Analogue and digital electronics are becoming more intertwined
Analogue domain becomes more complex and itself needs digital control
Motivation: Power electronics context

- Efficient implementation of power converters is paramount
  - Extending the battery life of mobile gadgets
  - Reducing the energy bill for PCs and data centres (5% and 3% of global electricity production respectively)
- Need for responsive and reliable control circuitry - *little digital*
  - Millions of control decisions per second for years
  - An incorrect decision may permanently damage the circuit
- Poor EDA support
  - Synthesis is optimised for data processing - *big digital*
  - *Ad hoc* solutions are prone to errors and cannot be verified
In the textbook buck a diode is used instead of NMOS transistor
Basic buck converter: Informal specification

- **no ZC** – under-voltage without zero-crossing
- **late ZC** – under-voltage before zero-crossing
- **early ZC** – under-voltage after zero-crossing
Multiphase buck converter: Informal specification

- Normal mode
  - Phases are activated sequentially
  - Phases may overlap

- High-load mode
  - All phases are activated simultaneously

- Benefits
  - Faster reaction to the power demand
  - Heat dissipation from a larger area
  - Decreased ripple of the output voltage
  - Smaller transistors and coils
Synchronous design

- Two clocks: phase activation (~5MHz) and sampling (~100MHz)
  - Easy to design (RTL synthesis flow)
  - Response time is of the order of clock period
  - Power consumed even when idle
  - Non-negligible probability of a synchronisation failure

- Manual ad hoc design to alleviate the disadvantages
  - Verification by exhaustive simulation
Asynchronous design

- Event-driven control decisions
  - 🌅 Prompt response (a delay of few gates)
  - ☀️ No dynamic power consumption when the buck is inactive
  - 😞 Other well known advantages
  - 🤷‍♂️ Insufficient methodology and tool support

- Our goals
  - Formal specification of power control behaviour
  - Reuse of existing synthesis methods
  - Formal verification of the obtained circuits
  - Demonstrate new advantages for power regulation (power efficiency, smaller coils, ripple and transient response)
No need for phase activation clock
A2A components

- Interface analogue world of “dirty” signals
- Provide hazard-free “sanitised” digital signals for asynchronous control

Library of A2A components

- **WAIT / WAIT0** – wait for stable 1 / 0 on analogue input and latch it until explicit release signal
- **RWAIT / RWAIT0** – WAIT element with a possibility to persistently cancel the waiting request
- **WAIT01 / WAIT10** – wait for a rising / falling edge
- **WAITX / WAITX2** – wait for one of analogue signals in a mutually exclusive way using 4-phase / 2-phase control signalling
- **SAMPLE** – sample the state of analogue signal
A2A components: WAIT element

- STG specification

Read-consume conflict between output and input.

- ME-based solution

- Gate-level implementation

sig+ \rightarrow san+ \rightarrow ctrl-
sig- \rightarrow ctrl+ \rightarrow san-

sig0 \rightarrow sig1

r1 \rightarrow g1

r2 \rightarrow g2

ME

can be removed
Synthesis flow

- Manual decomposition of the system into modules
  - To create formal specification from informal requirements (feedback loop with engineers)
  - To simplify specification and synthesis
  - Some modules are reusable
  - Some modules (A2A components, Opportunistic Merge) are potential standard components

- Each component is specified using STGs
- Automatic synthesis into speed-independent circuits (arbitrary gate delays and some forks must be isochronic)
Formal verification

- STG verification
  - All standard speed-independence properties (consistency, output-persistency, complete state coding)
  - PMOS and NMOS are never ON simultaneously (to prevent from short circuit)
  - Some timers are used in a mutually exclusive way and can be shared

- Circuit verification
  - Conforms to the environment
  - Deadlock-free and hazard-free under the given environment
Tool support: WORKCRAFT

- Framework for *interpreted graph models* (STGs, circuits, FSMs, dataflow structures, etc.)
  - Interoperability between models
  - Elaborated GUI

- Includes many backend tools
  - PETRIFY – STG and circuit synthesis, BDD-based
  - PUNF – STG unfolder
  - MPSAT – unfolding-based verification and synthesis
  - PCOMP – parallel composition of STGs
Tool support: WORKCRAFT

Message:
Under the given environment (stg-ZCH.work) the circuit is:
- conformant
- deadlock-free
- hazard-free
Simulation results

- Verilog-A model of the 3-phase buck
- Control implemented in TSMC 90nm
- AMS simulation in CADENCE NC-VERILOG
- Synchronous design
  - Phase activation clock – 5 MHz
  - Clocked FSM-based control – 100 MHz
  - Sampling and synchronisation

- Asynchronous design
  - Phase activation - token ring with 200 ns timer (= 5 MHz)
  - Event-driven control (input-output mode)
  - Waiting rather than sampling (A2A components)
AMS Trends & Challenges

- **Key drivers**
  - Internet of Things
  - Mobile computing
  - Automotive electronics

- **Trends**
  - Technology scaling
  - Multiple power and time domains
  - Analog and digital integration

- **Challenges**
  - Tighter reliability margins
  - Concurrent analog and digital analysis
  - Short development cycle

Based on slide from DAC-2014 by ANSYS
What this means for AMS?

- Achieving better verification of analog and digital blocks
- Verifying the increasing amount of digital logic in analog designs
- Creating a higher level of abstraction for analog and mixed signal blocks
- Automating the manual custom design steps
- Adopting circuit analytics that tell why and where the circuit is failing to perform

Based on slide from ISQED-2013 by Mentor Graphics
AMS flow: Tool support

- **WORKCRAFT** – synthesis and verification of async. circuits
- **LEMA** – modelling and verification of AMS circuits
  - Model generation from simulation traces
  - Property expression and checking
AMS flow: Basic buck

- over-current (oc)
- under-voltage (uv)
- $I_{\text{max}}$
- control
  - gp_ack
  - gp
  - oc
  - uv
  - gn_ack
  - gn
- buck
  - PMOS
  - NMOS
- $R_{\text{load}}$
- $V_{\text{ref}}$

Events:
- gp_ack
- oc
- uv
- gn_ack
- gn
- over-current (oc)
- under-voltage (uv)

Nodes:
- gp_ack+
- gp_ack-
- gp+
- gp-
- oc+
- oc-
- uv+
- uv-
- gn+
- gn-
- gn_ack+
- gn_ack-
- p0
- p1
- p2
- p3
- p4
- p5
AMS flow: Model generation example

Initial conditions:
- gp_ack = false
- gp = false
- gp_gate = 10000
- gp_gate_rate = 0
AMS flow: Optimised specification

- Concurrency reduction

- Scenario elimination
ADC: Sampling schemes

- Synchronous

- Asynchronous

ADC: Asynchronous design

![Diagram of Asynchronous Design]

- **Slope Detector**
- **Ramp Gen**
- **Comp**
- **Async controller**
- **Time to Digital Converter**

**Waveforms**:
- $V_{in}$
- $V_{ramp}$
- $V_{change}$
- $V_{pulse}$

**Timings**:
- $t_1$
- $t_2$
- $t_3$
- $t_4$
- $t_5$
- $t_6$

Terms:
- rise
- fall
- $refp$
- $refm$
- $refn$
ADC: Specification and implementation

- STG specification
- Speed-independent implementation
Conclusions

- Fully asynchronous design of multiphase buck controller
  - Quick response time: few gate delays, all mutexes are outside the critical path
  - Reliable: no synchronisation failures

- Design flow is automated to large extent
  - Automatic logic synthesis
  - Formal verification at the STG and circuit levels
  - Library of A2A components

- Vision for an advanced AMS design flow
Acknowledgements

- A4A team
  - Newcastle University: Vladimir Dubikhin, Victor Khomenko, Andrey Mokhov, Danil Sokolov, Prof. Alex Yakovlev
  - Dialog Semiconductor: David Lloyd
  - University of Utah: Prof. Chris Myers

- EPSRC for funding A4A project (EP/L025507/1)
- Design automation
  - WORKCRAFT – http://workcraft.org/
  - LEMA – http://www.async.ece.utah.edu/LEMA

- Recent publications
  - D. Lloyd, R. Illman: “Scan insertion and ATPG for C-gate based asynchronous designs”, SNUG- 2014.