Paper Title: **Performance Evaluation of Dynamic Circuit Specialization on Xilinx FPGAs**
Authors: Prof. Dirk Stroobandt, Tom Davidson, Karel Heyse, Amit Kulkarni
Venues: Copenhagen - Sept. 11th; Stockholm - Sept. 9th
Abstract: Dynamic Circuit Specialization (DCS) is a technique used to optimize FPGA applications when some of the inputs, called parameters, are infrequently changing compared to other inputs. For every change of parameter input values, a specialized FPGA configuration is generated during run time and the FPGA is reconfigured with a specialized bitstream. We examine how the performance of the DCS technique evolves with the advent of newer Xilinx FPGA architectures. The performance of the DCS technique is evaluated on three different Xilinx FPGA architectures: Virtex-II Pro, Virtex-5 and Zynq SoC. We have used a 16-tap, 8-bit FIR filter as a parameterized design, with the filter coefficients as the parameters of the FIR design.

Paper Title: **Optimizing Memory Power in Hybrid ARM-FPGA Chips With Lossless Data Compression**
Authors: Peng Sun, Jose Nunez-Yanez
Venues: Copenhagen - Sept. 11th; Stockholm - Sept. 9th
Abstract: In current electronic systems the amount of power needed by the memory components can represent a large percentage of overall power requirements, and while modern DRAM memories offer very low idle power states, the reduction in active power is much more modest. Motivated by these observations, this paper presents a system architecture in which a hardware lossless data compressor/decompressor is connected to the application processor present in the same chip. The compressor increases the amount of time that the DRAM memories can remain in low power state by reducing the number of memory accesses and hence reducing the DRAM memory power consumption. The data compressor is instantiated in the programmable logic side of a ZYNQ device and is controlled by the ARM processors present in this chip moving data between the on-chip local memory and the off-chip DDR memory through the AXI interconnect. Memory active time and power are monitored in the board while different tests are run under the Linux operating system. The presence of the compressor enables the memory to move to a low power mode more frequently and it achieves an overall system power reduction of 12.4%. This figure includes the power overhead introduced by the presence of the compressor itself and it is limited by the efficiency of the low power modes of the considered DDR3 devices and data compressibility.

Paper Title: **The Track Engine – an FPGA Implementation of a Track-finding Algorithm for the IceCube Neutrino Telescope**
Authors: Carl WERNHOFF
Venues: Stockholm - Sept. 9th
Abstract: The Track Engine is an additional low-energy trigger and track reconstructor for the IceCube Neutrino Telescope. IceCube is built within the ice on the geographical South Pole, Antarctica. A neutrino interacting with an ice molecule might produce a muon, in turn emitting a trace of photons as it travels through the ice. Those photons can be detected by some 5000 optical modules, contained within the 1 km3 detector volume. Photons from neutrinos will be

embedded in a background of noise hits. Triggers and reconstruction algorithms search the detector data for signatures of hits associated with particle tracks. Low-energy neutrinos produce low-energy muons leaving dim tracks (few photon hits), and the Track Engine has been developed to improve the triggering sensibility for those tracks. For that, complex computations in realtime are required and this required an FPGA implementation since a software implementation would be too slow. Although the original intention of the Track Engine was to act as an additional trigger, each trigger message contains additional information: angle estimates of the indicated track in zenith and azimuth (accurate to roughly 10 degrees) and an indication of its strength. The Track Engine hence offers a rough reconstruction of the track as well. The Track Engine Algorithm has been developed to find dim tracks within a strong noise background. The stream of hits is divided into time windows, and for each time window, all possible pairs of the hits are formed. The zenith and azimuth angles (in a spherical coordinate system) is computed for each pair, and the angles are histogrammed. For time windows with only noise hits, the histogram is expected to be flat, whereas for a time window with a photon trace, an excess of angles consistent with the direction of the muon is expected. Hence, for a peak in the histogram, the TrackEngine reports a potential track, including the angles triggered on. The FPGA implementation of the algorithm consists of a long (approx. 1500 clock cycles) pipeline. The pipelined structure, although complex, gives an unsurpassed computing performance. A design choice of sticking with integers was made. This reduced area usage and improved timing (and hence performance), but at the cost of complexity. The histograms were implemented in BlockRAM memory. Handling pipelined population and read-out of the histograms required means of stalling the pipeline in case of consecutive increments of the same bin. The Track Engine hardware has been installed both on a test system and in the detector computer hall on the South Pole. The Track Engine has been thoroughly tested, on the production hardware, with simulated data. The relative improvement of number of tracks found by the Track Engine, compared to other existing triggers, is 6% for cosmic ray-induced muons, and 12% for atmospheric neutrinos. Only about 0.1% of the Track Engine triggers are caused by triggering on the noise background.

Paper Title: **Going for Brain-Scale Integration – using FPGAs, TSVs and NOC based Artificial Neural Networks: A Case Study**

Authors: Nowshad Painda Mand, Johnny Öberg

Venues: Stockholm - Sept. 9th

Abstract: With better understanding of brain massive parallel processing, brain-scale integration has been announced as one of the key research area in modern times and numerous efforts has been done to mimic such models. Multicore architectures, Network-On-Chip, 3D stacked ICs with TSVs, FPGA's growth beyond Moore's law and new design methodologies like high level synthesis will ultimately lead us toward single- and multi-chip solutions of Artificial Neural Net models comprising of millions or even more neurons per chip. Historically ANNs have been emulated by either software models, ASICs or a hybrid of both. Software models are very slow while ASICs based designs lacks plasticity. FPGA consumes a little more power but offer the flexibility of software and performance of ASICs along with basic requirement of plasticity in the form of reconfigurability. However, the traditional bottom up approach for building large ANN models is no more feasible and wiring along with memory becomes major bottlenecks when considering networks comprised of large number of neurons. The aim of this paper is to present

a design space exploration of large-scale ANN models using a scalable NOC based architecture together with high level synthesis tools to explore the feasibility of implementing brain-scale ANNs on FPGAs using 3D stacked memory structures.

Paper Title: **A Multi-Paradigm Approach to Teaching Students Embedded Systems Design using FPGAs and CPLDs**
Authors: David C. Dyer, Yan Lin Aung
Venues: Copenhagen - Sept. 11th
Abstract: To create optimal embedded electronic systems, it is essential to ensure all implementation options are considered, and students of electronics and computer engineering must be educated in hardware, software and firmware. We begin by reviewing in an educational context various implementation techniques. These include commercial microcontrollers, bespoke instruction set architectures (ISA), Field programmable Gate Arrays (FPGAs) for 'soft-core' processors and dedicated digital 'engines', as well as Complex Programmable Logic Devices (CPLDs) for interface management. Thereafter, we describe our work to create a platform that incorporates the above but is extended to include software development and tools. Regarding ISAs, we use an FPGA configured as an ARM Cortex-M1 32-bit processor but also introduce a bespoke hybrid RISC/CISC 12-bit processor called VIP. This helps students explore and compare soft-core implementation issues. Furthermore, unlike most proprietary platforms, we can provide students with the HDL code of all our peripherals and interfaces. Especially those for the address and data lines used communicate with devices on our associated bespoke Teaching Auxiliary Board (TAB); which itself uses a CPLD programmed to provide features such as bus handshake, protocol conversions, timers, interrupts and simulation of 'slow memory locations'. We believe that our holistic approach provides exceptional learning opportunities to show how implementations may be partitioned across FPGAs and CPLDs acting as dedicated programmed logic or programmable soft-core processors.

Paper Title: **An FPGA-Like Ultra Low-Power Network-On-Chip for Multicore Embedded Systems**
Authors: Mehdi, Meisam
Venues: Copenhagen - Sept. 11th; Stockholm - Sept. 9th
Abstract: Packet switching is the most popular switching method in networks-on-chips (NoCs) due to its high resource utilization, good scalability, and superior performance in dealing with various traffic patterns. However, these benefits come at the price of complex multi-stage pipelined routers which increase NoC area and power consumption. On the other hand, most typical embedded applications designed for multicores have predictable inter-core traffic behavior that can be extracted at design time. For such applications, many design-time traffic management and network optimization methods can be adopted to enhance NoC power/performance. Our proposal is an FPGA-like NoC that relies on Spatial-Division Multiplexing (SDM) technique to assign a subset of NoC wires to the communication flows of a target application. To further decrease the communication power, we propose a new router architecture that utilizes a mixture of hard-wired and traditional programmable crossbar switches. We then develop an algorithm to map different tasks of an input application onto a

mesh NoC and assign an SDM route with sufficient bit width to each inter-task communication flow. The results show about 38% reduction in NoC power consumption, 19% smaller area, and 12% shorter packet latency. Due to the similarities between the proposed NoC and the communication infrastructure of PFGAs, the algorithm can also be used to solve the FPGA routing problem.

Paper Title: **A Reconfigurable On-chip Network Architecture for the Dark Silicon Era**
Authors: Mehdi, Hamid
Venues: Copenhagen - Sept. 11th; Stockholm - Sept. 9th
Abstract: Future chips are expected to feature hundreds of ondie resources, but a considerable portion of silicon area of these chips will be dark that cannot be powered and provided their required bandwidth. As a result, only a limited number of cores of future chips can be powered on simultaneously. Such an arrangement of cores requires some special on-chip inter-core communication treatment to be able to bypass dark cores. In this paper, we propose a reconfigurable network-on-chip that leverages the routers of the dark portion of the chip as an FPGAlike switch box to customize the topology for the powered cores at any time. To this end, routers of the dark parts of the chip are used as bypass switches that can directly connect distant active nodes in the network. Our experimental results reveal considerable improvements in energy consumption and latency of on-chip communication when compared to state-of-the art NoCs