INSTIGATE OVERVIEW
EDA Experience

Number of pages 23
March 13, 2013

www.instigatedesign.com

Contents

1 Introduction

1.1 About This Document

The purpose of this document is to provide high-level overview of Instigate's experience in EDA (mainly FPGA tool-chains and IDE) and related fields, such as RTL design, ESL design, Embedded SW programming, HW-SW co-design, FPGA programming, HW modeling, verification, validation.

1.2 Definition of Terms and Acronyms

CGRA - coarse-grain reconfigurable architecture - reconfigurable devices with nodes that are much larger/complex than LUT's and DFF's typically found in FPGA. E.g. this could be a matrix of ALU's or reconfigurable DSP modules (multipliers, adders) interconnected via switching fabric, with one or few CPU nodes and RAM blocks in each cluster, or could be just a matrix of multi-port RAM and CPU nodes (distributed-memory computer). Main complexity of programming such devices is a need for new paradigm of computing, because neither HW - design flows (Verilog → bit-stream) nor SW design flows (program → parallel algorithm) can be fully automated. Such compute fabrics assume development of parallel algorithms (netlists) from the very beginning.

Core Generators - SW modules, which generate HW IP or FPGA IP (RTL/Netlist) blocks, based on user-specified parameters. Example is a FFT/iFFT generator or Memory Core generator, which instantiates DSP and DPRAM modules available in FPGA fabric to synthesize an FFT (or DPRAM) with specified parameters.

DSP - digital signal processing, refers to set of algorithms (computations) which are performed on digital representations of (usually) analog signals. Can also refer to devices which perform such algorithms: e.g. a HW module embedded into FPGA that can compute the butterfly operation (typically found in FFT implementations) can be considered a DSP module. Can also refer to specially built CPU's, which have special instructions (typically SIMD instructions) for DSP operations.

EDA - Electronic Design Automation, sector of CAD (computer-aided-design) SW industry focused on building CAD tools for automation of various design phases of integrated circuits (ICs). Classical phases for digital design are Front-End: RTL design, synthesis, technology-mapping; and Back-End: floor-planning, place & route. There is also verification flow: Static Timing Analysis **(STA)** and RTL simulation for front-end, and for back-end the STA and Design Rule Checking **(DRC)**. Most of these are also applicable for FPGA-based design flow.

Embedded SW - also referred to as Firmware, is a SW that runs on embedded processors, which are CPU/DSP/Micro-controllers embedded into SoC subsystems (peripherals) and therefore are not managed by the main Operating System scheduler as regular CPU cores. The subsystems are attached to system bus, and appear to the main OS as peripheral devices (RAM blocks, or Character devices), however internally they have own micro-architecture: BUS or Network-on-Chip, a CPU core or micro-controller, and a FLASH or other non-volatile memory, where the FW is stored. In principle the CPU can be a Cortex A processor, and FW can include full-blown operating system kernel, such as GNU/Linux. Example of such device is a high-performance Ethernet card such as KillerNIC, which runs embedded GNU/Linux installation inside.

ESL - electronic system level - refers to several stages in standard SoC design flow. First of all these are initial design phases, starting from algorithmic/functional design (reference implementations in C/Matlab), followed by **structural decomposition** of the system into communicating state-machines (sequential processes). Next step is **Architecture**.

Exploration, as a result of which the system is partitioned between Hardware, Firmware, Software subsets, and architecture specification is defined. This phase is also called **HW-SW partitioning** or trade-off. ESL frequently also includes the subsequent phases of modelling the Hardware part in SystemC/TLM or other modelling languages, to enable **SW co-design** before the HW prototype (e.g. FPGA emulation) is available. This phase is called **Virtual Platform Prototyping**.

FPGA - Field Programmable Gate Array a reconfigurable IC, which consists of universal logical elements (e.g. 4-input 1-output universal boolean functions) called Lookup Tables (LUT), and sequential elements such as D-Flip-Flops (DFF). Elements are grouped into clusters and interconnected both within clusters and between clusters via network-on-chip, which is also programmable. Most of programming technologies for FPGAs are based on ASIC digital design flow: RTL Bit-stream, with synthesis, STA, in front-end, and floor- planning/P&R in the back-end. These tools, however are different due to specifics of the FPGA. FPGA may also have built-in HW accelerators, DSPs, CPU cores, etc. For example, every cluster may have a multiplier, or adder, or other configurable ALU device or DSP modules, as well as one or several dual ported RAM blocks (DPRAM).

GPGPU - General Purpose Graphical Processing Units. NVIDIA and AMD produce GPU devices which are programmable for general purpose computing, using special programming languages: NVIDIA CUDA and Ap-

ple's OpenCL. OpenCL is supported on both platforms, and is also useful for programming regular CPU-s, thus becoming an abstraction layer for programming heterogeneous devices such as NVIDIA Tegra and AMD APU.

Heterogeneous Compute Fabric - compute fabrics which consist of nodes (network) with different compute model, different programming model, such as combination of FPGA fabric with CPU (for example Xilinx Zynq) or combination of GPGPU fabric with CPU (for example NVIDIA Tegra) and combinations of those, for example multi-core/multi-CPU server machine with FPGA and GPGPU cards plugged into PCIE slots. Programming such computers is a multi-discipline process, similar to SoC design, because it involves Architecture Exploration and Partitioning phases, like in ESL flow.

HLS - High Level Synthesis refers to technologies which allow automatic conversion of the algorithm/computation specification in high-level programming languages such as C/C++ and SystemC/TLM to HDL such as Verilog RTL or even directly synthesized into netlist. For Heterogeneous devices such as Xilinx Zynq this would allow targeting the embedded CPU's and the embedded FPGA fabric using the same programming language (e.g. C++).

HW - hardware - refers to HW platforms. Sometimes could refer to subsystems (e.g. SoC audio sub-system) which contain embedded CPU and Flash RAM with **software (SW)** installed in it. This is referred to as **firmware (FW)**.

IDE - integrated development environment, typically refers to a GUI cockpit providing integrated text-editor, debugger/simulator, profiler/wave-form-viewer, and other development tools under unified coherent graphical user interface (GUI). Most IDEs made in the EDA industry have also embedded scripting engine, providing user with scripting interface, to perform most/all of the operations available in GUI via scripting commands, using scripting languages such as TCL, Python, LUA, Scheme, Skill, JavaScript, etc. For programmers the IDE may also provide an Application Programmers Interface (**API**), which allows to customize/automate the GUI even further, by implementing **Plug-Ins** that enhance/modify the functionality of the IDE. In these cases the IDE can also be referred to as **Framework**, i.e. a semi-fabricate SW platform, on which the SW developers can build their own IDE for specific CAD tool. Typically Frameworks also have libraries supporting parallel programming, inter-process communication, communication across network. EDA oriented frameworks may also include HDL parser for RTL, netlist, and other formats.

IFS - Infrastructure - refers to EDA SW infrastructure (IDE + Database Management Layer) which provides data-structures for in-memory representation of design data, import/export and save/load utilities for on-disk representation, and connection of that with Graphical User Interface (GUI Cockpit) and Scripting API for automation. Essential parts of IFS may be provided by underlying SW frameworks such as IBM Eclipse, Nokia QT, or Instigate Application Framework.

RTL - Register Transfer Level - refers to abstraction level used in defining digital circuits in one of commonly used Hardware Description Languages (**HDL**) such as Verilog or VHDL, that is suitable as input for Synthesis tools. Synthesis tools read-in RTL description and produce **netlists**, which are HDL descriptions of the same circuit in much lower abstraction level: instantiating library modules (e.g. LUT/DFF on FPGA) and specifying their interconnect. Each synthesis tool defines it's own subset of HDL syntax and other constraints, which together define the synthesizable subset of the HDL.

SoC - an ASIC that has complete compute system inside: one or many CPU cores, a Bus or Network On Chip (**NoC**) fabric, Memory Controller, and other HW-accelerators or peripheral controllers. Note that SoC can be designed not only using standard ASIC technology, but also can be built on top of a heterogeneous universal chip such as Xilinx Zynq or Nvidia Tegra. In this case the HW-accelerated operations are mapped to the FPGA or GPGPU fabric respectively. For example, some mobile device manufacturers (e.g. Motorola) have started producing smart-phones with Nvidia Tegra used instead of classical SoC

2 Company Overview

2.1 Founders

Enno Wein, Co-Founder and President, Germany [1]

- 20 years of experience in EDA and Semiconductor
- LSI Logic (Europe, USA), Monterey Design, Element CXI (USA)
- Multiple entrepreneur (Germany, Armenia)
- 15+ patents in Semiconductor, Reconfigurable computing, CPU/CGRA (USA and Europe)

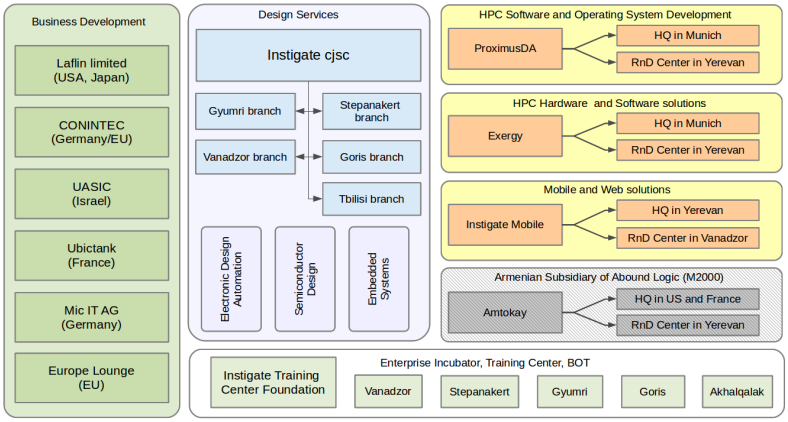
Vahagn Poghosyan, Ph.D, Co-Founder and CTO, Armenia [2]

- 15 years of experience in EDA and Embedded development
- Monterey Design (USA), Synopsys (USA), Element CXI (USA),
- Patents in EDA and Embedded SW development (USA and Europe)

For details see also [3] [4] [5]

2.2 Corporate structure

Overall Instigate ecosystem is depicted below:



The headquarters of the company is in Yerevan, Armenia. The engineering centers are located in Yerevan and 4 other cities in Armenia.

Main business of Instigate lies in design services in EDA, with strong emphasis on FPGA tool-chain and IDE, as well as CGRA, including custom

DSP/ALU/CPU fabrics. The company has a strong experience in writing parsers for RTL, parsers for higher level languages such as SystemC/TLM and C/C++, using various parser/compiler front-ends, and optimizing compilers. Some teams worked on advanced HLS tools, such as power-optimizing C/C++ to Verilog compilers. Complex HW/FW/SW modeling, validation and verification is another major strength of our teams.

The daughter companies, subsidiaries and spin-outs were created to cover other related areas, as well as VC-funded startups to build EDA and semiconductor or SW products in these and related areas.

2.3 Capacity

Overall there are more than 200 engineers working in Instigate and its subsidiaries. Most of the engineers come from Yerevan universities, which have strong history of CS/EE training, with strong emphasis on EDA (especially the Applied Mathematics faculty of the Yerevan State University). They pass through special training program organized by our NGO subsidiary the Instigate Training Center Foundation. Instigate cjsc (the design service company) employs more than 100 engineers experienced in EDA, Semiconductor, and Embedded SW projects. There EDA/Semiconductor experience is mainly related with FPGA tool-chain and IDE development, and relatively less for ASIC.

With proper initial planning Instigate can ramp-up a team of 10-20 engineers within several months in all areas of its expertise, and all types of work: R&D, Application Engineering, Software/Hardware/System-Level Quality Assurance, Configuration Management, IT.

2.4 Example of Capacity Building

Amtokay cjsc is an example of Instigate's capability to ramp-up an entire R&D center for US/European companies in Armenia. It is a turn-key subsidiary created for Abound Logic (M2000) - an FPGA startup located in France and California - which was outsourcing some of it's R&D and SQA projects to Instigate in 2006-2007.

At some point the scope of the work done at Instigate reached the level where Abound Logic was concerned about security of it's R&D progress, schedules, and investment in general, in the case that Instigate undergoes M&A process.

Abound Logic asked for special contract, which would protect their R&D team from being acquired by a third party. The solution was a subsidiary

created and operated by Instigate to exclusively contract with Abound Logic, and protected by special contractual agreement, under which Abound Logic would be able to purchase this subsidiary at certain points in time, with certain predefined price, between 2007-2010.

The project was extremely successful for both sides: Instigate raised R&D, QA, and IT/CM teams, total of 25 engineers, which operated successfully until 2010.

Unfortunately, the Amtokay subsidiary is currently shut-down, following the bankruptcy of Abound Logic in 2010.

Nevertheless Instigate retained significant know-how, experience and skills in this area, and gained valuable knowledge and ability to ramp-up such teams for advanced EDA/Semiconductor projects.

2.5 Instigate recruitment/HR strategy

Instigate Training Center is the foundation of Instigate unique HR/recruiting strategy.

It starts by bringing in interns into Instigate Training Center and teaching them basics of Production System, Quality Management System, Collaboration tools, IT/CM tools, and Industry overview (Semiconductor, EDA, Embedded, Mobile world).

Then selected graduates continue internship by participating in multiple of advanced research projects and open-source projects. Some of these become real products and Instigate spins them out as standalone business, or continues their development in professional environment.

More than 400 new hires passed through Instigate Training Center program. More than 50% of them works at Instigate or it's subsidiaries/spin-outs (200+ engineers) and more than 30% have found jobs in other IT companies in Armenia and world-wide (many moved to USA/Canada/EU and work in companies such as BMW and Nokia).

2.6 Presence and Capacity in Europe

Instigate has partners in Germany through which it provides design services to local customers. Under this partnership it is possible to ramp-up teams in Germany (primarily Munich).

For local support, delivery and training, it is also possible to organize long-

term business trips from Yerevan.

For example in 2012-2013 several developers are working on 1+ year on-site projects in offices of a major car manufacturer in Germany.

2.7 Partners

Instigate has business development partners and sales representatives in several countries world-wide. As the ramping up of teams in Europe may take more time and be limited by the lack of available profiles, Instigate offers the possibility to relocate some of its own developers. In such cases the selection is made from those developers, who are already planning to go abroad for long-term or permanent assignments.

3 Instigate History

3.1 A note on history of IT in Armenia

Being only 2% of population of USSR, Armenia was designing, producing and exporting 40% of the electro-technical equipment of USSR. Most of production was secret factories working for USSR defense industry. EDA algorithms for P&R used by most of EDA tools where developed by discrete-mathematicians of Yerevan Institute of Mathematical Machines.

This and several other institutions have developed USSR's own compute platforms' Nairi and Hrazdan, before USSR decided to switch to IBM360-compatible architecture and started cloning IBM computers.

3.2 Current state of IT in Armenia

After collapse of USSR most of the institutes and production factories were shut-down, privatized or destroyed. Most of the work-force left Armenia to find jobs in Europe and USA. In 1990's and 2000s they started coming back and launching companies and foreign company subsidiaries in Armenia. Currently all together there are about 10000 employees in IT sector, with strong emphasis on complex algorithmic SW development.

EDA leaders such as Synopsys and Mentor Graphics have large (several hundreds engineers) subsidiaries in Yerevan. Recently Synopsys announced expansion in Stepanakert, by starting a special faculty in State University. Synopsys and National Instruments also have large faculty/labs in Yerevan Polytechnic State University.

Note that Instigate Training Center is also currently in process of establishing a LAB in the Yerevan State Polytechnical University, with strong emphasis on DSP, FPGA programming, EDA and Embedded SW development.

See also [6] for information on IT sector in Armenia.

3.3 Incubation: Instigate Training Center (2004)

Enno and Vahagn met and worked together between 2001-2004 at Monterey Design, an EDA startup in Silicon Valley, California, with large remote R&D team in Yerevan, Armenia.

In 2004 Monterey was acquired by Synopsys, which inherited Monterey's Armenian subsidiary, and described this as a talent acquisition, rather than technology acquisition [7]. Enno and Vahagn decided to start their own

business in Armenia, which eventually have grown to multiple companies in Armenia and Germany, some with VC-funding, and some created with private investment and profit generated from the consulting business.

Instigate started in 2004 as training/incubation center in Yerevan, Armenia, with strong emphasis on EDA and Semiconductor development, first of all because of background of its founders, but also because Armenia was one of the leading EDA and Semiconductor development centers of former USSR.

3.4 Original Product Vision

The goal of founders in 2004-2005 period was to rise funding for own EDA tool and framework, including GUI Cockpit (IDE) with plug-in API (similar to Eclipse) that would enable EDA start-ups to build their point-tools in a high-quality GUI environment. The Framework would cover all design phases, starting from early Algorithmic/ESL phases, down to P&R.

This instigated deep study of HW-SW partitioning and co-design technologies, and established tradition at Instigate to pass new hires though combined HW/SW/FW development trainings and career-path.

3.5 Incorporation (2005)

When the team was well trained and capable to develop products, and funds where needed to support it, the founders decided to offer design-services in the same fields: ESL design, Reconfigurable Computing, etc.

The company was incorporated in 2005 in Yerevan, Armenia. First customers where fabless CGRA vendors from Silicon Valley, and HLS vendors from USA and Europe.

In 2006 Vahagn moved back to Armenia, to manage the growing R&D activity as the CTO. Enno is frequently travelling between the USA, Europe, and Armenia and is mainly responsible for the business development as well as strategic technology development.

3.6 FPGA experience centers in Yerevan and Gyumri (2006-2007)

In 2006 Instigate started a subsidiary in Gyumri, the second largest city in Armenia, where we found post-graduate school that provides good FPGA-design classes, and hired first graduates of this school.

In 2007 Instigate engaged in subsidiary creation service agreement with EU/USA based FPGA startup M2000 (Abound Logic) and built a large

team of FPGA fabric & tool-chain validation experts, as well as R&D (P&R, Logic Optimization/Mapping, STA) and CM/IT teams. As part of subsidiary creation service, we have also built separate infrastructure, with completely isolated Internet service, LAN and VPN, as well as exclusive office access for the employees of this team.

Thanks to this, and other customer projects with FPGA involvement, there is a large and well trained team currently available in both Yerevan and Gyumri with good experience in FPGA fabric and FPGA tool-chain validation and verification.

3.7 Instigate Application Framework for EDA SW Development

Between 2006-2008 Instigate has developed two products at it's Training Center: the Instigate Application Framework set of C/C++ rapid application development tools and libraries for development of rich GUI and TCL-based embedded scripting, with strong emphasis on CAD tool development, and Proximus the HW/SW partitioning and co-design tool. Instigate Application Framework versions 2.0 and 3.0 ware used by several external and internal customers to build modern CAD tools with rich documentation, automatic QA (including automatic GUI tests) and automatic generation of on-line documentation and tutorials, as part of build process and regression test system.

3.8 Design Services as Marketing Strategy

Later, when it was possible to bring investment, support own development and build the products as planned, Instigate decided to continue the design-service business in parallel, because it was a great source of marketing and knowledge on customer needs.

In many cases the design service provided by Instigate to customers was accompanied with products offering: Proximus or Framework licenses were provided as part of design service to customers, where it was possible to significantly reduce the HW modeling effort, or IDE/GUI coding effort by using these products.

This enabled early validation of our products on real-life projects, even in the phase when they were not fully functional, but were already mature enough to significantly reduce R&D and SQA team efforts, significantly cutting costs for customers, and increasing Instigate's competitiveness as R&D, QA, Application Engineering and Tech-Writer service provider.

3.9 Proximus - ESL spin-out (2008)

In 2008 Proximus was spun-off into Germany (Munich) based VC funded startup, with R&D Center in Armenia. Since then it expanded it's product line to cover heterogeneous computing, High Performance Computing and parallel system design in general.

Currently it has one major SoC customer STMicroelectronics, in Grenoble, and is now in process of validating the Proximus as runtime environment for cluster-computing and heterogeneous computing.

3.10 Mobile and Web experience centers (2010-2012)

Since then establishing branches in regions of Armenia became one of the main points of our HR strategy. Currently Instigate has subsidiaries in Yerevan, Gyumri, Goris, Vanadzor, Stepanakert (Armenia) as well as subsidiary in Tbilisi and Akhalqalak (Georgia). For design services and solutions where Mobile and Web technologies are involved, we also established subsidiary specialized in iOS, Android, and other mobile platforms and Web technologies Instigate Mobile cjsc, based in Yerevan, Vanadzor, and Stepanakert (Armenia).

3.11 Exergy - Energy Efficient High Performance Computing (2012)

In 2012 Instigate started spin-out of second start-up, Exergy cjsc, which is specialized in building High Performance Compute servers and workstations, with up to 64-core AMD CPUs and up to 8-core GPGPUs in single water cooled high-quality server box, produced in partnership with German server manufacturer, using our water cooling technology, and using Proximus as Operating System for heterogeneous/cluster computing. It is also possible to equip Exergy servers with FPGA boards attached to PCIE or other interfaces and use Proximus to develop applications for such computers. Currently Proximus is developing technologies for hot-swap of modules on the FPGA fabric without rebooting main operating system. Main experiments are done using Xilinx Zynq platform.

3.12 Yerevak - H.265/HEVC Technology startup (2013)

Next internal product which is currently in process of funding is an H.265 technology and product startup, which researches modern video-coding standards, including up-coming H.265/HEVC and VP9 standards, and builds universal parallel implementations and models of these algorithms using Proximus technology and Exergy servers.

Then it applies the gained knowledge of intrinsics and parallelization possibilities, to implement these algorithms on various platforms and compute fabrics, to build various products, ranging from ASIC/SOC implementations, which require HW-SW trade-off decisions and architecture exploration, to Heterogeneous chips such as NVIDIA Tegra and Xilinx Zynq, to high-performance computing heterogeneous servers, such as Exergy computer with 64 AMD CPUs, 6000+ NVIDIA GPGPU cores, and possibility to add FPGA cards for further optimization and off-loading of computation-intensive parts. Latter is applicable in high-end products such as 8K digital cinema servers, teleconferencing solutions, as well as IP-TV back-end servers.

4 Coarse-grain reconfigurable computing

Instigate had multiple CGRA customers, involving both development and verification of both the HW fabric and R&D tool-chain.

4.1 Tool-chain development and verification

For several CGRA devices we have developed various parts of R&D tool-chain:

Compiler for embedded custom processors (16 bit architecture, custom instruction set). Compiler is developed completely from scratch, using simplified version of C, with special hooks to access additional HW accelerators and I/O modules available on system bus via special instructions. I/O modules allow communication with other nodes in the CGRA array.

Compiler for 32-bit RISK processor on a distributed-memory architecture (Mesh of CPU and RAM modules) with special hooks to access multi-port RAMs and send/receive information to remote nodes, and remote RAM blocks, that are not directly accessible from current node.

Design and implementation of custom Assembler-like language for CGRA fabric, providing combination of declarative language (netlist description) and imperative language (DSP configurations, CPU programs). Implementation of compiler, linker, loader.

4.2 Early evaluation of the architecture

Instigate was involved in early modeling of CGRA fabric for a customer in a very early phase of architecture specification. Early SW models of the half-specified fabric enabled validation of the future HW fabric from performance and programmability (expressiveness, convenience) point of view in very early phase of the architecture specification.
Applications engineering team was using an IDE/GUI front-end on top of this early model to develop applications. This enabled early feedback on (dynamic) routing resources availability and other hard-to-estimate performance metrics in a phase when the HW, ALU specifications, instruction-sets, and BUS/NoC arbitration logic was not fully designed/chosen.

4.3 Bit-accurate HW models

The early HW model was maintained in parallel with refinement of the architecture specification, and eventually turned into cycle-accurate and bit-accurate model of the actual chip. It was integrated into IDE with rich GUI, displaying the entire device hierarchy, and allowing to program and

debug the model interactively, with single clock stepping, register probing/modification, waveform analysis, and many other convenient features.

4.4 HW Verification

For this and other CGRA customers we did HW Verification as well, using functional and performance tests, developed for verification purposes, and tried first of all on HW models, then on FPGA emulated prototypes, and finally on the actual HW. In some cases HW was provided to Instigate locally (in Yerevan/Gyumri) and in other cases it was made available over the network, with special power management system controllable over VPN.

4.5 Functional models

In order to speedup execution of functional tests, a special model was developed, which was fully compatible with bit-accurate simulator, but excluded all the timing characteristics except temporal dependencies between events. Optimized netlist was generated based on the HW configuration, using graph arrangement and other combinatorial optimization algorithms. The result was 100-1000x simulation speedup for running regression tests.

4.6 Core Generators

Instigate developed rich set of Core Generators for several CGRA fabrics. These are functionally similar to Matlab DSP block-set, providing automatic synthesis and mapping of the netlist onto the CGRA fabric, instantiating HW-accelerators and ALU modules available in the fabric, and configuring the switching fabric according to the interconnect.

4.7 IDE development

All above mentioned tools were integrated into GUI cockpit (IDE) providing convenient access to these tools, integrated with source code editor with syntax highlighting, visualization of the compute fabric and it's configuration bits, possibility to visually program both the logical interconnect and mapping of it onto physical device.

The IDE also integrated library of Core Generators, and Schematic Editor to build hierarchical designs from generated/instantiated cores.

5 FPGA IDE/Tool-chain development, Validation, Testing

5.1 FPGA Test designs

For several FPGA customers we have developed test-designs, starting from simple library-modules to full-blown systems (e.g. several million LUT/DFF), instantiating also the DSP, DPRAM, and other coarse-grain reconfigurable nodes available on the fabric. Example of such designs is a multi-core micro-architecture using LEON3 CPU and AMBA bus fabric, fully operational: with boot-loader and application running inside the system as part of the simulation.

This was used to validate both the Synthesis/P&R tool-chain, as well as to validate the actual fabric. IO testing was also performed by Instigate for several custom IO blocks. Engineers have good skills in reading complex specs (e.g. SERDES) and developing good test-plans based on these.

5.2 FPGA IP development

IP libraries such as parametric modules for instantiation of DSP, DPRAM and IO modules were developed by Instigate for customer's FPGA fabric.

5.3 System Level validation and verification

Instigate offered system level validation services to CGRA fabrics in the past. Similar project can be implemented for FPGA vendors. Currently we are building such solution based on Xilinx Zynq for Proximus-DA. The goal is to prove that Proximus is useful for programming CPU + FPGA combination. Full blown applications are being developed partially in C/C++ and partially in Verilog and mapped to an Exergy server with PCIE-attached Xilinx FPGA board, and/or to Zynq card, utilizing embedded ARM fabric.

Instigate has all necessary knowledge to implement complex full-blown real-life applications, which would be mapped to a CPU + FPGA combination and prove usability of the FPGA fabric and Synthesis/P&R tool-chain in the complete flow, taking into account all nuances of collaboration between HW, FW and SW teams in such project, and providing feedback to the R&D team on usability, functionality, performance and documentation of the tool-chain.

5.4 Core Generators

For complex IPs, rather than implementing them as parametric RTL modules, we have developed SW core-generators, which produce optimized fully

or partially synthesized netlists based on the user-specified parameters of the IP.

5.5 P&R tool-chain

Instigate has experience in building various P&R tools either completely standalone, or as part of distributed team, with leads working remotely in the US and/or EU. Due to background of founders, and strong school in Yerevan State University, Instigate, and Armenian EDA companies in general, are traditionally specialized in Back-End: Floor-planning, Partitioning, P&R. For several customers we have also worked on STA and elements of Front-End: synthesis and logic optimization.

5.6 FPGA IDE and IFS development

Instigate has good experience in developing various IDEs and GUI Cockpits for EDA tools, with and without using it's Application Framework for EDA SW development. This also includes database/IFS development, import-export utilities, HDL parsers, etc.

5.7 HW validation and verification

In addition to IDE/Tool-chain verification and validation, Instigate also provides services in development of functional models of HW components (DSP, I/O, LUT/DFF, etc), as well as their usage in SQA tool-chain, in validation of HW, as well as other tasks related to HW fabric validation, bit-stream validation, etc, which require good understanding of all aspects: HW fabric, Tool-Chain, and application domain (RTL development, simulation).

6 Electronic System Level Design

6.1 Modeling heterogeneous fabrics

As it has been mentioned in chapter 4, while working on reconfigurable and heterogeneous fabrics, Instigate had to develop not only tool-chain for configuring those fabrics, but also SW models of the HW, to enable SW co-design in the early phases of the HW Fabric development, when the actual chip, or the FPGA emulation was not yet available.

Later on the model was still useful for regression testing, as well as for HW validation to run unit-tests of the HW IP blocks. Multiple models have been built by Instigate for different devices at multiple abstraction levels: fast functional models for functional SW co-design, verification, and HW validation, approximately timed models for SW architecture exploration, HW performance analysis, and overall system-level performance validation, and clock-accurate models for detailed SW/FW validation, performance analysis, and SQA.

6.2 Algorithm parallelization and mapping to reconfigurable devices

Instigate's main experience is parallelization of algorithms and functional reference implementations with the purpose of subsequent mapping of the resulting netlist onto various target fabrics.

These fabrics are ranging from parallel multi-core/many-core CPU fabrics, to heterogeneous devices such as NVIDIA Tegra and Xilinx Zynq, as well as pure FPGA implementations. Such experience was put into different EDA tools, ranging from ESL design and modeling environments such as Proximus, to HLS tools, such as C RTL translators. Instigate has experience in developing C/C++/SystemC/TLM code parsers, and analysis tools, which help the user to semi-automatically or fully automatically parallelize the sequential code.

We believe, however, that systems such as Xilinx Zynq should be targeted in the same manner, as classical SoC design: it's a multi-discipline team work, requiring all phases of classical SoC design: algorithmic phase, structural decomposition phase, HW-FW-SW partitioning (Architecture Exploration), and subsequent implementation of HW, FW, and SW by corresponding design teams, including the TLM modeling team which would implement the SW models of the future Hardware, to enable early SW co-design.

6.3 Proximus - new paradigm for parallel/reconfigurable computing

Targeting wide variety of reconfigurable, parallel, and heterogeneous fabrics, Instigate has come to conclusion that hierarchical netlist with blocking I/O (FIFO semantics) is the universal approach to logical design of future electronic systems. Our proposed flow has Logical Design (Front-End) step based on this paradigm, followed by Physical design (Back-End) which allows actual mapping of this logical hierarchical netlist onto the existing or future (yet to be designed) platform, such as FPGA, or heterogeneous reconfigurable devices such as Xilinx Zynq, Nvidia Tegra, or regular computers/servers with FPGA or GPGPU cards plugged into their PCIE slots.

Based on this methodology, Instigate has developed a tool-chain and IDE for universal parallel design of systems, called Proximus. In 2008 this was spun-off into independent company, which continues buying services from Instigate, as well as being the main user of it's EDA Application Framework, thanks to which Proximus has saved multiple millions of $ on SQA, Tech-Writers, and GUI/IFS team costs.

7 Embedded SW Development

7.1 GNU/Linux kernel and driver porting

While working with various CPU's embedded into heterogeneous/reconfigurable fabrics, Instigate had multiple projects involving FW development, as well as OS/Driver porting, testing, fixing. This includes both work on SW models (instruction-set-simulators) of the CPU's, as well as actual hardware.

7.2 Development of drivers for IP's mapped to reconfigurable fabric

One of currently active projects is an ambitious large-scale platform development for Xilinx Zynq platform, as part of Proximus Runtime Environment. The goal is to enable HW-accelerated IP mapping to the FPGA fabric available on the Zynq, and activation/deactivation of these HW peripherals without rebooting GNU/Linux or Android/Linux platform running on the embedded ARM cores. This includes development of special drivers in combination with bit-stream loading/unloading and partial reconfiguration of the FPGA.

7.3 Mobile SW development

When the solutions need to include applications for Mobile devices or Web-technologies, then Instigate Mobile cjsc can provide necessary resources. For example, for one of our EDA customers our web technology experts are building fully customized feature-rich automatic build system, with web-interface for control and interactive browsing of consolidated reports, simulation statistics, profile results, etc.

References

[1] Enno Wein, Linked-In profile -
`http://www.linkedin.com/pub/enno-wein/2/b/661`

[2] Vahagn Poghosyan, Linked-In profile -
`http://www.linkedin.com/pub/vahagn-poghosyan/4/284/787`

[3] Patents, Enno Wein -
`http://patent.ipexl.com/inventor/Enno_Wein_1.html`

[4] Patents, Vahagn Poghosyan -
`http://patent.ipexl.com/inventor/Vahagn_Poghosyan_1.html`

[5] Patent, Proximus technology -
`http://www.highbeam.com/doc/1P3-2251949181.html`

[6] Armenian IT sector overview -
`http://www.caps.am/UserFiles/File/EIF-ENG.pdf`

[7] Commentary on EDA, including reference to Monterey Design's acquisition by Synopsys (link)

[8] Instigate Web-Site -
`http://www.instigatedesign.com`