Design of Mixed-signal Systems with Asynchronous Control

Vladimir Dubikhin†, Danil Sokolov†, Chris J. Myers‡, Alex Yakovlev†
† Newcastle University, Newcastle upon Tyne, NE1 7RU, UK
‡ University of Utah, Salt Lake City, UT 84112, USA

***Abstract* - This paper presents a novel workflow for the design of mixed-signal systems. Current methods rely on synchronous control logic and full-system simulation, which might lead to suboptimal results and even project respins due to critical errors. The proposed workflow aims to combine state-of-the-art tools for asynchronous circuit design and formal verification of analogue systems in a unified environment. The effectiveness of this methodology is demonstrated by the analysis of a buck converter.**

INTRODUCTION

While numerous tools have been developed for automation and verification of digital design, analogue tool development has not kept pace. To cope with this problem, analogue designers have turned to using digital alternatives whenever possible. Existing methods for digital design are based on synchronous circuits, which results in suboptimal solutions for mixed-signal systems [1].

Digital design methods and tools are optimized for synchronous circuits that are governed by a global clock signal. The clocked operation mode, natural for the data processing, might lead to either low responsiveness or power consumption overheads in control modules of mixed-signal systems. On the one hand, the operating frequency must be sufficiently high to promptly react to changes in analogue sensor readings. On the other hand, high clocking frequency can potentially result in wasted clock cycles if the sensors' readings change slowly.

Asynchronous circuits can provide greater robustness, reactivity, and power efficiency. However, due to the lack of necessary computer-aided design tools, engineers have to rely on ad hoc development approaches and use extensive simulation to prove correctness of their designs. Furthermore, not only simulation time considerably increases with system's complexity, but also simulation driven verification is prone to human error and depends on the number and diversity of tests. This may result in longer development times and even the necessity to restart the whole project from the start due to some critical errors found at the final phase. To cope with these problems, a number of methodologies for design and synthesis of asynchronous circuits, as well as formal verification of mixed-signal systems has been developed.

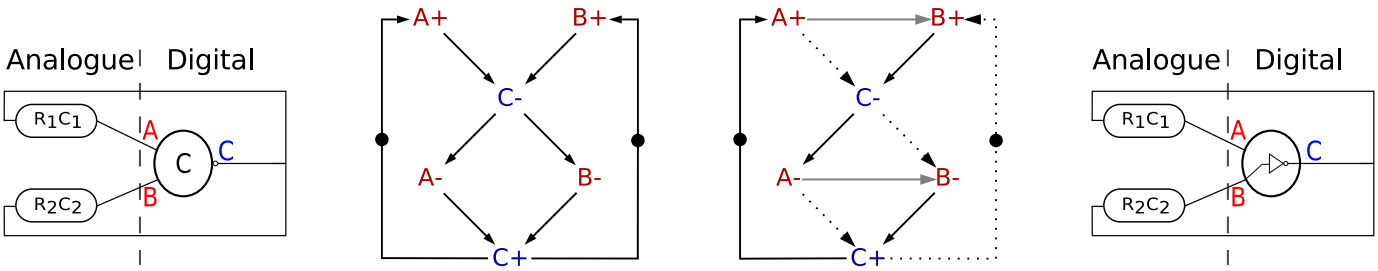
An established method for specification of self-timed circuits in the asynchronous community are *signal transition graphs* (STGs). They provide excellent capabilities for capturing concurrent behaviour of asynchronous circuits, as well as a necessary design notation [2]. An STG is a *Petri net* (PN), in which transitions are labeled with the rising and falling edges of circuit signals. A number of tools, such as PETRIFY[1] or ATACS[2], support verification of STG models and logic synthesis, thus paving the way towards automated design of asynchronous circuits. However, STGs have no means to describe the behaviour of the analogue environment, making full-system verification problematic.

Several approaches, such as hybrid automata and hybrid Petri nets, have been proposed to construct abstract models of mixed-signal systems [3]. These models provide formal verification methods for *analogue/mixed-signal* (AMS) designs, reducing the need for conventional simulation methods and improving robustness of the whole system. One particular example of hybrid Petri nets, *labelled Petri net* (LPN) [4], can specify timing behavior, discrete events, and continuous dynamics. LPNs include continuous variables that can be sampled in an enabling condition and delay assignment labels on the transitions in the LPN. These continuous variables or their rates of change can be modified by transition firings. In addition, conversion between STG and LPN formats can be done in a straight forward manner, thus making possible analysis of asynchronous control under analogue environment with formal methods.

This paper introduces a workflow for the design of mixed-signal systems with asynchronous control. As a motivating example, consider the C-element example shown in Figure 1a. This AMS system consists of a C-element, which feeds its output through an inverter to two RC circuits with different time constants. Without knowledge of the analogue environment, the designer has to use the complete STG specification from Figure 1b. However, using the proposed workflow, it is possible to discover that A changes before B, leading to the updated STG with added timing assumptions (shown as grey arcs) in Figure 1c. These timing assumptions make other arcs obsolete (shown as dashed arcs) introducing the possibility of control simplification. As a result, it is possible to use an inverter instead of a C-element as shown in Figure 1d.

The main goal of this paper is to introduce the novel automated workflow, which enables formal verification of AMS systems with asynchronous control that has been optimized with correct timing assumptions extracted from the full-system model. The workflow has been applied to generate an LPN model of a buck converter with asynchronous control and identify possibilities for control optimization. The resulting optimized module proves to be more area and delay efficient.

[1] http://www.cs.upc.edu/∼jordicf/petrify/
[2] http://www.async.ece.utah.edu/ATACS/



(a) Original system. (b) Original STG. (c) Optimized STG. (d) Optimized system.

Figure 1: C-element example.

Design Workflow

Recent research has addressed existing problems in the design of *analogue/mixed-signal* (AMS) systems. The tool WORKCRAFT[3] aims to provide formal methods for specification and synthesis of asynchronous circuits. Formal verification of AMS circuits is performed by the tool LEMA [5]. The proposed workflow leverages both of these tools to design mixed-signal circuits with asynchronous digital control.

WORKCRAFT

WORKCRAFT is a toolset for editing, simulation, synthesis, and verification of interpreted graph models. The tool provides a cross-platform front-end to established synthesis and verification back-end tools. A plugin-based architecture enables new models and back-end tools to be integrated into the framework.

One of the key features is the ability to create and manipulate *signal transition graphs*. WORKCRAFT provides a convenient mechanism for verification of constructed STGs and subsequent high-level synthesis, using one of its back-end tools, such as, PETRIFY [6] or MPSAT [7].

In [8], this software is used to create an asynchronous controller for a buck converter based on its timing diagram (TD) specification. The resulting circuit is more power efficient, as well as, containing fewer complex gates than an equivalent synchronous design. Moreover, since the input-output latency of the new circuit depends only on the delay of a single gate, the resulting responsiveness is also improved.

LEMA

LEMA[4] is a tool for the modeling, analysis, and verification of AMS circuits [5]. The formal model utilized by LEMA is a *labelled Petri net*. A feature of LEMA is that it can automatically create LPNs from simulation traces [9]. LEMA also includes a property specification language, LAMP, that allows easy expression of model properties, making formal verification accessible to users unfamiliar with formal methods [10]. Finally, verified models can be converted into SystemVerilog for use in digital only simulations.

LEMA has been successfully used to model the behaviour and verify the design of several AMS designs [5], [4], [10]. Formal verification has been shown to discover problems in both the models and circuit designs that are difficult to find in simulation-only methodologies.

[3] http://www.workcraft.org/
[4] http://www.async.ece.utah.edu/LEMA/

Proposed Methodology

WORKCRAFT and LEMA solve two major problems in the design of AMS systems, but there is currently no convenient means to exchange data between them. As a result, the verification process becomes cumbersome and does not provide any information on optimization possibilities for the digital circuit. In order to overcome these shortcomings, a workflow that integrates these two tools, shown in Figure 2, is under development.

Initially, only an informal specification of the AMS system is given. This specification contains high-level information on system behaviour and does not represent internal structure. The first step is system formalization. The digital part is expressed in the STG format, and the analogue environment is implemented as a behavioural or circuit model. Next, the STG is verified and used to synthesize a Verilog netlist, using WORKCRAFT. In order to generate a formal model, the system model must be simulated to produce a number of simulation traces. There are two possible ways to do so:

- **Full-system simulation:** In this approach, the Verilog netlist, obtained from an STG, is combined with the analogue circuit environment model, and the system undergoes transient simulation. Although this is an easy and straightforward method, it has a serious drawback. Namely, the resulting trace might not cover all possible states of the environment, making verification limited.
- **Simulation of individual modules:** In this approach, the analogue part of the design is split into individual modules, which are simulated intensively to provide better state coverage. However, special care has to be taken during system partitioning to limit module complexity, while keeping a sufficient level of detail.

After simulation data is obtained, LEMA is able to generate an LPN model for the analogue portion of the AMS system. The STG can be automatically converted into the LPN format, and it can be merged with the AMS LPN model to create a model of the entire system. This composite LPN model enables the checking of important properties of the design. If formal verification reveals no problems, then the digital control can be optimized by comparing states in the standalone STG and the one used in combination with the analogue

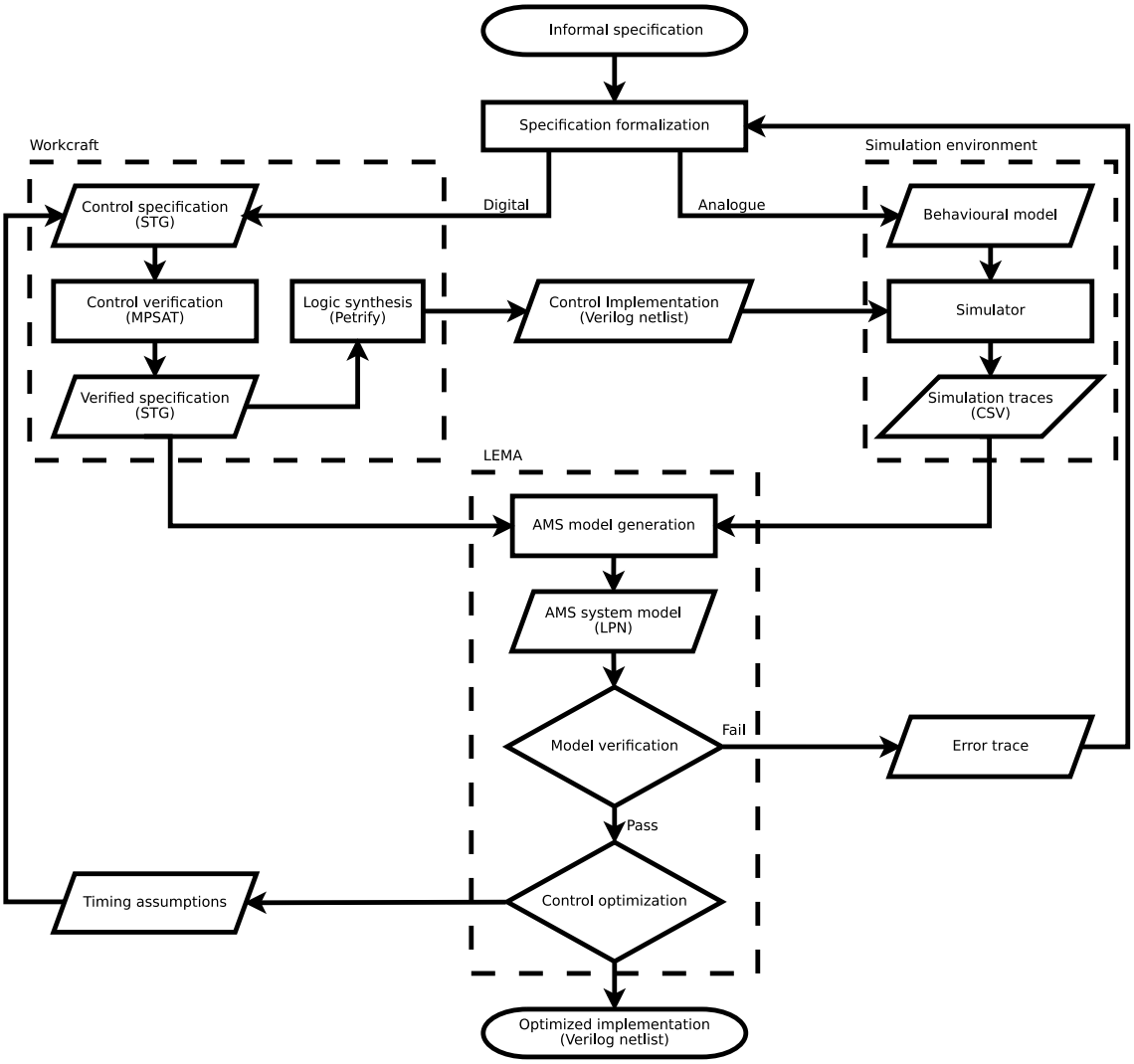


Figure 2: Workcraft and LEMA integrated workflow.

environment. Unreachable states can be removed, reducing the complexity of the control circuit. When no optimizations are possible, the designer can proceed to layout implementation.

The proposed methodology aims to integrate both tools in a unified environment and to provide a joint workflow for the synthesis and verification of AMS systems with asynchronous control.

Buck Control Optimization

The described methodology has been applied to the asynchronous control module of a buck converter. The circuit, synthesized from the STG specification, is used in mixed-signal simulation with an analogue environment. Using the set of generated simulation traces, a full-system model was generated, which shows possibilities for concurrency and scenario optimizations in the original specification.

Buck converter

DC-DC converters are an important part of modern digital circuits and are required to provide a stable power supply over long periods of time. A basic power regulator consists of an analogue block and a digital controller, as shown in the schematic in Figure 3. The controller determines the state of NMOS and PMOS transistors as a reaction to *acknowledgment* (ACK), *undervoltage* (UV), and *over-current* (OC) conditions. These conditions are detected and signaled by a number of special sensors, implemented as comparators in combination with buffering latches.

Initially, specification of the control module is given as timing diagrams with causal relations between signals. Two possible scenarios are considered: stable state when output capacitor charges up to threshold value during one charge cycle (see Figure 4a), and start-up operation mode during which multiple charge cycles are needed to charge the output capacitor (see Figure 4b).

The formal specification is derived from the provided diagrams. The resulting STG, Figure 5, captures behaviour of both scenarios. In addition, special care is taken to incorporate the concurrent nature of the transistors' acknowledgments and over-current signals.

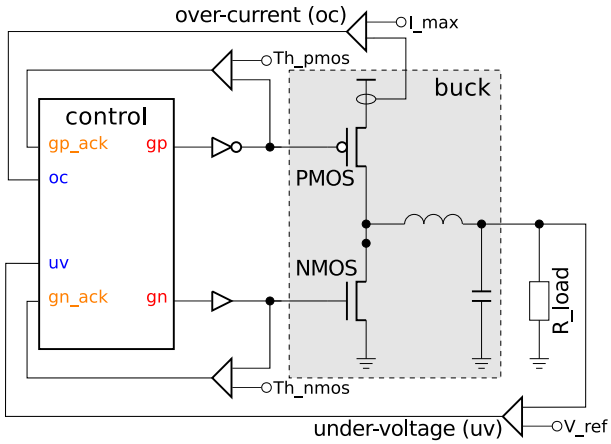
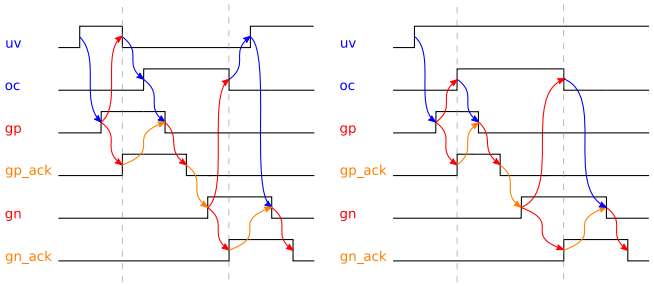


Figure 3: Buck converter schematic.



(a) OC with UV. (b) OC without UV.

Figure 4: Informal specification.

Model generation

The control circuit, synthesized from the STG, is combined with a Verilog-AMS model of the analogue part of buck converter to undergo a series of simulations, using the VIRTUOSO AMS simulation environment. The dynamic resistive load is used to ensure that the system works under different operating conditions.

In order to generate abstract models of the analogue components, causal relations between the digital and the corresponding analogue signals have to be established. There are two possible types of causality that have to be identified:

- **Direct causality**: An analogue signal is directly affected by a digital control signal.
- **Indirect causality**: A digital control signal affects an analogue signal transitively via some intermediate events.

The voltage on the transistors' gates is in direct correlation

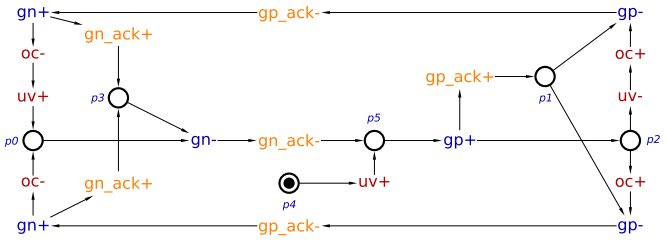


Figure 5: Buck control STG.

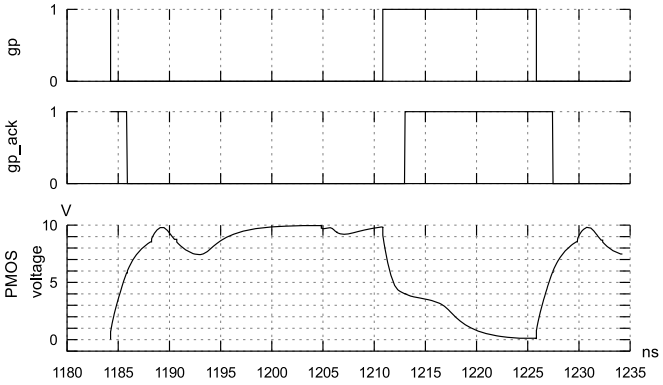


Figure 6: PMOS acknowledgment signals.

with control outputs *gp* and *gn* and determines the state of the corresponding acknowledgment signals *gp_ack* and *gn_ack*, as shown on Figure 6. The process of model generation revolves around determining states with unique variable encodings. Values of continuous variables are assigned to different regions or *bins*, according to the specified thresholds and linearly approximated with ranges of rates. The construction of the LPN is performed by creating transitions with proper variable assignments and guard conditions for input signals. These transitions are linked together in accordance with their evolution in the waveform.

The resulting LPN model shown in Figure 7 captures the presented behaviour for signals of the PMOS transistor. The model decides upon voltage evolution rate, depending on input signal state and current voltage value. Transitions *charging{1,2}* and *discharging{1,2}* represent charging and discharging processes of the gate capacitor with threshold points specified at the change of the acknowledgment signal. For example, the annotation of the *discharging1* transition means that when the *gp* is true and the voltage of the *gp_gate* signal is above or equal to 4V, the change of this voltage must be relatively fast (within the shown range between -0.27V/ns and -0.28V/ns), and then the *discharging2* shows that as soon as the *gp_gate* voltage drops below 4V the rate of discharge becomes slower (within the shown range between -0.03V/ns and -0.048V/ns). Special transitions *corner{1,2}* are essential to prevent voltage from reaching values not present in the original waveform. The model of the acknowledgment signal(upper part of the Figure 7) communicates with the voltage model(lower part of the Figure 7) via guard conditions and assigns output values in accordance with the voltage value. For example, the *discharge 2* transition is synchronized with the *ack_pos* transition and *charge2* with *ack_neg* (cf. corresponding events in the waveforms of Figure 6). This unique feature of LPNs allows one to construct complex models as a set of small Petri nets with implicit communication via guard variables. A model of the acknowledgment signal of the NMOS transistor is derived in a similar manner.

There is, however, no direct dependency (see Figure 8) between digital outputs and the signals responsible for generating UV and OC inputs. The output voltage, as well as current through PMOS transistor, are affected by the inductor current. Thus, an intermediate model of the inductor has to be created

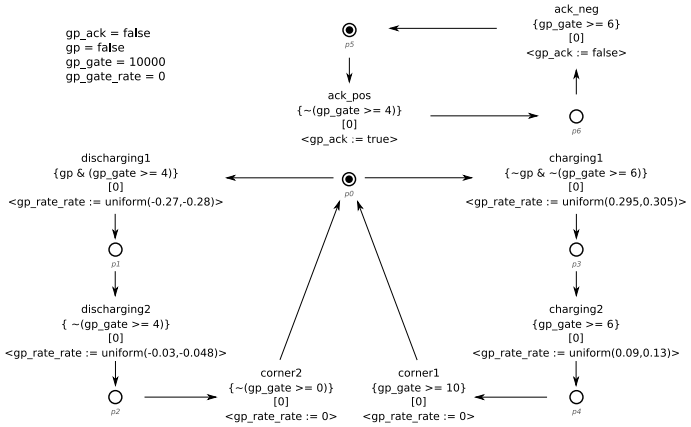


Figure 7: PMOS acknowledgment model.

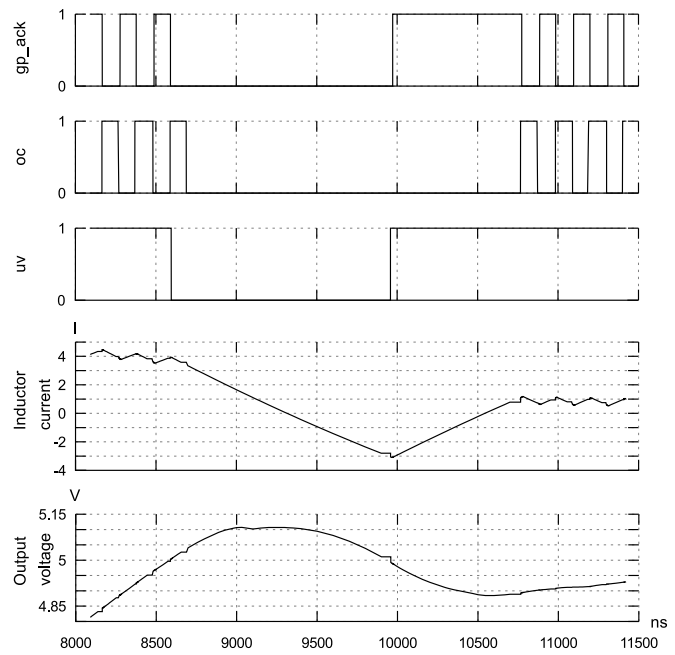
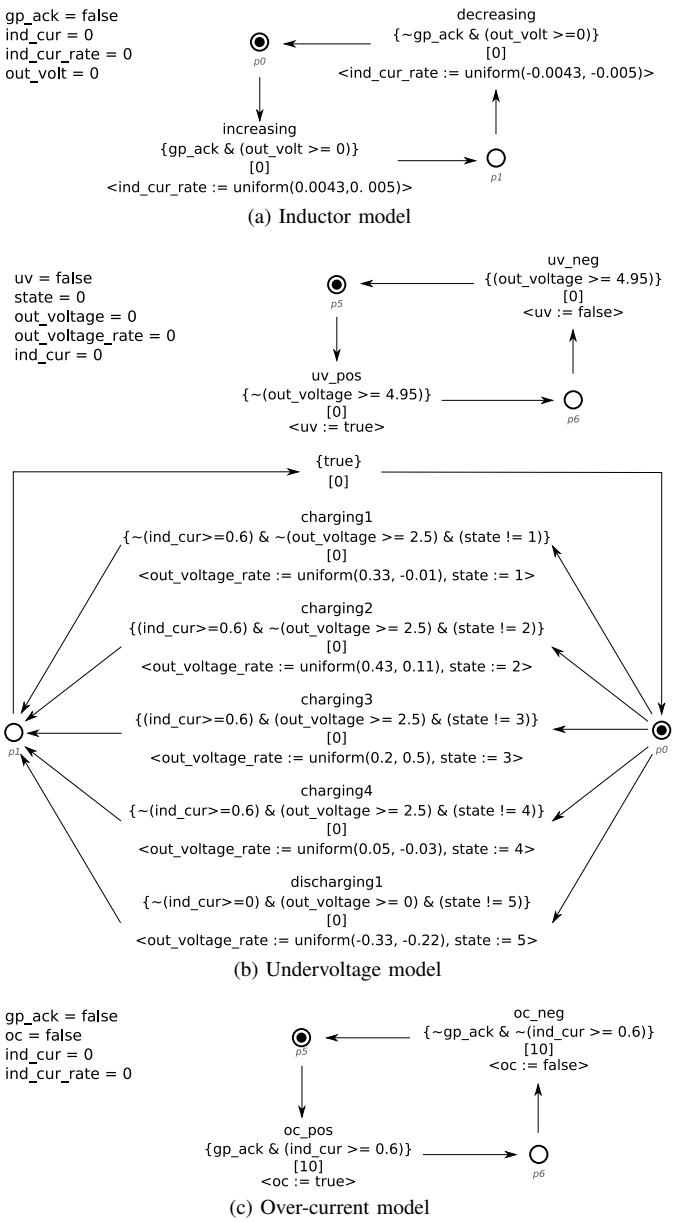


Figure 8: Over-current and undervoltage signals.



(a) Inductor model

(b) Undervoltage model

(c) Over-current model

Figure 9: Over-current and undervoltage models.

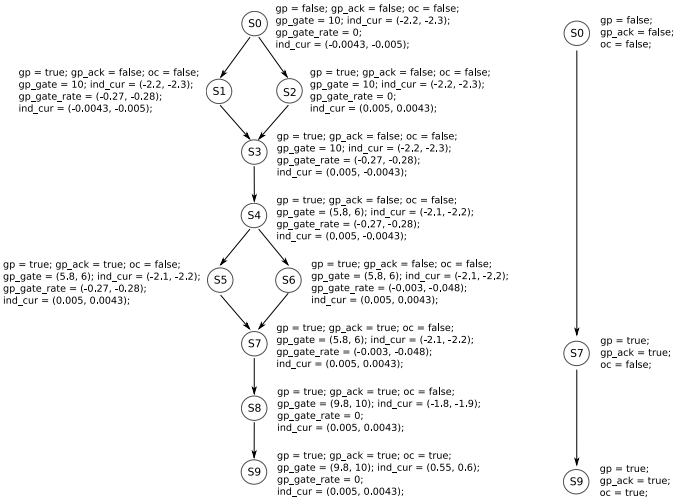
first. The model presented in Figure 9a describes the behavior of the inductor's ripple current. Transitions *increasing* and *decreasing* set the current rate according to the transistor's state. Over-current and undervoltage models are derived in a similar style to the acknowledgment models with inductor current as one of the input signals. The UV model (see Figure 9b) sets output capacitor charge rate, depending on inductor current. A wide range of rates is needed to model dynamic load.

Optimization method

Once the models of the analogue blocks are created, it is possible to obtain a full-system model by directly converting the control STG into LPN format. The resulting model can be used to find possibilities for optimization in the control module.

As a first step, the state graph of the system is reduced via a node contraction algorithm. Connected states with similar vectors of control variables are merged together to reduce the state graph size while maintaining the original graph structure. An example of this reduction process is shown in Figure 10. Initially full-state graph is traversed and states, where digital control signals(*gp*, *gp_ack*, *oc*) do not change in comparison with all of the preset states, are marked. After that these marked places are removed and extra arcs are added to the remaining places to preserve the original graph structure.

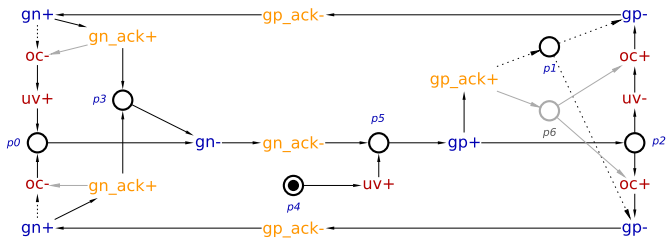
The reduced state graph is later analyzed to determine the timing relations of the input signals. This timing information can be used in the synthesis process with PETRIFY. Additionally, the reduced state graph can be converted into a STG form, which can be used in WORKCRAFT during the established design flow. Although the original structure of the STG may not be preserved, the new version can show greater optimization potential in the form of scenario elimination.



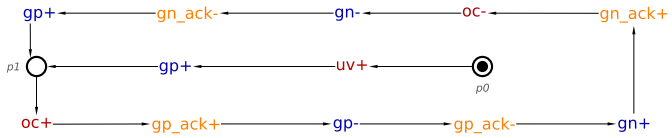
(a) Full state graph.

(b) Reduced state graph.

Figure 10: State graphs.



(a) Concurrency reduction



(b) Scenario elimination

Figure 11: Optimized control models.

Results

The proposed workflow has been applied to the optimization of the buck converter control. The optimization method yields a timing dependency between transistors' acknowledgment and over-current signals, thus reducing all concurrent places in the original model (see Figure 11a). While the achieved area and latency reduction are not considerable, these results are achieved in an automated manner, thus promising greater results for more complex projects.

Additionally, a special environment with a small buck capacitor is used to identify extra optimization capabilities. The small capacitance ensured that only one charge cycle is needed for the output voltage to reach the threshold value. As a result it was possible to eliminate scenario in the original STG (see Figure 11b).

Optimized models are synthesized using PETRIFY and compared against the original model. Results, reported in abstract units, are summarized in the Table I.

Buck control	Total area	Average delay
Original	240	0.83
Reduced concurrency	144	0.58
Removed scenario	112	0.57

Table I: Optimization results.

CONCLUSION AND FUTURE WORK

The design of mixed-signal systems is a complex task, which requires expert knowledge in both analogue and digital domains. With growing system's complexity, as well as decreasing development cycle, designers are required to use new design methodologies. The usage of asynchronous logic and formal verification can greatly enhance the development process of AMS systems, however, necessary tool support is needed in order for these approaches to become an industrial standard.

The described workflow seeks to combine existing methodologies in an automated solution to streamline the design of AMS systems. The development process is ongoing, and it is organized into several key stages:

- **Model generation.** The problem of generalizing observed behaviour and constructing compact models thereof is well-studied in the *process mining* community [11]. Our future work includes the integration of existing process mining theories and software tools in our workflow.
- **Abstract models.** The quality of generated models greatly depends on simulation traces. Alternative methods to obtain a system's models from schematic representation are considered.
- **Optimization method.** Currently, the existing optimization algorithm requires a full state graph of the mixed-signal system, which can lead to excessive time and memory consumption. An improved version would construct an optimized STG on the fly during the verification process.
- **Additional examples.** For the methodology to be accepted, additional examples of AMS systems have to be investigated. Therefore, the primary focus for the future work is the analysis of asynchronous control of multiphase buck converters [12].
- **Control models.** Incorporate additional formal models of digital control module into the workflow to support full-system verification and optimization of synchronous circuits.

ACKNOWLEDGEMENTS

This material is based upon work supported by EPSRC research grant 'A4A: Asynchronous design for Analogue electronics' (EP/L025507/1) and National Science Foundation Grant No. CCF-1117515.

REFERENCES

[1] T. Towers, "Practical design problems in transistor dc/dc converters and dc/ac inverters," *Proc. IEE*, 1959.

[2] J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavango, and A. Yakovlev, *Logic Synthesis of Asynchronous Controllers and Interfaces*. Springer-Verlag Berlin Heidelberg, 2002.

[3] M. H. Zaki, S. Tahar, and G. Bois, "Review: Formal verification of analog and mixed signal designs: A survey," *Microelectron. J.*, vol. 39, no. 12, pp. 1395–1404, Dec. 2008. [Online]. Available: http://dx.doi.org/10.1016/j.mejo.2008.05.013

[4] S. Little, D. Walter, C. Myers, R. Thacker, S. Batchu, and T. Yoneda, "Verification of analog/mixed-signal circuits using labeled hybrid petri nets," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 4, pp. 617–630, April 2011.

[5] A. Fisher, S. Batchu, K. Jones, D. Kulkarni, S. Little, D. Walter, and C. Myers, "Lema: A tool for the formal verification of digitally-intensive analog/mixed-signal circuits," in *Circuits and Systems (MWSCAS), 2014 IEEE 57th International Midwest Symposium on*, Aug 2014, pp. 1017–1020.

[6] J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavagno, E. Pastor, and A. Yakovlev. (1999) Petrify. [Online]. Available: http://www.cs.upc.edu/ jordicf/petrify/

[7] V. Khomenko, "A usable reachability analyser," Newcastle University, Tech. Rep., 2009.

[8] D. Sokolov, A. Mokhov, A. Yakovlev, and D. Lloyd, "Towards asynchronous power management," in *EEE Faible Tension Faible Consommation (FTFC)*, 2014, pp. 1–4.

[9] S. Little, D. Walter, K. Jones, C. Myers, and A. Sen, "Analog/mixed-signal circuit verification using models generated from simulation traces," *The Inernational Journal of Foundations of Computer Science*, vol. 21, no. 2, pp. 191–210, 2010.

[10] A. N. Fisher, D. Kulkarni, and C. J. Myers, "A new assertion property language for analog/mixed-signal circuits," in *Languages, Design Methods, and Tools for electronic System Design: Selected Contributions from FDL 2013*, M.-M. Louërat and T. Maehne, Eds., vol. 311, Oct. 2014.

[11] W. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer-Verlag Berlin Heidelberg, 2011.

[12] D. Sokolov, V. Khomenko, A. Mokhov, A. Yakovlev, and D. Lloyd, "Design and verification of speed-independent multiphase buck controller," in *Asynchronous Circuits and Systems (ASYNC), 2015 21st IEEE International Symposium on*, May 2015, pp. 29–36.

PLACE PHOTO HERE

Vladimir Dubikhin is a Phd student in the School of Electrical and Electronic Engineering, Newcastle University, UK. He received his master degree in Applied Physics and mathematics from Moscow Institute of Physics and Technology. In the period 2011-2014 he has been employed as a software intern in Intel to develop methodology for a microprocessor power and area estimation during early stages of development. His research interests include asynchronous circuit design and formal verification of analogue/mixed signal circuits.

PLACE PHOTO HERE

Danil Sokolov is a Senior Research Associate and Research Fellow in the School of Electrical and Electronic Engineering, Newcastle University, UK. He received PhD from Newcastle University in 2006. In the period 2007-2010 he has been employed as a consultant by an EDA start-up Elastix Corp to lead R&D in the area of ultra low power synthesis. Dr. Sokolov is the author of 10 journal articles, over 30 conference papers and a chapter in a textbook *System On Chip: Next Generation Electronics*. His research expertise is in modelling of self-timed heterogeneous systems, synthesis methods for energy-efficient circuits, development of CAD tools and their integration into industry-level design flows. He is the lead developer of Workcraft – a toolset for capture, simulation, synthesis and verification of interpreted graph models.

PLACE PHOTO HERE

Chris J. Myers received the B.S. degree in Electrical Engineering and Chinese history in 1991 from the California Institute of Technology, Pasadena, CA, and the M.S.E.E. and Ph.D. degrees from Stanford University, Stanford, CA, in 1993 and 1995, respectively. He is a Professor in the Department of Electrical and Computer Engineering, University of Utah, Salt Lake City, UT. Dr. Myers is the author of over 140 technical papers and the textbooks Asynchronous Circuit Design and Engineering Genetic Circuits. He is also a co-inventor on 4 patents. His research interests include asynchronous circuit design, formal verification of analogue/mixed signal circuits and cyber-physical systems, and modeling, analysis, and design of genetic circuits. Dr. Myers received an NSF Fellowship in 1991, an NSF CAREER award in 1996, and best paper awards at the 1999 and 2007 Symposiums on Asynchronous Circuits and Systems. Dr. Myers is a Fellow of the IEEE, and he has served on the Editorial Boards for the IEEE Transactions on VLSI Systems, IEEE Design & Test Magazine, IEEE Life Sciences Letters, and Springer journal on Formal Methods in System Design.

PLACE PHOTO HERE

Alex Yakovlev is a Professor of Computing Systems Design at the School of Electrical and Electronic Engineering. He was a Dream Fellow of the prestigious UK Engineering and Physical Sciences Research Council in 2011-12 to investigate different aspects of energy-modulated computing. He received DSc from Newcastle University in 2006, and PhD from St. Petersburg Electrical Engineering Institute in 1982, both in the field of asynchronous systems. Since 1991 he has been at Newcastle University, where he is currently leading the microSystems research group at the School of Electrical and Electronic Engineering. His main interests and publications are in the field of modelling and design of asynchronous, concurrent, real-time and real-power systems, and autonomous systems for survival. He has published eight authored and edited monographs and more than 300 papers in academic journals and conferences, has managed over 30 research contracts. He has supervised more than 40 PhD students.