JAMES DUNCAN LYALL

Email | jim.lyall@gmail.com
Address | 515-1888 Victoria Diversion, Vancouver BC
Nationality | British (Canadian PR)
Telephone | +1 604 754 2653 | Canada, V5N 0C4

CORE TECHNICAL SKILLS

Languages C/embedded C, Java (J2SE, JNI, JUnit), Python, C++, MATLAB

Engineering SW design – System Architecture, OOD, Design Patterns, UML
Engineering maths/algorithm development
UI design, virtualisation/containerisation
Digital systems/electronics debugging/hardware drivers

EMPLOYMENT

July 2018 – Present **DMGblockchain, Vancouver BC**
Lead Software Engineer

- Built software team
- Prototyped containerised AWS data collection pipeline
- Released feature rich bitcoin mine management product

Sept 2014 – Jan 2018 **Broadcom Canada Ltd, Richmond BC**
Principal Development Software Engineer

- Video expert on Cisco 8845/65 videophone project
- Embedded audio/video driver and algorithmic development
- Linux kernel development

Jun 2010 – Dec 2013 **Broadcom Europe Ltd, Cambridge**
Senior Applications Software Engineer

- Software applications engineer on Cisco VGA videophone (8945) project
- Capable of working to very tight deadlines within a multinational team, delegating when necessary
- Responsible for ensuring final customer acceptance

Senior Development Software Engineer

- Enhanced quality and efficiency of existing H264 video encoder
- Completed algorithmic research for next generation H264 video encoder
- Held sole responsibility for development of next generation hardware driver, low level device drivers and FPGA software platform required for hardware verification

Technical Experience

- Embedded multi-threaded C, RTOS, embedded processors, device drivers, assembly code, hardware debugging, data analysis/algorithmic development, Threadx, Perforce, H264 video codecs, customer support, subversion, CVS, Linux, git, SVN

Apr 2010 – May 2010 **Cintel International Ltd, Ware**
Software Contractor

- Implemented, documented and tested a TCP/IP interface to an image manipulation device
- Developed a C++ server on the device to allow control via an XML based protocol and developed a Java application for testing and documentation

Feb 2007 – Nov 2009 **Microsaic Systems, Woking**
Software Development Engineer

- Designed and developed software for MEMS mass spectrometers
- Captured in-house and pharmaceutical client requirements, produced a formal specification and executed a phased plan to convert prototype software into final product in just 3 months
- Used UML design principles and built on previous experience of design patterns to ensure the extensibility of the core software
- Designed an innovative UI and demonstrated it to customers and investors

Sept 2001 – Feb 2007 **Cintel International Ltd, Ware**
Senior Design Engineer

- Developed two high-resolution film scanners, two colour correctors and an image manipulation platform for use in the film industry
- Enhanced film scanner value through introduction of greater levels of automation and implementation of an intuitive UI
- Specified complete image manipulation platform including processor, OS, toolchain architecture and programming language
- Introduced UML design methods and C++ to company, resulting in sufficiently extensible architecture that could support multiple applications
- Interacted with customers at client sites and trade shows

June 2000 – July 2001 **Tarragon Embedded Technologies, Cambridge**
Control Software Engineer – Automotive control and tools

EDUCATION

Sept 1996 – July 2000 **University of Sheffield**
MEng (Hons) Computer Systems Engineering (2:1)

- A dual honours, taught masters combining a core of applied maths (systems engineering) with computing

Sept 1995 – June 1996 **University of Sheffield**
Engineering Foundation Year (1st)

Sept 1988 – June 1990 **Grantham College of Further Education**
BTEC OND Electrical and Electronic Engineering (Merit)

PERSONAL INTERESTS

- Hiking/trekking, cross-country cycling, badminton, travelling, learning Mandarin, philosophy

CASE STUDIES

Bitcoin Mine Management Platform

Design Requirements:
Implement a data logging and control system for an ever increasing number of ASIC bitcoin miners. The system would require the ability to scale and be highly available for interrogation of "real-time" data as well as archiving the datastream for batch analysis.

Design Considerations:
The system would be based on several unsuccessful prototypes to enable reuse of hardware, with the design optimised for scaling and reliability.

The Design:
Reliable - the design has no single point of failure*. Inexpensive hardware network controllers are paired and can be optimised to any level of performance during a single controller failure. The data gathering is performed by a pub-sub cluster in an AWS virtual private cloud spread across multiple availability zones to ensure performance across major service outages. Telemetry from inside the system's most vital components is also available in the event stream. To ensure reliability all aspects of the system deployment were automated providing reproducibility.
Scalability - Scaling was identified early on as a major factor, in the mine management system this is enabled through automation. Using parameterisation during deployment and the ability to transition network controllers between different clusters the management software can satisfy most customer use cases*

Pre-Silicon Production GPU Video Encoder Driver

Design Requirements:
Implement a hardware video encoder driver on an FPGA emulation of the hardware.

Design Considerations:
The ability to test use cases early with a production driver is valuable in the semiconductor industry, but simulation environments are generally slow and validation is often focussed on proving many small use cases which may miss complex corner cases. Automatically synthesizing the RTL allowing the hardware to be emulated on an FPGA would provide a platform to develop the production driver in parallel with hardware validation

The Design:
A test driven approach was used to validate the hardware in simulation. Test data was available from a C model of the new encoder, this was used to validate the hardware model. Building the production driver inside a test harness with access to the test data accelerated the development of the driver through easy validation and early bug discovery. Different behaviour between the simulation and the FPGA synthesis could be examined in several resolutions to determine the nature of the bug, in this way process and synthesis bugs were reduced.