APPLICATION OF THE "METHODOLOGY ASSESSMENT FRAMEWORK (MAF)" ON AN EDA TOOL DEVELOPMENT PROJECT

Ebru Dalbudak
Computer Science and Engineering, Université du Québec en Outaouais
Gatineau, QC, J8Y 3G5, Canada
dale02@uqo.ca

Karim Baratli
Computer Science and Engineering, Université du Québec en Outaouais
Gatineau, QC, J8Y 3G5, Canada
bark07@uqo.ca

Yousef Fouzar
Computer Science and Engineering, Université du Québec en Outaouais
Gatineau, QC, J8Y 3G5, Canada
yfouzar@yahoo.ca

Bachir Lakhssassi
Computer Science and Engineering, Université du Québec en Outaouais
Gatineau, QC, J8Y 3G5, Canada
Lakb01@uqo.ca

Karim El Guemhioui
Computer Science and Engineering, Université du Québec en Outaouais
Gatineau, QC, J8Y 3G5, Canada
Karim.elguemhioui@uqo.ca

Ahmed Lakhssassi
Computer Science and Engineering, Université du Québec en Outaouais
Gatineau, QC, J8Y 3G5, Canada
Ahmed.lakhssassi@uqo.ca

Abstract—With the increased level of complexity and market pressures to deliver better results faster and cheaper, Integrated Circuit (IC) companies are looking to innovate and introduce efficiencies into their development process. To improve IC design productivity and time-to-market, this paper introduces an innovative Electronic Design Automation (EDA) tool in order to replicate an existing layout in different technology nodes by automatically porting analog and mixed-signal circuits. Furthermore, the paper presents a case study of the application of the Methodology Assessment Framework (MAF) as an evaluation tool to choose the best suited Software Development Methodology (SDM) for the EDA tool development project.

Keywords— IP Porting, IC design productivity, EDA, Technology Migration, Layout Migration, SDM, agile

I. INTRODUCTION

Integrated Circuit (IC) design complexity has been exponentially increasing since the appearance of transistors in 1958 [1]. Each generation of IC design process introduces improvements such as reducing the size of the transistors and wires and while increasing the number that can fit on a chip. Consequently, this allows the chips to run at higher frequencies and then the new designs get more sophisticated.

As the process technology advances, IC design rules[1] that are provided by semiconductor manufacturers to determine what can and cannot be manufactured also get more complicated and usually they are not compatible with previous generations [2]. According to Zhu et al.[3], "manufacturing processes are updated every 18 months each with a different set of design rules", and companies need to offer different versions for different foundries. The standard cell library, which consists of standardized layout specific components, for each process update also increases [4]. As a result, regenerating a new standard cell library for each process update gets more costly and time consuming [4, 5]. Moreover, the circuits need to migrated to work with the new target technology and this requires a tedious, often manual process of iterating layout changes, extraction and simulation until desired characteristics are met [1].

To maximize their design investments and reduce their design duration, Intellectual Property (IP) providers reuse blocks and hard IP whenever possible. At the beginning of or during a new IC design project, it is often necessary to use and re-use available IP and circuits from previous projects, instead of designing everything from scratch [6, 7]. Design reuse makes it faster and cheaper to design and the time lost by completely re-designing an existing circuit in another technology process can then be spent on new parts of the system that add extra functionality and/or increase performance and productivity [8].

Furthermore, since IC companies choose to outsource the fabrication of their semiconductor chips to foundries (by going fabless), they need to migrate their existing circuit designs "between process technologies and within internal and external foundry processes" because, frequently, foundries differ in their versions even if they use same process technology node [7]. Particularly, the number of metal layers, material table, etc.

Hence, there is a growing need in the semiconductor market to migrate an existing circuit layout between different technology

[1] Design rules determine whether the physical layout of a given circuit satisfies a series of recommended parameters to ensure that most of the parts will work correctly.

Copyright © AJSE Publishers

nodes and foundries in a reliable, fast and cost efficient manner [2, 3, 8, 9]. To produce designs quickly and to address design challenges, IC companies have been increasingly making use of the EDA tools [1].

This paper introduces two innovative approaches that will improve IC design productivity and hence will facilitate faster semiconductor development and time-to-market. First, it introduces a new EDA tool in microelectronics which aims to automatically port analog and mixed-signal circuits between technologies nodes in the hard IP market. Next, it presents the application of a new framework called Methodology Assessment Framework (MAF) on this EDA tool development project to demonstrate how the agile methodology was chosen to be the best suited SDM to promote the adaptive planning and evolutionary development and delivery of a module of this tool using short and iterative development cycles.

The article begins with introducing concepts relevant to the study as an overview. Next, it provides a detailed discussion on layout porting and its challenges. The paper continues with the introduction of the new EDA tool called OPIC that can help better manage the complexity in microelectronics for IP design porting of analog and mixed signal circuits between technology nodes. It proceeds with the case study on the use of agile methods during the development of a Design Rule Check (DRC) clean-up module of the OPIC tool. And it ends with concluding remarks.

II. Background / Related Work

There are studies that were focused on IP reuse and technology migrations due to their expected benefits.

Sobe et al. [10] presents a method, that reflects common practice of analog design divided into design data conversion and sizing by optimization, to convert and optimize a circuit topology. Fu et al. [11] introduce a new rectangular topological layout for topology-driven cell migration. Kar et al. [12] talks about the technology migration process and introduce a new layout preserving migration tool called: TECHMIG. Francken et al. [6] presents a methodology for the migration of analog circuit design from one technology process to another, considering both the sizing and the layout phase. Diodato et al. [13] discuss porting of 2.5 micron CMOS VLSI design into 1.75 micron design rules. They were able to implement the new design in 33% less time and achieved smaller size in design rules by 19% and higher operating frequency by 43%. Dornelas, Helga et al. [5] introduced a new technology migration methodology for analog IC design and instead of doing the migration manually or at netlist level, they performed automated migration on a schematic level, followed by a robustness verification with the usage of MunEDA tools [5, 10].

However, literature review has revealed that even though there is abundant research in the field of IP reuse, design migration and layout porting, their use in analog and mixed-signal circuits is not as common [2].

The literature review on the EDA tools and the use of agile in IC development projects has proven that the use of agile methods in hardware development projects is not as common as in software development projects [14]. Innovation is a great challenge in the EDA industry and to provide faster, better, and cheaper products, the major players in the industry focus on improving existing tools and methodologies [15]. According to Bryant et al. “The quality of the design tools and associated methodologies determine the design time, performance, cost and correctness of the final system product” [16].

III. Layout Porting

As Francken et al. stated “porting successful designs from one technology to another can distinctly reduce the design cycle” [6]. The aim of layout porting between technological process nodes is to be able replicate an existing layout (reference design) in different technology nodes by modifying or adapting it to a newer process design rules by taking advantage of the reusability of the layout resources accumulated from the initial process. Semiconductor foundries continue to innovate by continuous scaling of transistors. There is a growing need to redesign mixed signal circuits for new technological processes. The analog domain still lacks automation, which resulted in a manual design of analog circuits each time to migrate from one technology to another. While doing that the ultimate goal is to “achieve high quality layout with high integration and low power consumption and to shorten the design-time-to-market” [8]. This way, instead of spending time and effort from scratch to completely redesign an entire circuit, semiconductor companies can focus on innovation by adding new functionalities into their designs and focusing on increasing performance [6].

IC products are usually mixed signal designs, in other words, they comprise of a combination of digital and analog functional blocks and circuits and the analog portions of these ICs are the most difficult portions of the chip to migrate between foundries and technology nodes [2, 17, 18]. In digital designs, reuse of IPs has already been well established thanks to fully automated digital workflow [8]. However, in analog circuit designs, reuse of IP have challenges, such as “coping with trade-offs among analog specific requirements, such as noise, linearity, gain, supply voltage, speed, power consumption, self-heating, etc” [8].

Porting an existing design from technology A to technology B usually follows a two step design flow. In the first step, the given schematics and topologies are converted from the source to target technology and existing layout and mapping layers are migrated by recognizing different structure and device placements; preserving all floorplan, matching, routing and interconnect within the given layout [7, 19]. In the second step, these circuits and IPs are sized for the new target specifications and optimized for the new target process

Copyright © AJSE Publishers

technology. Then, Design Rule Check (DRC) verification is conducted to preserve all matching within given layout and all routing and interconnect and Layout Versus Schematic (LVS) will be performed to ensure that the layout conforms to the rules required for faultless fabrication and performance [4].

At the present moment, in the IC design process, schematics and layout design phases are carried out sequentially. We start with the original reference design (for example 65nm), when the technology changes and we want for example 45nm technology node, we need to first migrate the specs (voltage is different, Vref(min, max)); get the schematics and circuit topologies for the new technology; translate schematics and topologies from 65nm process technology to 45nm process technology (via script); port the schematics from 65nm to 45nm; verify schematics; then use the schematics to generate/port the layout from 65nm to 45nm and verify the layout created in 45nm. As can be seen, in this type of a design, we do not take advantage of IP-reuse; design that works in one technology will not work in the other technology. So, IP has to be completely redesigned when the technology changes, which leads to more errors in the new design, added time, cost and human resources. Just to give an idea, today, about 100 engineers work together to design a new IP.

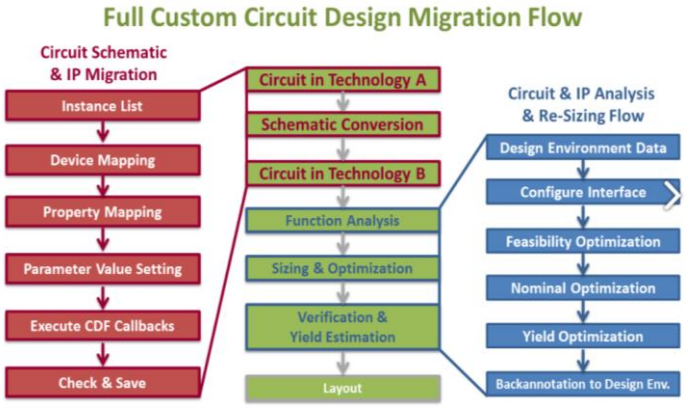


Figure 1: 2 steps of an IP Migration & Porting Flow [4]

IV. New Layout Porting Tool: OPIC

This paper introduces a new automated layout porting tool to increase the effectiveness and efficiencies of the design conversion process by reducing the human resources required and by shortening the porting time while reducing the failures during conversion.

There are some target layout recognition and conversion tools available commercially. However, these tools perform porting only when the reference design and the target technology is highly specialized (e.g. microcontrollers, etc.) and homogeneous (e.g. FPGA to FPGA).

Available commercial tools do not include features like automatic porting between technology nodes and foundries for analog and mixed-signal circuits; they do not re-use the existing layout by making changes on it; instead they regenerate the layout from scratch using traditional manual porting methods. Given the size and complexity of each migration, traditional manual porting methods are too time-consuming and costly. They are also risky and human intervention could impact the quality of the IP resulting in the loss of work that was already done in previous technology. IC designers need to reduce a risk of having costly design bug during the manual porting process from scratch. Therefore, the new layout needs to be retested rigorously for reliability, which impacts the IP delivery timelines. Moreover, instead of losing their precious time on repetitive manual tasks, IC designers should rather focus on using their creativity and precision crafting their designs.

A tool that would automatically migrate any IP design and porting flow with minimum human intervention from one foundry specific technology to another would be invaluable for any IP provider. By using this kind of an automated tool, semiconductor companies can increase the efficiency of their IC design process by reducing the porting time and the risk of creating bugs or errors, as well as the costs.

OPIC, which stands for "Outil de Portabilite entre technologies des Circuits Integrés/ Porting Tool for Integrated Circuits", is a new EDA tool that is designed for hard IP market to automatically perform the porting tasks with minimum human intervention and replicate an existing analog/mixed-signal circuit design (reference design) within different foundries and technology nodes.

The OPIC tool responds as well to an immediate need in the field of semiconductors and historical transition from traditional planar CMOS transistors to FinFETs, which is pending any adequate solution. The tool will be particularly useful in a variety of different important sectors, including IP providers, silicon foundries etc. within the semiconductor industry.

This automated porting tool is a type of back-end design tool, which aims to provide a quick and robust analysis of IP using existing layout, floorplan and routing between the blocks and will generate a DRC (Design Rule Check) and LVS (Layout Versus Schematic) clean layout in the targeted technology with minimum human intervention. The tool will port the graphical patterns (polygons) for each transistor and wire on the IC from one technology to another while preserving the key characteristics of an existing layout including matching of critical components and their relative placement in order to create a layout design that will function properly when fabricated.

The tool follows two major steps; first it will create a DRC and LVS database in technology B and then will perform circuit optimization and reflect schematic changes onto a clean layout.

Copyright © AJSE Publishers

The OPIC tool can be used to do the circuit and IP porting of a layout in source technology A to target technology B by converting the given schematics and topologies from the source to the target technology. It checks the geometries against geometry design rules, detects design rule violations according to specified IC parameters and automatically cleans them iteratively. As a final product, it generates an IC design in GDSII format, which is ready to be shipped to foundry for manufacturing.

The tool takes an existing design layout in a given technology (technology A), which consists of network of transistors and described by a geometrically exact Graphics Data System II (GDSII) file, which is the de facto industry standard for data exchange of IC design layout, and generates a new layout design in a different technology node (technology B) by using the design spec for the new layout in technology B and the technical mappings (for both A and B). which can be reviewed by a physical layout designer before it gets sent out to the foundry for manufacturing.

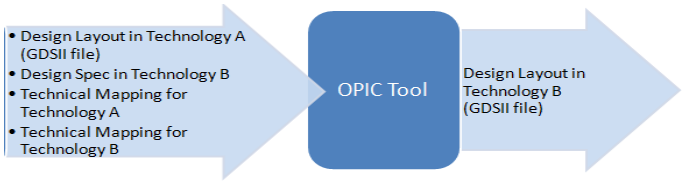


Figure 2: OPIC tool

To perform porting between two technological process nodes (from A to B), we need to have the following as input and follow the following steps listed below.

Input: Existing Layout Design in technology A in GDSII format, Design spec for the new layout in technology B, Technical mapping for technology A, Technical Mapping for technology B

Steps:

1. Open the given layout design in technology A with spec B and Existing Layout Design in technology A in GDSII format, Design spec for the new layout in technology B, Technical mapping for technology A, Technical Mapping for technology B migrate the layout into new layout design in technology B
2. Run the EDA tool called Design Rule Check (DRC) on Layout of technology B to identify the violations of design rules
3. Correct all the violations automatically and iteratively until the violations are cleaned
4. Obtain 100% accurate (target) layout for new design in technology B

Output: New layout design in technology B in GDSII format (binary file), which is ready to be given to IC foundries

We can run simulations on this new layout design in technology B to test it further and send this layout to a Physical layout designer for verification before the design gets sent to the foundry.

V. Case Study: The Use Of methodology assessment framework (MAF) In the OPIC Project

A. *The MAF framework*

Selecting the appropriate SDM for a project is one of the most important steps to ensure the project success [20, 21]. There are so many SDMs in the market but how do we choose the right one? How do we determine and measure the concept of "fit" between the chosen SDM and the project? According to Mullaly & Thomas, "attaining fit suggests that there is an alignment between what is being implemented and the environment and situation of an organization"[22].

The literature review has proven that there are several frameworks that have been used to define, compare and evaluate SDMs [23-26]. However, there is yet no comprehensive framework that uses the crucial factors to determine whether the traditional/plan-driven/waterfall, agile or hybrid type of system development methodology would be most suited for a project. Existing evaluation frameworks lack several aspects. Most of them have not considered evaluating agility. The nature and the complexity of the project as well as the extent and enrollment of the project's constituent base have been neglected or only partially addressed. To address the need of a comprehensive evaluation framework, this paper has identified seven factors that are critical to providing the project's executive sponsor and governance body with a dashboard view over the project landscape. They are as follows:

1. Outcomes being addressed by the project (OUTCOMES)
2. Scope /features of the project (SCOPE)
3. Nature and complexity of the project - CYNEFIN framework (CYNEFIN)
4. Extent and enrollment of the project's constituent base (CONSTITUENTS)
5. Applicability of Agile principles to the project (AGILE PRINCIPLES)
6. Team expertise and experience in system development methodologies (TEAM)
7. Maturity of the organizations involved on the project (ORGANIZATION)

This paper introduces a new framework called "Methodology Assessment Framework (MAF)" as a tool that is based on these seven decision factors to come up with an overall assessment of a project to help determine the software development methodology approach that is best suited.

The OPIC project was evaluated against the seven factors of MAF (Outcomes, Scope, CYNEFIN, Constituents, Principles, Team, and Organization), using a series of self-evaluation tools that have been developed for each.

Copyright © AJSE Publishers

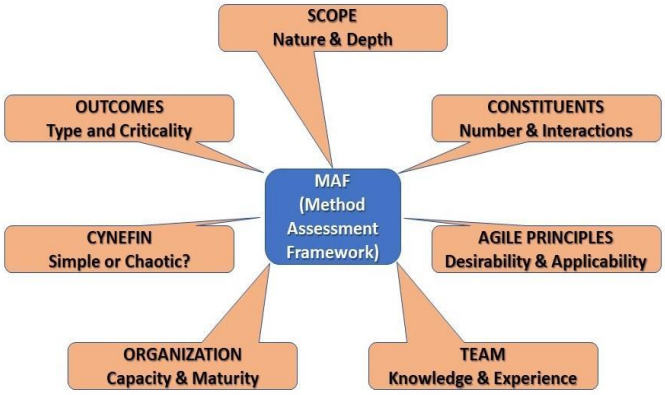


Figure 3: MAF framework and its 7 decision factors

The MAF suggested that an agile methodology is best suited SDM for this project because the project scored very low on Outcomes, Scope and Constituents dimensions while reasonably high on the CYNEFIN, Organization, Team and Agile Principles dimensions as demonstrated in the table below and using the radar chart in Figure 4.

MAF factors	Description
Outcomes	Important - The number and criticality of outcomes have a bearing on the development method selected.
Scope	Small project with low complexity
Constituents	Small project, few constituents with basic role (*Inform*, *Consult* or *Educate)*
CYNEFIN	Complex, with a bit of uncertainty, innovation and experimentation are required, problems and solutions are evolving
Organization	Established in terms of organizational capacity and maturity
Team	Advanced, in terms of skills and expertise
Principles	100% supports agile values and principles

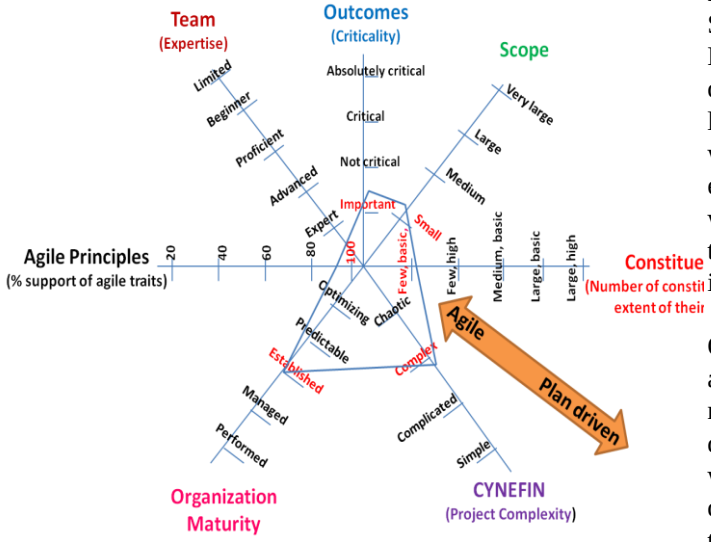


Figure 2: The MAF evaluation of the OPIC project using Radar Chart

B. The Use of Agile SDM in the OPIC Project

OPIC tool is a long term project being developed in phases, module by module. DRC rule cleanup is the first module of OPIC that was implemented successfully tested. DRC checks if the layout satisfies a set of design rules required for manufacturing. If the design rules are violated, the chip may not be functional.

OPIC DRC cleanup tool was implemented using JAVA and it is compatible with major EDA tools, such as Cadence, Mentor Graphics and Synopsys. The tool has been tested to ensure that it can correct DRC errors on a given layout (in GDSII file format) and the results have shown that the tool could reduce the tedious work of DRC checking from 25 hours of manual porting labor to 30 sec as demonstrated in the table below.

Reference Design	# and type of Errors	Manual correction	OPIC DRC Clean-up Tool
BIAS_GENERATOR	56 (7 type violations: Overlap, spacing, width, extension, clearance, etc.)	50 minutes	30 seconds (24 CPU)
Oscillator (7x7 inverters, 0,18nm)	378 (7 type violations: Overlap, spacing, width, extension, clearance, etc.)	240 minutes (no MACRO used, assuming different errors and no compaction problem)	183 seconds (24 CPU)

65nm and 28nm technology nodes were the two platforms used for verification. A reference design including various analog and mixed-signal circuits was built in 65nm technology and OPIC DRC tool was used to determine if the layout satisfied a set of rules required for manufacturing to migrate it to 28nm.

OPIC DRC clean-up is based on a proprietary algorithm (similar to the backtracking/shrinkage algorithm) and it includes 10 categories of common DRC errors that are related to spacing between metals, minimum width, etc. These errors are PO.C.2 (Polysilicon Oxide layer, Clearance), M1.W.1 (Metal Layer – Width), NW.W.1 (N-Well Layer, Width), NW.W.2 (N-Well Layer, Width), NW.S.1 (N-Well Layer, Spacing), NW.S.2 (N-Well Layer, Spacing), M1.S.1 (Metal Layer, Spacing), CO.E.2 (Contact Enclosure), NP.E.1 (N-doped Poly/substrate, Enclosure), PO.O.1 (Polysilicon Oxide layer, Overlap). An engine for correcting each error category was created. The tool recognizes these 10 different DRC errors and based on the error category it selects the engine that will correct the error. Until all the errors are cleaned-up and the DRC is clean, the tool runs iteratively, without human intervention.

OPIC DRC tool was developed by using a combination of agile software development and agile project management methodologies. The project was evaluated against the seven dimensions of the MAF framework and agile methodology was identified as the best suited SDM. As a result, instead of defining the whole requirements, resources, technologies and tools upfront by using a plan-driven methodology like a waterfall, an agile development process was adopted, where

Copyright © AJSE Publishers

the requirements, design and the process emerge in parallel during short, iterative cycles called *sprints.*

As shown in Figure 5 [27], the software development was based on an iterative life cycle and in one iteration cycle, an increment of the software was analyzed, designed, built, tested and integrated on a continuous basis until the full solution was realized [28]. At the end of each iteration, a subset of functionality was delivered to stakeholders for review and the software was built incrementally, piece by piece [29]. This way, the overall risk was minimized and the project could

In the OPIC DRC tool project, the best practices recommended by XP were used for agile software development, such as simple design, pair programming, and small, frequent releases. SCRUM project management framework was used to manage the steps taken to develop software in conjunction with the use of XP to ensure the quality of the software.

This project was led by a project manager, who was responsible for coordinating the work of the cross-functional, self-organizing Scrum team of 5 people and for ensuring that

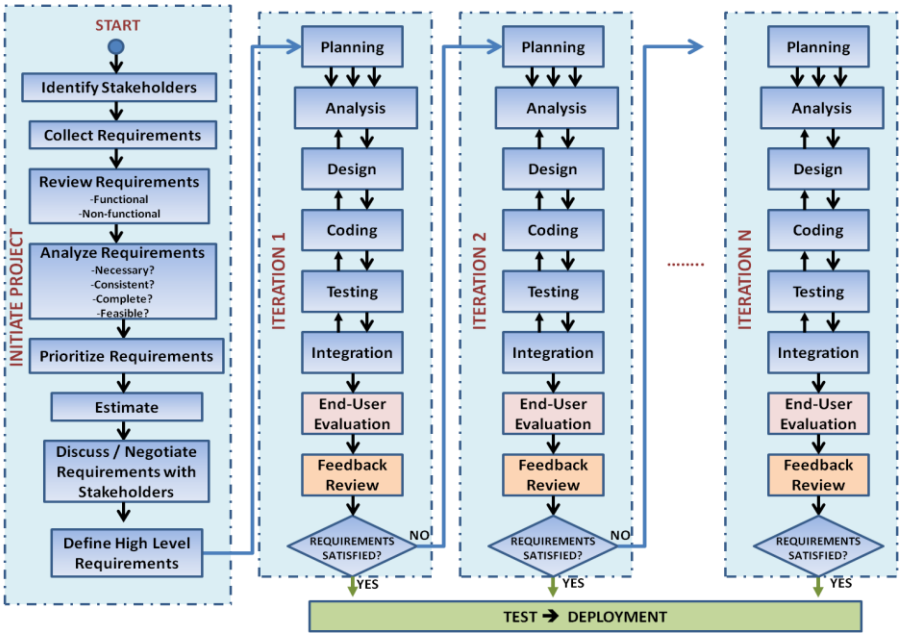


Figure 5: Agile Development methodology used in the OPIC project

adapt to changes quickly.

To be more specific, a hybrid agile approach of eXtreme Programming (XP) and SCRUM (where XP engineering practices are implemented in SCRUM sprints) were chosen and adopted in this project. According to Agile Alliance[2], "XP is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team" while SCRUM is a project management approach.

XP focuses on the developer side of the work with a set of engineering practices it mandates such as test driven development, pair programming, continuous integration, etc. which are very essential for building good quality software. SCRUM focuses on the project management side on what needs to be done. Since SCRUM does not have any engineering practices and XP does not have any management practices, XP and Scrum are well aligned and complement each other.

tasks were completed within the scheduled time. The work was broken down to fit 3-week long sprints. Each sprint includes the traditional phases of software development: requirements analysis, design, coding, testing.

As shown in Figure 6, at the beginning of each sprint, a Sprint Planning meeting was held to plan the work to be done during the given sprint. In this meeting, a Sprint backlog that contains all the user stories (work items) to be implemented during the sprint was created, reviewed and prioritized. The team had daily scrum meetings of maximum 15 minutes to review the progress and productivity, to synchronize the information and to address the blocking issues. Team members worked on the user stories until completed. At the end of each sprint, a Sprint review meeting was held to demonstrate the work that was completed during the sprint and to gather feedback. The Sprint retrospective meeting was held shortly after that to reflect upon the last sprint in order to identify what went well and possibilities for improvement.

[2]A non-profit organization dedicated to promoting agile methodologies at agilealliance.org

Copyright © AJSE Publishers

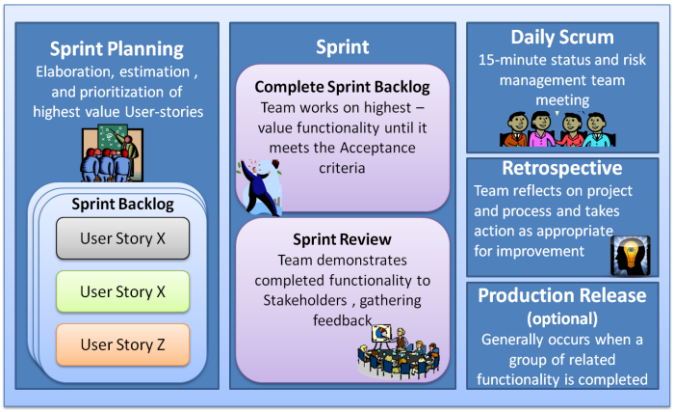


Figure 6: Scrum model that was followed in the OPIC project

Agile methodology was successfully adopted in this project to deliver the OPIC DRC tool in an iterative, incremental approach. Combining two agile approaches XP (for development practices) with Scrum (for project management) proved to be effective. Planning and big-picture view was crucial. Everyone in the team needed to understand what they were working towards for each piece to be able to come together in the end. Establishing an upfront, common understanding of the criteria for done-ness for each user story, for each sprint and for the release was very necessary. Clear and open communication was paramount. Note that agile mindset had to be understood and embraced by everyone in the team it took some time to refine the Scrum model to the point where team members were comfortable and productive.

VI. Conclusion

Due to the complexity of recent advanced technology nodes, no automatic or pseudo-automatic tool has been developed for IP porting yet. This paper has introduced a new EDA tool called OPIC that will automate porting of analog and mixed signal circuits from one technology node to another.

To select the appropriate SDM in order to maximize the chance of project success, the paper also introduced a new framework called MAF and used this framework to assess and evaluate an IC development project by using the OPIC DRC tool project as a case study. As per the MAF framework evaluation, which identified agile methodology as the best suited SDM for the project, the paper also presented a real-life example of adopting a hybrid agile methodology (combination of XP and SCRUM) in the OPIC DRC tool project.

Automated OPIC DRC cleanup tool represents a major step for layout porting. This tool could be used as a standalone product and integrated with major commercially available tools to perform a DRC cleanup. It will allow designers save a considerable amount of time and use this time to focus on precision crafting their designs without sacrificing creativity to repetitive manual tasks.

Once fully implemented, OPIC tool will perform both the DRC and LVS corrections, automatically. So, the next step is to create an LVS tool, which takes a schematic to identify and check IC electrical connectivity against IC schematics and makes corresponding changes on the layout.

The need for increased capacity for more complex chips, speedups and new capabilities is growing and the time-to-market is getting shorter, increasing design productivity by utilizing IP-reuse and EDA tools will continue to play a critical role in semiconductor design and development.

As for the MAF framework, further empirical research is recommended to refine and update the proposed framework and apply it to assess various projects of different size and complexity.

References

1. Birnbaum, M.D., *Essential Electronic Design Automation (EDA)*. Prentice Hall PTR/Pearson Education, 2004.
2. Scheible, J. and J. Lienig. *Automation of analog IC layout: challenges and solutions*. in *Proceedings of the 2015 Symposium on International Symposium on Physical Design*. 2015. ACM.
3. Zhu, J., F. Fang, and Q. Tang, *Calligrapher: a new layout-migration engine for hard intellectual property libraries.* IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2005. **24**(9): p. 1347-1361.
4. ME, M.N.R.K., *DEVELOPMENT OF HIGH PERFORMANCE STANDARD CELL LIBRARY IN UMC180nm TECHNOLOGY.* International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), 2014. **3**(9).
5. Dornelas, H., A. Schmidt, G. Strube, and E. Fabris, *New Technology Migration Methodology for Analog IC Design.*
6. Francken, K. and G. Gielen. *Methodology for analog technology porting including performance tuning*. in *Circuits and Systems, 1999. ISCAS'99. Proceedings of the 1999 IEEE International Symposium on*. 1999. IEEE.
7. MunEDA *Automated Design Migration & IP Porting Flow for Custom Circuits*. 2014. https://www.muneda.com/Applications_Custom-Circuit-Design-Migration-&-IP-Porting.
8. Nojima, T., S. Nakatake, T. Fujimura, K. Okazaki, Y. Kajitani, and N. Ono. *Adaptive porting of analog IPs with reusable conservative properties*. in *IEEE Computer Society Annual Symposium on Emerging VLSI Technologies and Architectures (ISVLSI'06)*. 2006. IEEE.
9. Qian, L., Z. Bi, D. Zhou, and X. Zeng, *Automated technology migration methodology for mixed-signal circuit based on multistart optimization framework.* IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2015. **23**(11): p. 2595-2605.
10. Sobe, U., A. Graupner, E. Böhme, A. Ripp, and M. Pronath. *Analog ip porting by topology conversion and optimization*. in *IP-ESC09 Conference*. 2009.
11. Fu, D.-S., Y.-Z. Chaung, Y.-H. Lin, and Y.-L. Li. *Topology-driven cell layout migration with collinear constraints*. in *Computer Design, 2009. ICCD 2009. IEEE International Conference on*. 2009. IEEE.
12. Kar, P.K. and S.K. Roy. *TECHMIG: A layout tool for technology migration*. in *VLSI Design, 1999. Proceedings. Twelfth International Conference On*. 1999. IEEE.
13. Diodato, P., A. Goksel, F. La Rocca, and W. Troutman. *Porting of a CMOS VLSI Chip from 2.5 Micron to 1.75 Micron Design Rules*. in *Solid-State Circuits Conference, 1984. ESSCIRC'84. Tenth European*. 1984. IEEE.
14. Johnson, N. *Agile hardware development–nonsense or necessity*. EE Times, 2013. https://www.eetimes.com/agile-hardware-development-nonsense-or-necessity/#.
15. Sangiovanni-Vincentelli, A., *The tides of EDA.* IEEE Design & Test of Computers, 2003. **20**(6): p. 59-75.

Copyright © AJSE Publishers

16. Bryant, R.E., K.-T. Cheng, A.B. Kahng, K. Keutzer, W. Maly, R. Newton, L. Pileggi, J.M. Rabaey, and A. Sangiovanni-Vincentelli, *Limitations and challenges of computer-aided design technology for CMOS VLSI.* Proceedings of the IEEE, 2001. **89**(3): p. 341-365.
17. López, R.C., F.V. Fernández, Ó. Guerra-Vinuesa, and Á. Rodríguez-Vázquez, *Reuse-based methodologies and tools in the design of analog and mixed-signal integrated circuits*2007: Springer Science & Business Media.
18. Macintosh, M., *Analog Hard IP made portable.* Chip Design Magazine, 2014.
19. Payne, D. *Three Steps for Custom IC Design Migration and Optimization*. 2016. https://www.semiwiki.com/forum/content/6140-three-steps-custom-ic-design-migration-optimization.html.
20. Chow, T. and D.-B. Cao, *A survey study of critical success factors in agile software projects.* Journal of Systems and Software, 2008. **81**(6): p. 961-971.
21. Misra, S.C., V. Kumar, and U. Kumar, *Identifying some important success factors in adopting agile software development practices.* Journal of Systems and Software, 2009. **82**(11): p. 1869-1890.
22. Mullaly, M. and J.L. Thomas, *Exploring the dynamics of value and fit: Insights from project management.* Project Management Journal, 2009. **40**(1): p. 124-135.
23. Micic, L. *Agile methodology selection criteria: IT start-up case study*. in *IOP Conference Series: Materials Science and Engineering*. 2017. IOP Publishing.
24. Taromirad, M. and R. Ramsin. *Cefam: Comprehensive evaluation framework for agile methodologies*. in *Software Engineering Workshop, 2008. SEW'08. 32nd Annual IEEE*. 2008. IEEE.
25. Griffiths, M., *Agile suitability filters.* Retrieved January, 2013.
26. Alqudah, M.K. and R. Razali, *Key factors for selecting an Agile method: A systematic literature review.* International Journal on Advanced Science, Engineering and Information Technology, 2017. **7**(2): p. 526-537.
27. Dalbudak, E., El Guemhioui Karim, Lakhssassi Ahmed, *On the Use of Agile Methods for IC Design*, in *28th International Conference on Computer Applications in Industry and Engineering (CAINE 2015)*2015: San Diego, California.
28. Highsmith, J., A. *What is Agile Software Development?* CROSSTALK The Journal of Defense Software Engineering, 2002. http://www.adaptivesd.com/articles/cross_oct02.pdf.
29. Fowler, M. and J. Highsmith, *The agile manifesto.* Software Development, 2001. **9**(8): p. 28-35.

Copyright © AJSE Publishers