

Roy Spliet, MSc.

Web: <https://roy.spliet.org>

E-mail: roy@spliet.org

Tel: op verzoek

Als PhD-student aan de Universiteit van Cambridge onderzoek ik architecturale oplossingen voor safety-critical, hard real-time systemen. Ik ben gefascineerd door de low-level fundamenteën van computers (architectuur, besturingssystemen, compilers), met een specifieke interesse in data-parallelle architecturen als GPUs. Problemen pak ik het liefst hands-on aan, en ik pak snel de benodigde vaardigheden op om systematisch problemen te ontleden en op te lossen.

Publicaties

"**The case for limited-preemptive scheduling for GPUs in hard real-time systems**", R. Spliet, R.D. Mullins, in *Operating System Platforms for Embedded Real-Time Systems (OSPERT) 2018*

"**Motivating preemptive GPU scheduling for real-time systems**", R. Spliet, presentatie in *X.org Developers Conference 2016*

"**Conquering the complexity mountain: Full-stack computer architecture teaching with FPGAs**", A.T. Marketos, S.W. Moore, B.D. Jones, R. Spliet, V.A. Gavrilu, in *European Workshop on Microelectronics Education (EWME) 2018*

"**Fast on Average, Predictable in the Worst Case: Exploring Real-Time Futexes in LITMUS^{ART}**", R. Spliet, M. Vanga, B.B. Brandenburg, S. Dziadek, in *Real-Time Systems Symposium 2014*

"**KMA: A Dynamic Memory Manager for OpenCL**", R. Spliet, A. Varbanescu, B.R. Gaster, L.W. Howes, in *GPGPU7 Workshop 2014*

Opleiding

2015-present: PhD in Computer Science, University of Cambridge (funding: lowRISC C.I.C.)

Proefschrift: A SIMD architecture for hard real-time systems

Opkomende safety-critical systemen hebben behoefte aan high-performance data-parallelle processoren welke *tight* en *safe* worst-case execution times kunnen garanderen. Gezien de complexiteit van de huidige systemen, bijv. GPUs, is het onwaarschijnlijk dat er in de nabije toekomst modellen en WCET-algoritmes van voldoende precisie zullen worden ontworpen. In mijn proefschrift beschrijf ik een efficiënte SIMD accelerator voor hard real-time systemen. Deze architectuur, vanaf de grond ontworpen en geïmplementeerd in SystemC als een cycle-accuraat performance-model, staat het toe om *safe* en *tight* bovengrenzen voor de run-tijd van data-parallelle programma's af te leiden.

2010-2013: Master of Science in Computer Engineering, TU Delft

Proefschrift: A comprehensive study of Dynamic Memory Management in OpenCL kernels.

Specialisatie: General Purpose and High Performance Systems, omvat vakken in computerarchitectuur, compilers, parallele computatie en real-time systemen.

2006-2010: Bachelor of Science in Technische Informatica, TU Delft

Specialisation: Software Technology, omvat vakken in besturingssystemen, embedded systems en algoritmes.

2000-2006: *VWO natuur/techniek, Cals College IJsselstein/Nieuwegein*

Onderwijs

2016-2019: University of Cambridge, Computer Lab

Digital Electronics, eerstejaars bachelors vak, supervisies (onderwijsassistentie),

Object-Oriented Programming, eerstejaars bachelors vak, supervisies,

Compiler Construction, tweedejaars bachelors vak, supervisies,

Electronic Computer Aided Design and Architecture, tweedejaars bachelors practicum,

Algorithms, Sutton Trust zomerschool voor examenjaar middelbare scholieren, supervisies.

Werkervaring

Jul-Sep 2015 **Stagiair, University of Cambridge Computer Lab**

- Ontwerp van practica-opdrachten voor het tweedejaars vak “Electronic Computer Aided Design and Architecture”, waarin studenten worden getest op hun vaardigheden in Verilog, assembly en computerarchitectuur,
- Implementatie van een prototype (FPGA logica, Linux device driver) voor interactie tussen de programmeerbare logica en de ARM Cortex A9 processorkern aanwezig op de Altera Cyclone V FPGA.

Mar-Jun 2015 **Software Engineer, Ultimaker B.V.**

- Ontwikkeling van commercial-grade embedded software in C++ voor een prototype. Specificatie van de interface tussen client software en prototype in een team van 3.
- Diverse bijdragen aan het bootproces (Linux, U-boot) van een op Allwinner A20-gebaseerd ARM board met NAND storage.
- Rapid-prototyping van verbeteringen aan de NAND storage driver ten behoeve van de performance en boot-tijd, met diverse upstream Linux contributies als resultaat.

Jan-Mar 2015 **Uitzendkracht Customer Due Diligence, FGH bank**

Jul-Sep 2014 **Student, X.org Endless Vacation of Code**

- Reverse-engineeren van de “voltage- and frequency scaling hardware” van diverse NVIDIA GT2x0 GPUs, inclusief de procedures voor het aanpassen van de DRAM klok en timings,
- Implementatie van een mechanisme voor het aanpassen van de kloksnelheid in the Nouveau open-source driver voor NVIDIA hardware, met performanceverbeteringen tot resultaat.

2013-2014 **Research Intern, Max Planck Institute for Software Systems**

- Onderzoek van snelle, praktische implementaties van generieke real-time synchronisatie-primitieven, volgens de principes van fast userspace mutexes (futex).

Sep-Dec 2012 **Stagiair, ARM Ltd.**

- Ontwerp en implementatie van een Linux kernel driver en trace decode toepassing voor de CoreSight self-hosted trace en debug hardware aanwezig op de meeste ARM-based SoCs.

2010, 2011 **Part-time Developer (PHP, Drupal CMS), Sapito BV**

2008-2009 **Junior Developer (JAVA/ABAP), Super de Boer Supermarkets**

Geselecteerde open-source contributies

Linux kernel	<i>Nouveau driver</i> : mechanismen voor het aanpassen van voltage en kloksnelheid van diverse NVIDIA GPUs, <i>Allwinner SoC</i> : NAND storage driver verbeteringen, <i>LITMUS^RT downstream real-time scheduling extensions</i> : ARM-ondersteuning en real-time futex implementaties.
Ramulator	DDR3 en DDR4 timing model fixes.
GPGPU-Sim	Verbeteringen aan OpenCL-ondersteuning.

Kerncompetenties

Programmeren	C, C++, Java, Assembly, HTML/CSS, PHP, SQL
Parallel programmeren	OpenCL, pthreads, OpenMP
Hardwaremodellieren	SystemVerilog, SystemC, VHDL
Revisiebeheer	GIT, SVN
Talen	<i>Nederlands</i> moedertaal <i>Engels (UK)</i> vloeiend, ECRTS 8.0/CEFR C1 <i>Frans</i> basis, CEFR B2 <i>German</i> basis