Bridging Asynchronous Circuits and Mixed-Signal Design

Alex Yakovlev
School of Engineering
Newcastle University, UK
Contact: Alex.Yakovlev@ncl.ac.uk
async.org.uk; workcraft.org

Nano-Network 2018, Tjome, 13th June 2018
• What are Asynchronous (digital) Circuits?
• Why are Async circuits needed?
  – Tolerant to process variations
  – Tolerant to environmental variations
• Success and Obstacles with Async Design
• “Little digital”: Async design for Analog electronics
  – Design of Power converters
  – Tool Workcraft.org
• Messages and Open challenges
Asynchronous Behaviour

• Synchronous vs Asynchronous behaviour in general terms, examples:
  – Orchestra playing with vs without a conductor
  – Party of people having a set menu vs a la carte
• Synchronous means all parts of the system acting globally in tact, even if some or all part ‘do nothing’
• Asynchronous means parts of the system act on demand rather than on global clock tick
• Acting in computation and communication is, generally, changing the system state
• Synchrony and Asynchrony can be in found in CPUs, Memory, Communications, SoCs, NoCs etc.
Synchronous clocking

**How we think**

- $R_1$ → $R_2$ → $CL_3$ → $R_3$ → $CL_4$ → $R_4$

**What we design**

- Clock signal
- Clock gate signal

- $R_1$ → $R_2$ → $CL_3$ → $R_3$ → $CL_4$ → $R_4$
Asynchronous handshaking

What we design

How we think

"Channel" or "Link"

Handshake latch

Handshake CL
Tolerance to
Process variations
Performance/energy/yield trade off

From Asenov, UKDF’10
Asynchronous to cope with uncertainty

Delay insensitive

Timing robustness

Gate delay (ps)

Vdd (V)

Timing assumptions

Clocked

Technology node: 90nm

Source of variability analysis:
Yu Cao, Clark, L.T., 2007
Async for energy efficiency

Asynchronous (self-timed) logic can provide completion detection and thus reduce the interval of leakage to minimum, thereby doing nothing well!

Source: Akgun et al, ASYNC’10
Tolerance to Environmental conditions
Piezo-experiment with asynchronous counter

- Vibration
- Piezoelectric Transducer
- (AC)
- 6-bit Self-timed Counter
- Full-bridge Rectifier
- Vdd (DC)

(A)

(B)
Traditional system
Energy-modulated system

![Diagram of an energy-modulated system](image-url)
Async (digital) behaviour

• Triggered by events (e.g. level-crossing)
• Modelled by
  – cause-effect relations
  – token flow
  – handshakes
  – data-flow
• Power-driven timing
• Applications: interfacing, control, pipeline
Example 1: speed-independent SRAM

Self-timed SRAM chip: UMC CMOS 90nm

Low Vdd – slow response
High Vdd – fast response
SRAM testing and results

- SRAM operations modulated by Vdd from a Capacitor Bank
- When Vdd goes below 0.75v, the ack signal is not generated by SRAM
- The circuit automatically wakes up when Vdd goes up
Example 2: Power-proportional CPU design (8051)

- CMOS 130nm CMP process, 2013
- 0.89V to 1.5V: full capability mode
- 0.74V to 0.89V: at 0.89V the RAM starts to fail, so the chip operates using
  - 0.22V to 0.74V: at 0.74V the program counter starts to fail, however the control logic synthesised using the CPOG model continues to operate correctly down to 0.22V
  - 67 MIPS at 1.2 V
  - ~2700 instructions per second at 0.25V
So, if Asynchronous Design is so cool, why does mainstream industry not use it!?
Mainstream industry is Big Digital and has an “established Design Process” (with commercial tools for globally clocked designs)

$$$

Design Cost is the main factor! TOOLS are a key!
It’s hard to beat Synchronous Design for BIG Digital!

What about Analog-Mixed-Signal?
Analog and Mixed Signal (AMS) design becomes more complex:

- More functionality
- Move to deep submicron after all!

- According to Andrew Talbot from Intel (2016) “transistors are very fast switches, netlists are huge, parasitics are phenomenally difficult to estimate, passives don’t follow Moore’s law, reliability is a brand new landscape”
Emergence of little digital electronics

- Analog and digital electronics are becoming more intertwined
- Analog domain becomes complex and needs digital control
Example: Buck (DC-DC) converter control
Example: Buck converter

Building asynchronous circuits in Analog-Mixed Signal context requires extending traditional assumptions about speed-independence ...

Phase diagram specification:

Buck conditions:
- under-voltage (UV)
- over-current (OC)
- zero-crossing (ZC)

Operating modes:
- no zero-crossing
- late zero-crossing
- early zero-crossing
Starting point – a New Behaviour Capture Language:

Signal Transition Graph

similar to

Traditional Waveforms

so easy to grasp by industry!
Design Case Study:
Multiphase Buck Converter
Sync vs Async
Multiphase Buck: Sync Control

- Two clocks: phase activation (slow) and sampling (fast)
- Need for multiple synchronizers (grey boxes) - latency & metastability
- Conventional RTL design flow
Multiphase Buck: Async Control

- Token ring architecture, no need for phase activation clock
- No need for synchronisers - all signals are asynchronous
- A4A design flow

D. Sokolov, et al. DATE 2017
Simulation results: Comparison

- 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)
Simulation results

synchronous @ 333MHz

asynchronous
### Reaction time

<table>
<thead>
<tr>
<th>Buck controller</th>
<th>HL (ns)</th>
<th>UV (ns)</th>
<th>OV (ns)</th>
<th>OC (ns)</th>
<th>ZC (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYNC @ 100MHz</td>
<td>25.00</td>
<td>25.00</td>
<td>25.00</td>
<td>25.00</td>
<td>25.00</td>
</tr>
<tr>
<td>SYNC @ 333MHz</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
</tr>
<tr>
<td>SYNC @ 666MHz</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
</tr>
<tr>
<td>SYNC @ 1GHz</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
</tr>
<tr>
<td>ASYNC</td>
<td>1.87</td>
<td>1.02</td>
<td>1.18</td>
<td>0.75</td>
<td>0.31</td>
</tr>
<tr>
<td>Improvement over 333MHz</td>
<td>4x</td>
<td>7x</td>
<td>6x</td>
<td>10x</td>
<td>24x</td>
</tr>
</tbody>
</table>

Synchronous buck controllers exhibit latency of 2.5 clock cycles.
Peak current

The graph shows the peak inductor current (mA) plotted against coil inductance (μH) for different frequencies: 100MHz, 333MHz, 666MHz, and 1GHz. The ASYNC line is also included. The peak current decreases as the coil inductance increases for all frequencies tested.
Inductor losses

![Graph showing inductor losses vs coil inductance for different frequencies.](image-url)
Async Design Tools:

Workcraft.org
Logic synthesis and formal verification of asynchronous circuits.

The tool is now used in education, research and industry. Open source and well supported by tutorials.
Messages to take away (for IC design research)

• Asynchronous circuits began their life (in the 50s) for ‘little digital’ and today is the right time for them
• Analog and mixed-signal is a good application – it combines:
  – Need for low latency and high range of feedback types
  – Non-traditional benefits (inductor size, peak current) can be gained
• Productivity in industry is a good drive – and we have tools accepted by industry
• Interesting research problems are there – tech mapping, holistic analog-mixed signal verification, behavioural mining, dealing with complexity
• Think about power and timing and their relationship
• Thanks to Asynchronous logic and Power-modulated computing we can:
  • Tolerate imperfections of technology (e.g. Variability), so don’t push yourself in optimising devices
  • Tolerate environmental variations (e.g. Power doesn’t need to be perfectly regulated)
• This is a good way towards building trillions of devices without batteries
THANK YOU!