



Asynchronous Design Research or Building Little Clockless Universes

Alex Yakovlev

uSystems Group

School of Engineering, Newcastle University

https://www.ncl.ac.uk/engineering/research/eee/microsystems/ http://async.org.uk/

RAEng Event 28 January 2019

Scientific Curiosity Practical Needs Engineering Challenges

My research is about ...

... designing microelectronic systems.

I am not developing drugs to cure cancer I am not constructing new bridges I am not designing spacecrafts I am not building electric cars Etc.

So what am I doing?

Building "little universes" ...

... where life can be measured in 10^15 or more time units!

- Well to be precise, I am helping to build such little universes in semiconductor technologies.
- Imagine their complexity and how to make sure that they do what we want them to !?

So to paint this picture let's look at building blocks and numbers!





nanoseconds







microseconds



milliseconds

) 🔿 🔿 blink.c	
**************************************	comments area
lefine F_CPU 1000000 nclude ⊲avr/io.h> nclude ⊲util/delay.h> nclude "/leah_library/pin_macros.h"	setup area 1
t main (void) b00utput(); for(;;)	setup area 2
<pre>{</pre>	main action happens here



seconds to days

1/28/2019

Newcastle

Life span of the little universes

Transistors (basic switches) - picoseconds (10^-12) Logic gates (AND, OR, NOT) - 10ps (10^-11) Logic circuits (Compare, Decode) – 100ps (10⁻¹⁰) Arithmetic circuits (Adder/Shifter) – nanoseconds (10^-9) Processors (arithmetic circuits and registers), Memory, Communication links - 10ns (10⁻⁸) Instructions (ADD, MOVE, MULT, JUMP - 100ns (10^-8) Programs (Sort, Search) – microseconds (10^-6) Software tasks (input track, compute PID control, display, manage battery) – milliseconds (10^-3) Applications (mail, phone, video, sensing, ECG monitor) seconds to days (1 to 10⁺⁶)

Total life span is about 10^18 transistor switchings!

How do we manage?

It's all about design process:

- understanding of technology
- discovering and using the laws of physics, electronics and computing
- mathematical modelling
- sometimes inventing new mathematics
- developing design tools
- developing testing and validation methods
- sharing engineers expertise
- ... and constant and continuous innovation

Curiosity ...

In my research I ask many questions:

- how to design logic circuits that don't glitch (hazard-free)
- how to design a frequency divider with the same response time regardless of the ratio?
- how to construct a logic circuit from its intended behaviour in a given set of constraints (time, power, energy, area) ?
- how to reach the minimum energy per operation point for a data processing pipeline?
- what is the decay law of a capacitor discharge via digital circuit?
- how to reduce the probability of synchronization failure?
- how to mask power dissipation that reveals data values?

1/28/**2012**

Practicality ...

We should make systems:

- Autonomous (surviving and adapting)
- Energy frugal (long battery life, low heat)
- Low cost in design (automation) and maintenance (battery-free)
- Smaller size (capacitors and inductors)
- More reliable (longer time between errors)
- More secure (hard to attack by hackers)

Scientific Curiosity ┿ **Practical Needs Engineering Challenges**

Swarm of devices – Future of ICT



Trillions of ubiquitous systems (sensors, probes, monitors, actuators, controllers) are being deployed to operate in myriad of places (organisation, human, body part, household, offices, pets) using harvested energy or micro-batteries



Timing is a key issue everywhere ...

- To deal with complexity
- To deal with performance
- To deal with reliability
- To deal with resource optimisation (e.g. energy)

Synchronous vs Asynchronous circuits

What is it all about?

Synchronous circuit



Synchronous circuit



AYakovlev - RAEng event - Jan 2019 -Newcastle Drop the (global) clock and use Handshaking

Handshaking – a way to asynchronous



Variations of asynchronous

- Aperiodic
- Self-timed
- Self-synchronized
- Elastic
- Globally async locally sync (GALS)
- And more mathematical names
- Fundamental mode FSMs
- Speed-independent
- Delay-insensitive
- Quasi-delay-insensitive

Asynchronous design research glimpses ...

Inventing new models for logic circuits: Signal Transition Graphs (1980-1983)

Logic circuit



Timing Diagram



STG model



AYakovlev - RAEng event - Jan 2019 -Newcastle

Early days of using STG model for circuit design (1984-1989)



VME instrumentation interface controller

Figure 8.

Decomposition of SLAVE controller: second step (refining SYNCHRO)





Figure 11.

Implementation of SYNCLOGIC: normal signal graph (a), deriving logical functions (b)



1/28/2019

AYakovlev - RAEng event - Jan 2019 -Newcastle

Fault-Tolerant Computing: Self-Timed Ring (1981-1986)

For an on-board airborne computer-control system which tolerates up to two faults. Self-timed ring provided **self-checking and selfrepair at the hardware level**



Glitch-free circuit design (1994-2000)



Newcastle

112010

Secure smart card design: Self-timed dual rail circuits (2003-2008)

เงษพบสรเษ



Single Rail power trace

Key property of dual-rail alternating spacer logic: For every operational cycle of the circuit all logic nodes fire => maximum power balancing



1/28/2019

Async-for-Analog design automation: Workcraft toolkit (2010-now)

Asynchronous Power Management ICs



Energy-modulated computing (EPSRC Dream f'ship 2011 - now)



Key Challenges

- Theory and design tools for wide-band powered electronic systems
- Introducing non-volatility (seamless stateretention) into electronics systems for on-chip power management and survival
- Working at near-noise levels (e.g. for CMOS: <=50mV); scavenging from inner and external noise
- Design automation for mixed-signal electronics (avoiding many hours of simulations)

Power efficiency and regularity

- Modern systems rely on highly regular (periodic) power sources they "invest" some power into power regulation
- Future systems will have to operate in a wide dynamic range, paying the price in efficiency in a particular band



Vision for Future

Power-modulated multi-layer system

- Multiple layers of the system design can turn on at different power levels (analogies with living organisms' nervous systems or underwater life, layers of different cost labour in resilient economies)
- As power goes higher new layers turn on, while the lower layers ("back up") remain active
- The more active layers the system has the more power resourceful it is



Newcastle Async Chip Designs











- Arbitration: HADIC1 (2000, CMOS 0.6um) asynchronous arbiters, ADC, Async communication mechanism
- Time measurement: TMC (2003, CMOS 0.2um, collab. with Sun) time amplifier and time-to-digital converter
- Synchronization: SYRINGE1 (2006, UMC 0.18um), SYRINGE2 (2008, UMC, 0.13um) metastability measurement, robust synchronizer
- Security: SCREEN1 (2005, AMS 0.35um), SCREEN2 (2006, AMS 0.18um) AES cores with power balancing; SURE (2008, UMC 0.13um), Galois Encoded Logic, Script1 &2 (2010, TSMC 90nm)
- Networks on Chip: NEGUS1 (2007, UMC 0.13um), phaseencoding signalling, NeuroNOC (2009, TSMC 90nm) reconfigurable neural network, 3D-chip (2011, MIT Lincoln Labs, 150nm with TSVs)
- Energy harvesting: Holistic1 (2011, UMC 90nm), self-timed SRAM, Holistic 2 (2012, UMC 0.18um), reference free voltage sensor
- Power proportionality: Async 8051 (2013, STM CMP 0.13um)
- SAVVIE and A4A chips: to be reported











Examples of Impact: last REF

Quoting one of Intel's Chief Scientists (2013), "the tool Petrify and STGs have become a fantastic means of teaching and explaining asynchronous design concepts which is essential to enabling asynchronous circuits to have wide-spread adoption".

Some of the asynchronous circuits designed with these tools were used by Intel in building their Switch fabric chips for wide deployment in the Financial Services industry to facilitate transactions on the NYSE and NASDAQ.

http://www.ncl.ac.uk/research/impact/casestudies/powe ringindustry/#discovermore

Examples of impact tracking

"Newcastle's work on Asynchronous Synthesis Tools has been of great benefit to Dialog Semiconductor (http://www.dialogsemiconductor.com). We've have been using his tools since 2012. They have allowed us to automate some key stages of our design flow. We are now using his tools to produce one of our key products. This product will ship in extremely large volumes."

From Dialog Semiconductor, March 2015

"In the last year Newcastle design method and tools have been increasingly used by the company's engineers, which has significantly boosted the design productivity and quality of the integrated circuits that were put into final products of the company. Dialog Semiconductor continues to further advance its technology on the basis of the constantly improving design tools coming from Professor Yakovlev's team and ship its products in large volumes."

From Dialog Semiconductor, March 2016

People I am working with at Newcastle University

• Microsystems Group at Engineering:

Dr Alex Bystrov, Dr Patrick Degenaar, Dr Andrey Mokhov, Dr Reza Ramezani, Dr Rishad Shafik, Dr Delong Shang (now Manchester), Dr Danil Sokolov, Dr Ahmed Soltan, Dr Fei Xia and many former and current PhD students

• Computing:

Dr Victor Khomenko, Prof Maciej Koutny, Dr Marta Koutny, Prof. Brian Randell, Prof Alexander Romanovsky

External collaborations

- UK: Southampton, Manchester, Bristol, Imperial, Cambridge ...
- International: Barcelona, Turin, Utah, Southern California, Berkeley, Grenoble, Oslo, Tokyo ...
- Industrial: Intel, Sun/Oracle, Cadence, MBDA, Atmel, ARM, Dialog, SMEs: Elastix, Tiempo, Temporal Computing
- R&D: IMEC (Belgium), CEA/LETI (France), IHP (Germany) ...

Research KPIs

- 500+ research papers and 8 monographs
- Lectures, Tutorials, New conferences and workshops
- 50+ PhD graduated students
- 30+ Research grants (EPSRC, EU, Royal Society, British Council, Leverhulme, Industry), total funding: over £15M
- Design flows and tools: STGs, Petrify, Workcraft (used in academia and industry)
- Industrial impact examples: Dialog, Intel, Atmel (power converters for mobile devices, processors for NASDAQ servers, smartcards)

Career steps

- 1979: MSc Computer Engineering (St. Petersburg Elect. Eng. Inst.)
- 1982: PhD Computer Science (ibid)
- 1982/1987: Assistant Prof/Associate Prof (ibid)
- 1984-85: Postdoc Fellow of British Council (Newcastle)
- 1990-91: Senior Lecturer (Uni. of Glamorgan)
- 1991: Lecturer at CS (Newcastle)
- 1997: Reader head of VLSI group (CS)
- 2000: Chair co-founder of Async Systems Lab (ASL)
- 2002: moved to EEEng head of Microsystems Design group
- 2006: DSc Engineering (Newcastle)
- 2011/13: EPSRC Dream Fellow
- 2015/16/17: FIET/FIEEE/FREng

Tribute to my teachers and mentors

Victor Varshavsky (1933-2005): pioneer of asynchronous design



Leonid Rosenblum: my mentor and co-inventor of STGs



Leonid Rosenblum introduced Petri nets into asynchronous circuit modelling and popularized Petri nets in USSR in the 70s and 80s 1/28/2019 David Kinniment (1940-2012): pioneer of metastability and synchronisation research



Thank you!

More information:

https://www.ncl.ac.uk/engineering/research/eee/microsystems/

http://async.org.uk/

http://workcraft.org

https://blogs.ncl.ac.uk/alexyakovlev/