does not start from scratch. It stronly relies on the SoCLib virtual prototyping platform [8] for prototyping, (DSX, component library), operating systems (MUTEKH, DNA/OS). It also leverages on several existing technologies: on the GAUT [16] and UGH [12] tools for HLS, on the ROMA [35] project for ASIP, on the SYNTOL [26] and BEE [11] tools for source-level analysis and transformations and on the XILINX and ALTERA IP core libraries. Finally it will use the XILINX and ALTERA logic and physical synthesis tools to generate the FPGA configuration bitstreams.

The COACH proposal has been prepared during one year by a technical working group involving the 5 academic partners (one monthly meeting from january 2009 to february 2010). The objective was to analyse the issues of integrating and enhancing the existing tools and tecnnologies into a unique framework. Most of the general software architecture of the proposed design flow (including the exchange format specification) has been define by this working group. Because the COACH project leanes on the ANR SoCLib platform, it may be described as an extension of the SoCLib platform.

Two major FPGA companies are involved in the project: XILINX will contribute as a contractual partner providing documentation and manpower; ALTERA will contribute as a supporter (see letter page 55) providing documentation and development boards. These two companies are strongly motivated to help the COACH project to generate efficient bitsreams for both FPGA families. The role of the industrial partners BULL, THALES, NAVTEL-SYSTEM and FLEXRAS is to provide real use cases to benchmark the COACH design environment and to analyze the designer productivity improvements.

Following the general policy of the SoCLib platform, the COACH project will be an open infrastructure, available in the framework of the SoCLib server. The architectural templates, and the COACH software tools will be distributed under the GPL license. The VHDL synthesizable models for the neutral architectural template (SoCLib IP core library) will be freely available for non commercial use. For industrial exploitation the technology providers are ready to propose commercial licenses, directly to the end user, or through a third party.

Finally, the COACH project is already supported by a large number of PMEs, as demonstrated by the ”letters of interest” (see Annex B), that have collected during the preparation of the project : ADACSYS, MDS, INPIXAL, CAMKA System, ATEME, ALSIM, SILICOMP-AQL, ABOUND Logic, EADS-ASTRIUM.

2 Context and relevance to the call

Embedded systems (SoC and MPSoC) became an inevitable evolution in the microelectronic industry. Due to the exploding fabrication costs, the ASIC technology (Application Specific Integrated Circuit) is not an option for SMEs (Small and Medium Enterprises). Fortunately, the new FPGA (Field Programmable Gate Array) components, such as the Virtex5 family from XILINX, or the Stratix4 family from ALTERA can implement a complete multi-processor architecture on a single device. But the design of embedded system is a long and complex task that requires expertise in software,

software/hardware partionning, operating system, hardware design, VHDL/Verilog modeling. Only very few SMEs have these multiple expertises and are present on the embedded system market.

The major objective of COACH is to provide to SMEs an open-source framework to design embedded systems on FPGA devices by system designers.

The COACH project will leverage on the expertise gained in the field of virtual prototyping with the SoCLib platform, to propose a new design flow based on a small number of architectural templates. An architectural template is a generic, parameterized architecture, relying on a predefined library of IP cores. Besides using a specific collection of general purpose IP cores (such as processors cores, embedded memory controllers, system bus controllers, I/O and peripheral controllers), each architectural template can be enriched by dedicated hardware coprocessors, obtained by high level synthesis (HLS) tools. During this project, the COACH partners will develop three different architectural templates:

1. An ALTERA architectural template based on the ALTERA IP core library, the AVALON system bus and the NIOS processor.

2. A XILINX architectural template based on the XILINX IP core library, the PLB system bus and the Microblaze processor.

3. A Neutral architectural template based on the SoCLib IP core library and the VCI/OCP communication infrastructure.

The proposed design flow starts from a high level description of the application, specified as a set of parallel tasks written in C, without any assumption on the hardware or software implementation of these tasks. It lets the system designer in charge of expressing the coarse grain parallelism of the application, gives the designer the possibility to explore various mapping of the application on the selected template architecture, and offers a high predictability of results with respect to cost and performance objectives.
When this interactive, system level, design space exploration is completed (converging to a specific mapping on a specific version of the selected architectural template), the rest of the flow is fully automated: The synthesisable VHDL models for the various hardware components, as well as the binary code for the software running on the embedded processors, and the bit-stream to program the the target FPGA will be automatically generated by the COACH tools.

The strength of the COACH approach is the strong integration of the high-level synthesis tools in a platform based design flow supporting virtual prototyping and design space exploration. Most building blocks already exist (resulting from previous projects): the GAUT or UGH synthesis tools, the MUTEKH or DNA embedded operating systems, the ASIP technology, the DSX exploration tool, the MWMR hardware/software communication middleware, the BEE parallelisation tool, as well as the SoCLib library of systemC simulation models. They must now be enhanced and integrated in a consistent design flow.

In HPC (High Performance Computing), the targeted application is an existing application running on a PC. The COACH framework helps designer to accelerate it by migrating critical parts into a SoC embedded into an FPGA device plugged to the PC PCI/X bus.

The second objective of COACH is to extend the framework to HPC.

This will allow SMEs to enter HPC market for the applications that are unadapted to the current GPU based solutions.

In summary, the COACH project is clearly oriented toward industry, even if most technology building blocks have been previously developed by academic laboratories.

Segment	2010	2011	2012
Communications	1,867	1,946	2,096
High end	467	511	550
Consumer	550	592	672
High end	53	62	75
Automotive	243	286	358
High end	-	-	-
Industrial	1,102	1,228	1,406
High end	177	188	207
Military/Aereo	566	636	717
High end	56	65	82
Total FPGA/PLD	4,659	5,015	5,583
Total High-End FPGA	753	826	914

Table 1: Gartner estimation of worldwide FPGA/PLD consumption (Millions $)

2.1 Economic and societal issues

Microelectronic components allow the integration of complex functions into products, increases commercial attractivity of these products and improves their competitivity. Multimedia and tele-communication sectors have taken advantage from microelectronics facilities thanks to the developpment of design methodologies and tools for embedded systems. Unfortunately, the Non Recurring Engineering (NRE) costs involded in the design and manufacturing ASICs is very high. An IC foundry costs several billions of euros and the fabrication of a specific circuit costs several millions. For example a conservative estimate for a 65nm ASIC project is 10 million USD. Consequently, it is more and more unaffordable to design and fabricate ASICs for low and medium volume markets.

Today, FPGAs become important actors in the computational domain that was originally dominated by microprocessors and ASICs. Just like microprocessors, FPGA based systems can be reprogrammed on a per-application basis. For many applications, FPGAs offer significant performance benefits over microprocessors implementation. There is still a performance degradation of one order of magnitude versus an equivalent ASIC implementations, but low cost (500 euros to 10K euros), fast time-to-market and flexibility of FPGAs make them an attractive choice for low-to-medium volume applications. Since their introduction in the mid eighties, FPGAs evolved from a simple, low-capacity gate array to devices (ALTERA STRATIX III, XILINX Virtex V) that provide a mix of coarse-grained data path units, memory blocks, microprocessor cores, on chip A/D conversion, and gate counts by millions. This high logic capacity allows to implement complex systems like multi-processors platform with application dedicated coprocessors. Table 1 shows the estimation of the FPGA worldwide market in the next years in various application domains. The "high end" lines concern only FPGA with high logic capacity for complex system implementations. This market is in significant expansion and is estimated to 914 M$ in 2012.

Today, several companies (Atipa, blue-arc, Bull, Chelsio, Convey, CRAY, DataDirect, DELL, hp, Wild Systems, IBM, Intel, Microsoft, Myricom, NEC, nvidia etc) are making systems where demand for very high performance (HPC) primes over other requirements. They tend to use the highest performing devices like Multi-core CPUs, GPUs, large FPGAs, custom ICs and the most innovative architectures and algorithms. These companies show up in different "traditional" applications and market segments like computing clusters (ad-hoc), servers and storage, networking and Telecom, ASIC emulation and prototyping, military/aereo etc. The HPC market size is estimated today by FPGA providers at 214 M$. This market is dominated by Multi-core CPUs and GPUs based solutions and the expansion of FPGA-based solutions is limited by the lack of design automation.

Nowadays, there are no commercial or academic tools covering the whole design flow from the system level specification to the bitstream generation neither for embedded system design nor for HPC.

> *The aim of the COACH project is to integrate all these design steps into a single design framework and to allow* ***pure software*** *developpers to design embedded systems.*

The COACH project proposes an open-source framework for mapping multi-tasks software applications on Field Programmable Gate Array circuits (FPGA). It aims to propose solutions to the societal/economical challenges by providing SMEs novel design capabilities enabling them to increase their design productivity with design exploration and synthesis methods that are placed on top of the state-of-the-art methods. We believe that the combination of a design environment dedicated to software developpers and FPGA targets, will allow small and even very small companies to propose embedded system and accelerating solutions for standard software applications with attractive and competitive prices. This new market may explode in the same way as the micro-computer market in the eighties, whose success was due to the low cost of the first micro-processors (compared to main frames) and the advent of high level programming languages which allowed a high number of programmers to launch start-ups in software engineering.

2.2 Relevance of the proposal

COACH will contribute to build an open design and run-time environment, including communication middleware and tools to support developers in the production of embedded software, through all phases of the software lifecycle, from requirements analysis downto deployment and maintenance. More specifically, COACH focuses on:

- High level methods and concepts (esp. requirements and architectural level) for system design, development and integration, addressing complexity aspects and modularity.
- Open and modular design environments, enabling flexibility and extensibility by means of new or sector-specific tools and ensuring consistency and traceability along the development lifecycle.
- Light/agile methodologies and adaptive workflow providing a dynamic and adaptive environment, suitable for co-operative and distributed development.

COACH outcome will contribute to strengthen Europe's competitive position by developing technologies and methodologies for product design, focusing (in compliance with the scope of the above program) on technologies, engineering methodologies, novel tools which facilitate resource use efficiency. The COACH approaches and tools will enable new and emerging information technologies for the development, manufacturing and integration of devices and related software into end-products.

The COACH project will benefit from a number of previous recent projects:

SOCLIB The SoCLib ANR platform (2007-2009) is an open infrastructure developped by 10 academic laboratories (TIMA, LIP6, Lab-STICC, IRISA, ENST, CEA-LIST, CEA-LETI, CITI, INRIA-Futurs, LIS) and 6 industrial companies (Thales Communications, Thomson R&D, STMicroelectronics, Silicomp, MDS, TurboConcept). It supports system level virtual prototyping of shared memory, multi-processors architectures, and provides tools to map multi-tasks software application on these architectures, for reliable performance evaluation. The core of this platform is a library of SystemC simulation models for general purpose IP cores such as processors, buses, networks, memories, IO controller. The platform provides also embedded operating systems and software/hardware communication middleware. The synthesisable VHDL models of IPs are not part of the SoCLib platform, and COACH will enhance SoCLib by providing the synthesisable VHDL models required for FPGA synthesis.

ROMA The ROMA ANR project [35] involving IRISA (CAIRN team), LIRMM, CEA List, THOMSON France R&D, proposes to develop a reconfigurable processor, exhibiting high silicon density and power efficiency, able to adapt its computing structure to computation patterns that can be speed-up and/or power efficient. The project will borrow from the ROMA ANR project and the ongoing joint INRIA-STMicro Nano2012 project to adapt existing pattern extraction algorithms and datapath merging techniques to ASIP synthesis.

TSAR The TSAR MEDEA+ project (2008-2010) involving BULL, THALES and LIP6 targets the design of a scalable, coherent shared memory, multi-cores processor architecture, and uses the SoCLib plaform for virtual prototyping. COACH will benefit from the synthesizable VHDL models developped in the framework of TSAR (MIPS32 processor core, and RING interconnect).

BioWic On the HPC application side, we also hope to benefit from the experience in hardware acceleration of bioinformatic algorithms/workfows gathered by the CAIRN group in the context of the ANR BioWic project (2009-2011), so as to be able to validate the framework on real-life HPC applications.

The laboratories involved in the COACH project have a well estabished expertise in the domains:

- In the field of High Level Synthesis (HLS), the project leverages on know-how acquired over the last 15 years with the GAUT [16] project developped by the LAB-STICC laboratory, and with the UGH [12] project developped by the LIP6 and TIMA laboratories.
- Regarding system level architecture, the project is based on the know-how acquired by LIP6 and TIMA in the framework of various projects in the field of communication architectures for shared memory multi-processors systems (COSY [30], DISYDENT [29] or DSPIN [36] of MEDEA-MESA). As an example, the DSPIN project is now used in the TSAR project.
- Regarding Application Specific Instruction Processor (ASIP) design, the CAIRN group at INRIA Rennes – Bretagne Atlantique benefits from several years of expertise in the domain of retargetable compiler (Armor/Calife [14] since 1996, and the Gecos compilers [33] since 2002).
- In the field of compilers, the ENS Lyon/LIP Compsys group was founded in 2002 by several senior researchers with experience in high performance computing and automatic parallelization. They have been among the initiators of the polyhedral model, a theory which serve to unify many parallelism detection and exploitation techniques for regular programs. It is expected that the techniques developped by ENS Lyon/LIP for parallelism detection, scheduling [22, 23], process construction [25] and memory management [11] will be very useful as a front-end for HLS tools.

The COACH project answers to several of the challenges found in different axis of the call for proposals.

Axis 1 *Architectures des systèmes embarqués*
COACH will address new embedded systems architectures by allowing the design of Multi-Core Systems-on-Chip (possibly heterogeneous) on FPGA according to the design constraints and objectives (real-time, low-power). It will permit designing complex SoC based on IP cores (memory, peripherals...), running Embedded Software, as well as an Operating System with associated middleware and API and using hardware accelerator automatically generated. It will also permit to use efficiently different dynamic system management techniques and reconfiguration mechanisms. **Thereby COACH well corresponds to axis 1**.

Axis 2 *Infrastructures pour l'Internet, le calcul intensif ou les services*
COACH will address High-Performance Computing (HPC) by helping designers to accelerate an application running on a PC. By providing tools that translate high level language programs to FPGA configurations, COACH will allow to easily migrate critical parts into an FPGA plugged to the PC bus (through a communication link like PCI/X). Moreover, Dynamic Partial Reconfiguration will be used for improving HPC performance as well as reducing the required area. **Thereby COACH partially corresponds to axis 2**.

Axis 3 *Robotique et contrôle/commande*:

COACH will address robotic and control applications by allowing to design complex systems based on MPSoC architecture. Like in the consumer electronics domain, future control applications will employ more and more SoC for safety and security applications. Application domains for such systems are for example automotive or avionics domains (e.g. collision-detection, intelligent navigation...). Manufacturing technology will also increasingly need high-end vision analysis and high-speed robot control. **Thereby COACH indirectly answers to axis 3**.

Axis 5 *Sécurité et sureté*:

The results of the COACH project will help users to build cryptographic secure systems implemented in hardware or both in software/hardware in an effective way, substantially enhancing the process productivity of the cryptographic algorithms hardware synthesis, improving the quality and reducing the design time and the cost of synthesised cryptographic devices. **Thereby COACH indirectly answers to axis 5**.

Finally, it is worth to note that this project covers priorities defined by the commission experts in the field of Information Technolgies Society (IST) for Embedded Systems: <<*Concepts, methods and tools for designing systems dealing with systems complexity and allowing to apply efficiently applications and various products on embedded platforms, considering resources constraints (delays, power, memory, etc.), security and quality services*>>.

3 Scientific and technical Description

3.1 State of the Art

Our project covers several critical domains in system design in order to achieve high performance computing. Starting from a high level description we aim at generating automatically both hardware and software components of the system.

3.1.1 High Performance Computing

The High-Performance Computing (HPC) world is composed of three main families of architectures: many-core, GPGPU (General Purpose computation on Graphics Unit Processing) and FPGA. The first two families are dominating the market by taking benefit of the strength and influence of mass-market leaders (Intel, Nvidia). In this market, FPGA architectures are emerging and very promising. By adapting architecture to the software, FPGAs architectures enable better performance (typically between x10 and x100 accelerations) while using smaller size and less energy (and heat). However, using FPGAs presents significant challenges [34]. First, the operating frequency of an FPGA is low compared to a high-end microprocessor. Second, based on Amdahl law, HPC/FPGA application performance is unusually sensitive to the implementation quality [19]. Finally, efficient design methodology are required in order to hide FPGA complexity and the underlying implantation subtleties to HPC users, so that they do not have to change their habits and can have equivalent design productivity than in others families [38].

HPC/FPGA hardware is only now emerging and in early commercial stages, but these techniques have not yet caught up. Industrial (Mitrionics [5], Gidel [4], Convey Computer [2]) and academic (CHREC) researches on HPC-FPGA are mainly conducted in the USA. None of the approaches developed in these researches are fulfilling entirely the challenges described above. For example, Convey Computer proposes application-specific instruction set extension of x86 cores in FPGA accelerator, but extension generation is not automated and requires hardware design skills. Mitrionics has an elegant solution based on a compute engine specifically developed for high-performance execution in FPGAs. Unfortunately, the design flow is based on a new programming language (mitrionC) implying important designer efforts and poor portability.

Thus, much effort is required to develop design tools that translate high level language programs to FPGA configurations. Moreover, as already remarked in [21], Dynamic Partial Reconfiguration [37] (DPR, which enables changing a part of the FPGA, while the rest is still working) appears very interesting for improving HPC performance as well as reducing required area.

3.1.2 System Synthesis

Today, several solutions for system design are proposed and commercialized. The existing commercial or free tools do not cover the whole system synthesis process in a full automatic way. Moreover, they are bound to a particular device family and to IPs library. The most commonly used are provided by ALTERA and XILINX to promote their FPGA devices. These representative tools used to synthesize SoC on FPGA are introduced below.

The XILINX System Generator for DSP [10] is a plug-in to Simulink that enables designers to develop high-performance DSP systems for XILINX FPGAs. Designers can design and simulate a system using MATLAB and Simulink. The tool will then automatically generate synthesizable Hardware Description Language (HDL) code mapped to XILINX pre-optimized algorithms. However, this tool targets only DSP based algorithms, XILINX FPGAs and cannot handle a complete SoC. Thus, it is not really a system synthesis tool.

In the opposite, SOPC Builder [9] from ALTERA and XILINX Platform Studio XPS from XILINX allows to describe a system, to synthesis it, to program it into a target FPGA and to upload a software application. Both SOPC Builder and XPS, allow designers to select and parameterize components from an extensive drop-down list of IP cores (I/O core, DSP, processor, bus core, ...) as well as incorporate their own IP. Nevertheless, all the previously introduced tools do not provide any facilities to synthesize coprocessors and to simulate the platform at a high level (SystemC). System designer must provide the synthesizable description of its own IP-cores with the feasible bus interface. Design Space Exploration is thus limited and SystemC simulation is not possible neither at transactional nor at cycle accurate level.

In addition, XILINX System Generator, XPS and SOPC Builder are closed world since each one imposes their own IPs which are not interchangeable. Designers can then only generate a synthesized netlist, VHDL/Verilog simulation test bench and custom software library that reflect the hardware configuration.

Consequently, a designer developing an embedded system needs to master four different design environments:

1. a virtual prototyping environment (in SystemC) for system level exploration,
2. an architecture compiler to define the hardware architecture (Verilog/VHDL),
3. one or several third-party HLS tools for coprocessor synthesis (C to RTL),
4. and finally back-end synthesis tools for the bit-stream generation (RTL to bitstream).

Furthermore, mixing these tools requires an important interfacing effort and this makes the design process very complex and achievable only by designers skilled in many domains.

3.1.3 High Level Synthesis

High Level Synthesis translates a sequential algorithmic description and a set of constraints (area, power, frequency, ...) to a micro-architecture at Register Transfer Level (RTL). Several academic and commercial tools are today available. The most common tools are SPARK [28], GAUT [16], UGH [12] in the academic world and CATAPULTC [1], PICO [7] and CYNTHETIZER [3] in the commercial world. Despite their maturity, their usage is restrained by [18] [13] [17]:

- HLS tools are not integrated into an architecture and system exploration tool. Thus, a designer who needs to accelerate a software part of the system, must adapt it manually to the HLS input dialect and perform engineering work to exploit the synthesis result at the system level,

- Current HLS tools can not target control AND data oriented applications,
- HLS tools take into account mainly a unique constraint while realistic design is multi-constrained. Low power consumption constraint which is mandatory for embedded systems is not yet well handled or not handled at all by the HLS tools already available,
- The parallelism is extracted from initial specification. To get more parallelism or to reduce the amount of required memory in the SoC, the user must re-write the algorithmic specification while there is techniques such as polyedric transformations to increase the intrinsic parallelism,
- While they support limited loop transformations like loop unrolling and loop pipelining, current HLS tools do not provide support for design space exploration neither through automatic loop transformations nor through memory mapping,
- Despite having the same input language (C/C++), they are sensitive to the style in which the algorithm dis written. Consequently, engineering work is required to swap from a tool to another,
- They do not respect accurately the frequency constraint when they target an FPGA device. Their error is about 10 percent. This is annoying when the generated component is integrated in a SoC since it will slow down the whole system.

Regarding these limitations, it is necessary to create a new tool generation reducing the gap between the specification of an heterogeneous system and its hardware implementation [17] [18].

3.1.4 Application Specific Instruction Processors

ASIP (Application-Specific Instruction-Set Processor) are programmable processors in which both the instruction and the micro architecture have been tailored to a given application domain or to a specific application. This specialization usually offers a good compromise between performance (w.r.t a pure software implementation on an embedded CPU) and flexibility (w.r.t an application specific hardware co-processor). In spite of their obvious advantages, using/designing ASIPs remains a difficult task, since it involves designing both a micro-architecture and a compiler for this architecture. Besides, to our knowledge, there is still no available open-source design flow for ASIP design even if such a tool would be valuable in the context of a System Level design exploration tool.

In this context, ASIP design based on Instruction Set Extensions (ISEs) has received a lot of interest [6], as it makes micro architecture synthesis more tractable [1], and help ASIP designers to focus on compilers, for which there are still many open problems[27]. This approach however has a severe weakness, since it also significantly reduces opportunities for achieving good speedups (most speedups remain between 1.5x and 2.5x), since ISEs performance is generally tied down by I/O constraints as they generally rely on the main CPU register file to access data.

To cope with this issue, recent approaches [32, 31, 15] advocate the use of micro-architectural ISE models in which the coupling between the processor micro-architecture and the ISE component is tightened up so as to allow the ISE to overcome the register I/O limitations. However these approaches generally tackle the problem from a compiler/simulation point of view and do not address the problem of generating synthesizable representations for these models.

We therefore strongly believe that there is a need for an open-framework which would allow researchers and system designers to :

- Explore the various level of interactions between the original CPU micro-architecture and its extension (for example through a Domain Specific Language targeted at micro-architecture specification and synthesis).
- Retarget the compiler instruction-selection pass (or prototype new passes) so as to be able to take advantage of this ISEs.

[1] ISEs rely on a template micro-architecture in which only a small fraction of the architecture has to be specialized

- Provide a complete System-level Integration for using ASIP as SoC building blocks (integration with application specific blocks, MPSoc, etc.)

3.1.5 Automatic Parallelization

The problem of compiling sequential programs for parallel computers has been studied since the advent of the first parallel architectures in the 1970s. The basic approach consists in applying program transformations which exhibit or increase the potential parallelism, while guaranteeing the preservation of the program semantics. Most of these transformations just reorder the operations of the program; some of them modify its data structures. Dependences (exact or conservative) are checked to guarantee the legality of the transformation.

This has lead to the invention of many loop transformations (loop fusion, loop splitting, loop skewing, loop interchange, loop unrolling, ...) which interact in a complicated way. More recently, it has been noticed that all of these are just changes of basis in the iteration domain of the program. This has lead to the introduction of the polyhedral model [24, 20], in which the combination of two transformations is simply a matrix product.

Since hardware is inherently parallel, finding parallelism in sequential programs in an important prerequisite for HLS. The large FPGA chips of today can accomodate much more parallelism than is available in basic blocks. The polyhedral model is the ideal tool for finding more parallelism in loops.

As a side effect, it has been observed that the polyhedral model is a useful tool for many other optimization, like memory reduction and locality improvement. Another point is that the polyhedral domain *stricto sensu* applies only to very regular programs. Its extension to more general programs is an active research subject.

3.2 S & T objectives, progress beyond the state of the art

The design steps are presented figure 1.

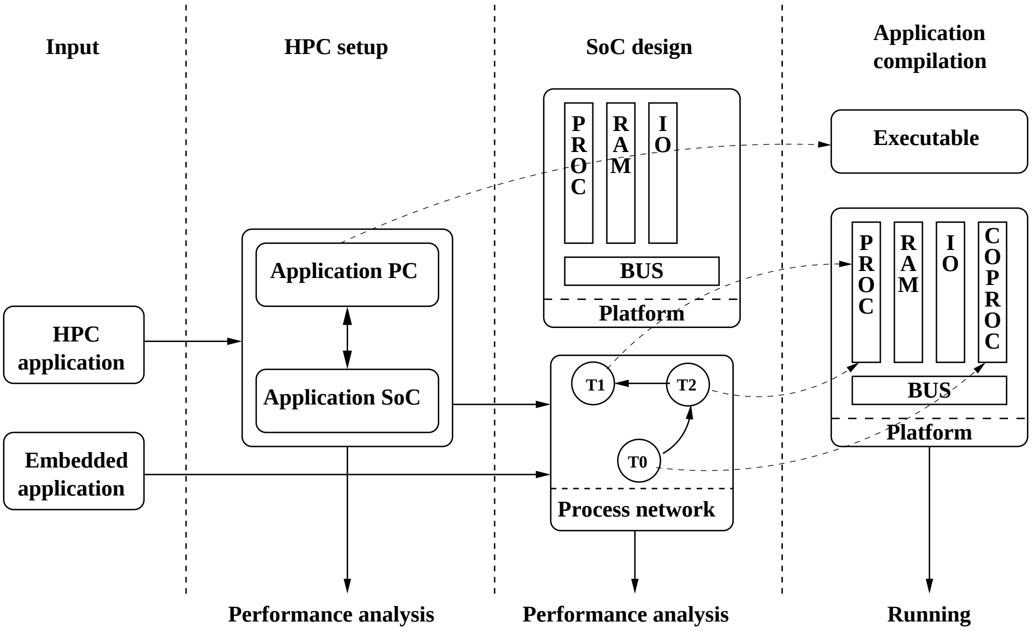


Figure 1: COACH design flow

HPC setup: During this step, the user splits the application into 2 parts: the host application which remains on the PC and the SoC application which is mapped on the FPGA. COACH will provide a complete simulation model of the whole system (PC+communication+FPGA-SoC) which will allow performance evaluation.

SoC design: In this phase, COACH will allow the user to obtain virtual prototypes for the SoC at different abstraction levels. The user input will consist of a process network describing the coarse grain parallelism of the application, an instance of a generic hardware platform and a mapping of processes on the platform components. COACH will offer different targets to map the processes: software (the process runs as a software task on a SoC processor), ASIP (the process runs as a software task on a SoC processor enhanced with dedicated instructions), and hardware (the process is implemented as a synthesized hardware coprocessor).

Application compilation: Once the SoC architecture is validated through performances analysis, COACH will generate automatically an executable containing the host application and the FPGA bitstream. This bitstream contains both the hardware architecture and the SoC application software. The user will be able to launch the application by loading the bitstream on an FPGA and running the executable on PC.

Hardware/Software co-design is a very complex task. To simplify it, COACH will address the following scientific and technological barriers:

Design Space Exploration by Virtual Prototyping : The COACH environment will allow to easily map a parallel application described as a process network Model of Computation (MoC) on a shared-memory, MPSoC architecture. COACH will permit to explore the design space by allowing system designer to select and parameterize the target architecture, and to define the best hardware/software partitioning of the application.

High-Level Synthesis : COACH will allow the automatic generation of hardware accelerators when required by using High-Level Synthesis (HLS) tools. These HLS tools will be fully integrated into a complete system-level design environment. Moreover, COACH will support both data and control dominated applications, and the HLS tools of COACH will support a common language and coding style to avoid re-engineering by the designer. COACH will provide a tool which will automatically explore the micro-architectural design space of coprocessor.

High-level code transformation : COACH will allow to optimize the memory usage, to enhance the parallelism through loop transformations and parallelization. The challenge is to identify the coarse grained parallelism and to generate, from a sequential algorithm, application containing multiple communicating tasks. To this aim, one may adapt techniques which were developed in the 1990 for the construction of distributed programs. However, in the context of HLS, there are still several original problems to be solved, mainly to do with the construction of FIFO communication channels and with memory optimization. Additionnal preprocessing, source-level transformations, are thus required to improve the process. Particularly, this includes parallelism exposure and efficient memory mapping. COACH will support code transformation by providing a source to source C2C tool.

Hardware/Software communication middleware : COACH will implement an homogeneous HW/SW communication infrastructure and communication APIs (Application Programming Interface), that will be used for communications between software tasks running on embedded processors and dedicated hardware coprocessors. This will allow explore the design space by mapping the tasks of the application (described as a process network) on a shared-memory, MPSoC architecture.

Processor customization : ASIP design will be addressed by the COACH project. COACH will allow system designers to explore the various level of interactions between the original CPU micro-architecture and its extension. It will also allow to retarget the compiler instruction-selection pass. Finally, COACH will integrate ASIP design in a complete System-level design framework.

The main result is the framework. It is composed concretely of: a communication middleware for HPC, 5 HAS tools (control dominated HLS, data dominated HLS, Coarse grained HLS, Memory optimisation HLS and ASIP), 3 architectural templates that are synthesizable and that can be prototyped, one design space exploration tool, 2 operating systems (DNA/OS and MUTEKH).
The framework fonctionality will be demonstrated with the demonstrators (see task-7 page 31) and the tutorial example (see task-8 page 4.3.8).

4 Scientific and technical objectives / project description

4.1 Scientific Programme, Project structure

Figures 2, 3 and 4 summarize the software architecture of the COACH framework we will develop. In figures, the dotted boxes are the softwares or formats that COACH has to provide and to support.

For the system generation presented in figure 2, the conductor is the tool `CSG` (COACH System Generator). Its inputs are a process network describing the target application and the synthesis parameters. The main parameters are the target hardware architectural template with its instantiation parameters, the hardware/software mapping of the tasks, the FPGA device and design constraints. `CSG` thus requires an architectural template library, an operating system library, two system hardware component (CPU, memories, BUS...) libraries (one for synthesis, one for simulation). For generating the coprocessor of a task mapped as hardware, `CSG` controls the HAS tools described below. From these inputs `CSG` can generate the entire system (both software and hardware) either as a SystemC simulator (cycla accurate and/or TLM) to prototype and explore quickly the design space or as a bitstream[2] directly downloadable on the FPGA device[3].

The software architecture for HAS is presented in figure 3. The input is a single task of the process network. The HAS tools do not work directly on the C++ task description but on an internal format called `xcoach` generated by a plugin into the GNU C compiler (GCC). This will allow on the one hand to insure that all the tools will accept the same C++ description and on the other hand make possible their chaining. The front-end tools read a `xcoach` description and generate a new `xcoach` description that exibits more parallelism or implement specific instructions for ASIP. The back-end tools read an `xcoach` description and generate an `xcoach+` description. This is an `xcoach` description annotated with hardware information (scheduling, binding) required by the VHDL and systemC drivers. Furthermore, the back-end tools uses a macro-cell library (functional and memory unit).

In addition to digital system design, HPC requires a supplementary partitioning step presented in figure 4. The designer splits the initial application (tag 1) in two parts: one still on the PC and the other running in a FPGA plugged on the PCI/X PC bus. The two parts exchange data through communication primitives (tag 2) implemented in a library. To evaluate the relevance of the partitioning, the designer can build a simulator. Once the partitioning is validated, the design of the FPGA part is done through `CSG` (figure 2).

The project is split into 8 tasks numbered from 1 to 8. They are described in short below and in detail in section 4.3.

Task-1: ***Project management*** This task relates to the monitoring of the COACH project.

Task-2: ***Backbone infrastructure*** This task tackles the fundamental points of the project such as the defintion of the COACH inputs and outputs, the internal formats (i.e. `xcoach` and `xcoach+`) and their associated tools, the architectural templates and the design flow.

Task-3: ***System generation*** This task addresses the prototyping and the generation of digital system. Apart from HAS that belongs to task 3 and 4, its components are those presented figure 2 (e.g. `CSG`, operating systems).

[2] COACH generates synthesizable VHDL, and launch the XILINX or ALTERA RTL synthesis tools.
[3] Additional partial bitstreams are generated in case of dynamic partial reconfiguration

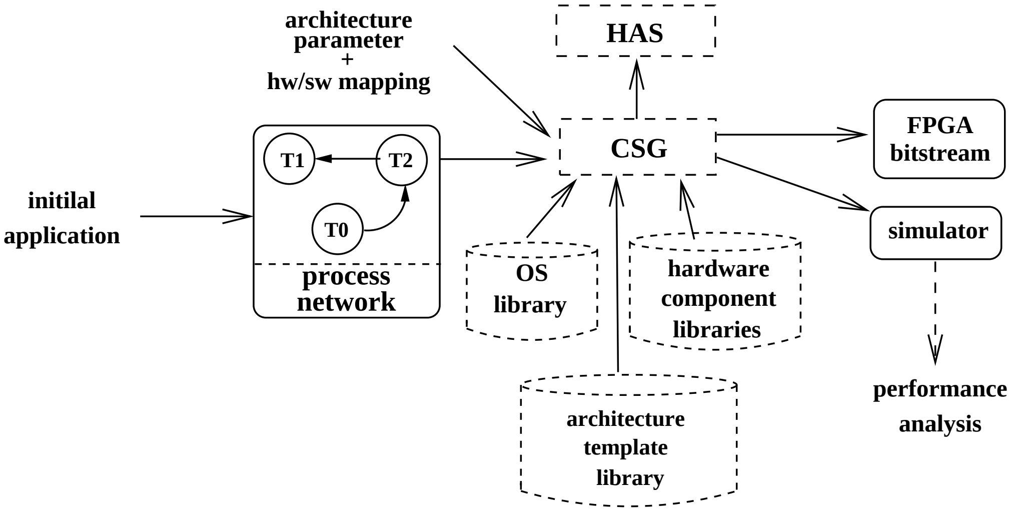


Figure 2: Software architecture for digital system generation

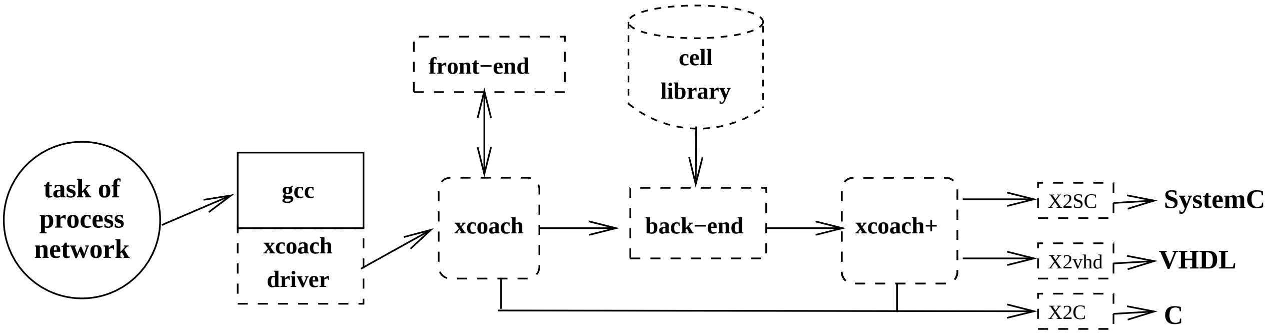


Figure 3: Software architecture of hardware accellerator synthesis

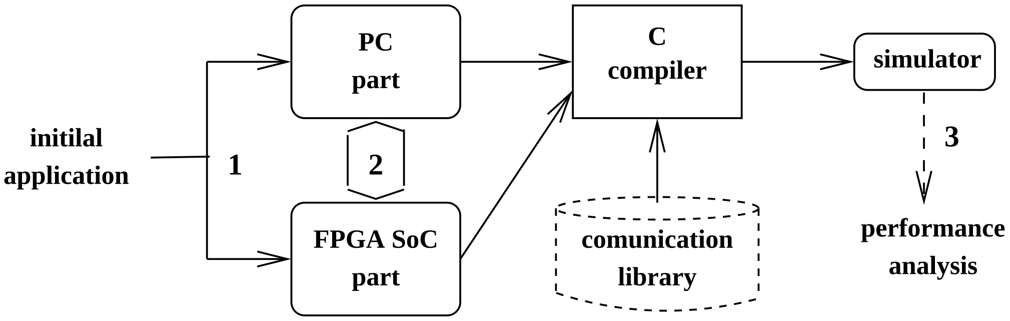


Figure 4: Software architecture of HPC

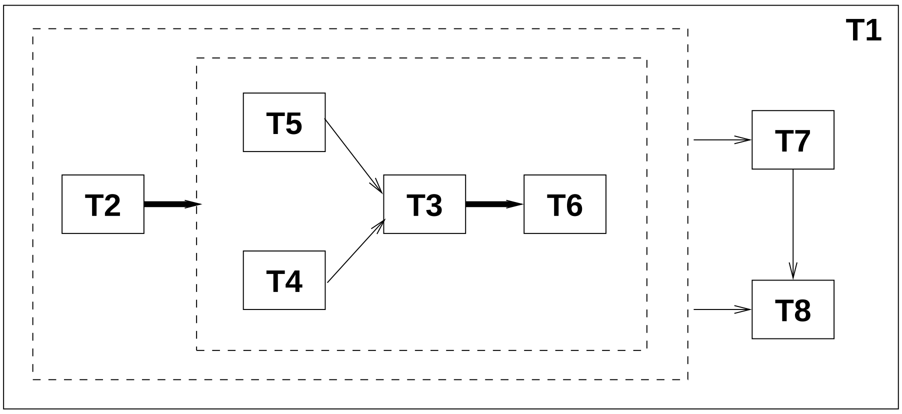


Figure 5: Task dependencies

Task-4: ***HAS front-end*** This task mainly focusses on four functionalities: optimization of the memory usage, parallelism enhancement through loop transformations, coarse grain parallelization and ASIP generation.

Task-5: ***HAS back-end*** This task groups two functionalities: High-Level Synthesis of data dominated description and HLS of control dominated description. This task contains also the development of a frequency adaptator that will allow the coprocessors to respect the processor and the bus frequency.

Task-6: ***PC/FPGA communication middleware*** This task pools the features dedicated to HPC. These are mainly the validation of the partitioning (see figure 4), the sytem drivers for both PC and FPGA-SoC sides, the hardware communication components and the support for dynamic partial reconfiguration.

Task-7: ***Industrial demonstrators*** This task groups the demonstrators of the COACH project. Most of them are industrial applications that will be developped within the COACH framework. Others consist in integrating the COACH framework as a driver of industrial proprietary design tools.

Task 8: ***Dissemination*** This task concerns the diffusion of the project results. It mainly consists of the production of 4 COACH releases (`T0+12`, `T0+18`, `T0+24` and `T0+36`), the publication of a tutorial and user manuals on a WEB site, the publication of research papers in international journals and conferences and the organization of workshops and tutorials in international conferences.

Figure 5 presents the tasks dependencies. "$T_N \longrightarrow T_M$" means that T_N impacts the T_M. The more bold the arrow, the more important is the impact. The graph shows:

- Even though $T4$ and $T5$ functionalities are complementary, their developments are independent (thanks to the `xcoach` internal format).
- $T3$ slightly depends on $T4$ and $T5$. Indeed, $T3$ may work without $T4$ and $T5$ if targeted digital systems do not include hardware accelerators.
- $T3$ strongly impacts $T6$ but $T3$ does not depend at all on $T6$. Hence demonstrators ($T7$) of embedded system would not be impacted if $T6$ would fail.
- $T2$ drives all the tasks ($T3$, $T4$, $T5$, $T6$) and is at the heart of the COACH project.
- The demonstrators developped in $T7$, of course strongly depend on the achievements of the previous tasks ($T2$, $T3$, $T4$, $T5$, $T6$).

- $T8$ and $T1$ depend on and impact all the other tasks.

This organisation offers enough robustness to insure the success of the project except for the specification task $T2$. The only critical task in this chart is $T2$. However, the partners met 12 times (a one-day meeting per month) during the last year: 10 meetings to exchange and work on scientific and technical aspects and 2 meetings to prepare the project proposal. This gives us a high degree of confidence that $T2$ will be completed in time.

4.2 Project management

Project management structure Each task is assigned to a Task Leader. The Task Leaders assist the project leader in the technical organization, effort management, of the co-operation and the reporting of the progress. A steering committee is composed by task leaders and the project leader. The steering committee has a monthly conference call and is in charge of conflict management if necessary. Each task leader has to report on the main high-lights, major opportunities and problems according to the work-plan. The redaction of the 6-month reports is the responsability of the steering committee. Therefore, each Partner has the responsibility to monthly inform the task Leaders of the current development of the sub-task he has in charge. COACH will be organized in 8 tasks whose interactions are presented in Figure 5.

Scientific and Technical Reports For every yearly review, a written progress report for each deliverable has to be provided by the task leader to the coordinator for integration in the contractual reports.

Management of knowledge, Intellectual Property Right (IPR) and Results Exploitation The partners will have to work under eventual NDA constraints. Prior Intellectual Property remains property of the concerned partners. The exploitation of the results obtained in the project and by each partner involved in the consortium will follow the rules written in the articles of the Consortium Agreement accepted and signed by each partner at most 6 months after the project kick-off. To manage the exploitation and dissemination plan within the project, six monthly meetings will analyze the intentions from the consortium (patent, publication...).

Management Tools In order to permit a good management, before the kick-off meeting, each partner will have to identify (name, address, phone, fax and e-mail):

- the financial and administrative contact person,
- the scientific and technical contact person,
- all participants to the project.

A complete and detailed list will be communicated to each partner and to the public Authority. The partners will construct mailing lists for day-to-day communication.

The first task will be the redaction of a Consortium Agreement, dealing mainly with all aspects of the relations between partners, including legal aspects, property rights and further exploitation of the results. This document will be submitted to the partner's financial and legal departments, and will define the management rules (decision level, reporting systems, red flag cases). A first draft of this document will be submitted to each partner during the kick-off meeting.

Project follow-ups The basic communication between single project partners will be carried out by means of an Information System (web site), which will be developed and introduced at the very beginning of the project implementation. All scientific and administrative data related to the project will be collected and treated within a specific e-management plate-form accessible directly by the project web site by an individual login and pass-word. The web site will have a few levels of accessibility starting with completely free access, open to broad public up to internal materials available only for members of the consortium for the e-management area.

This communication tools will permit to perform all the reports and to follow as well as possible all the tasks.

Project monitoring For this project format and size, a 12 months review by ANR, based on a yearly progress report incorporating milestones reports and deliverables, seems optimum. The internal consortium meetings will be every six months, including a kick-off meeting at the start of the project, in our eyes the most important of all, as it phases the partners for the start of the project.

4.3 Description of the tasks

In this document, we use the following abbreviations in the tables and Gantt diagrams:

partner INRI for INRIA/CAIRN, LIP for ENS Lyon/LIP, TIMA for TIMA, UBS for LAB-STICC, LIP6 for LIP6, XILX for XILINX, BULL for BULL, TRT for THALES, FLEX for FLEXRAS and NAV for NAVTEL-SYSTEM.

kind of the deliverable `x` for a software, `d` for a document and `h` for a hardware component.

task contribution ”lead.” for leader and ”part.” for participant.

other abbreviations ”resp.” for responsible patrner, ”kd” for kind of deliverable.

4.3.1 Task 1: *Project management*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
part.	part.	part.	part.	lead.	part.	part.	part.	part.	part.

Objectives This task relates to the monitoring of the COACH project. Its main objectives are:

- To ensure the appropriate progress of the project,
- To coordinate the scientific and technical cooperation between the partners,
- To manage and monitor the scientific and technical work and progress in the tasks,
- To verify the conformance to agreed budget and time scales,
- To prepare periodic progress reports in order to control the overall progress of the project,
- To organize the project meetings,
- To set up a shared development infrastructure as a version control system and development WEB site.

ST1-1 This sub-task consists in writing and ratifying the consortium agreement.

number	date	type	resp.	description
D110-VF	T0+6	D	LIP6	A document describing the consortium agreement, signed by all the partners.

ST1-2 This sub-task concerns the global management of the deliverables and of the global organization of the project at all the levels.

number	date	type	resp.	description
D120-VF	T0+36	D	LIP6	Global management of the project at all the levels: progress monitoring, record keeping, meeting organization, review organization, the writing of the review reports.

ST1-3 This sub-task consists in managing the project at the partner level. It includes mainly the progress monitoring, the record keeping the participation to the project meetings and the communication with the project leader and the other partners.

number	date	type	resp.	description
D130-VF	T0+36		LIP6	Project management at the partner level.

ST1-4 This sub-task consists firstly in the building, and next in the administration and the maintenance of the development and dissemination infrastructure. It is also in charge of the COACH releases distribution.

number	date	type	resp.	description
D140-V1	T0+6	x	LIP6	Setup of the development infrastructure (version control system configuration, wiki).
D140-VF	T0+36	x	LIP6	Standard management of a development infrastructure (adding & suppressing account, retrieving forgotten passwords, creation and closing development branch, ...)

4.3.2 Task 2: *Backbone infrastructure*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
part.	part.	part.	part.	lead.	part.	part.	part.	part.	part.

Objectives This task deals with the main features of digital systems. Its objectives are the specification of the designer input, the definition of the hardware architectural templates and of all the features that the HAS tools will share.

ST2-1 This sub-task specifies the COACH environment for the system designer. At this level the COACH framework is a black box. The deliverables are documents specifying: how to feed COACH (the inputs), how to use COACH (the design flow), what is generated (the outputs).

number	date	type	resp.	description
D210-V1	T0+6	D	LIP6	The first version of the COACH specification. This document contains the general description of the framework, the design flow and the description of the architectural templates. It refers to the HAS specification (deliverable D212-VF) and to the CSG specifcation (deliverable D211-VF) for the COACH input descriptions.
D210-VF	T0+12	D	LIP6	The final version of the D210-V1 deliverable updated with the first feed-backs of the demonstrator sub-tasks.
D211-V1	T0+6	D	TIMA	The first version of the CSG (COACH System Generator) specification. It specifies how the task graph is described, the communication schemes and its associated API (Application Programming Interface). The base is the SRL library and the MWMR communication component defined by the SocLib ANR project. Nevertheless, these basic schemes will be enhanced to allow more efficient synthesis.
D211-VF	T0+12	D	TIMA	The final version of the D210-V1 deliverable updated with the first feed-backs of the demonstrator sub-tasks.

D212-V1	T0+6	D	UBS	The first version of the HAS (Hardware Accelerator Synthesis) specification. It specifies how tasks must be written (C/C++ subset) and how communication schemes defined in the D211-VF deliverable must be described for coprocessor synthesis.
D212-VF	T0+12	D	UBS	The final version of the D210-V1 deliverable updated with the first feed-backs of the demonstrator sub-tasks.

ST2-2 This sub-task specifies the software COACH structure. The deliverable is a document listing all the COACH software components and how they cooperate.

number	date	type	resp.	description
D220-VF	T0+6	D	LIP6	Description of the software list and the data flow among the tools.

ST2-3 This sub-task specifies the `xcoach` and the `xcoach+` formats.

number	date	type	resp.	description
D230-V1	T0+6	D+X	LIP	First release of the XML specification of the `xcoach` format (DTD) and its associated documentation allowing to start HLS tools development.
D230-V2	T0+12	D+X	LIP	Second release of XML specification of the `xcoach` format taking into account the corrections and modifications that the developers of HAS tools suggested.
D230-VF	T0+18	D+X	LIP	Last release of XML specification of the `xcoach` format enhanced with the expression of loop potential parallelism.
D231-V1	T0+12	X	UBS	A GCC plugin C2X that generates a `xcoach` description (defined in D230-V1 deliverable) from a C/C++ task description (defined in D212-VF deliverable).
D231-VF	T0+18	X	UBS	An updated version of C2X (D231-V1) which supports the `xcoach` format defined in the D230-VF deliverable and the HAS input format defined in the D212-VF deliverable.
D232-V1	T0+12	X	UBS	This second tool X2C regenerates a C description from a `xcoach` description.
D232-VF	T0+18	X	UBS	The same software as the former (D232-V1) but for the `xcoach` format as defined in the D230-VF deliverable and HAS input as defined in the D212-VF deliverable.
D233-V1	T0+18	X	LIP6	The first release of the software tool X2SC that translates `xcoach+` description to CABA and TLM-DT SystemC module.
D233-VF	T0+24	X	LIP6	Final release of the former software (D233-V1).
D234-V1	T0+18	X	UBS	The first release of the software tool X2VHDL that translates `xcoach+` description to synthesizable VHDL description.
D234-VF	T0+24	X	UBS	Final release of the former software (D234-V1) and integration of enhancements proposed in D235 deliverable.

D235-VF	T0+21	D	XILX	This deliverable consists in optimizing the VHDL generated from `xcoach+` format (deliverable D234) for the XILINX RTL synthesis tools. LAB-STICC will provide several examples of VHDL source files generated from `xcoach+`, with explanations about generation process of main data structures used in VHDL sources, XILINX will provide back a documentation listing that proposes VHDL generation enhancements.

ST2-4 This sub-task aims to define a tool in order to pilot the GCC/xcoach compiler.

number	date	type	resp.	description
D240-VF	T0+3	D	UBS	Specification of the GCC driver tool.
D241-V1	T0+9	X	UBS	First release of the GCC driver tool.
D241-VF	T0+12	X	UBS	Final release of the GCC driver tool.

ST2-5 Backend HLS tools use a characterized macro-cell library to build the micro-architecture of a coprocessor. The characterisation of a cell depends on the target device. The role of this sub-task is to define the macro-cells and to provite a tool that characterizes them automatically by synthesizing them and by extracting their delays. This is done by using RTL synthesis.

number	date	type	resp.	description
D250-VF	T0+6	D	UBS	Definition of the macro cells and the file format describing them.
D251-VF	T0+12	X	UBS	Final release of the software tool that generates automatically the characterized macro-cell library for a FPGA device.

4.3.3 Task 3: *System generation*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
part.		part.	part.	lead.	part.				

Objectives This task deals with the prototyping and the generation of FPGA-SoC digital systems. Its is described on figure 2. Its objective is to allow the system designer to explore the design space by quickly prototyping and then to automatically generate the FPGA-SoC systems. This task consists of

- The development of all the missing components (SytemC models and/or synthesizable VHDL models of the IP-cores),
- The configuration and the development of drivers of the operating systems (Board Support Package, HAL),
- The CSG software that generates the SystemC simulators for prototyping and the FPGA-SoC system including its bitstream and software executable code,
- The specification of enhanced communication schemes and their sofware and hardware implementations.

This task being based on the SoCLib platform, a first release will be delivered at $T0 + 12$ to allow the demonstrators to start working. This release will include the standard communication schemes (based on SoCLib MWMR component) and support the neutral architectural template for prototyping and hardware generation.

ST3-1 This sub-task corresponds to the COACH System Generator (CSG) software.

number	date	type	resp.	description
D310-V1	T0+12	X	LIP6	The first software release of the CSG tool that will allow demonstrators to start working by using the neutral architectural template.
D310-V2	T0+18	X	LIP6	The second release of CSG supports the XILINX and ALTERA architectural templates and the enhanced communication system, but only for SystemC prototyping. This release integrates a first integration of HLS tools.
D310-V3	T0+24	X	LIP6	This milestone extends CSG (D310-V2) to FPGA-SoC generation for the XILINX and ALTERA architectural template.
D310-VF	T0+36	X	LIP6	Final release of CSG.

ST3-2 This sub-task deals with the components of the architectural templates.

For the neutral architectural template, it consists of the development of the VHDL synthesizable description of the missing communication components (MWMR) in order to support the process network communication model. Notice that the SystemC models comes from the SocLib ANR project, the processor with its cache comes from the TSAR ANR project.

For the XILINX and ALTERA architectural templates, we use the XILINX and ALTERA IPs (NIOS, Microblaze, memories, busses...).

number	date	type	resp.	description
D320-VF	T0+12	H	LIP6	The VHDL synthesizable descriptions of the SocLib MWMR, TokenRing components.
D321-VF	T0+15	D	XILX	This deliverable consists in optimizing the VHDL descriptions of the components of the neutral architectural template (deliverable D320) to the XILINX RTL synthesis tools. LIP6 will provide the VHDL descriptions, XILINX will provide back a documentation listing that proposes VHDL generation enhancements.
D322-V1	T0+18	X	TIMA	The SystemC simulation module of the MWMR component with a PLB bus interface plus the SystemC modules of the components of the XILINX architectural template currently not available in the SocLib component library.
D322-VF	T0+24	H	TIMA	The synthesizable VHDL description of the MWMR component corresponding to the SystemC module of the former deliverable (D322-V1).
D323-VF	T0+27	D	XILX	This deliverable consists in optimizing the MWMR VHDL description (deliverable D322) of the XILINX architectural template. TIMA will provide MWMR VHDL description, XILINX will provide back a documentation listing that proposes VHDL generation enhancements.
D324-V1	T0+18	X	INRI	The SystemC simulation module of the MWMR component with an AVALON bus interface plus the SystemC modules of the components of the ALTERA architectural template currently not available in the SocLib component library.
D324-VF	T0+24	H	INRI	The synthesizable VHDL description of the MWMR component corresponding to the SystemC module of the former deliverable (D324-V1);

D325-VF	T0+12	D	UBS	Specification of an optimized communication adapter (space and time) component to handle data interleaving. This evolution aims to solve out of order communication weakness of the classical MWMR.
D326-V1	T0+24	X	UBS	0:6:0 First release of the tool that generates the VHDL description of the optimized communication adapter and its corresponding SystemC module.
D326-VF	T0+30	X	UBS	Final release of the tool that generates the VHDL description of the optimized communication adapter and its corresponding SystemC module (D325-VF).
D327-VF	T0+27	D	XILX	This deliverable consists in optimizing the communication adapter VHDL description (deliverable D325). LAB-STICC will provide communication adapter VHDL description, XILINX will provide back a documentation listing that proposes VHDL generation enhancements.

ST3-3 This sub-task consists of the configuration of the SocLib MUTEKH and DNA operating system and the development of drivers for the hardware architectural templates and enhanced communication schemes defined in D211 deliverable. For the ALTERA and XILINX architectural templates, the OSs must also be ported on the NIOS2 and MICROBLAZE processors.

number	date	type	resp.	description
D330-V1	T0+8	X	LIP6	The drivers required for the first CSG milestone (deliverable D310-V1).
D330-V2	T0+18	X	LIP6	The drivers required for the second CSG milestone (D310-V2).
D330-VF	T0+33	X	LIP6	Final release of the MUTEKH OS drivers.
D331-VF	T0+18	X	LIP6	Porting of MUTEKH OS on the NIOS2 and MICROBLAZE processors.
D332-V1	T0+8	X	TIMA	The drivers required for the first CSG milestone (deliverable D310-V1).
D332-V2	T0+18	X	TIMA	The drivers required for the second CSG milestone (D310-V2).
D332-VF	T0+33	X	TIMA	Final release of the DNA OS drivers.
D333-VF	T0+18	X	TIMA	Porting of DNA OS on the NIOS2 and MICROBLAZE processors.

4.3.4 Task 4: *HAS front-end*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
part.	lead.	part.	part.	part.					

Objectives The objective of this task is to convert the input specification of an hardware accelerator, which must be written in a familiar language (C/C++) with as few constraints as possible, into a form suitable for the HLS tools (i.e. HAS back-end tools of the COACH project). If the target is an ASIP, the frontend has to extract patterns from the source code and convert them into the definition of an extensible processor. If the target is a process network, the front end has to distribute the workload and the data sets as fairly as possible, identify communication channels, and output an `xcoach` description.

ST4-1 This sub-task aims at providing compiler support for custom instructions within the HAS front-end. It will take as input the COACH intermediate representation, and will output an annotated COACH IR containing the custom instructions definitions along with their occurrence in the application.

number	date	type	resp.	description
D410-V1	T0+18	X	INRI	In this first version of the software, the computations patterns corresponding to custom instructions are specified by the user, and then automatically extracted (when beneficial) from the application intermediate representation.
D410-VF	T0+24	X	INRI	In this second version, the software will also be able to automatically identify interesting pattern candidates in the application code, and use them as custom instructions.

ST4-2 In this sub-task, we provide micro-architectural template models for the two target processor architectures (NIOS-II and MIPS) supported within the COACH-ASIP design flow. For each processor, we provide a simulation model (System-C) and a synthesizable model (VHDL) of the architecture, along with its architectural extensions

number	date	type	resp.	description
D420-V1	T0+12	X	INRI	A SystemC simulation model for a simple extensible MIPS architectural template
D420-VF	T0+20	X	INRI	A SystemC simulation model for an extensible MIPS with a tight architectural integration of its instruction set extensions
D421-VF	T0+12	X	INRI	A SystemC simulation model for an extensible NIOS processor template, the VHDL model being already available from ALTERA
D422-V1	T0+18	H	INRI	A synthesizable VHDL model for a simple extensible MIPS architectural template
D422-VF	T0+24	H	INRI	A synthesizable VHDL model for an extensible MIPS with a tight architectural integration of its instruction set extensions
D423-VF	T0+36	D	INRI	An evaluation report with quantitative analysis of the performance/area trade-off induced by the different approaches

ST4-3 Extraction of parallelism in polyhedral loops and conversion into a process network.

number	date	type	resp.	description
D430-V1	T0+6	D	LIP	Description and specification of a process construction method for programs with polyhedral loops.
D430-VF	T0+36	D	LIP	Final assessment of the method and improved version of the specification.
D431-V1	T0+12	X	LIP	Preliminary implementation in the Syntol framework. At this step the software will just implement a single constructor.
D431-V2	T0+18	X	LIP	Implementation of the array contraction and FIFO construction algorithm. Conversion of the input and output to the `xcoach` format.
D431-V3	T0+30	D+X	LIP	Extension of automatic parallelization and array contraction to non-polyhedral loops. Implementation in the Bee framework.
D431-VF	T0+36	X	LIP	Final release taking into account the feedbacks from the demonstrator sub-tasks.

4.3.5 Task 5: *HAS back-end*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
		part.	lead.	part.	part.				

Objectives The objectives of this task are to provide the two HAS back-ends of the COACH project and a tool that adapt the coprocessor frequency to the FPGA-SoC frequency as required by the processors and the system BUS.
The HAS back-ends as shown in figure 3 reads an `xcoach` description and provides an `xcoach+` description, i.e. an `xcoach` description annotated with hardware information such as variables binding to registers, operations bindings to cells/fonctional units, operation scheduling... The `xcoach` format being generated by the D231 deliverable and the `xcoach+` being treated by the D233 and the D234 deliverables, this task strongly depends on task 1.

For the two HAS front-end, this task is based on the already existing HLS tools GAUT and UGH. These tools are complementary and not in competition because they cover respectively data and control dominated designs. The organization of the task is firstly to quickly integrate the existing HLS to the COACH framework. Secondly these tools will be improved to allows to treat data dominated application with a few control for GAUT and control dominated application with a few data processing for UGH. This will enlarge the domain the HLS can cover which is a strong limitation of the tools currently avilable [17] [18] [13].

ST5-1 The goal of this sub-task is to integrate the UGH HLS tool to the COACH framework. It consists of suppressing the C compiler and the SystemC and VHDL drivers and replacing them by `xcoach` and `xcoach+` drivers i.e. C2X, X2SC and X2VHDL.

number	date	type	resp.	description
D510-VF	T0+12	x	TIMA	Release of the UGH software that reads `xcoach` format.
D511-V1	T0+18	x	LIP6	Release of the UGH software that writes `xcoach+` format.
D511-VF	T0+33	x	LIP6	Final release of the UGH software.

ST5-2 The goal of this sub-task is to integrate the GAUT HLS tool to the COACH framework. It consists of suppressing the C compiler and the SystemC and VHDL drivers and replacing them by `xcoach` and `xcoach+` drivers.

number	date	type	resp.	description
D520-VF	T0+12	x	UBS	Release of the GAUT software that is able to read `xcoach` format.
D521-VF	T0+18	x	UBS	Release of the GAUT software that is able to read `xcoach` format and to write `xcoach+` format.

ST5-3 The goal of this sub-task is to improve the UGH and GAUT HLS tools. UGH and GAUT experimentations have shown respectively usefull enhancements.

number	date	type	resp.	description
D530-VF	T0+24	x	TIMA	Release of the UGH software with support for treating automatically data dominated sections included into a control dominated application.
D531-VF	T0+27	x	TIMA	Release of the UGH software able to generate a micro-architecture without the variable binding currently done by the designer.
D532-VF	T0+24	x	UBS	Release of the GAUT software that supports the `xcoach` model during the binding and the scheduling steps.

D533-VF	T0+33	X	UBS	Release of the GAUT software that supports the `xcoach` model during the binding and the scheduling steps and also supports new constraints and objectives.
D534-V1	T0+24	D	UBS	Specification of a Design Space Exploration framework for the HAS Back-end: The high level specification tools, such as GAUT, have to be able to use synthesis feed-back informations in order to explore the design space and to generate optimized architectures.
D534-VF	T0+36	X	UBS	Release of the GAUT software that supports the features defined in D534-V1

ST5-4 In FPGA-SoC, the frequency is given by the processor(s) and the system BUS. The coprocessors generated by HLS synthesis must respect this frequency. However, the HLS tools can not guarantee that the micro-architectures they generate accurately respect this frequency. This is especially the case when the target is a FPGA device, because the delays are really known only after the RTL synthesis and that estimated delays used by the HLS are very inaccurate. The goal of this sub-task is to provide a tool that adapts the coprocessors frequency to the FPGA-SoC frequency after the coprocessor RTL synthesis.

number	date	type	resp.	description
D540-V1	T0+12	D	LIP6	A document describing the set up of the coprocessor frequency calibration.:
D540-V2	T0+24	X	LIP6	A VHDL description of hardware added to the coprocessor to enable the calibration.
D540-VF	T0+33	X	LIP6	The frequency calibration software consists of a driver in the FPGA-SoC operating system and of a control software.
D541-VF	T0+27	D	XILX	This deliverable consists in optimizing the VHDL description provided in D540. LIP6 will provide the VHDL description, XILINX will provide back a documentation listing that proposes VHDL generation enhancements.

4.3.6 Task 6: *PC/FPGA communication middleware*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
		part.		part.	part.	lead.	part.		

Objectives This task pools the features dedicated to HPC system design. It is described on figures 1 and 4. It consists in

- Providing a software tool that helps the HPC designer to find a good partition of the initial application (figure 4).
- specification of the communication schemes between the software part running on the PC and the FPGA-SoC.
- Implementing the communication scheme at all levels: partition help, software implementation both on the PC and in the operating system of the FPGA-SoC, hardware.
- Providing support for dynamic partial reconfiguration of XILINX FPGA in order to optimize FPGA ressource usage.

The low level hardware transmission support will be the PCI/X bus which allows high bit-rate transfers. The reasons of this choice are that both ALTERA and XILINX provide PCI/X IP for their FPGA and that GPU HPC softwares use also it.

ST6-1 This sub-task deals with the COACH HPC feature that consists in accelerating an existing apllication running on a PC by migrating critical parts into a SoC implemented on an FPGA plugged to the PC PCI/X bus. The main steps and components of this sub-task are:

- The definition of the communication middleware as a software API (Application Programing Interface) between the application part running on the PC and the application part running on the FPGA-SoC.
- A software for helping the end-user to partition applications (figure 4). This software is a library implementing the communication API with features to profile the partitioned application.
- The implementation of the communication API on the both sides (PC part and FPGA-SoC).

number	date	type	resp.	description
D610-VF	T0+6	D	BULL	Specification describing the API.
D611-VF	T0+12	X	LIP6	A library implementing the communication API defined in the D610-VF deliverable. This library is dedicated to help the end-user to partition an applicattion for HPC.
D612-VF	T0+21	X	LIP6	The PC part of the HPC communication API that comminicates with the FPGA-SOC, a library and probably a LINUX module.
D613-VF	T0+21	X	LIP6	The FPGA-SoC part of the communication API, a driver.
D614-VF	T0+24	X	TIMA	Port of the D613-VF driver on the DNA OS.
D615-VF	T0+33	X	LIP6	Bug corrections and enhancements of communication middleware (D610, D611, D612, D613, D614).

ST6-2 This sub-task deals with the implementation of hardware and SystemC modules required by the neutral architectural template for using the PCI/X IP of ALTERA and XILINX.

number	date	type	resp.	description
D620-VF	T0+18	H	TIMA	The synthesizable VHDL description of a PLB/VCI bridge and its corresponding SystemC model.
D621-VF	T0+18	H	LIP6	The synthesizable VHDL description of an AVALON/VCI bridge and its corresponding SystemC model.
D622-VF	T0+24	H	LIP6	The SystemC description of a component that generates PCI/X traffic. It is required to prototype FPGA-SoC dedicated to HPC.

ST6-3 This sub-task consists in integrating dynamic partial reconfiguration of XILINX FPGA in the CSG design flow. It also includes appropriate SoC-FPGA OS drivers and a modification of the profiling library.

number	date	type	resp.	description
D630-VF	T0+36	X	LIP6	Modification of the CSG software to support statically reconfigurable tasks.
D631-VF	T0+36	X	TIMA	This livrable is a CSG module allowing to partition the task graph along the dynamic partial reconfiguration regions. The resulting task-region assignement is directly used for generation of bitstreams. The module also produces reconfiguration management software to be run on the SoC-FPGA.

D632-VF	T0+30	X	TIMA	The drivers required by the DNA OS in order to manage dynamic partial reconfiguration inside the SoC-FPGA.
D633-VF	T0+36	X	LIP6	Port of the D632-VF drivers on the MUTEKH OS.
D634-VF	T0+36	X	TIMA	Extension of the HPC partionning helper in order to integrate dynamic partial reconfiguration dedicated features (reconfiguration time of regions, variable number of coprocessors).
D635-VF	T0+36	D	XILX	XILINX will work with TIMA in order to better take into account during partitioning decisions specific constraints due to partial reconfiguration process. The deliverable is a document describing the XILINX specific constraints.

4.3.7 Task 7: *Industrial demonstrators*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
						part.	lead.	part.	part.

Objectives This task groups the demonstrators of the COACH project. The demonstrators cover various domains and application types to drive the specification choices and to check most of the COACH features.

ST7-1 The application that BULL proposes is HPC oriented. The domain of the application is the treatment of medical images (image noise reduction and segmentation or registration). Our expectation from COACH project is to enhance the BULL HPC solutions that are currently based on multi-cores and GPUs with fine grain parallelism on FPGA.

number	date	type	resp.	description
D710-V1	T0+6	D	BULL	The deliverable is a document that describes the application that will be use as demonstrator.
D710-V2	T0+12	X	BULL	The deliverable is the specification of the demonstrator in COACH input format defined in the D210-VF deliverable.
D710-VF	T0+36	D	BULL	Validation of the demonstrator, the deliverable is a document describing the result of the experimentations.

ST7-2 The objective of this sub-task is to specify the THALES application and to develop the high level code. This application is in the domain of surveillance of critical infrastructures. The objective is to detect and classify the presence of humans in the restricted area. The algorithm is based on the work of Viola and Jones [39]. It implements in particular a cascade of classifiers operating on Haar like features, where simple weak classifiers at the beginning of the cascade reject a majority of void sub-windows, before more complex classifiers concentrate on potential regions of interest. This application is computation intensive and also makes an intensive use of binary decision trees to cascade the filters, which makes it a good candidate to assess the COACH platform.

Moreover, the higher levels of computing can involve tracking and data fusion between several camera streams and some other informations. The targeted system will be composed of one camera connected to a PC. All the computing part of the application is executed on a FPGA board connected to the PC.

number	date	type	resp.	description
D720-V1	T0+6	D	TRT	This deliverable is a document that specifies the application.
D720-VF	T0+12	X	TRT	This deliverable is the code of the application spcecified former deliverable (D720-V1).

ST7-3 THALES will use its internal software environment tool SPEAR DE to describe the application. The tool is able to partition and to generate the code for the target.
In this task, we will adapt SPEAR DE to generate the application description input of COACH framework. We will also describe the three templates of architecture in order to be able to partition the application on the architecture.

number	date	type	resp.	description
D730-VF	T0+18	X	TRT	Adaptation of SPEAR-DE for COACH framework.

ST7-4 In this sub-task, THALES will evaluate the COACH platform. In particular, THALES will verify its ability to generate a whole VHDL of an embedded system on FPGA for an application mixing control and data flow aspects. THALES will evaluate the performance of the generated system in terms of GOPS, and the design time from a high level description.

number	date	type	resp.	description
D740-V1	T0+24	D+X	TRT	This deliverable is a document describing the result got for the application (D720-V1) with SPEAR-DE (D730-VF) using COACH milestone of T0+18. The updated code of the application will be also provide.
D740-V2	T0+30	D+X	TRT	This deliverable is a document describing the result got for the application (D720-V1) with SPEAR-DE (D730-VF) using COACH milestone of T0+24. The updated code of the application will be also provide.
D740-VF	T0+36	D+X	TRT	This deliverable is a document that validates and evaluates COACH (final release) for the THALES demonstrators (D720-V1). The updated code of the application will be also provide.

ST7-5 FLEXRAS will design an application based on M-JPEG video standard. FLEXRAS will propose a SoC architecture integrating an embedded FPGA (eFPGA). The architecture is composed essentially of a processor, a bus and several RAMs. The embedded FPGA is connected to the bus and communicates with the other components. The (eFPGA) works in 2 modes:

Slave mode As a DMA, the processor will send the configuration bitstream stored on the RAM to the eFPGA. In this mode, the eFPGA is considered as a writeable memory and is configured by the processor.

Master mode Once the FPGA is programmed, it becomes a coprocessor achieving the aimed task.

The top architecture of this SoC based-platform will be generated using COACH framework. The application that will be run on the SoC corresponds initially to a graph of software tasks. Critical tasks will be identified and transformed automatically to hardware tasks using COACH high level synthesis feature. While software tasks will be run on the processor, hardware ones will be mapped on eFPGA to take advantage of its optimized resources and parallelism. FLEXRAS provides all the flow from RTL synthesis to bitstream generation.

number	date	type	resp.	description
D750-VF	T0+6	D	FLEX	FLEXRAS will use IPs provided by LIP6 (vhdl models of SoCLIB) and its eFPGA IP to generate the SoC architecture. This deliverable is a document that describes this architecture.

D751-VF	T0+18	H	FLEX	FLEXRAS has to adapt the eFPGA interface to connect it to the VCI bus. This deliverable is a VHDL description.
D752-VF	T0+24	X	FLEX	FLEXRAS will propose to test COACH framework and the FLEXRAS architecture template throught an application based on M-JPEG video standard. This applicattion will containt 3 communicating tasks under the COACH format specified in D210 deliverable. The first one is a hardware task generated by the HAS tools and transformed into a bit stream by the FLEXRAS tools. The second is a bitstream loader that will load the bitstream of the first task on the eFPGA. The third is a software task that communicates with the hw task for testing it.
D753-VF	T0+30	X	FLEX	This deliverable is a file under the format defined by the deliverable D250-VF that characterizes the eFPGA. This will allow the COACH HLS tools to take into account the eFPGA delays.
D754-VF	T0+36	D	FLEX	This deliverable is a document that describes the tests, the validation and the evaluation of COACH with the FLEXRAS architecture and tools.

ST7-6 The NAVTEL-SYSTEM Embedded Supper Computing (ESC) project is based on simple hardware but tightly coupled module between a embedded processor and an FPGA both on a board. By using the COACH environment, NAVTEL-SYSTEM will automatically synthetize two cores: one for software radio through a polyphase resampler and one for an industrial control application through an embedded PID controller. The objective is to sequence the cores in realtime in FPGA using partial configuration methods proposed in the COACH project. This will allow us to gain experience on automatic multi core sequencing at system level. The specification for our first work package will concern this aspect.

The ESC can function on different topologies: Single, parallel or Grid computing modes for industrial and scientific applications. The processor and FPGA configuration also facilitate the co-simulation which allows to gain time on the development and integration phase. The architecture consists of a wrapper that encapsules computing units depending on the application and a real time kernal for task switching and partial reconfiguration of FPGA on run time environment.

To day NAVTEL-SYSTEM develops these computing units manually. NAVTEL-SYSTEM expects to benefit from the COACH project especially the HLS tools for generating the computing unit.

number	date	type	resp.	description
D760-VF	T0+6	D	NAV	A document that will define the requirements for automatic RTL generation for signal processing units of our market sector such as digital communication, imaging and industrial control. This document will include the description of some already handmade processing units.
D761-VF	T0+18	H	NAV	The adaptation of our wrapper to support coprocessor generated by COACH.

D762-VF	T0+36	D	NAV	NAVTEL-SYSTEM will test the COACH HLS tools on the processing units that are described in the D760-VF deliverable. A document will be written that describes the results obtained taking into account: 1) the performance in terms of space, 2) the performance in terms of time, 3) the friendlyness of the environment.

4.3.8 Task 8: *Dissemination*

INRI	LIP	TIMA	UBS	LIP6	XILX	BULL	TRT	NAV	FLEX
part.	part.	part.	part.	lead.	part.	part.	part.	part.	part.

Objectives This task relates to the diffusion of the project results. The objective is to ensure the COACH dissemination by publishing on a public WEB site all the information that a COACH user requires. The main information features are:

- The COACH releases (milestones and final release) and their associated installation manuals.
- The COACH user reference manual.
- The user manual of the various tools.
- A COACH tutorial.
- The conference publication.
- A user wiki.

ST8-1 This sub-task relates to the management of the WEB site and to the distribution of the COACH releases.

number	date	type	resp.	description
D810-V1	T0+6	D	LIP6	This deliverable consists firstly in providing a WEB site (name, HTTP server setup, wiki) and secondly in defining the site map and finally in writting and installing the pages.
D810-VF	T0+36	D	LIP6	This deliverable corresponds to the standard management of a WEB site (modifying, adding, suppressing, replacing pages). Especialy the user reference manuals provided in the other tasks will be published on this site. The published articles will be also be installed in this site.
D811-VF	T0+36	D+X	LIP6	This deliverable deals with the elaboration of the COACH software milestones and final releases with their installation manuals and to publish then into the WEB site.

ST8-2 This sub-task consists of making a COACH tutorial and to publish it on the public WEB site. The tutorial example will also be used as reference demonstrator of the framework. The application of this tutorial will be a Motion JPEG application.

number	date	type	resp.	description
D820-V1	T0+6	X	LIP6	Choice of the application and its implementation as a C/C++ program.
D820-V2	T0+12	D+X	LIP6	The application is split into two communicating parts, the PC part and FPGA-SoC part. By using the features the T0+12 milestone provides, the tutorial describes how this efficient partionning was obtained. The FPGA-SoC part is described as communicating task graph. The tutorial also describes how a promising task graph can be obtained.
D820-V3	T0+24	D	LIP6	This tutorial shows how a task can be migrated to coprocessor using HAS tools and how FPGA-SoC can be generated and run to FPGA. This for HAS tools and and architectural template available in T0+24 milestone.
D820-VF	T0+36	D	LIP6	The final release of the tutorial.
D821-VF	T0+33	D	XILX	XILINX will check that the developped tutorial works well with XILINX tools, and will propose corrections or enhancements if needed into a document.

ST8-3 This sub-task consists of making the COACH user reference manuals. They will be published on the public WEB site.

number	date	type	resp.	description
D830-VF	T0+24	D	TIMA	This user manual shows how to generate a complete HW/SW system by using CSG tool.
D831-VF	T0+24	D	LIP	This user manual shows how to apply loop transformations to a task.
D832-VF	T0+36	D	INRI	This user manual shows how to customize a processor to obtain an ASIP.
D833-VF	T0+24	D	UBS	This user manual shows how a task can be synthesized by using HLS tools developped in the COACH project.

4.4 Tasks schedule, deliverables and milestones

The figures 6 & 7 present the Gantt diagram of the project. Before the final release (T0+36), there are 4 milestones (red lines on the figures) at $T0+6$, $T0+12$, $T0+18$ and $T0+24$ that are rendez-vous points of the precedent deliverables.

Milestone 1 ($T0+6$) Specification of COACH inputs, of the `xcoach` format and of the demonstators as a reference software.

Milestone 2 ($T0+12$) The first COACH release. At this step the demonstrators are written in the COACH input format. This COACH release allows to prototype and to generate the FPGA-SoC. The main restrictions are: 1) Only the neutral architectural template is supported, 2) HAS is not available (but prototyping with virtual coprocessors is available), 3) Enhanced communication schemes are not available. 4) ASIP compilation flow is not available.

Milestone 3 ($T0+18$) The second COACH release. At this step most of the COACH features are availables. A preliminary version of the ASIP synthesis flow is supported, for a simple extensible MIPS model. The main restriction is that COACH can not yet generate FPGA-SoC for ALTERA and XILINX architectural templates. The others restriction is that the HAS tools are not yet fully operational.

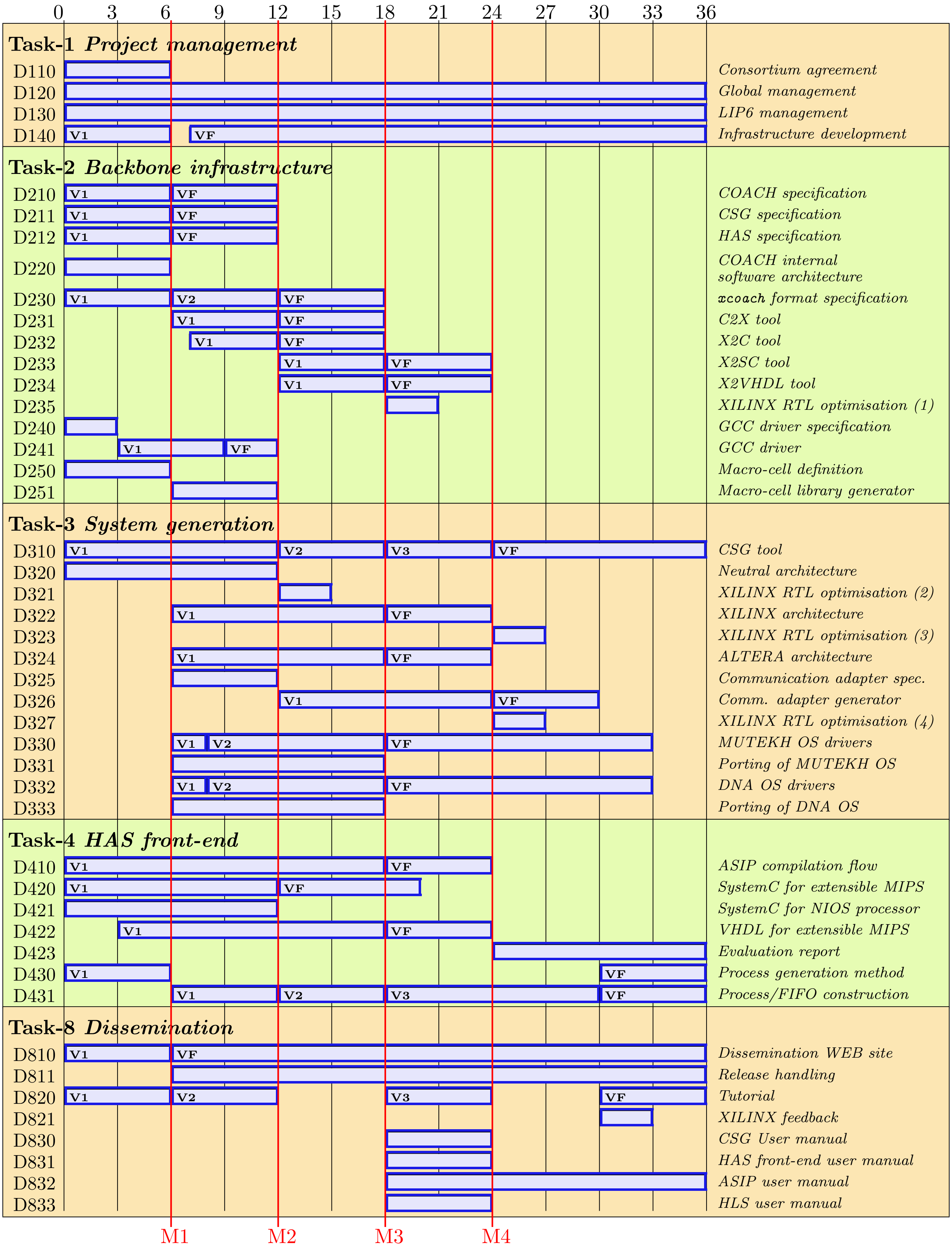


Figure 6: Gantt diagram of deliverables (task-1 to task-4 and task-8)

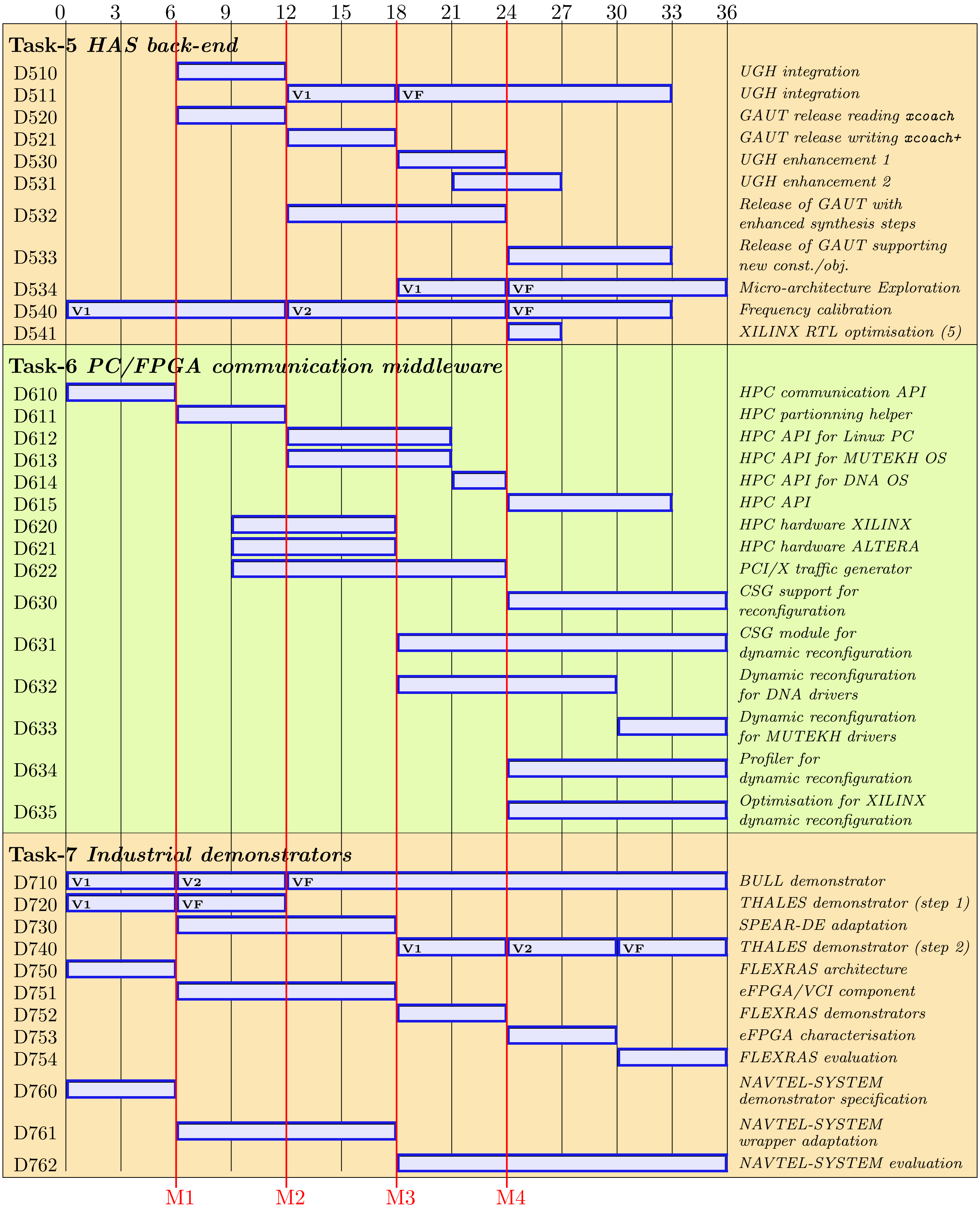


Figure 7: Gantt diagram of deliverables (task-5, task-6 and task-7)

Milestone 4 ($T0 + 24$) The pre-release of the COACH project. The full design flow is supported. The main restriction are: 1) The backend HAS tools have not been yet enhanced, 2) Dynamic partial reconfiguration is not supported, 3) NIOS processor instruction set extension is supported, but only for user specified patterns.

Final Release ($T0 + 36$)

This organisation allows the project to globally progress step by step mixing development and demonstrator deliverables. Hence, demonstrator feed-back will arrive early and so the risk to point out incompatibility at the integration phase is significantly reduced.

The risks that have been identified at the beginning of the project are the following:

`xcoach` **format (D230, D231)** Partners have to agree on a convenient exchange format for all tools involved. Because all the HAS tools rely on it, the `xcoach` format specification is a crucial step. There are no work-around but as mentionned in section 4.1 (page 17) the five academic partners have worked on it for a full year and a preliminary document already exists.

Virtual prototyping of ALTERA & XILINX architectural templates (D324-V1, D322-V1) The SoCLib component library contains several SystemC models used for the virtual prototyping of the ALTERA and XILINX architectural templates (NIOS and Microblaze processor cores). Nevertheless, at this time we do not know how many IP cores SystemC simulation models have to be developped. If the workload of this simulation model development is too important, virtual prototyping of those architectural templates will not be directly supported. The three architectural templates being quite similar, the virtual prototyping will use the neutral architectural template.

VCI/AVALON & VCI/PLB bridges (D621, D620) If one of these tasks is impossible or too important or leads to inefficiency, it will be abandoned. In this case, the neutral architectural template will not be available for HPC and a SystemC VCI model corresponding to the PCI/X IP will be developped to allow virtual prototyping.

Finally the list of all the deliverables is presented on figure 8.

5 Dissemination and exploitation of results. Management of intellectual property

5.1 Dissemination

The COACH project will bring new scientific results in various fields, such as high level synthesis, hardware/software codesign, virtual prototyping, hardware oriented compilation techniques, automatic parallelisation, etc. These results will be published in relevant International Conferences, namely DATE, DAC, or ICCAD.

More generally, the COACH infrastructure and the design flow supported by the COACH tools and libraries will be promoted by proposing tutorials on FPGA oriented system level synthesis in various worshops and conferences (DATE, DAC, CODES+ISSS...).

Several COACH partners being members of the HiPEAC European Network of Excellence (High Performance and Embedded Architecture and Compilation), courses will be proposed for the HiPEAC summer school on Advanced Computer Architecture and Compilation for Embedded Systems.

Following the general policy of the SoCLib platform, the COACH project will be an open infrastructure, and the COACH tools and libraries will be available in the framework of the SoCLib WEB server. This server will be maintened by the UPMC/LIP6 laboratory.

number	resp.	T0+	kind	description
D110	LIP6	6	d	Consortium agreement
D120	LIP6	36	d	Global management
D130	LIP6	36		LIP6 management
D140-V1	LIP6	6	x	Infrastructure development
D140-VF	LIP6	36	x	
D210-V1	LIP6	6	d	COACH specification
D210-VF	LIP6	12	d	
D211-V1	TIMA	6	d	CSG specification
D211-VF	TIMA	12	d	
D212-V1	UBS	6	d	HAS specification
D212-VF	UBS	12	d	
D220	LIP6	6	d	COACH internal software architecture
D230-V1	LIP	6	d+x	`xcoach` format specification
D230-V2	LIP	12	d+x	
D230-VF	LIP	18	d+x	
D231-V1	UBS	12	x	C2X tool
D231-VF	UBS	18	x	
D232-V1	UBS	12	x	X2C tool
D232-VF	UBS	18	x	
D233-V1	LIP6	18	x	X2SC tool
D233-VF	LIP6	24	x	
D234-V1	UBS	18	x	X2VHDL tool
D234-VF	UBS	24	x	
D235	XILX	21	d	XILINX RTL optimisation (1)
D240	UBS	3	d	GCC driver specification
D241-V1	UBS	9	x	GCC driver
D241-VF	UBS	12	x	
D250	UBS	6	d	Macro-cell definition
D251	UBS	12	x	Macro-cell library generator
D310-V1	LIP6	12	x	CSG tool
D310-V2	LIP6	18	x	
D310-V3	LIP6	24	x	
D310-VF	LIP6	36	x	
D320	LIP6	12	h	Neutral architecture
D321	XILX	15	d	XILINX RTL optimisation (2)
D322-V1	TIMA	18	x	XILINX architecture
D322-VF	TIMA	24	h	
D323	XILX	27	d	XILINX RTL optimisation (3)
D324-V1	INRI	18	x	ALTERA architecture
D324-VF	INRI	24	h	
D325	UBS	12	d	Communication adapter spec.
D326-V1	UBS	24	x	Comm. adapter generator
D326-VF	UBS	30	x	
D327	XILX	27	d	XILINX RTL optimisation (4)
D330-V1	LIP6	8	x	MUTEKH OS drivers
D330-V2	LIP6	18	x	
D330-VF	LIP6	33	x	
D331	LIP6	18	x	Porting of MUTEKH OS
D332-V1	TIMA	8	x	DNA OS drivers
D332-V2	TIMA	18	x	
D332-VF	TIMA	33	x	
D333	TIMA	18	x	Porting of DNA OS
D410-V1	INRI	18	x	ASIP compilation flow
D410-VF	INRI	24	x	
D420-V1	INRI	12	x	SystemC for extensible MIPS
D420-VF	INRI	20	x	
D421	INRI	12	x	SystemC for NIOS processor
D422-V1	INRI	18	h	VHDL for extensible MIPS
D422-VF	INRI	24	h	
D423	INRI	36	d	Evaluation report
D430-V1	LIP	6	d	Process generation method
D430-VF	LIP	36	d	
D431-V1	LIP	12	x	Process/FIFO construction
D431-V2	LIP	18	x	
D431-V3	LIP	30	d+x	
D431-VF	LIP	36	x	
D510	TIMA	12	x	UGH integration
D511-V1	LIP6	18	x	UGH integration
D511-VF	LIP6	33	x	
D520	UBS	12	x	GAUT release reading `xcoach`
D521	UBS	18	x	GAUT release writing `xcoach+`
D530	TIMA	24	x	UGH enhancement 1
D531	TIMA	27	x	UGH enhancement 2
D532	UBS	24	x	Release of GAUT with enhanced synthesis steps
D533	UBS	33	x	Release of GAUT supporting new const./obj.
D534-V1	UBS	24	d	Micro-architecture Exploration
D534-VF	UBS	36	x	
D540-V1	LIP6	12	d	Frequency calibration
D540-V2	LIP6	24	x	
D540-VF	LIP6	33	x	
D541	XILX	27	d	XILINX RTL optimisation (5)
D610	BULL	6	d	HPC communication API
D611	LIP6	12	x	HPC partionning helper
D612	LIP6	21	x	HPC API for Linux PC
D613	LIP6	21	x	HPC API for MUTEKH OS
D614	TIMA	24	x	HPC API for DNA OS
D615	LIP6	33	x	HPC API
D620	TIMA	18	h	HPC hardware XILINX
D621	LIP6	18	h	HPC hardware ALTERA
D622	LIP6	24	h	PCI/X traffic generator
D630	LIP6	36	x	CSG support for reconfiguration
D631	TIMA	36	x	CSG module for dynamic reconfiguration
D632	TIMA	30	x	Dynamic reconfiguration for DNA drivers
D633	LIP6	36	x	Dynamic reconfiguration for MUTEKH drivers
D634	TIMA	36	x	Profiler for dynamic reconfiguration
D635	XILX	36	d	Optimisation for XILINX dynamic reconfiguration
D710-V1	BULL	6	d	BULL demonstrator
D710-V2	BULL	12	x	
D710-VF	BULL	36	d	
D720-V1	TRT	6	d	THALES demonstrator (step 1)
D720-VF	TRT	12	x	
D730	TRT	18	x	SPEAR-DE adaptation
D740-V1	TRT	24	d+x	THALES demonstrator (step 2)
D740-V2	TRT	30	d+x	
D740-VF	TRT	36	d+x	
D750	FLEX	6	d	FLEXRAS architecture
D751	FLEX	18	h	eFPGA/VCI component
D752	FLEX	24	x	FLEXRAS demonstrators
D753	FLEX	30	x	eFPGA characterisation
D754	FLEX	36	d	FLEXRAS evaluation
D760	NAV	6	d	NAVTEL-SYSTEM demonstrator specification
D761	NAV	18	h	NAVTEL-SYSTEM wrapper adaptation
D762	NAV	36	d	NAVTEL-SYSTEM evaluation
D810-V1	LIP6	6	d	Dissemination WEB site
D810-VF	LIP6	36	d	
D811	LIP6	36	d+x	Release handling
D820-V1	LIP6	6	x	Tutorial
D820-V2	LIP6	12	d+x	
D820-V3	LIP6	24	d	
D820-VF	LIP6	36	d	
D821	XILX	33	d	XILINX feedback
D830	TIMA	24	d	CSG User manual
D831	LIP	24	d	HAS front-end user manual
D832	INRI	36	d	ASIP user manual
D833	UBS	24	d	HLS user manual

Figure 8: All the deliverables

5.2 Exploitation of results

The main goal of the COACH project is to help SMEs (Small and Medium Enterprises) to enter the world of MPSoC technologies. For small companies, the cost is a primary concern. Moreover, these companies have not always in-home expertise in hardware design and VHDL modelling. As the fabrication costs of an ASIC is generally too high for SMEs, the COACH project focus on FPGA technologies. Regarding the design tools, the cost of advanced ESL (Electronic System Design) tools is an issue, and the COACH project will follow the same general policy as the SoCLib platform :

- All software tools supporting the COACH design flow will be available as free software. All academic partners contributing to the COACH project agreed to distribute the ESL software tools under the same GPL license as the SoCLib tools.
- The SystemC simulation models for the hardware components used by the SoCLib architectural template will be distributed as free software under a non-contaminant LGPL license.
- The synthesizable VHDL models supporting the neutral architectural template (corresponding to the SocLib IP cores library), will have two modes of dissemination. A typical MPSoC contains not only dedicated, synthesized coprocessors. It contains also general purpose, reusable components, such as processor cores, memory controllers optimised cache controllers, peripheral controllers, or bus controllers. For non commercial use (i.e. research or education in an academic context, or feasbility study in an industrial context), the synthesizable VHDL models will be freely available. For commercial use, commercial licenses will be negociated between the owners and the customers.
- The proprietary ALTERA, XILINX and FLEXRAS IP core libraries are commercial products that are not involved by the free software policy, but these libraries will be supported by the synthesis tools developped in the COACH project.

This general approach is supported by a large number (10) of SMEs, as demonstrated by the ”letters of interest” that have been collected during the preparation of the project and presented in annexe B.

5.3 Indusrial Interest in COACH

Partner: *BULL*

The team of BULL participating to the COACH project is from the Server Development Department who is in charge of developing hardware for open servers (e.g. NovaScale) and HPC solutions. The main expectation from COACH is to derive a new component (fine-grain FPGA parallelism) to add to existing Bull HPC solutions.

Partner: *XILINX*

Computing power potential of our FPGA architectures growing very quickly on one side, and complexity of designs implemented using our FPGAs dramatically increasing on the other side, it is very interesting for us to get high level design methodologies progressing quickly and targetting our FPGAs in the most possible efficient way.

XILINX goal is to get COACH to generate bitstream optimized as much as possible for XILINX FPGAs in order to both, validate the methodology on our FPGA families, and ease future work of our customers.

Partner: *THALES*

THALES has two main reasons to use the COACH platform:

- The huge increase of the complexity of the systems in particular by their heterogeneity, raises the issues of design cost and time in the same proportion. The divisions need a design tool which supports the implementation of the applications from algorithm description to the executable code on platforms composed of several general purpose processors and dedicated IPs.
- The applications are more and more complex and adaptable to the environment which leads to a mixture of control aspects and data stream computing aspects. A new approach is necessary to be able to describe this type of application and manage the high level synthesis of system embedding control and data flow aspects.

TRT (Thales Research and Technology) has the mission to assess and de-risk the emerging technologies in its domains of expertise. Specifically in COACH, the studied technology is a method and associated tools to make the bridge between application capture at system level and the implementation on heterogeneous distributed computing architectures. The main stake for Thales behind this is the future design process that will be applied to its system teams in the future for the computation-intensive sensor applications. In a context of very instable market of tools for parallel programming, it is important to experiment and demonstrate the candidate technologies.

In its role of internal dissemination, TRT will make the demonstration of the full design flow within Thales, and will keep available a platform to later evaluate additional applications coming from the Business Units.

The COACH platform will be used in the new THALES products in which the algorithms are more and more dependent of the environment and have to permanently adapt their behavior in varying environments. The target markets are the critical infrastructures security and border monitoring.

Partner: *FLEXRAS*

FLEXRAS is developing a new architecture for embedded system. Our interest in using COACH are:

- firstly, to validate our new architecture by emulating it with COACH.
- Secondly, to use this emulator and the COACH potential to quickly setup demonstrator to our customer.

Partner: *NAVTEL-SYSTEM*

NAVTEL-SYSTEM has a platform for high performence computation based on ARM processor and FPGAs that embedde coprocessors. Currently, the coprocessors are handmade and their designs constitute an important part of our product cost. We have try free HLS tools to diminish them but the quality of the generated designs was not sufficient to be useable. So our interest in COACH is mainly the HLS tools.

Industrial supports

The following SMEs demonstrate interest to the COACH project (see the ”letters of interest” in annexe B) and will follow the COACH evolution and will evaluate it: ALTERA Corporation (page 55), ADACSYS (page 57), MAGILLEM Design Services (page 58), INPIXAL (page 59), CAMKA System (page 60), ATEME (page 61), ALSIM Simulateur (page 62), SILICOMP-AQL (page 63), ABOUND Logic (page 64), EADS-ASTRIUM (page 65).

5.4 Management of Intellectual Property

A global consortium agreement will be defined during the first six monts of the project. As already stated, the COACH project has been prepared during one year by a monthly meeting involving the five academic partners. The general free software policy described in the previous section has been agreed by academic partners and has been approved by all industrial participants. This free software policy will simplify the definition of the consortium agreement.

6 Consortium Description

6.1 Partners description & relevance, complementarity

6.1.1 INRIA/CAIRN

INRIA, the French national institute for research in computer science and control, operating under the dual authority of the Ministry of Research and the Ministry of Industry, is dedicated to fundamental and applied research in information and communication science and technology (ICST). The Institute also plays a major role in technology transfer by fostering training through research, diffusion of scientific and technical information, development, as well as providing expert advice and participating in international programs.

By playing a leading role in the scientific community in the field and being in close contact with industry, INRIA is a major participant in the development of ICST in France. Throughout its eight research centres in Rocquencourt, Rennes, Sophia Antipolis, Grenoble, Nancy, Bordeaux, Lille and Saclay, INRIA has a workforce of 3 800, 2 800 of whom are scientists from INRIA and INRIA's partner organizations such as CNRS (the French National Center for Scientific Research), universities and leading engineering schools. They work in 168 joint research project-teams. Many INRIA researchers are also professors and approximately 1 000 doctoral students work on theses as part of INRIA research project-teams.

The CAIRN group of INRIA Rennes – Bretagne Atlantique study reconfigurable system-on-chip, i.e. hardware systems whose configuration may change before or even during execution. To this end, CAIRN has 13 permanent researchers and a variable number of PhD students, post-docs and engineers. CAIRN intends to approach reconfigurable architectures from three angles: the invention of new reconfigurable platforms, the development of associated transformation, compilation and synthesis tools, and the exploration of the interaction between algorithms and architectures. CAIRN is a joint team with CNRS, University of Rennes 1 and ENS Cachan.

6.1.2 ENS Lyon/LIP/Compsys

The Compsys group of Ecole Normale Supérieure de Lyon is a project-team of INRIA Rhône-Alpes and a part of Laboratoire de l'Informatique du Parallélisme (LIP), UMR 5668 of CNRS. It has four permanent researchers and a variable number of PhD students and post-docs. Its field of expertise is compilation for embedded system, optimizing compilers and automatic parallelization. Its members were among the initiators of the polyhedral model for automatic parallelization and program optimization generally. It has authored or contributed to several well known libraries for linear programming, polyhedra manipulation and optimization in general. It has strong industrial cooperations, notably with ST Microelectronics and THALES.

6.1.3 TIMA

The TIMA laboratory ("Techniques of Informatics and Microelectronics for integrated systems Architecture") is a public research laboratory sponsored by Centre National de la Recherche Scientifique (CNRS, UMR5159), Grenoble Institute of Technology (Grenoble-INP) and Université Joseph Fourier (UJF). The research topics cover the specification, design, verification, test, CAD tools and design methods for integrated systems, from analog and digital components on one end of the spectrum, to multiprocessor Systems-on-Chip together with their basic operating system on the other end.

Currently, the lab employs 124 persons among which 60 PhD candidates, and runs 32 ongoing French/European funded projects. Since its creation in 1984, TIMA funded 7 startups, patented 36 inventions and had 243 PhD thesis defended.

The System Level Synthesis Group (25 people including PhDs) is involved in several FP6, FP7, CATRENE and ANR projects. Its field of expertise is in CAD and architecture for Multiprocessor SoC and Hardware/Software interface.

6.1.4 LAB-STICC

The Lab-STICC (Laboratoire des Sciences et Techniques de l'Information, de la Communication, et de la Connaissance), is a French CNRS laboratory (UMR 3192) that groups 4 research centers in the west and south Brittany: the Université de Bretagne-Sud (UBS), the Université de Bretagne Occidentale (UBO), and Telecom Bretagne (ENSTB). The Lab-STICC is composed of three departments: Microwave and equipments (MOM), Digital communications, Architectures and circuits (CACS) and Knowledge, information and decision (CID). The Lab-STICC represents a staff of 279 peoples, including 115 researchers and 113 PhD students. The scientific production during the last 4 years represents 20 books, 200 journal publications, 500 conference publications, 22 patents, 69 PhDs diploma.

The UBS/Lab-STICC laboratory is involved in several national research projects (e.g. RNTL : SystemC'Mantic, EPICURE - RNRT : MILPAT, ALIPTA, A3S - ANR : MoPCoM, SoCLib, Famous, RaaR, AFANA, Open-PEOPLE, ICTER ...), CMCU project (COSIP) and regional projects (e.g. ITR projects PALMYRE ...). It is also involved in European Project (e.g. ITEA/SPICES, IST/AETHER ...). These projects are conducted through tight cooperation with national and international companies and organizations (e.g. France Telecom CNET, MATRA, CEA, ASTRIUM, THALES Com., THALES Avionics, AIRBUS, BarCo, STMicroelectronics, Alcatel-Lucent ...). Results of those or former projects are for example the high-level synthesis tool GAUT, the UHLS syntax and semantics-oriented editor, the DSP power estimation tool Soft-explorer or the co-design framework Design Trotter.

The CACS department of the Lab-STICC (also referred as UBS/Lab-STICC), located in Lorient, is involved in COACH. The UBS/Lab-STICC is working on the design of complex electronic systems and circuits, especially but not exclusively focussing on real-time embedded systems, power and energy consumption optimization, high-level synthesis and IP design, digital communications, hardware/software co-design and ESL methodologies. The application targeted by the UBS/Lab-STICC are mainly from telecommunication and multimedia domains which enclose signal, image, video, vision, and communication processing.

6.1.5 LIP6

University Pierre et Marie Curie (UPMC) is the largest university in France (7400 employees,38000 students). The Laboratoire d'Informatique de Paris 6 (LIP6) is the computer science laboratory of UPMC, hosting more than 400 researchers, under the umbrella of the CNRS (Centre National de la Recherche Scientifique). The System on Chip Department of LIP6 consists of 80 people, including 40 PHD students. The research focuses on CAD tools and methods for VLSI and System on Chip design. The annual budget is about 3 M€, and 1.5 M€ are from research contracts. The SoC department has been involved in several european projects :IDPS, EVEREST, OMI-HIC, OMI-MACRAME, OMI-ARCHES, EUROPRO, COSY, Medea SMT, Medea MESA, Medea+ BDREAMS, Medea+ TSAR.

The public domain VLSI CAD system ALLIANCE, developped at UPMC is installed in more than 200 universities worldwide. The LIP6 is in charge of the technical coordination of the SoCLib national project, and is hosting the SoCLib WEB server. The SoCLib DSX component was designed and developped in our laboratory. It allows design space exploration and will the base of the *CSG*

COACH tools. Moreover, the LIP6 developped during the last 10 years the UGH tool for high level synthesis of control-dominated coprocessors. This tool will be modified to be integrated in the COACH design flow.

Even if the preferred dissemination policy for the COACH design flow will be the free software policy, (following the SoCLib model), the SoC department is ready to support start-ups : Six startup companies (including FLEXRAS) have been created by former researchers from the SoC department of LIP6 between 1997 and 2002.

6.1.6 XILINX

XILINX is the world leader in the domain of programmable logic circuits (FPGA). XILINX develops on one hand several FPGA architectures (CoolRunner, Spartan and Virtex families) and on the other hand a software solution allowing exploiting the characteristics of these FPGA.

The tools proposed allow the designer to describe his architecture from a modeling language (VHDL/Verilog) to an optimized architecture implemented to the selected technology. The team located at Grenoble is responsible of the logic synthesis tool development (XST) of the software solution, which aggregates all the steps allowing proceeding from a HDL model to a technological netlist:

- Compilation of HDL code and model generation at Register Transfer Level (RTL).
- RTL model optimizations.
- Inference and generation of optimized macro blocks (Finite states machine, counter).
- Boolean equations generation for random logic.
- Logical, mapping and timing optimizations.

The architectures developed by XILINX offer a collection of technological primitives (variable complexity) from simple Boolean generators (LUT) to complex DSP blocks or memory and even configurable processor cores (Pico and MicroBlaze families). This kind of architecture allows, therefore, the designer to validate different hardware/software possibilities in a High Level Synthesis (HLS) framework.

The classical optimization techniques focus, mainly, on the frequency aspects and on available resources use. The optimizations, taking into account the consumption criteria, become critical due to the fact of the increase of the architecture complexity and due to the use of FPGA component for low power applications.

6.1.7 BULL

BULL designs and develops servers and software for an open environment, integrating the most advanced technologies. It brings to its customers its expertise and know-how to help them in the transformation of their information systems and to optimize their IT infrastructure and their applications.

BULL is particularly present in the public sector, banking, finance, telecommunication and industry sectors. Capitalizing on its wide experience, the Group has a thorough understanding of the business and specific processes of these sectors, thus enabling it to efficiently advise and to accompany its customers. Its distribution network spreads to over 100 countries worldwide.

The team participating to the COACH project is from the Server Development Department based in Les Clayes-sous-Bois, France. The SD Department is in charge of developing hardware for open servers (e.g. NovaScale) and HPC solutions. Its main activities range from architecture specification, ASIC design/verification/prototyping to board design and include also specific EDA development to complement standard tools.

6.1.8 THALES

THALES is a world leader for mission critical information systems, with activities in 3 core businesses: aerospace (with all major aircraft manufacturers as customers), defence, and security (including ground transportation solutions). It employs 68000 people worldwide, and is present in 50 countries. THALES Research & Technology operates at the corporate level as the technical community network architect, in charge of developing upstream and THALES-wide R & T activities, with vision and visibility. In support of THALES applications, TRT's mission is also to anticipate and speed up technology transfer from research to development in Divisions by developing collaborations in R&T. THALES is international, but Europe-centered. Research & Development activities are disseminated, and corporate Research and Technology is concentrated in Centres in France, the United Kingdom and the Netherlands. A key mission of our R&T centres is to have a bi-directional transfer, or ”impedance matching” function between the scientific research network and the corresponding businesses. The TRT's Information Science and Technology Group is able to develop innovative solutions along the information chain exploiting sensors data, through expertise in: computational architectures in embedded systems, typically suitable for autonomous system environments, mathematics and technologies for decision involving information fusion and cognitive processing, and cooperative technologies including man system interaction.

The Embedded System Laboratory (ESL) of TRT involved in the COACH project is part of the Information Science and Technology Group. Like other labs of TRT, ESL is in charge of making the link between the needs from THALES business units and the emerging technologies, in particular through assessment and de-risking studies. It has a long experience on parallel architectures design, in particular on SIMD architectures used for image processing and signal processing applications and on reconfigurable architectures. ESL is also strongly involved in studies on programming tools for these types of architectures and has developed the SpearDE tool used in this project. The laboratory had coordinated the FP6 IST MORPHEUS project on reconfigurable technology, being highly involved in the associated programming toolset. The team is also involved in the FP6 IST FET AETHER project on self-adaptability technologies and coordinates national projects on MPSoC architecture and tools like the Ter@ops project (Pôle de Compétitivité System@tic) dedicated to the design of a MPSoC for intensive computing embedded systems.

6.1.9 FLEXRAS

FLEXRAS is an innovative start-up specialized in the conception of configurable circuits and the development of CAD tools. FLEXRAS provides a complete front-to-back-end generator of ”hardware” reprogrammable IP cores that can be embedded in ASIC and ASSP SoC designs. FLEXRAS solution is based on a patented FPGA architecture delivering an unprecedented level of logic density. This high capacity is accessible using a traditional RTL flow from Verilog/VHDL synthesis all the way to bitstream generation.

FLEXRAS is a spin-off from LIP6 (Laboratoire Informatique Paris 6) and was awarded at the French National Competition for Business Startup and Innovative Technology in 2007 and 2009 in ”emergence” and ”creation” categories respectively.

6.1.10 NAVTEL-SYSTEM

NAVTEL-SYSTEM was created in 1994 to develop flexible systems based on FPGAs and currently focuses on intelligent signal mining for knowlege based signal processing systems. The company main activity covers the following domains: satellite communication, aeronautics, imaging and security. NAVTEL-SYSTEM dedicates about 70% of its activity to client projects in satellite, aeronautical and imaging systems and 30% to its own research programmes in collaboration with French and international partners.

The multi disciplinary technical team comprises 6 engineers for signal processing and hardware development and one technician.

NAVTEL-SYSTEM has its own Ph.D program which includes in the past (classification technology and MIMO for FPGA implementation) and currently the preparation of a project for remote sensing with signal intelligence for satellite application. The company participates in national and European level projects contributing to a strategic alliance between academic and industrial partners.
The current research covers particle filter applications for communication and RADAR, Cognitive Radio, Satellite communication, embedded super computing and focuses on low power algorithms for implementation in FPGA and soft computing.

For manufacturing and industrialization, NAVTEL-SYSTEM works with ISO certified partners. The company clients include the CNES, Thalès Alenia Space, Thalès Communication, EADS, Eutelsat, AIRBUS, Schlumberger. NAVTEL-SYSTEM participates from the R&D phase up to the system delivery.

Recognitions:

- HEC Challenge+ program for innovative projects (promotion 9)
- Innovation and technology development Trophées Région Centre
- Recognition by the French Senate for company creation during the Semaine de l'entrepreneur 2005.

6.2 Relevant experience of the project coordinator

The COACH project will be coordinated by professor Alain Greiner from Université Pierre et Marie Curie. Alain Greiner is the initiator and the main architect of the SoCLib project. This ANR platform for virtual prototyping of MPSoCs involved 6 industrial companies (including ST Microelectronics and THALES) and ten academic laboratories (5 of them are involved in the COACH project). The SoCLib project was managed by THALES, but the technical coordination has been done by Alain Greiner, who has a good experience in coordinating large technical projects in both industrial and academic contexts:

- He received the "Docteur es Sciences" degree from University Denis DIDEROT in 1982 after working six years at Commissariat a l' Energie Atomique.
- From 1986 to 1990, he worked for the french BULL company, as team leader, in charge of designing the Basic Processing Unit for the BULL DPS7000 computer, the most powerfull mainframe of the family.
- In 1990, Alain Greiner joined UPMC, as Professor and became the head of the MASI laboratory in 1994. From 2000, he was head of the Hardware Department of the LIP6 laboratory.
- From 1990 to 2000, he was the leader of the the ALLIANCE project: This GPL based cooperative project developped a public domain VLSI/CAD system that has been used in more than 200 universities worlwide, for education and research. This project obtained the Seymour Cray award in 1994.

7 Scientific justification for the mobilisation of the resources

7.1 Partner 1: INRIA/CAIRN

Equipment No specific equipment acquisition.

Personnel costs The faculty members involved in the project are François Charot (INRIA researcher), Steven Derrien (associate professor), Christophe Wolinski (professor) and Charles

Wagner (research engineer). The non-permanent personal required is a PhD student that will mainly work on ASIP generation. We are looking for a profile with strong informatic skills and good knowledge in computer architecture.

The table below summarizes the manpower in men*months by tasks for both permanent and non-permanent personnels. The detail by deliverables is given in figure 9. The non-permanent personnels costs represent 48% of the personnal costs. The requested funding for non permanent personnels is 100% of the total ANR requested funding.

	title	years			total
		1	2	3	
Task-3	System generation	6.0	6.0	0.0	12.0
Task-4	HAS front-end	19.0	24.0	2.0	45.0
Task-8	Dissemination	0.0	1.0	1.0	2.0
	total	25.0	31.0	3.0	59.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to project meeting as well as participation to conferences. The travel costs are estimated to 7,5% of the total requested ANR funding.

Expenses for inward billing The costs justified by internal invoicing procedures are evaluated to 4% of the total requested ANR funding.

7.2 Partner 2: ENS Lyon/LIP

Equipment No specific equipment acquisition. The costs for depreciation of workstations is evaluated to 4% of the total requested ANR funding.

Personnel costs The faculty members involved in the project are an emeritus professor at ENS Lyon (Paul Feautrier) and a research associate (CR2) at INRIA Rhône-Alpes (Christophe Alias). The non-permanent personel required is a PhD student that will work on network process generation from polyhedral loops, then on extensions to non-polyhedral loops. We are looking for a student with both theoretical and practical skills, that will be able to get a sufficient understanding of the polyhedral techniques and to produce a working implementation.

The table below summarizes the men*months by deliverables and tasks for both permanent and non-permanent personnels. The non-permanent personnels costs represent 26% of the personnal costs. The requested funding for non permanent personnels is 100% of the total ANR requested funding.

number	title	years			total
		1	2	3	
	project management	1.0	1.0	1.0	3.0
	total Task-1	1.0	1.0	1.0	3.0
D230	*xcoach format specification*	7.0	3.0	0.0	10.0
	total Task-2	7.0	3.0	0.0	10.0
D430	*Process generation method*	10.0	0.0	9.0	19.0
D431	*Process/FIFO construction*	10.0	20.0	12.0	42.0
	total Task-4	20.0	20.0	21.0	61.0
D831	*HAS front-end user manual*	0.0	1.0	0.0	1.0
	total Task-8	0.0	1.0	0.0	1.0
	total	28.0	25.0	22.0	75.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to project meeting as well as participation to conferences. The travel costs are estimated to 20% of the total requested ANR funding.

Expenses for inward billing The costs justified by internal invoicing procedures are evaluated to 4% of the total requested ANR funding.

7.3 Partner 3: TIMA

Equipment No specific equipment acquisition.

Personnel costs The permanent personnels involved in the project are professor and assistant professor (Frédéric Pétrot and Olivier Muller). The non permanent personnels are Phd students and post-doc researchers. Related costs are estimated in men*months. One phd student (Adrien Prost-Boucle), funded by the french ministry of research, will be working on the project. One 100% funded phd student will be hired in september 2010. A post-doc researcher will be hired at the end of 2011 for one year and an half. The PhD student will mainly work on the evolution of UGH HLS tool. Thus, we are looking for a profile with strong informatic skills and good knowledge in computer architecture. The post-doc will mainly work on dynamic reconfiguration and HPC. The required profile will be more oriented on computer architecture and advanced digital design.

The table below sumarizes the man power in men*months by tasks for both permanent and non-permanent personnels. The detail by deliverables is given in figure 9. The requested funding for personnels represent 50% of the total personnal costs. The requested funding for non permanent personnels is 85% of the total ANR requested funding.

	title	years			total
		1	2	3	
Task-1	Project management	1.0	1.0	1.0	3.0
Task-2	Backbone infrastructure	1.0	0.0	0.0	1.0
Task-3	System generation	18.0	13.0	2.0	33.0
Task-5	HAS back-end	12.0	12.0	6.0	30.0
Task-6	PC/FPGA communication middleware	3.0	19.0	21.0	43.0
Task-8	Dissemination	0.0	3.0	1.0	4.0
	total	35.0	48.0	31.0	114.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to project meeting as well as participation to conferences. The travel costs are estimated to 11% of the total requested ANR funding.

Expenses for inward billing The costs justified by internal invoicing procedures are evaluated to 4% of the total requested ANR funding.

7.4 Partner 4: LAB-STICC

Equipment In order to validate the design flow project, the Lab-STICC laboratory will buy FPGA developpement boards. The cost for these FPGA boards is estimated to 3% of the total ANR funding.

Personnel costs The faculty members involved in the project are associate professors (Philippe COUSSY, Cyrille CHAVET) or research engineers (Dominique HELLER). All non-permanent personnel costs are estimated in men*months for senior researchers (post-doc or research engineers).

The table below sumarizes the man power in men*months by tasks for both permanent and non-permanent personnels. The detail by deliverables is given in figure 10. The non-permanent personnels costs represent 50% of the personnal costs. The requested funding for non permanent personnels is about 83% of the total ANR requested funding.

	title	years			total
		1	2	3	
Task-1	Project management	1.0	1.0	1.0	3.0
Task-2	Backbone infrastructure	13.0	5.0	0.0	18.0
Task-3	System generation	1.0	6.0	3.0	10.0
Task-5	HAS back-end	6.0	19.0	11.0	36.0
Task-8	Dissemination	0.0	3.0	2.0	5.0
	total	21.0	34.0	17.0	72.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to management and meeting as well as participation to conferences. The travel costs are estimated to 10% of the total requested ANR funding.

Expenses for inward billing The costs justified by internal invoicing procedures are evaluated to 4% of the total requested ANR funding.

7.5 Partner 5: LIP6

Equipment No specific equipment acquisition is required for this project. The costs for depreciation of workstations and pre-existing FPGA boards are evaluated to 7% of the total requested ANR funding.

Personnel costs The permanent personnels involved in the project are professors or assistant processors (Alain Greiner and Ivan Augé). All non permanent personnel costs are estimated in men*months for senior researchers (post-doc or research engineers). The table below sumarizes the man power by tasks in men*months for both permanent and non-permanent personnels. The detail by deliverables is given in figure 10. The non-permanent personnels costs represent 50% of the personnal costs. The requested funding for non permanent personnels is 79% of the total ANR requested funding.

	title	years			total
		1	2	3	
Task-1	Project management	4.0	2.5	2.5	9.0
Task-2	Backbone infrastructure	2.0	2.0	0.0	4.0
Task-3	System generation	9.0	7.5	7.5	24.0
Task-5	HAS back-end	2.0	2.5	7.5	12.0
Task-6	PC/FPGA communication middleware	3.0	8.0	4.0	15.0
Task-8	Dissemination	4.0	2.0	2.0	8.0
	total	24.0	24.5	23.5	72.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to management and coordination meeting as well as participation to conferences. The travel costs are estimated to 10% of the total requested ANR funding.

Expenses for inward billing The costs justified by internal invoicing procedures are evaluated to 4% of the total requested ANR funding.

7.6 Partner 6: XILINX

Equipment No specific equipment acquisition is required for this project.

Personnel costs XILINX employees involved in the project are permanent Software Engineers. The man power detail in men*months by deliverables is given in figure 9 and a sumary by task in the following table.

	title	years			total
		1	2	3	
Task-2	Backbone infrastructure	0.0	3.0	0.0	3.0
Task-3	System generation	0.0	2.0	3.0	5.0
Task-5	HAS back-end	0.0	0.0	1.5	1.5
Task-6	PC/FPGA communication middleware	0.0	0.0	2.0	2.0
Task-8	Dissemination	0.0	0.0	0.5	0.5
	total	0.0	5.0	7.0	12.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to project meeting as well as participation to conferences. The travel costs are estimated to 2% of the total requested ANR funding.

Expenses for inward billing none

Other working costs none

7.7 Partner 7: BULL

Equipment Acquisition of a FPGA development board will represent the main equipment cost for Bull in COACH. It is estimated at about 5% (tbc) of the total funding.

Personnel costs A permanent engineer will be assigned full time to the project for a duration of 20 months as shown in the table below that gives the man power in men*months:

number	title	years			total
		1	2	3	
	User specification	3.0	0.0	0.0	3.0
	total Task-2	3.0	0.0	0.0	3.0
D610	*HPC communication API*	3.0	0.0	0.0	3.0
	total Task-6	3.0	0.0	0.0	3.0
D710	*BULL demonstrator*	2.0	6.0	10.0	18.0
	total Task-7	2.0	6.0	10.0	18.0
	total	8.0	6.0	10.0	24.0

Subcontracting No subcontracting costs.

Travel Application of a standard 10% of the total funding to travel costs.

Expenses for inward billing Costs justified by inward billing are estimated to about 5% of the total funding.

Other working costs none

7.8 Partner 8: THALES

Equipment In order to validate the design flow,TRT will buy FPGA developpement boards. The cost for these FPGA boards is estimated to 10 k€(6% of the total ANR funding).

Personnel costs The effort to adapt SPEAR DE to generate the input files to COACH framework is estimated to 13 men*months. The effort to describe and develop the application is estimated to 14 men*months. Finally we need one men*months for the partiticipation to the global specification in task 2. This is sumarized in the table below:

number	title	years			total
		1	2	3	
	User specification	1.0	0.0	0.0	1.0
	total Task-2	1.0	0.0	0.0	1.0
D720	*THALES demonstrator (step 1)*	4.0	0.0	0.0	4.0
D730	*SPEAR-DE adaptation*	6.0	7.0	0.0	13.0
D740	*THALES demonstrator (step 2)*	0.0	5.0	5.0	10.0
	total Task-7	10.0	12.0	5.0	27.0
	total	11.0	12.0	5.0	28.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to meeting, plenaries as well as participation to conferences. The travel costs are estimated to 10 k€. The travel costs are estimated to 5% of the total requested ANR funding.

Expenses for inward billing none

Other working costs none

7.9 Partner 9: FLEXRAS

Equipment No equipement costs.

Personnel costs The effort to define SoC architecture and adapt eFPGA interface to generate is estimated to 9.6 men*months. The effort to develop demonstrator and to extract eFPGA timing characteristics is estimated to 4.8 men*months. Finally we need one 3.6 man.month for the evaluation of the FLEXRAS solution. The table below summarizes the these manpower costs in men*months for the deliverables and by tasks.

number	title	years			total
		1	2	3	
D750	*FLEXRAS architecture*	2.4	0.0	0.0	2.4
D751	*eFPGA/VCI component*	3.6	3.6	0.0	7.2
D752	*FLEXRAS demonstrators*	0.0	2.4	0.0	2.4
D753	*eFPGA characterisation*	0.0	0.0	2.4	2.4
D754	*FLEXRAS evaluation*	0.0	0.0	3.6	3.6
	total Task-7	6.0	6.0	6.0	18.0
	total	6.0	6.0	6.0	18.0

Subcontracting No subcontracting costs.

Travel No travel costs.

Expenses for inward billing none

Other working costs none

7.10 Partner 10: NAVTEL-SYSTEM

Equipment Navtel will use FPGA board with ARM processors for the validation. The costs for depreciation of the board and the instrument of test are evaluated to 7% of the total requested ANR funding.

Personnel costs A permanent engineer will be assigned on average 1/3 time for all the duration of the project. The table below shows the estimated manpower cost in men*months for the deliverables and by tasks.

number	title	years			total
		1	2	3	
D760	*NAVTEL-SYSTEM demonstrator specification*	4.0	0.0	0.0	4.0
D761	*NAVTEL-SYSTEM wrapper adaptation*	1.0	1.0	0.0	2.0
D762	*NAVTEL-SYSTEM evaluation*	0.0	2.0	4.0	6.0
	total Task-7	5.0	3.0	4.0	12.0
	total	5.0	3.0	4.0	12.0

Subcontracting No subcontracting costs.

Travel The travel costs are associated to meeting, plenaries as well as participation to conferences. The travel costs are estimated to 3 k€.

Expenses for inward billing none

Other working costs none

number	title	years			total
		1	2	3	
D324	*ALTERA architecture*	6.0	6.0	0.0	12.0
	total Task-3	6.0	6.0	0.0	12.0
D410	*ASIP compilation flow*	6.0	9.0	0.0	15.0
D420	*SystemC for extensible MIPS*	2.0	3.0	0.0	5.0
D421	*SystemC for NIOS processor*	2.0	0.0	0.0	2.0
D422	*VHDL for extensible MIPS*	9.0	12.0	0.0	21.0
D423	*Evaluation report*	0.0	0.0	2.0	2.0
	total Task-4	19.0	24.0	2.0	45.0
D832	*ASIP user manual*	0.0	1.0	1.0	2.0
	total Task-8	0.0	1.0	1.0	2.0
	total	25.0	31.0	3.0	59.0

INRIA/CAIRN

number	title	years			total
		1	2	3	
D235	*XILINX RTL optimisation (1)*	0.0	3.0	0.0	3.0
	total Task-2	0.0	3.0	0.0	3.0
D321	*XILINX RTL optimisation (2)*	0.0	2.0	0.0	2.0
D323	*XILINX RTL optimisation (3)*	0.0	0.0	1.5	1.5
D327	*XILINX RTL optimisation (4)*	0.0	0.0	1.5	1.5
	total Task-3	0.0	2.0	3.0	5.0
D541	*XILINX RTL optimisation (5)*	0.0	0.0	1.5	1.5
	total Task-5	0.0	0.0	1.5	1.5
D635	*Optimisation for XILINX dynamic reconfiguration*	0.0	0.0	2.0	2.0
	total Task-6	0.0	0.0	2.0	2.0
D821	*XILINX feedback*	0.0	0.0	0.5	0.5
	total Task-8	0.0	0.0	0.5	0.5
	total	0.0	5.0	7.0	12.0

XILINX

number	title	years			total
		1	2	3	
	project management	1.0	1.0	1.0	3.0
	total Task-1	1.0	1.0	1.0	3.0
D211	*CSG specification*	1.0	0.0	0.0	1.0
	total Task-2	1.0	0.0	0.0	1.0
D322	*XILINX architecture*	9.0	9.0	0.0	18.0
D332	*DNA OS drivers*	6.0	3.0	2.0	11.0
D333	*Porting of DNA OS*	3.0	1.0	0.0	4.0
	total Task-3	18.0	13.0	2.0	33.0
D510	*UGH integration*	12.0	0.0	0.0	12.0
D530	*UGH enhancement 1*	0.0	9.0	0.0	9.0
D531	*UGH enhancement 2*	0.0	3.0	6.0	9.0
	total Task-5	12.0	12.0	6.0	30.0
D614	*HPC API for DNA OS*	0.0	3.0	0.0	3.0
D620	*HPC hardware XILINX*	3.0	9.0	0.0	12.0
D631	*CSG module for dynamic reconfiguration*	0.0	4.0	12.0	16.0
D632	*Dynamic reconfiguration for DNA drivers*	0.0	3.0	3.0	6.0
D634	*Profiler for dynamic reconfiguration*	0.0	0.0	6.0	6.0
	total Task-6	3.0	19.0	21.0	43.0
	dissemination	0.0	2.0	1.0	3.0
D830	*CSG User manual*	0.0	1.0	0.0	1.0
	total Task-8	0.0	3.0	1.0	4.0
	total	35.0	48.0	31.0	114.0

TIMA

Figure 9: Man power in men*months for the deliverables (1)

number	title	years			total
		1	2	3	
	project management	1.0	1.0	1.0	3.0
	total Task-1	1.0	1.0	1.0	3.0
D212	*HAS specification*	2.0	0.0	0.0	2.0
D231	*C2X tool*	2.0	1.0	0.0	3.0
D232	*X2C tool*	2.0	1.0	0.0	3.0
D234	*X2VHDL tool*	0.0	3.0	0.0	3.0
D240	*GCC driver specification*	1.0	0.0	0.0	1.0
D241	*GCC driver*	3.0	0.0	0.0	3.0
D250	*Macro-cell definition*	1.0	0.0	0.0	1.0
D251	*Macro-cell library generator*	2.0	0.0	0.0	2.0
	total Task-2	13.0	5.0	0.0	18.0
D325	*Communication adapter spec.*	1.0	0.0	0.0	1.0
D326	*Comm. adapter generator*	0.0	6.0	3.0	9.0
	total Task-3	1.0	6.0	3.0	10.0
D520	*GAUT release reading* `xcoach`	6.0	0.0	0.0	6.0
D521	*GAUT release writing* `xcoach+`	0.0	6.0	0.0	6.0
D532	*Release of GAUT with enhanced synthesis steps*	0.0	9.0	0.0	9.0
D533	*Release of GAUT supporting new const./obj.*	0.0	0.0	7.0	7.0
D534	*Micro-architecture Exploration*	0.0	4.0	4.0	8.0
	total Task-5	6.0	19.0	11.0	36.0
	dissemination	0.0	2.0	2.0	4.0
D833	*HLS user manual*	0.0	1.0	0.0	1.0
	total Task-8	0.0	3.0	2.0	5.0
	total	21.0	34.0	17.0	72.0

LAB-STICC

number	title	years			total
		1	2	3	
D110	*Consortium agreement*	1.0	0.0	0.0	1.0
D120	*Global management*	1.0	1.0	1.0	3.0
D130	*LIP6 management*	1.0	1.0	1.0	3.0
D140	*Infrastructure development*	1.0	0.5	0.5	2.0
	total Task-1	4.0	2.5	2.5	9.0
D210	*COACH specification*	1.0	0.0	0.0	1.0
D220	*COACH internal software architecture*	1.0	0.0	0.0	1.0
D233	*X2SC tool*	0.0	2.0	0.0	2.0
	total Task-2	2.0	2.0	0.0	4.0
D310	*CSG tool*	6.0	5.5	5.5	17.0
D320	*Neutral architecture*	1.0	0.0	0.0	1.0
D330	*MUTEKH OS drivers*	1.0	1.0	2.0	4.0
D331	*Porting of MUTEKH OS*	1.0	1.0	0.0	2.0
	total Task-3	9.0	7.5	7.5	24.0
D511	*UGH integration*	0.0	2.0	4.0	6.0
D540	*Frequency calibration*	2.0	0.5	3.5	6.0
	total Task-5	2.0	2.5	7.5	12.0
D611	*HPC partionning helper*	1.0	0.0	0.0	1.0
D612	*HPC API for Linux PC*	0.0	2.5	0.0	2.5
D613	*HPC API for MUTEKH OS*	0.0	2.5	0.0	2.5
D615	*HPC API*	0.0	0.0	1.0	1.0
D621	*HPC hardware ALTERA*	1.0	2.0	0.0	3.0
D622	*PCI/X traffic generator*	1.0	1.0	0.0	2.0
D630	*CSG support for reconfiguration*	0.0	0.0	2.0	2.0
D633	*Dynamic reconfiguration for MUTEKH drivers*	0.0	0.0	1.0	1.0
	total Task-6	3.0	8.0	4.0	15.0
D810	*Dissemination WEB site*	1.0	0.5	0.5	2.0
D811	*Release handling*	1.0	0.5	0.5	2.0
D820	*Tutorial*	2.0	1.0	1.0	4.0
	total Task-8	4.0	2.0	2.0	8.0
	total	24.0	24.5	23.5	72.0

LIP6

Figure 10: Man power in men*months for the deliverables (2)

A Bibliography

References

[1] CATAPULT-C Mentor HLS tool. http://www.mentor.com/products/esl/high_level_synthesis/, 2009.

[2] Convey computer. http://www.conveycomputers.com/, 2009.

[3] Forte's CYNTHESIZER. http://www.forteds.com/, 2009.

[4] Gidel. http://www.gidel.com/, 2009.

[5] Mitrionics. http://www.mitrionics.com/, 2009.

[6] *Nios II Processor Reference Handbook*. Altera, 2009.

[7] PICO. http://www.synfora.com/, 2009.

[8] Soclib. http://www.soclib.fr/, 2009.

[9] sopc builder support. http://www.altera.com/support/software/system/sopc/sof-sopc_builder.html, 2009.

[10] System Generator for DSP. http://www.xilinx.com/tools/sysgen.htm, 2009.

[11] Christophe Alias, Fabrice Baray, and Alain Darte. Bee+cl@k: An implementation of lattice-based array contraction in the source-to-source translator rose. In *LCTES*. ACM, 2007.

[12] Ivan Augé and Frédéric Pétrot. *User Guided High Level Synthesis*, chapter 10. Springer, 2008.

[13] Cluster for Application CATRENE and Technology Research in Europe on NanotElectronics. CATRENE, Cluster for Application and Technology Research in Europe on NanotElectronics, 2009.

[14] François Charot and Vincent Messé. A flexible code generation framework for the design of application specific programmable processors. In *CODES '99: Proceedings of the seventh international workshop on Hardware/software codesign*, pages 27–31, New York, NY, USA, 1999. ACM.

[15] Jason Cong, Guoling Han, and Zhiru Zhang. Architecture and compiler optimizations for data bandwidth improvement in configurable processors. *IEEE Trans. Very Large Scale Integr. Syst.*, 14(9):986–997, 2006.

[16] P. Coussy and al. *GAUT: A High-Level Synthesis Tool for DSP applications*. Springer, 2008.

[17] P. Coussy and A. Morawiec. Springer, 2008.

[18] Philippe Coussy and Andres Takach. *Special Issue on High-Level Synthesis*, volume 25, page 393. IEEE Computer Society, Los Alamitos, CA, USA, 2008.

[19] D. Buell. Programming Reconfigurable Computers. http://gladiator.ncsa.uiuc.edu/PDFs/rssi06/presentations 2006.

[20] Alain Darte, Yves Robert, and Frédéric Vivien. *Scheduling and automatic Parallelization*. Birkhäuser, 2000.

[21] I. Gonzalez E. El-Araby and T. El-Ghazawi. Virtual architecture and design automation for partial reconfiguration. In *HPRCTA*, 2008.

[22] Paul Feautrier. Some efficient solutions to the affine scheduling problem, I, one dimensional time. *Int. J. of Parallel Programming*, 21(5):313–348, October 1992.

[23] Paul Feautrier. Some efficient solutions to the affine scheduling problem, II, multidimensional time. *Int. J. of Parallel Programming*, 21(6):389–420, December 1992.

[24] Paul Feautrier. Automatic parallelization in the polytope model. In Guy-René Perrin and Alain Darte, editors, *The Data-Parallel Programming Model*, volume LNCS 1132, pages 79–103. Springer, 1996.

[25] Paul Feautrier. Distribution automatique des donnés et des calculs. *T.S.I.*, 15(5):529–557, 1996.

[26] Paul Feautrier. Scalable and structured scheduling. *Int. J. of Parallel Programming*, 34(5):459–487, May 2006.

[27] Carlo Galuzzi and Koen Bertels. The instruction-set extension problem: A survey. In *ARC '08: Proceedings of the 4th international workshop on Reconfigurable Computing*, pages 209–220, Berlin, Heidelberg, 2008. Springer-Verlag.

[28] S. Gupta and al. *SPARK: A Parallelizing Approach to the High-Level Synthesis of Digital Circuits*. Springer, 2004.

[29] Ivan Augé, Frédéric Pétrot, Franï£¡ois Donnet and Pascal Gomez. Platform-based design from parallel C specifications. In *IEEE Transaction on CAD of Integrated Circuits and Systems*, pages 1811–1826, December 2005.

[30] al J.Y Brunel. Cosy: a methodology for system design based on reusable hardware & software ip's. In *Technologies for the Information Society*, pages 709–716. IOS Press, 1998.

[31] Theo Kluter, Philip Brisk, Paolo Ienne, and Edoardo Charbon. Speculative dma for architecturally visible storage in instruction set extensions. In *CODES/ISSS '08: Proceedings of the 6th IEEE/ACM/IFIP international conference on Hardware/Software codesign and system synthesis*, pages 243–248, New York, NY, USA, 2008. ACM.

[32] Theo Kluter, Philip Brisk, Paolo Ienne, and Edoardo Charbon. Way stealing: cache-assisted automatic instruction set extensions. In *DAC '09: Proceedings of the 46th Annual Design Automation Conference*, pages 31–36, New York, NY, USA, 2009. ACM.

[33] Ludovic L'Hours. Generating Efficient Custom FPGA Soft-Cores for Control-Dominated Applications. In *ASAP '05: Proceedings of the 2005 IEEE International Conference on Application-Specific Systems, Architecture Processors*, pages 127–133, Washington, DC, USA, 2005. IEEE Computer Society.

[34] M.B. Gokhale and al. Promises and Pitfalls of Reconfigurable Supercomputing. In *Systems and Algorithms, CSREA Press*, pages 11–20, 2006.

[35] Daniel Menard, Emmanuel Casseau, Shafqat Khan, Olivier Sentieys, Stéphane Chevobbe, Stéphane Guyetant, and Raphael David. Reconfigurable operator based multimedia embedded processor. In *ARC '09: Proceedings of the 5th International Workshop on Reconfigurable Computing: Architectures, Tools and Applications*, pages 39–49, Berlin, Heidelberg, 2009. Springer-Verlag.

[36] Ivan Miro-Panades, Fabien Clermidy, Pascal Vivet, and Alain Greiner. Physical implementation of the dspin network-on-chip in the faust architecture. In *NOCS '08: Proceedings of the Second ACM/IEEE International Symposium on Networks-on-Chip*, pages 139–148, Washington, DC, USA, 2008. IEEE Computer Society.

[37] P. Lysaght and J. Dunlop. Dynamic reconfiguration of field programmable gate arrays. In *Field Programmable Logic and Applications, Oxford, England*, Sept 1993.

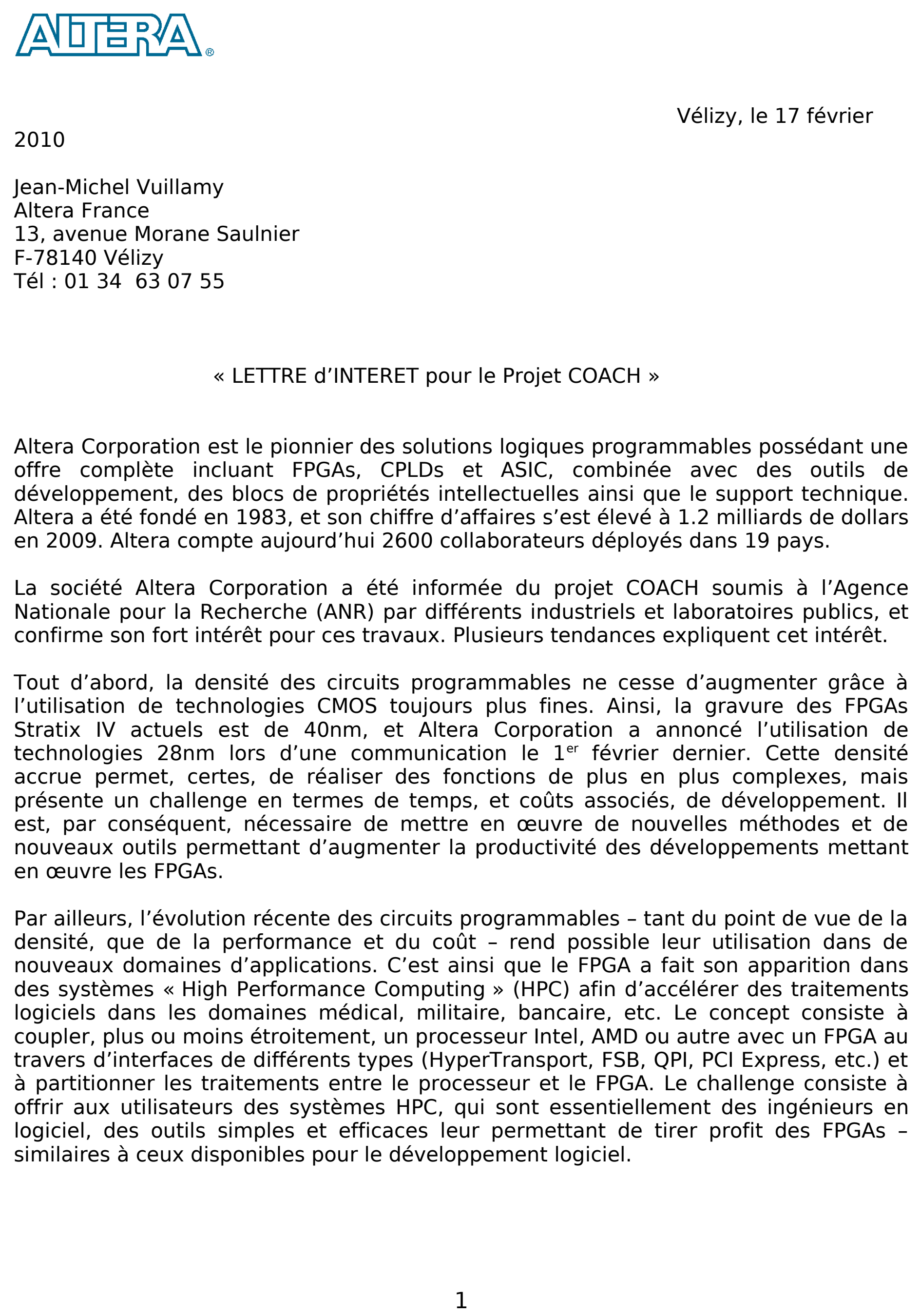
[38] T. Van Court and al. Achieving High Performance with FPGA-Based Computing. In *Computer, vol. 40, no. 3*, pages 50–57, mars 2007.

[39] Jones Viola. Rapid Object Detection using a Boosted Cascade of Simple Feature. In *Proceedings of Conference on Computer Vision and Pattern recognition*, 2001.

B Letters of interest

.

B.1 ALTERA Corporation



ALTERA®

Vélizy, le 17 février
2010

Jean-Michel Vuillamy
Altera France
13, avenue Morane Saulnier
F-78140 Vélizy
Tél : 01 34 63 07 55

« LETTRE d'INTERET pour le Projet COACH »

Altera Corporation est le pionnier des solutions logiques programmables possédant une offre complète incluant FPGAs, CPLDs et ASIC, combinée avec des outils de développement, des blocs de propriétés intellectuelles ainsi que le support technique. Altera a été fondé en 1983, et son chiffre d'affaires s'est élevé à 1.2 milliards de dollars en 2009. Altera compte aujourd'hui 2600 collaborateurs déployés dans 19 pays.

La société Altera Corporation a été informée du projet COACH soumis à l'Agence Nationale pour la Recherche (ANR) par différents industriels et laboratoires publics, et confirme son fort intérêt pour ces travaux. Plusieurs tendances expliquent cet intérêt.

Tout d'abord, la densité des circuits programmables ne cesse d'augmenter grâce à l'utilisation de technologies CMOS toujours plus fines. Ainsi, la gravure des FPGAs Stratix IV actuels est de 40nm, et Altera Corporation a annoncé l'utilisation de technologies 28nm lors d'une communication le 1er février dernier. Cette densité accrue permet, certes, de réaliser des fonctions de plus en plus complexes, mais présente un challenge en termes de temps, et coûts associés, de développement. Il est, par conséquent, nécessaire de mettre en œuvre de nouvelles méthodes et de nouveaux outils permettant d'augmenter la productivité des développements mettant en œuvre les FPGAs.

Par ailleurs, l'évolution récente des circuits programmables – tant du point de vue de la densité, que de la performance et du coût – rend possible leur utilisation dans de nouveaux domaines d'applications. C'est ainsi que le FPGA a fait son apparition dans des systèmes « High Performance Computing » (HPC) afin d'accélérer des traitements logiciels dans les domaines médical, militaire, bancaire, etc. Le concept consiste à coupler, plus ou moins étroitement, un processeur Intel, AMD ou autre avec un FPGA au travers d'interfaces de différents types (HyperTransport, FSB, QPI, PCI Express, etc.) et à partitionner les traitements entre le processeur et le FPGA. Le challenge consiste à offrir aux utilisateurs des systèmes HPC, qui sont essentiellement des ingénieurs en logiciel, des outils simples et efficaces leur permettant de tirer profit des FPGAs – similaires à ceux disponibles pour le développement logiciel.

1

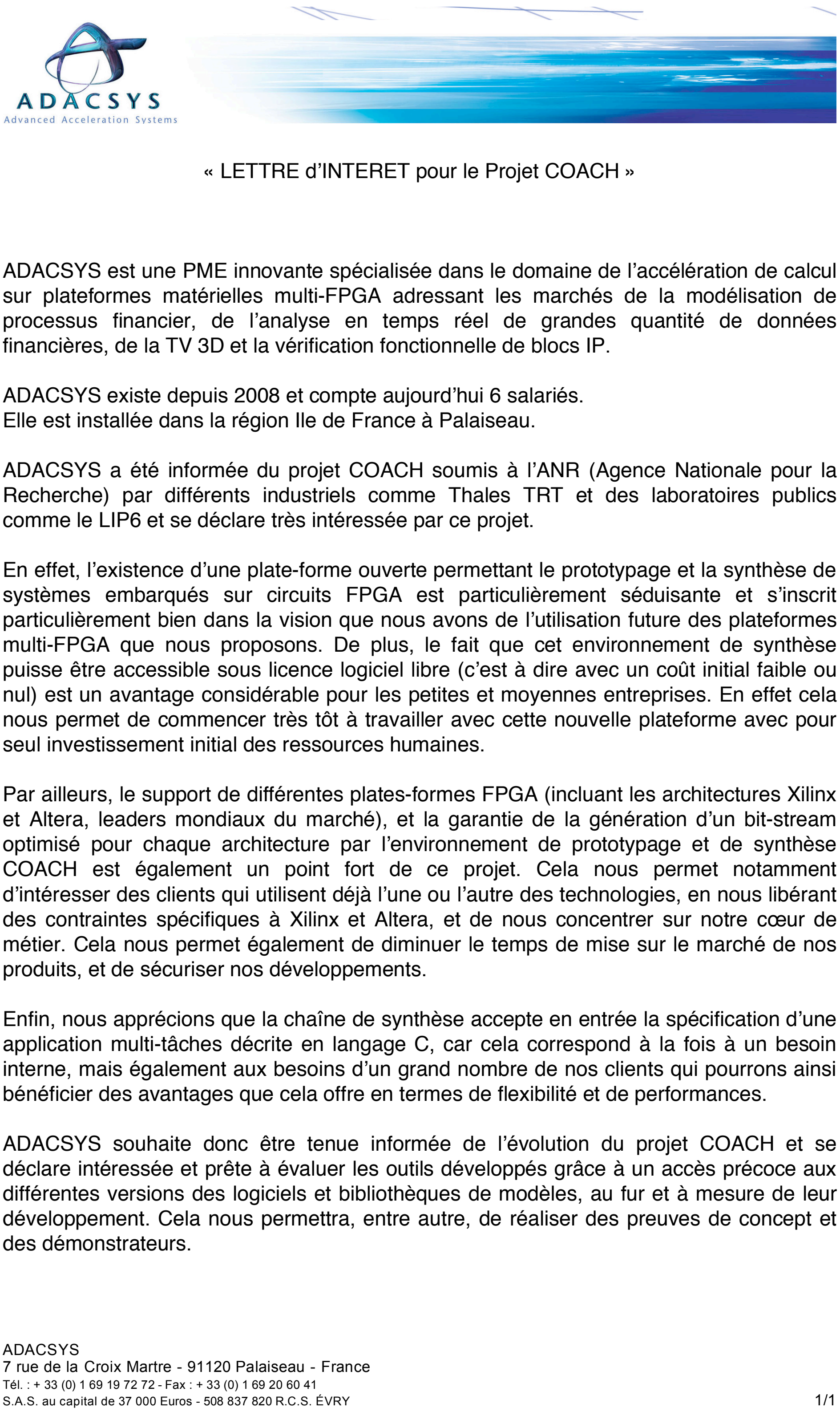
Le projet COACH, dont l'objectif principal est de simplifier la conception d'applications HPC et MPSoC (Multi-Processor System on Chip), répond, donc, tout à fait aux tendances et aux préoccupations actuelles d'Altera.

Par ailleurs, le projet COACH permettra aux concepteurs de FPGAs de disposer d'un outil de conception sous la forme d'un logiciel libre, ce qui ouvrira de nouvelles possibilités pour les petites structures (PME voire TPE). Ces dernières pourront, en effet, innover avec des produits à base de composants logiques programmables sans à avoir à investir des dizaines voire centaines de milliers d'euro dans des outils de conception de haut niveau.

Altera ne disposant pas de centre de recherche & développement en France, il n'a pas été possible de participer au projet COACH en tant que partenaire. Néanmoins, Altera souhaite soutenir ce projet sous la forme d'un engagement consistant à fournir deux cartes de développement à base de Stratix IV GX « Stratix IV GX FPGA development kit » pour un montant total de 8995 dollars. Ces cartes électroniques permettront aux équipes de développement, au sein de COACH, de valider les concepts sur une cible FPGA supportant des applications HPC conséquentes.

La société Altera Corporation souhaite donc être tenue informée des progrès du projet COACH, afin de communiquer à ce sujet auprès de ses clients.

B.2 ADACSYS



ADACSYS
Advanced Acceleration Systems

« LETTRE d'INTERET pour le Projet COACH »

ADACSYS est une PME innovante spécialisée dans le domaine de l'accélération de calcul sur plateformes matérielles multi-FPGA adressant les marchés de la modélisation de processus financier, de l'analyse en temps réel de grandes quantité de données financières, de la TV 3D et la vérification fonctionnelle de blocs IP.

ADACSYS existe depuis 2008 et compte aujourd'hui 6 salariés.
Elle est installée dans la région Ile de France à Palaiseau.

ADACSYS a été informée du projet COACH soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels comme Thales TRT et des laboratoires publics comme le LIP6 et se déclare très intéressée par ce projet.

En effet, l'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement séduisante et s'inscrit particulièrement bien dans la vision que nous avons de l'utilisation future des plateformes multi-FPGA que nous proposons. De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est à dire avec un coût initial faible ou nul) est un avantage considérable pour les petites et moyennes entreprises. En effet cela nous permet de commencer très tôt à travailler avec cette nouvelle plateforme avec pour seul investissement initial des ressources humaines.

Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures Xilinx et Altera, leaders mondiaux du marché), et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet. Cela nous permet notamment d'intéresser des clients qui utilisent déjà l'une ou l'autre des technologies, en nous libérant des contraintes spécifiques à Xilinx et Altera, et de nous concentrer sur notre cœur de métier. Cela nous permet également de diminuer le temps de mise sur le marché de nos produits, et de sécuriser nos développements.

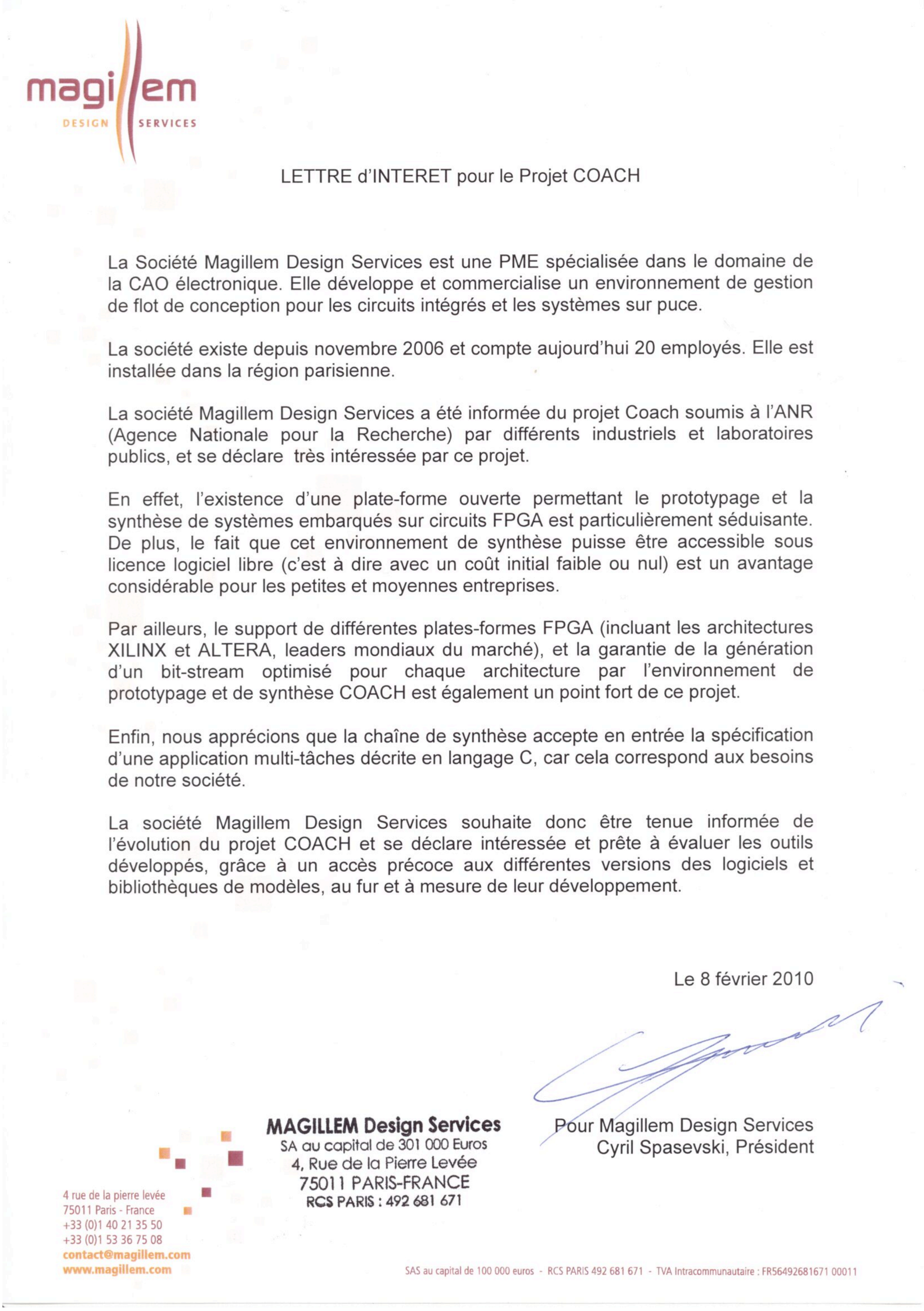
Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond à la fois à un besoin interne, mais également aux besoins d'un grand nombre de nos clients qui pourrons ainsi bénéficier des avantages que cela offre en termes de flexibilité et de performances.

ADACSYS souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement. Cela nous permettra, entre autre, de réaliser des preuves de concept et des démonstrateurs.

ADACSYS
7 rue de la Croix Martre - 91120 Palaiseau - France
Tél. : + 33 (0) 1 69 19 72 72 - Fax : + 33 (0) 1 69 20 60 41
S.A.S. au capital de 37 000 Euros - 508 837 820 R.C.S. ÉVRY

1/1

B.3 MAGILLEM Design Services



magillem
DESIGN SERVICES

LETTRE d'INTERET pour le Projet COACH

La Société Magillem Design Services est une PME spécialisée dans le domaine de la CAO électronique. Elle développe et commercialise un environnement de gestion de flot de conception pour les circuits intégrés et les systèmes sur puce.

La société existe depuis novembre 2006 et compte aujourd'hui 20 employés. Elle est installée dans la région parisienne.

La société Magillem Design Services a été informée du projet Coach soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels et laboratoires publics, et se déclare très intéressée par ce projet.

En effet, l'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement séduisante. De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est à dire avec un coût initial faible ou nul) est un avantage considérable pour les petites et moyennes entreprises.

Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures XILINX et ALTERA, leaders mondiaux du marché), et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond aux besoins de notre société.

La société Magillem Design Services souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés, grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

Le 8 février 2010

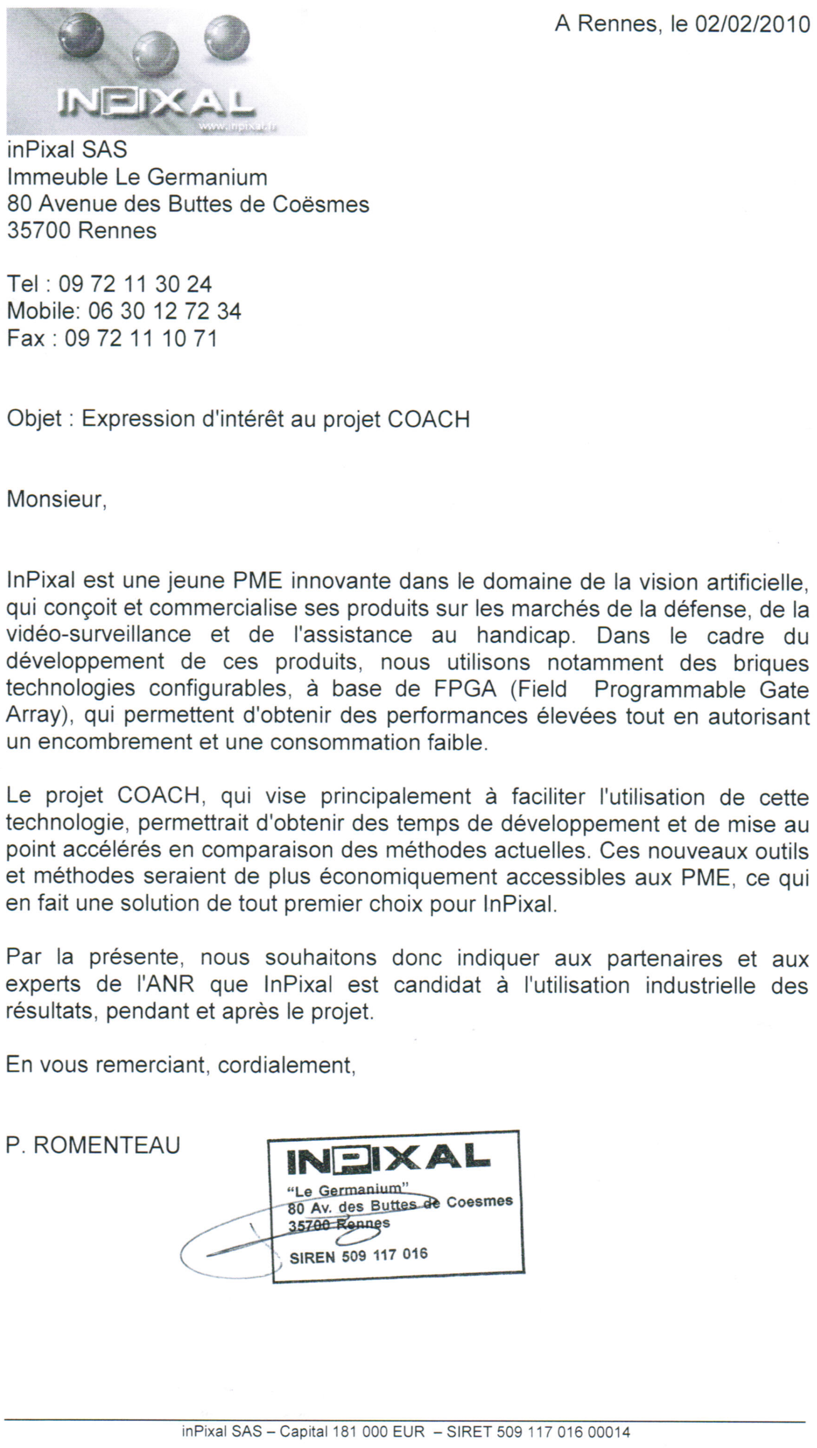
MAGILLEM Design Services
SA au capital de 301 000 Euros
4, Rue de la Pierre Levée
75011 PARIS-FRANCE
RCS PARIS : 492 681 671

Pour Magillem Design Services
Cyril Spasevski, Président

4 rue de la pierre levée
75011 Paris - France
+33 (0)1 40 21 35 50
+33 (0)1 53 36 75 08
contact@magillem.com
www.magillem.com

SAS au capital de 100 000 euros - RCS PARIS 492 681 671 - TVA Intracommunautaire : FR56492681671 00011

B.4 INPIXAL



A Rennes, le 02/02/2010

INPIXAL www.inpixal.fr

inPixal SAS
Immeuble Le Germanium
80 Avenue des Buttes de Coësmes
35700 Rennes

Tel : 09 72 11 30 24
Mobile: 06 30 12 72 34
Fax : 09 72 11 10 71

Objet : Expression d'intérêt au projet COACH

Monsieur,

InPixal est une jeune PME innovante dans le domaine de la vision artificielle, qui conçoit et commercialise ses produits sur les marchés de la défense, de la vidéo-surveillance et de l'assistance au handicap. Dans le cadre du développement de ces produits, nous utilisons notamment des briques technologies configurables, à base de FPGA (Field Programmable Gate Array), qui permettent d'obtenir des performances élevées tout en autorisant un encombrement et une consommation faible.

Le projet COACH, qui vise principalement à faciliter l'utilisation de cette technologie, permettrait d'obtenir des temps de développement et de mise au point accélérés en comparaison des méthodes actuelles. Ces nouveaux outils et méthodes seraient de plus économiquement accessibles aux PME, ce qui en fait une solution de tout premier choix pour InPixal.

Par la présente, nous souhaitons donc indiquer aux partenaires et aux experts de l'ANR que InPixal est candidat à l'utilisation industrielle des résultats, pendant et après le projet.

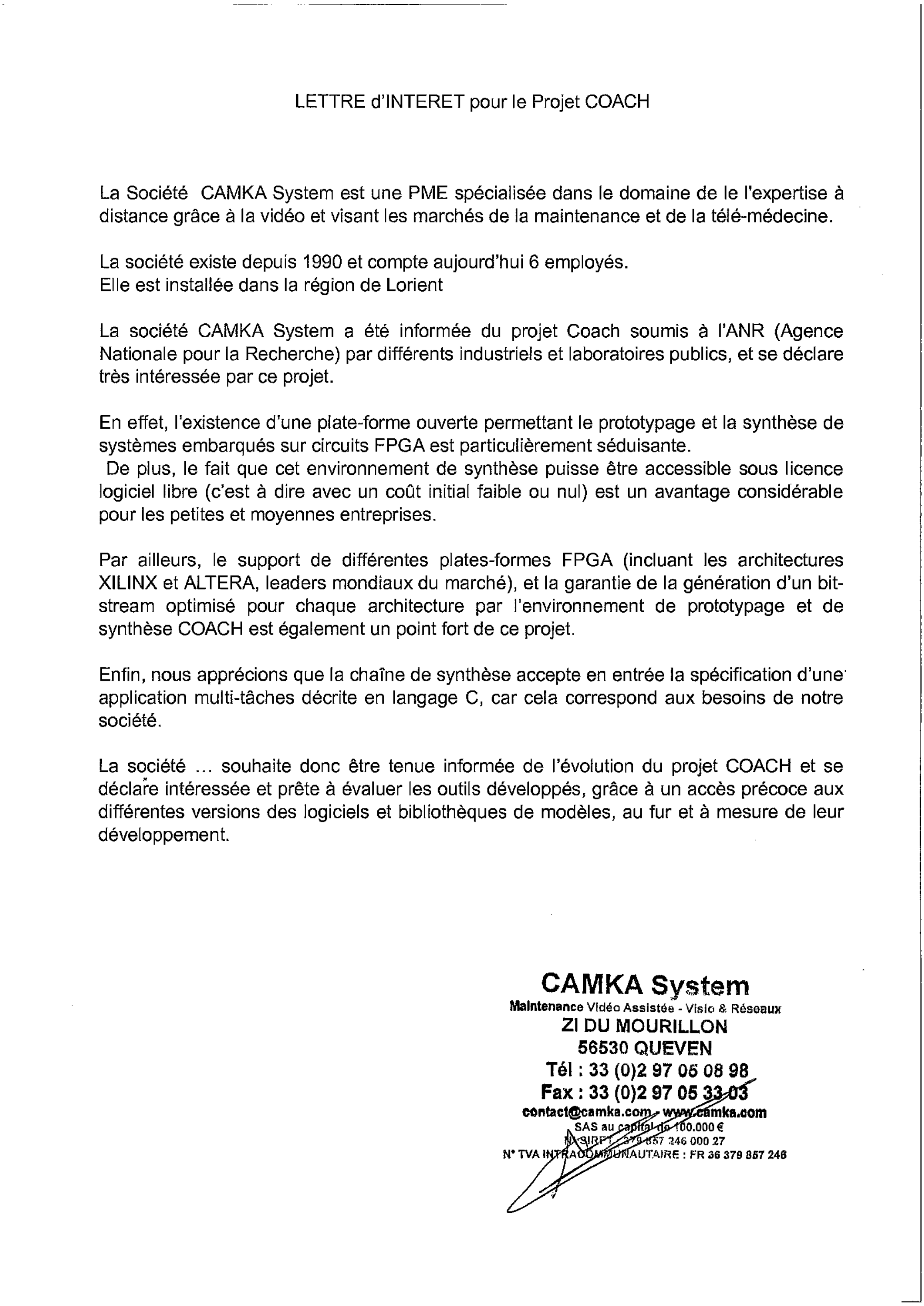
En vous remerciant, cordialement,

P. ROMENTEAU

INPIXAL
"Le Germanium"
80 Av. des Buttes de Coesmes
35700 Rennes
SIREN 509 117 016

inPixal SAS – Capital 181 000 EUR – SIRET 509 117 016 00014

B.5 CAMKA System



LETTRE d'INTERET pour le Projet COACH

La Société CAMKA System est une PME spécialisée dans le domaine de le l'expertise à distance grâce à la vidéo et visant les marchés de la maintenance et de la télé-médecine.

La société existe depuis 1990 et compte aujourd'hui 6 employés.
Elle est installée dans la région de Lorient

La société CAMKA System a été informée du projet Coach soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels et laboratoires publics, et se déclare très intéressée par ce projet.

En effet, l'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement séduisante.
De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est à dire avec un coût initial faible ou nul) est un avantage considérable pour les petites et moyennes entreprises.

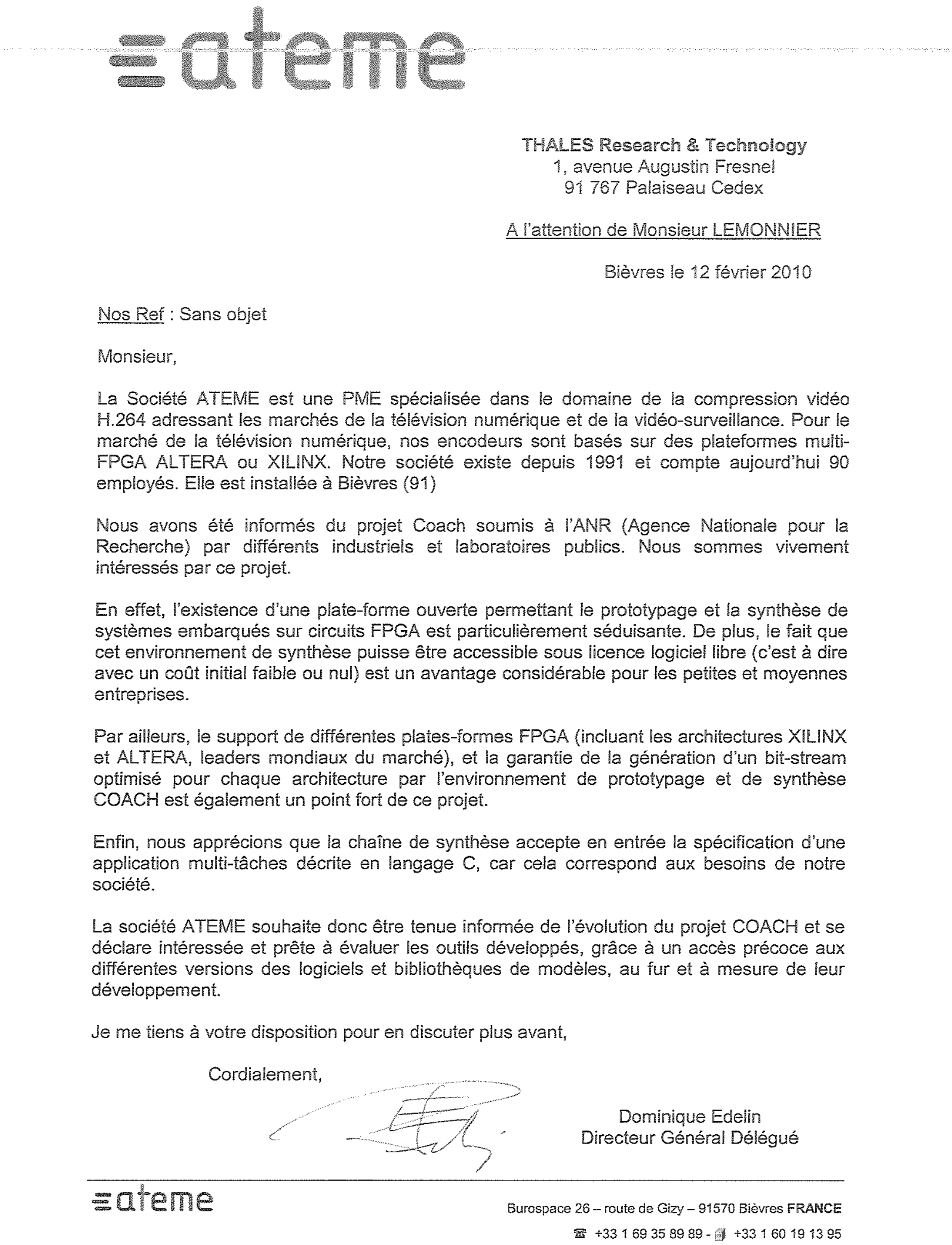
Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures XILINX et ALTERA, leaders mondiaux du marché), et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond aux besoins de notre société.

La société ... souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés, grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

CAMKA System
Maintenance Vidéo Assistée - Visio & Réseaux
ZI DU MOURILLON
56530 QUEVEN
Tél : 33 (0)2 97 05 08 98
Fax : 33 (0)2 97 05 33 03
contact@camka.com - www.camka.com
SAS au capital de 100.000 €
N° SIRET : 379 857 246 000 27
N° TVA INTRACOMMUNAUTAIRE : FR 36 379 857 246

B.6 ATEME



THALES Research & Technology
1, avenue Augustin Fresnel
91 767 Palaiseau Cedex

A l'attention de Monsieur LEMONNIER

Bièvres le 12 février 2010

Nos Ref : Sans objet

Monsieur,

La Société ATEME est une PME spécialisée dans le domaine de la compression vidéo H.264 adressant les marchés de la télévision numérique et de la vidéo-surveillance. Pour le marché de la télévision numérique, nos encodeurs sont basés sur des plateformes multi-FPGA ALTERA ou XILINX. Notre société existe depuis 1991 et compte aujourd'hui 90 employés. Elle est installée à Bièvres (91)

Nous avons été informés du projet Coach soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels et laboratoires publics. Nous sommes vivement intéressés par ce projet.

En effet, l'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement séduisante. De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est à dire avec un coût initial faible ou nul) est un avantage considérable pour les petites et moyennes entreprises.

Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures XILINX et ALTERA, leaders mondiaux du marché), et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond aux besoins de notre société.

La société ATEME souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés, grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

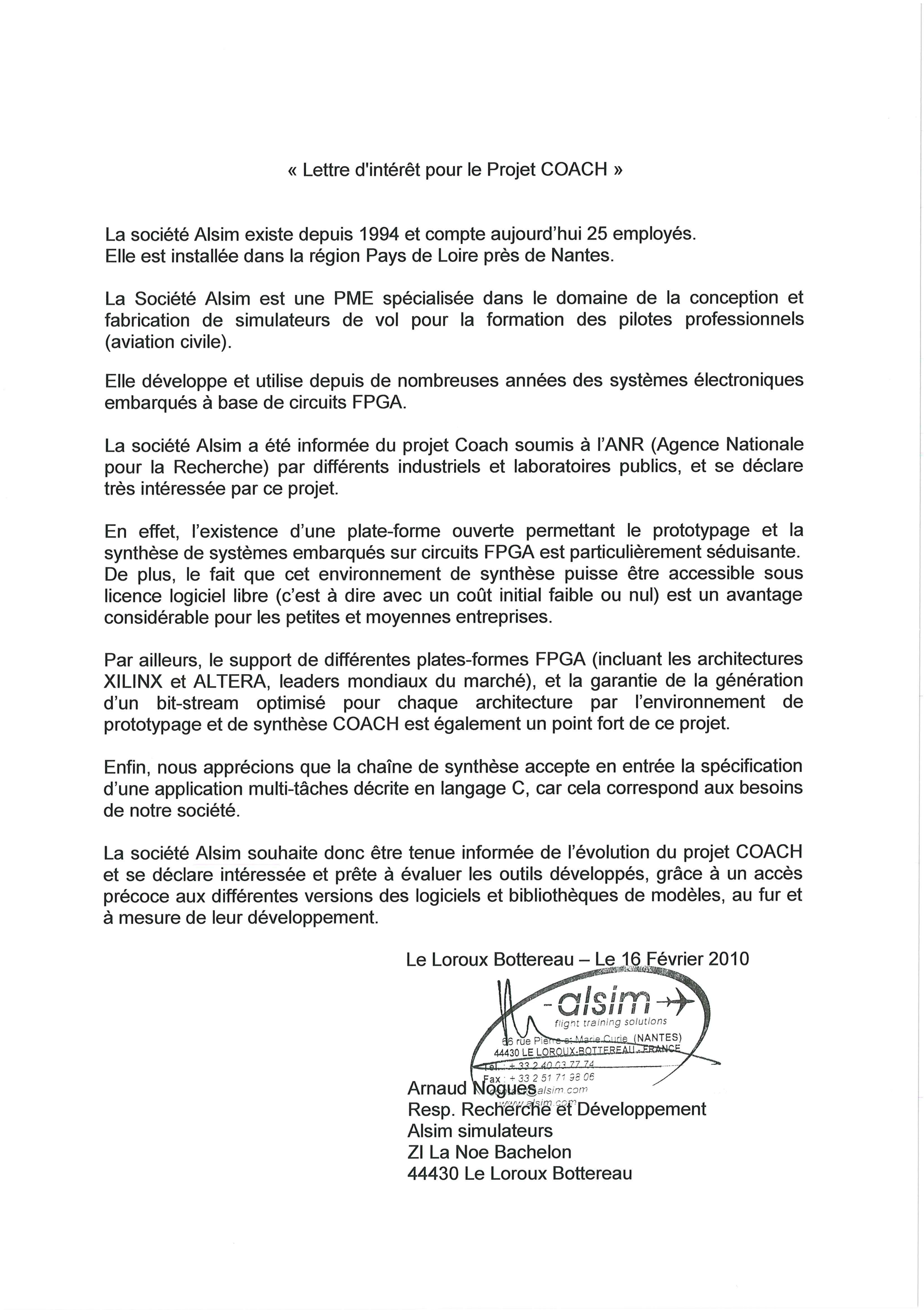
Je me tiens à votre disposition pour en discuter plus avant,

Cordialement,

Dominique Edelin
Directeur Général Délégué

ateme
Burospace 26 – route de Gizy – 91570 Bièvres FRANCE
☎ +33 1 69 35 89 89 - +33 1 60 19 13 95

B.7 ALSIM Simulateur



« Lettre d'intérêt pour le Projet COACH »

La société Alsim existe depuis 1994 et compte aujourd'hui 25 employés.
Elle est installée dans la région Pays de Loire près de Nantes.

La Société Alsim est une PME spécialisée dans le domaine de la conception et fabrication de simulateurs de vol pour la formation des pilotes professionnels (aviation civile).

Elle développe et utilise depuis de nombreuses années des systèmes électroniques embarqués à base de circuits FPGA.

La société Alsim a été informée du projet Coach soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels et laboratoires publics, et se déclare très intéressée par ce projet.

En effet, l'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement séduisante. De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est à dire avec un coût initial faible ou nul) est un avantage considérable pour les petites et moyennes entreprises.

Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures XILINX et ALTERA, leaders mondiaux du marché), et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond aux besoins de notre société.

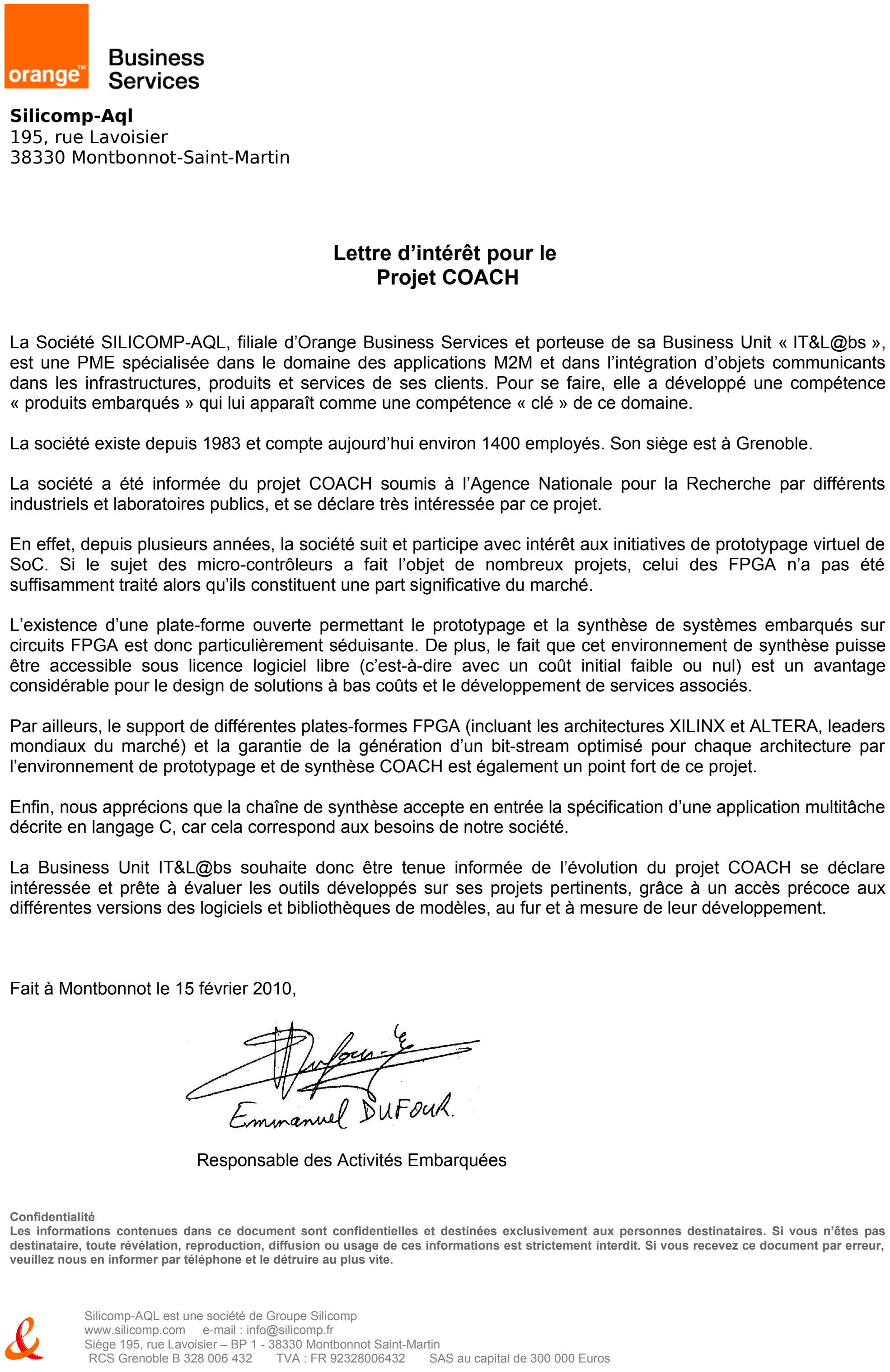
La société Alsim souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés, grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

Le Loroux Bottereau – Le 16 Février 2010

alsim
flight training solutions
66 rue Pierre et Marie Curie (NANTES)
44430 LE LOROUX-BOTTEREAU - FRANCE
Tel. : + 33 2 40 03 77 74
Fax : + 33 2 51 71 98 06
[illegible]alsim.com
www.alsim.com

Arnaud Nogues
Resp. Recherche et Développement
Alsim simulateurs
ZI La Noe Bachelon
44430 Le Loroux Bottereau

B.8 SILICOMP-AQL



orange Business Services

Silicomp-Aql
195, rue Lavoisier
38330 Montbonnot-Saint-Martin

**Lettre d'intérêt pour le
Projet COACH**

La Société SILICOMP-AQL, filiale d'Orange Business Services et porteuse de sa Business Unit « IT&L@bs », est une PME spécialisée dans le domaine des applications M2M et dans l'intégration d'objets communicants dans les infrastructures, produits et services de ses clients. Pour se faire, elle a développé une compétence « produits embarqués » qui lui apparaît comme une compétence « clé » de ce domaine.

La société existe depuis 1983 et compte aujourd'hui environ 1400 employés. Son siège est à Grenoble.

La société a été informée du projet COACH soumis à l'Agence Nationale pour la Recherche par différents industriels et laboratoires publics, et se déclare très intéressée par ce projet.

En effet, depuis plusieurs années, la société suit et participe avec intérêt aux initiatives de prototypage virtuel de SoC. Si le sujet des micro-contrôleurs a fait l'objet de nombreux projets, celui des FPGA n'a pas été suffisamment traité alors qu'ils constituent une part significative du marché.

L'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est donc particulièrement séduisante. De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est-à-dire avec un coût initial faible ou nul) est un avantage considérable pour le design de solutions à bas coûts et le développement de services associés.

Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures XILINX et ALTERA, leaders mondiaux du marché) et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multitâche décrite en langage C, car cela correspond aux besoins de notre société.

La Business Unit IT&L@bs souhaite donc être tenue informée de l'évolution du projet COACH se déclare intéressée et prête à évaluer les outils développés sur ses projets pertinents, grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

Fait à Montbonnot le 15 février 2010,

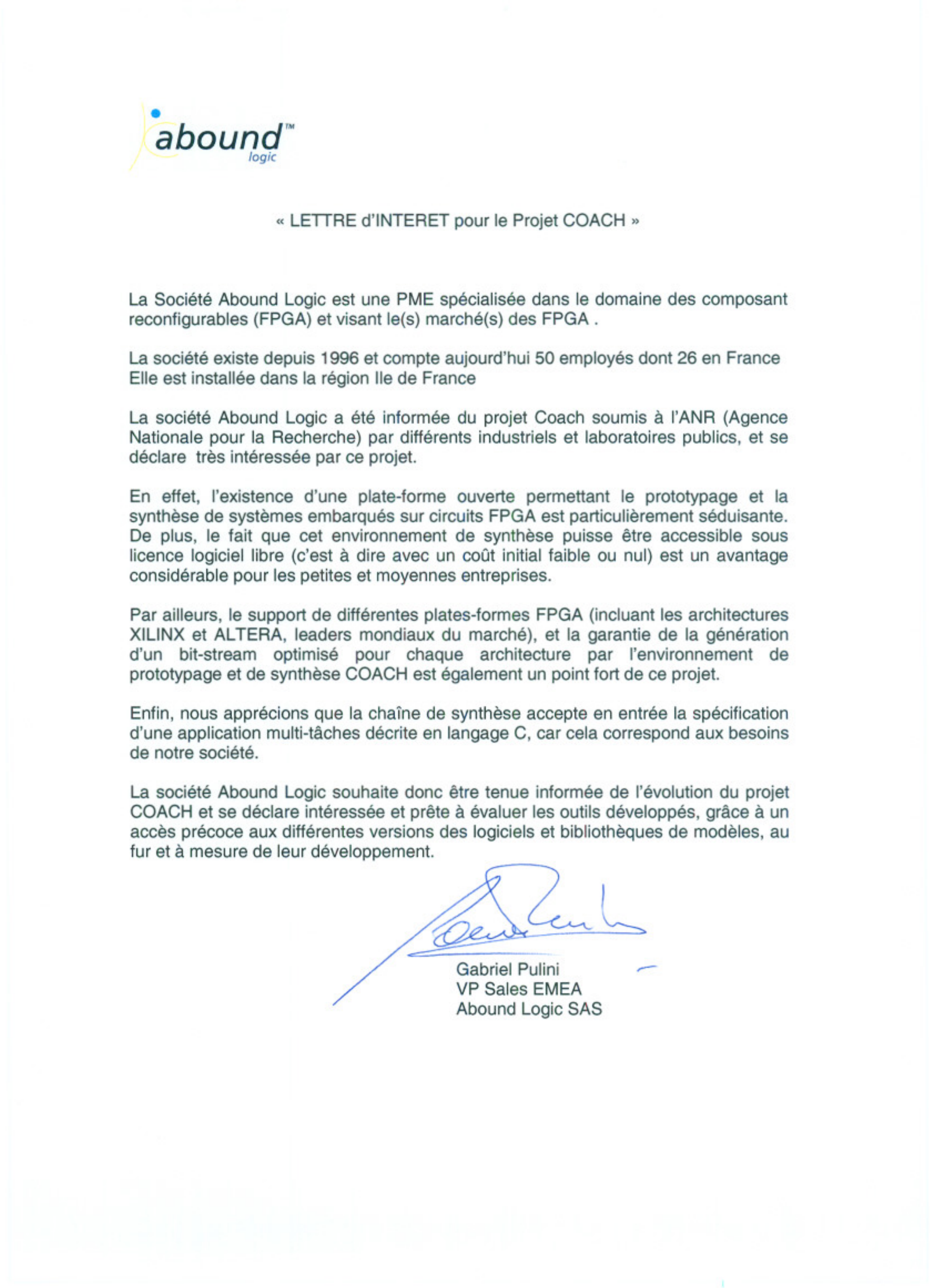
Emmanuel DUFOUR

Responsable des Activités Embarquées

Confidentialité
Les informations contenues dans ce document sont confidentielles et destinées exclusivement aux personnes destinataires. Si vous n'êtes pas destinataire, toute révélation, reproduction, diffusion ou usage de ces informations est strictement interdit. Si vous recevez ce document par erreur, veuillez nous en informer par téléphone et le détruire au plus vite.

Silicomp-AQL est une société de Groupe Silicomp
www.silicomp.com e-mail : info@silicomp.fr
Siège 195, rue Lavoisier – BP 1 - 38330 Montbonnot Saint-Martin
RCS Grenoble B 328 006 432 TVA : FR 92328006432 SAS au capital de 300 000 Euros

B.9 ABOUND Logic



abound™ logic

« LETTRE d'INTERET pour le Projet COACH »

La Société Abound Logic est une PME spécialisée dans le domaine des composant reconfigurables (FPGA) et visant le(s) marché(s) des FPGA .

La société existe depuis 1996 et compte aujourd'hui 50 employés dont 26 en France
Elle est installée dans la région Ile de France

La société Abound Logic a été informée du projet Coach soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels et laboratoires publics, et se déclare très intéressée par ce projet.

En effet, l'existence d'une plate-forme ouverte permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement séduisante. De plus, le fait que cet environnement de synthèse puisse être accessible sous licence logiciel libre (c'est à dire avec un coût initial faible ou nul) est un avantage considérable pour les petites et moyennes entreprises.

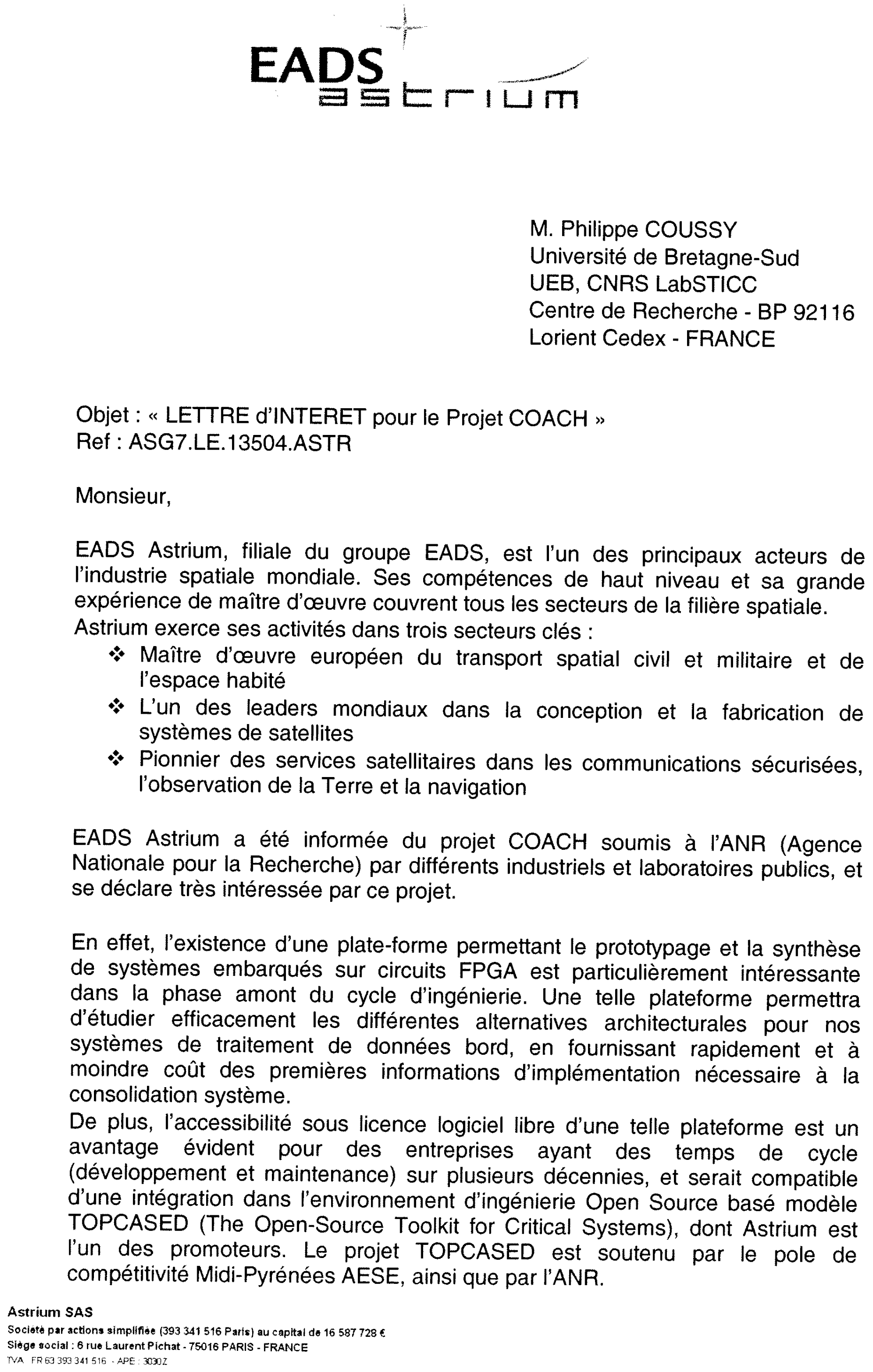
Par ailleurs, le support de différentes plates-formes FPGA (incluant les architectures XILINX et ALTERA, leaders mondiaux du marché), et la garantie de la génération d'un bit-stream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond aux besoins de notre société.

La société Abound Logic souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés, grâce à un accès précoce aux différentes versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

Gabriel Pulini
VP Sales EMEA
Abound Logic SAS

B.10 EADS-ASTRIUM



EADS astrium

M. Philippe COUSSY
Université de Bretagne-Sud
UEB, CNRS LabSTICC
Centre de Recherche - BP 92116
Lorient Cedex - FRANCE

Objet : « LETTRE d'INTERET pour le Projet COACH »
Ref : ASG7.LE.13504.ASTR

Monsieur,

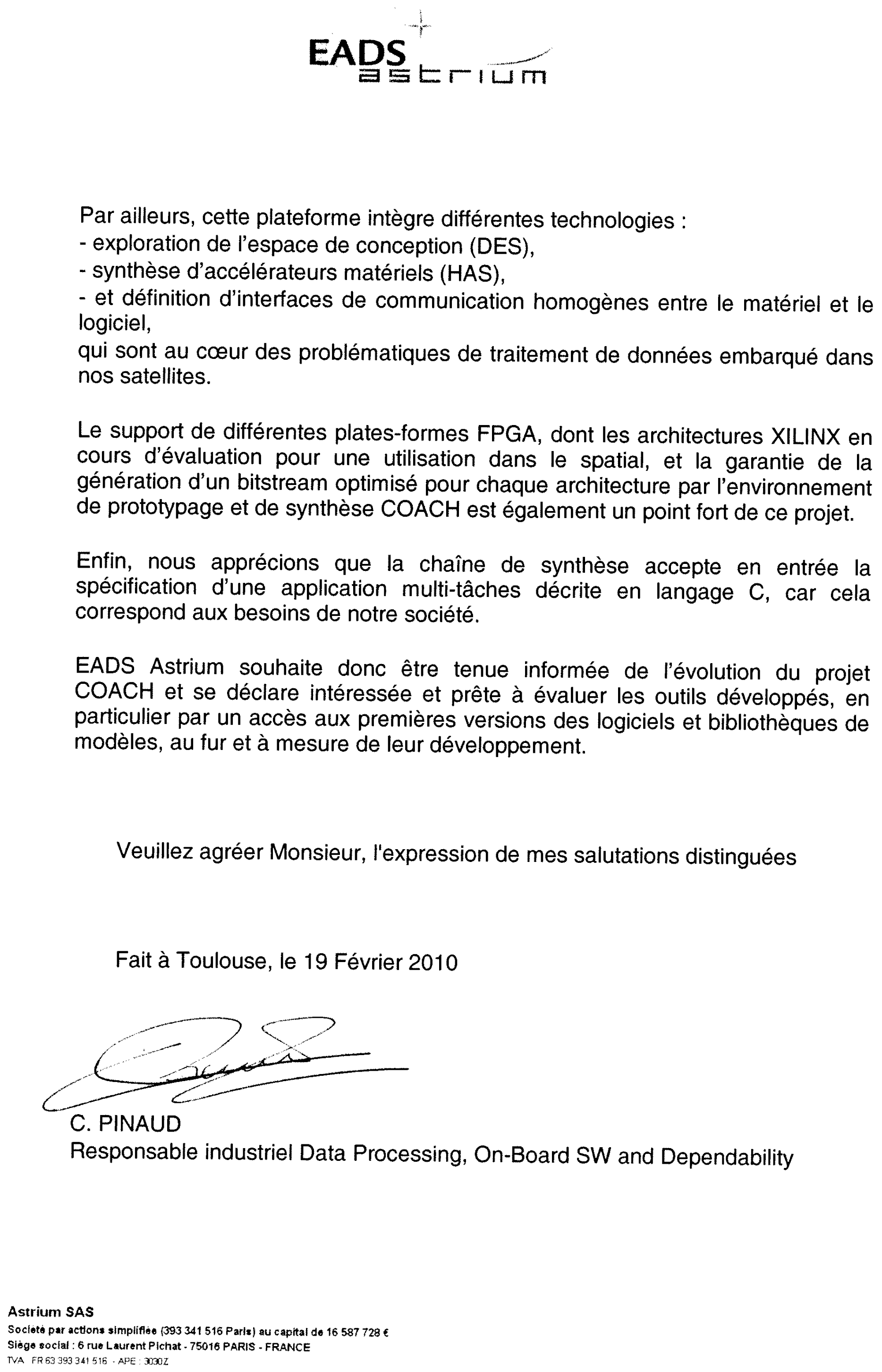
EADS Astrium, filiale du groupe EADS, est l'un des principaux acteurs de l'industrie spatiale mondiale. Ses compétences de haut niveau et sa grande expérience de maître d'œuvre couvrent tous les secteurs de la filière spatiale. Astrium exerce ses activités dans trois secteurs clés :

- ❖ Maître d'œuvre européen du transport spatial civil et militaire et de l'espace habité
- ❖ L'un des leaders mondiaux dans la conception et la fabrication de systèmes de satellites
- ❖ Pionnier des services satellitaires dans les communications sécurisées, l'observation de la Terre et la navigation

EADS Astrium a été informée du projet COACH soumis à l'ANR (Agence Nationale pour la Recherche) par différents industriels et laboratoires publics, et se déclare très intéressée par ce projet.

En effet, l'existence d'une plate-forme permettant le prototypage et la synthèse de systèmes embarqués sur circuits FPGA est particulièrement intéressante dans la phase amont du cycle d'ingénierie. Une telle plateforme permettra d'étudier efficacement les différentes alternatives architecturales pour nos systèmes de traitement de données bord, en fournissant rapidement et à moindre coût des premières informations d'implémentation nécessaire à la consolidation système.
De plus, l'accessibilité sous licence logiciel libre d'une telle plateforme est un avantage évident pour des entreprises ayant des temps de cycle (développement et maintenance) sur plusieurs décennies, et serait compatible d'une intégration dans l'environnement d'ingénierie Open Source basé modèle TOPCASED (The Open-Source Toolkit for Critical Systems), dont Astrium est l'un des promoteurs. Le projet TOPCASED est soutenu par le pole de compétitivité Midi-Pyrénées AESE, ainsi que par l'ANR.

Astrium SAS
Société par actions simplifiée (393 341 516 Paris) au capital de 16 587 728 €
Siège social : 6 rue Laurent Pichat - 75016 PARIS - FRANCE
TVA FR 63 393 341 516 - APE : 3030Z



EADS
astrium

Par ailleurs, cette plateforme intègre différentes technologies :
- exploration de l'espace de conception (DES),
- synthèse d'accélérateurs matériels (HAS),
- et définition d'interfaces de communication homogènes entre le matériel et le logiciel,

qui sont au cœur des problématiques de traitement de données embarqué dans nos satellites.

Le support de différentes plates-formes FPGA, dont les architectures XILINX en cours d'évaluation pour une utilisation dans le spatial, et la garantie de la génération d'un bitstream optimisé pour chaque architecture par l'environnement de prototypage et de synthèse COACH est également un point fort de ce projet.

Enfin, nous apprécions que la chaîne de synthèse accepte en entrée la spécification d'une application multi-tâches décrite en langage C, car cela correspond aux besoins de notre société.

EADS Astrium souhaite donc être tenue informée de l'évolution du projet COACH et se déclare intéressée et prête à évaluer les outils développés, en particulier par un accès aux premières versions des logiciels et bibliothèques de modèles, au fur et à mesure de leur développement.

Veuillez agréer Monsieur, l'expression de mes salutations distinguées

Fait à Toulouse, le 19 Février 2010

C. PINAUD
Responsable industriel Data Processing, On-Board SW and Dependability

Astrium SAS
Société par actions simplifiée (393 341 516 Paris) au capital de 16 587 728 €
Siège social : 6 rue Laurent Pichat - 75016 PARIS - FRANCE
TVA FR 63 393 341 516 - APE : 3030Z