

Optically-Clocked Instruction Set Extensions for High Efficiency Embedded Processors

Claudio Favi, Theo Kluter, Christian Mester, and Edoardo Charbon, *Senior Member, IEEE*

Abstract—We propose a technique to localize computation in Instruction Set Extensions (ISEs) that are clocked at very high speed with respect to the processor. In order to save power, data to and from Custom Instruction Units (CIUs) is synchronized via an optical signal that is detected through a Single-Photon Avalanche Diode (SPAD) capable of timing uncertainties as low as 50 ps.

The CIUs comprise a free-standing local oscillator serving a computing area of a few tens of square micrometers, thus resulting in extremely reduced power dissipations, since the distribution of a high frequency clock over long distances is avoided. This approach is based on the globally asynchronous locally synchronous concept, whereby the granularity of the local domains is reduced to a minimum, thus enabling extremely high local clock frequencies and low power, while minimizing substrate noise injection and intra-chip interference.

Thanks to this approach we can free ourselves from expensive synchronization techniques such as FIFOs, delays, or flip-flop based synchronizers by creating fixed synchronization points in time where data can be exchanged. The paradigm is demonstrated on a chip designed and fabricated in a standard 90 nm CMOS technology. A full characterization demonstrates the suitability of the approach.

Index Terms—Clock distribution, embedded systems, globally asynchronous locally synchronous (GALS), instruction set extensions (ISEs), optical clocking, optically clocked ISEs, single-photon avalanche diode (SPADs).

I. INTRODUCTION

OPTICAL clock distribution has been a subject of research for the past two decades. Even as early as the 1980s, with the rise of fiber-optics in telecommunications, Goodman *et al.* were the first to present a thorough analysis of optical interconnects for VLSI systems [1]. However, conventional electrical distribution remains the norm to date. The reasons for this trend until today were mainly the difficulty of integrating very fast detectors in standard CMOS processes, that operate in the $1.55\ \mu\text{m}$

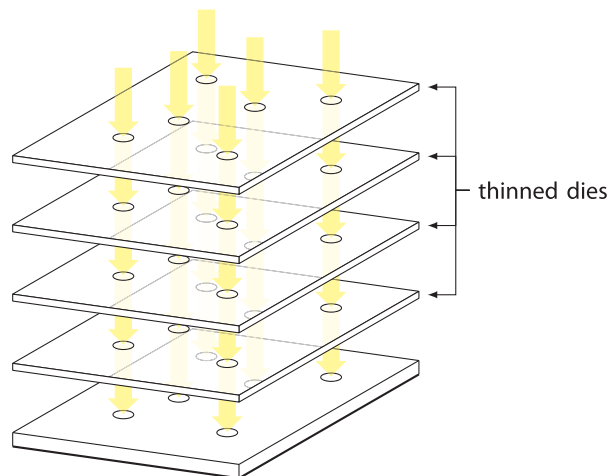


Fig. 1. Optical channels in a stack of thinned chip.

wavelength range. In addition, the need of *ad hoc* packages for optical distribution and fiber coupling deterred most manufacturers to pursue the optical clock route for cost and compatibility reasons.

Optical means for clock distribution and data transfer directly on chip are attractive for a number of reasons. In *primis*, an optically coupled network is less subject to the usual performance limitations of its electrical counterparts, such as skew, jitter, and power consumption, especially at high frequencies. In addition, with the emergence of 3D integration, fast through-chip communication and clocking has become a real issue, whereas through silicon vias, the currently proposed solution, are too bulky as of 2011 although they seem to start being effectively and reliably mass-produced. On the contrary, an optical channel can be implemented today through a stack of thinned silicon chips using conventional micro-optics techniques and air or dielectric based waveguides (Fig. 1).

Silicon dioxide and germanium waveguides can be used in a planar chip for horizontally pushing optical pulses. Their fabrication is becoming commonplace and it is already CMOS compatible at least in some SOI technologies [2].

Optical clock distribution can provide reduced skew and jitter in distributing the synchronization signal, though not necessarily slashing power [3]. A remaining problem is that of optical-to-electrical conversion [4] which [3], [5], and [6] have tried to solve with some success. However, much remains to be done in this field even with the encouraging advances achieved in optical clock distribution at the chip, package, board, and cabinet level [1], [7]–[10]. Interesting new directions are currently being pursued using alternative waveguide materials, such as

Manuscript received July 19, 2010; revised February 22, 2011; accepted September 03, 2011. Date of publication February 13, 2012; date of current version February 24, 2012. This work was supported by the Mobile Information and Communication System (MICS), the Swiss National Fund (SNF), and Xilinx University Program. This paper was recommended by Associate Editor Deming Chen.

C. Favi is with the School of Computer and Communication Sciences, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland (e-mail: claudio.favi@epfl.ch).

T. Kluter is with the Bern University of Applied Sciences, EKT, Microlab, 2501 Biel/Bienne, Switzerland.

C. Mester is with the School of Engineering, Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland.

E. Charbon is with the Delft University of Technology, Circuits and Systems Group, 2600 AA Delft, The Netherlands.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSI.2011.2169730

germanium that can be used horizontally and vertically. Fabrication of these waveguides can be performed even at low temperatures, thus making them compatible with a post-processing step on advanced deep-submicron CMOS technologies.

The development of purely electrical, high-performance clock networks has meanwhile progressed in the last years, yielding schemes to locally generate high-frequency clock signals in the spirit of a Globally Asynchronous Locally Synchronous (GALS) approach [11]. The solution generally adopted is that of a closed-loop-with-active-compensation that is implemented by means of phase-locked loops (PLLs) and delay-locked loops (DLLs) [12], [13]. In this context, much effort has been devoted to reducing jitter and power [4], [14]–[21]. However, these techniques are also generally power- and area-hungry. Besides PLL and DLL based circuits, ring oscillators have been proposed. With their simple design, compactness and predictable performance behavior [22], these circuits are commonly used for localized clock (re)generation [23]–[26].

We propose a CMOS optical clock distribution scheme named *Oscar*. Its application, detailed in Section II, is adapted to, but not limited to synchronization of an embedded processor with Instruction Set Extensions (ISEs) implemented on chip. The particular Custom Instruction Units (CIUs) proposed in this paper are circuitries that perform logic and arithmetic operations at an internal clock frequency that is significantly higher than that of the processor they serve. We believe that CIUs and in general application-specific integrated processors, are the best candidates to use the Oscar scheme since, as we will see later, they can run at a much higher speed than the host processor, due to the locality and relative simplicity of computation. The system can be thought of as fully synchronous but without the burden of global high-speed clocks that are replaced by ultra-low-power optical clock pick-ups based on single-photon detectors (Fig. 2). The single-photon detectors used in this work are CMOS compatible Single-Photon Avalanche Diode (SPADs) that were developed for the first time in a sub-100 nm CMOS technology by our group [27]. Due to the high speed and low jitter of these devices, they are equivalent to a high speed, low skew, and low jitter data/clock distribution network that at the same time requires no sophisticated, high-power techniques to achieve its performance. In addition, due to the versatility of these devices, the geographical localization of the CIUs on chip becomes irrelevant towards the achievable performance.

To understand the peculiarities of the proposed system, let us review those approaches that are relevant to it.

Distribution of clock signals, whether optical or electrical, has a major role not only in synchronous systems [28] but also in systems with limited or localized synchronicity. An example of such systems is the use of the GALS approach [29], where important power savings are realized by creating localized clock domains and replacing a global clock with asynchronous data exchange protocols. In the GALS approach, a core optimization lies in the selection of a sweet spot frequency for the localized clock domains. Another issue is that of the selection of the proper data exchange protocol to minimize the area and power

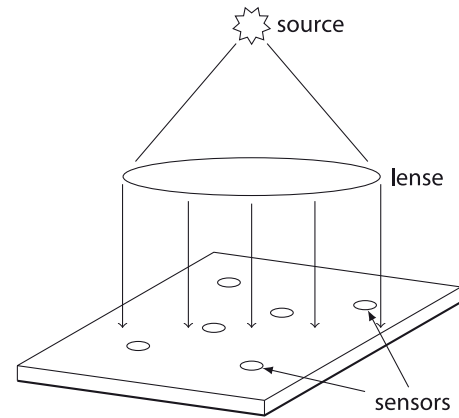


Fig. 2. Optical clock distribution example with a cone of light over the chip.

impact to the overall design. Reliable data transfer at high bandwidth between clock domains is addressed by several methods [29]–[31]. “Pausible Clocking” is one such mechanism of special interest to us. The idea is to “pause” the clock to allow safe latching of transmitted data between modules. The transmission can be done through FIFOs [32] or directly [33] but in any case synchronization is needed. Usually, these systems also suffer from non-deterministic execution which complicates testing and validation [34].

The demand of widely specialized processors has led to single-die, multi-die packaged, or multi-package heterogeneous systems. An example of the last category are coprocessor systems which were highly in vogue 20 years ago [35], [36] though some more recent work revisits the paradigm [37]–[40]. ISEs can be seen as an evolution of coprocessors. While coprocessors expand processor functionality with datapath and control logic through a defined external interface, ISEs with CIUs, are only an addition to the processor’s datapath. A careful choice of “accelerated” instructions is required and has been done manually until Clark *et al.* first demonstrated automatic selection [41]. Further research also confirmed the viability of automatic ISEs detection [42]–[50].

The system proposed in this paper builds on the experience of GALS, coprocessors, and ISEs. The clocking circuit implementing these ideas was designed into an integrated circuit and used to test a wide variety of trade-offs. The chip was implemented in a standard 90 nm digital CMOS technology. Chips in this process can be thinned to several tens of micrometers, thus enabling the *Oscar* technology to be used in 3D stacks where the optical clock would be transmitted through chips.

The use of SPADs for the optical pickups, instead of conventional photodiodes, has several advantages. First, due to the mechanism of self-amplification of SPADs, no amplifiers nor comparators are needed to convert optical onto electrical power. In addition, the avalanche process is very fast, thus enabling picosecond resolution in the synchronization edges. Second, thanks to SPAD sensitivity, it is possible to reduce the optical power used at the source and to use a combination of several parallel signals operating in close proximity. Finally, thanks to the miniaturization levels achieved in deep-submicron SPADs, the real estate overhead is negligible [27], [51]–[53].

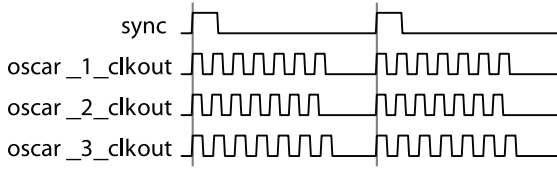


Fig. 3. Principle of operation of the non-PVT compensated oscillator. The three Oscar oscillators situated on different locations on chip are all started at the same time and run for $RV = 7$ cycles. Communication across clock domains is guaranteed only on the synchronization points.

As an alternative optical pickup technology Avalanche Photo diodes (APDs) could be used, the main advantage being an almost inexistent dead time that could enable operating frequencies in the gigahertz range at the price of a relatively complex amplification scheme and very strict bias control circuitry. However, in *Oscar*, global synchronization speeds are not critical, thus even nanosecond-long dead times are acceptable, as long as the timing resolution remains high, i.e., 100 ps or less. In SPADs, spurious firing (dark counts) and afterpulsing may occur [51]. However, these effects are inherently canceled by *Oscar* architecture.

The principle of operation of *Oscar*, illustrated in Fig. 3, is the following. A local oscillator is started by a pulse from the sensor and it is then stopped after an integer number RV of cycles. This mechanism ensures that the edges of all the generated clocks on the chip are aligned at these synchronization time-points. On the other hand, the clock edges in between might not be aligned due to process, voltage and/or temperature (PVT) variations. The limitation imposed by this clocking mechanism is that data can only be exchanged safely at the synchronization points. In order to demonstrate it, we propose to use the multi-cycle ISEs paradigm.

The paper is organized in the following manner. All the components of the architecture are described in Sections II-A–II-D. In particular, *Oscar* is described in Section II-B. The system and how it was validated is presented in Section III-A. Methodology of the tests at Section III-B introduces the results in Section III-C. Finally, the discussion in Sections III-D and IV covers both measurements, future work, and possible alternative applications of our system.

II. ARCHITECTURE

In this section, we will first present the demonstrator chip with all its components. The detailed description of the daughter/mother board ensemble and the software subsystem is outside of the scope of this publication but Fig. 5 and Section III-A give some hints on these subjects.

The chip, fabricated in TSMC 90 nm CMOS technology, comprises two OpenRISC processors and a custom bus interface (Fig. 5). All the peripherals such as main memory, VGA and USB interfaces, and BIOS are implemented in a field programmable gate array (FPGA). Note that having the main memory so far from the CPU is extremely inefficient however for the demonstration of *Oscar* functionality this is sufficient and cost effective. The micrograph in Fig. 4 shows the pad-limited chip design. The die size is $3940 \mu\text{m} \times 1875 \mu\text{m}$ for a total of 104 kGates.

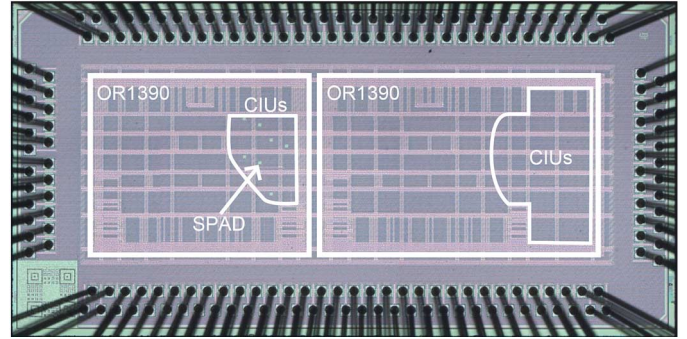


Fig. 4. Micrograph of the *Oscar* chip fabricated in TSMC 90 nm CMOS technology. The *Oscar* clocking circuitry is localized with the accompanying SPAD. Die size: $3950 \mu\text{m} \times 1875 \mu\text{m}$. Gate Count: 104k.

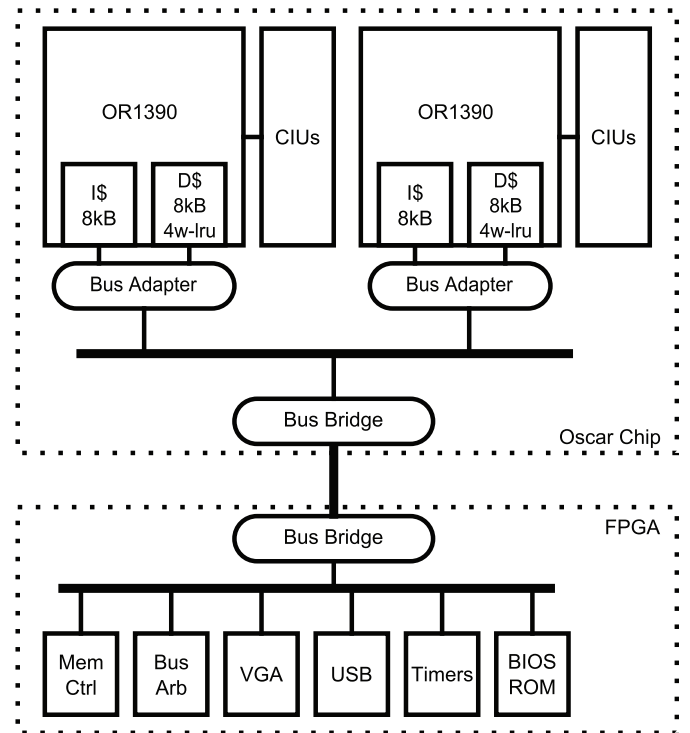


Fig. 5. Architecture of the *Oscar* chip and system's peripherals.

A. Single-Photon Avalanche Diode

A SPAD is an APD operated above breakdown voltage, in the so-called Geiger mode. In Geiger mode of operation, SPADs exhibit a virtually infinite optical-to-electrical gain, however a mechanism must be provided to quench the avalanche. There exist several techniques to accomplish quenching, classified as active and passive quenching. The simplest approach is the use of a ballast resistance. The avalanche current causes the diode reverse bias voltage to drop below breakdown, thus pushing the junction to linear avalanching and even pure accumulation mode. After quenching, the device requires a certain recovery time, to return to the initial state. The quenching and recovery times are collectively known as dead time. Fig. 6 shows the passive quenching scheme implemented in our design.

In 90 nm CMOS technology, SPADs exhibit a time resolution or jitter of 120 ps while the detection cycle, dominated by the dead time of the device, is generally of the order of 10 ns to

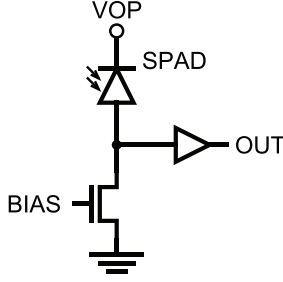
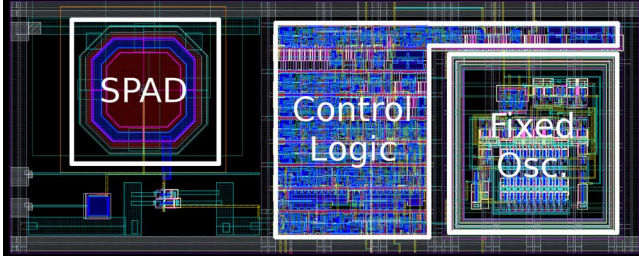
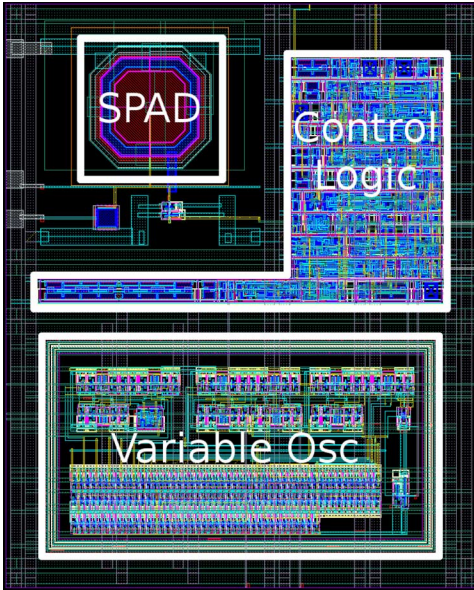
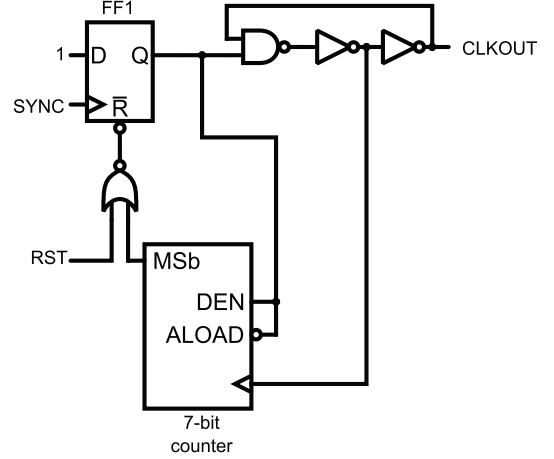
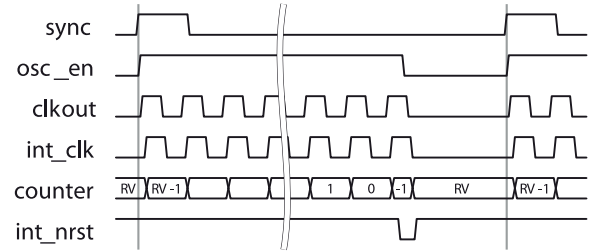


Fig. 6. The SPAD's ensemble: diode, quenching, and buffering circuitry.

Fig. 7. Layout of Oscar with fixed frequency oscillator 550 MHz. Dimensions: $68 \mu\text{m} \times 27 \mu\text{m}$.Fig. 8. Layout of Oscar variable frequency oscillator. Dimensions: $53 \mu\text{m} \times 66 \mu\text{m}$.

100 ns. The noise, known in SPADs as dark count rate (DCR), reaches a few kilohertz. The active area of the detector is less than $6 \mu\text{m}$ in diameter while the total size of the detector is of $400 \mu\text{m}^2$ due to the fact that a guard ring must be built around the active area to prevent premature edge breakdown. The $p^+ - n$ junction is designed to achieve a breakdown voltage at about 10 V. Operating the device at a few volts above breakdown, Geiger mode of operation is achieved, whereby this voltage is known as excess bias voltage. In these devices, photon detection probabilities up to 50% can be reached when an appropriate excess bias voltage is chosen.

At the time of the design, the SPAD ensemble was chosen at the same time that the first results of the 90 nm SPADs were

Fig. 9. Simplified schematic of *Oscar*. The D-flip-flop FF1 acts as a filter on the SYNC input that is driven by a SPAD and enables the non-PVT compensated ring oscillator. A 7-bit counter is used to reset the filter which in turn stops the clock generating oscillator.Fig. 10. Timing diagram of the inner workings of *Oscar*. Note that the internal asynchronous reset, is active when the counter underflows.

available. We discovered only later that the selected SPAD ensemble is not functional. However, for the purpose of the following discussion and without loss of generality, we used an external $0.35 \mu\text{m}$ SPAD.

B. *Oscar*

The constraints in designing the optically synchronized ring oscillator *Oscar* were twofold. First, it should be relatively small and simple so that to keep the area covered minimal. Second, it should generate a fixed number of rising clock edges without glitches at high frequencies. We designed two versions of *Oscar*: a fixed frequency and variable frequency one. Figs. 7 and 8 show the layout of the fixed and variable oscillator versions, respectively.

The designs are based on standard cells except for the sensor area. The oscillator was placed and routed by hand in contrast to the control logic, which was automatically synthesized, and placed and routed. The oscillator and control logic ensemble were validated in a transistor-level simulation.

The simplified schematic of *Oscar* is shown in Fig. 9. The output of the D-flip-flop FF1 is used to start the oscillator composed of the NAND-gate and the inverters when a rising edge appears on the SYNC input. The 7-bit down-counter starts from RV (the Reload Value of the counter) after reset and decrements based on a delayed version of the CLKOUT signal. When the counter underflows, the most significant bit is used to reset the D-flip-flop which in turn asynchronously loads the counter with

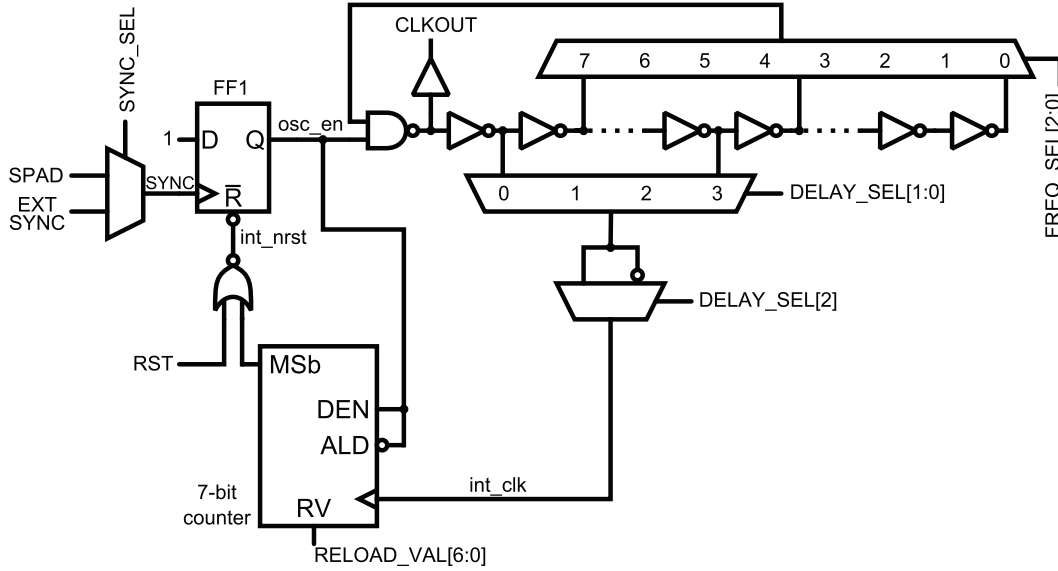


Fig. 11. Detailed schematic of Oscar. The D-flip-flop FF1 acts as a filter on the sync input that is driven by a SPAD and enables the non-PVT compensated ring oscillator. A 7-bit counter is used to reset the filter which in turn stops the clock generating oscillator.

the programmable RV value. Note also that the counter is active only when the oscillator is enabled.

A detailed version of the schematic and a timing diagram of the relevant signals in operation are shown in Figs. 11 and 10, respectively. Note in Fig. 11 the multiplexer that selects the SYNC signal, allowing the circuit to operate either using the SPAD or, in debug mode, with an external clock. The CLKOUT output is the first tap of the delay chain in order to minimize the delay and jitter between the SYNC pulse and the first edge of the generated clock signal. Glitches at the output may arise due to the counter-reaction control loop delay. To avoid glitches, the 3-bit DELAY_SEL enables a fine selection of delays in the control loop by selecting several readily available shifted versions of the clock. This mechanism is necessary on the variable oscillator version of *Oscar* (Fig. 11) because of the large difference between the oscillator's possible periods and the fixed delay of the counter-reaction loop from the counter to the oscillator's output.

The timing diagram of Fig. 10 illustrates the working operation. Note that for glitchless clock generation, the following relation must be true:

$$\delta_{fb} < \tau/2$$

where δ_{fb} is the delay of the feedback loop from/to the output of the NAND-gate passing through the underflowing counter, and τ is the clock period. When the clock frequency reaches the gigahertz range, $\tau/2$ approaches δ_{fb} . Fortunately, the feedback loop delay can be easily adjusted by selecting the proper feedback point in the delay line. Selecting odd or even taps, a delay values in the range of $[0, \tau/2]$ and $[\tau/2, \tau]$ respectively can be chosen. Note that in the preceeding discussion, the jitter of the SYNC signal and the generated clock was deliberately omitted for clarity. A safety margin is also required to cope with these signals uncertainties. The feedback loop delay mechanism is also suitable for this.

The layout of the variable frequency *Oscar* is shown in Fig. 8. The three constituting elements are the SPAD, control logic, and variable ring oscillator. The SPAD has been described in Section II-A. A much smaller device could be beneficial in terms of area, noise, and afterpulsing, due to the reduced carriers involved in an avalanche. Its active region is separated from the rest of the design by $10 \mu\text{m}$ in order to limit substrate noise injection. For the same reason, the ring oscillator has a triple guard ring to capture substrate charges generated by the mass of switching inverter gates that form the oscillator. The controller occupying the space between the SPAD and the ring oscillator, contains 58 digital cells. The total size of the cell is of $53 \mu\text{m} \times 66 \mu\text{m}$.

Both fixed and variable oscillator implementations suffer from metastability issues on the filtering flip-flop FF1. In fact if the recovery or removal times of this flip-flop are violated the system may become unstable or even oscillating. This is due to the fact that a metastable osc_en will propagate through the reset feedback loop to int_nrst . However, we force by design the SYNC signal to occur at a predefined interval δ_T . Therefore, for a given oscillator period δ_{osc} we choose a reload value RV such that

$$\delta_T - \delta_{\text{recovery}} - \delta_{\text{prop}} < RV \times \delta_{osc}$$

or

$$\delta_T + \delta_{\text{removal}} - \delta_{\text{prop}} > RV \times \delta_{osc}$$

where δ_{recovery} and δ_{removal} are the recovery and removal times of the reset signal with respect to the clock of the flip-flop. δ_{prop} is the propagation delay inherent to the feedback loop. Again, as discussed above, the jitter of the SYNC signal and the generated clock impact metastability and an extra safety margin should be taken for this.

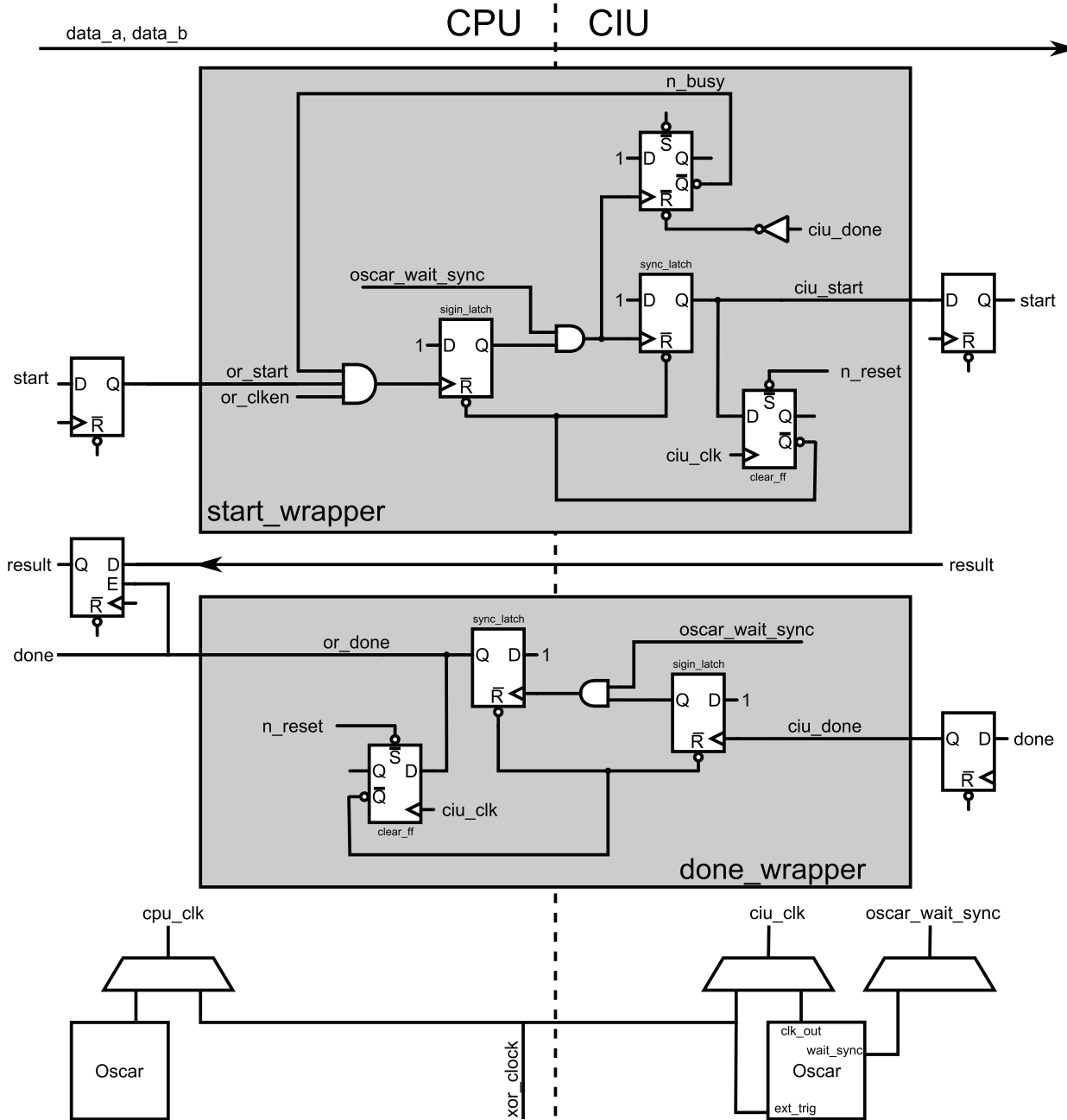


Fig. 12. CI call synchronization logic is used to ensure the *start* and *done* control signals are extended to the synchronization time-points.

A clock distribution network using *Oscar* must, like any other clock distribution network, control skew and jitter at all end-points. As already mentioned, skew can be reduced to almost zero, thanks to the optical distribution approach. However, a mismatch due to technology variations might introduce a systematic offset between the leading edge of two *Oscar* generated clocks trees. Note that, in the scheme proposed, only the skew of the leading edge of the first clock cycle is important. The same is true for jitter. The jitter of the first clock edge here is dominated by the SPAD's jitter. In fact, the filter flip-flop and NAND gate contributions are negligible. The sensor's jitter was not optimized in this design (400 ps for 90 nm SPAD and 80 ps for an

external 0.35 μm SPAD). For reference, commercial microprocessors have clock distribution jitter as low as a few picoseconds for multi-GHz clock frequencies at the cost of large silicon area.

C. Processor and Custom Instructions

The OpenRISC 1000 [54] instruction set-compliant processor used by Kluter in [55]–[57] was ported from FPGA fabric to ASIC. The ASIC derivation (OR1390) used in this work, was adapted for TSMC 90 nm CMOS semi-custom flow based on Low- V_t standard cells and memories. The processor has a 5-stage pipeline in-order architecture with 8 kilobytes 4-way set associative data cache and 8 kilobytes 2-way set

associative instruction cache. Both caches use a LRU replacement policy. The custom instruction interface is compliant with [58] allowing multi-cycle custom instructions to be added to the processor.

D. Custom Instruction Units and Oscar

The choice of the CIUs was made to demonstrate the *Oscar* clocking mechanism. As such the ideal CIU would be a unit that takes few input data as this is limited by the *Oscar* synchronization mechanism. It would then process the data for numerous local clock cycles and return few output data.

Three CIUs were manually implemented in order to test the *Oscar* clocking scheme. Each of these CIU contains control logic in form of a finite state machine (FSM) to provide multi-cycle execution. The first CIU is a textbook implementation of a radix-1 non-performing restoring 16-bit integer divider. This radix-1 divider takes 17 cycles to complete. The second CIU is a classic multi-cycle 32-bit integer multiplier with 32-bit integer result. This CIU takes 36 cycles to complete. Finally we implemented a shifter that supports arithmetic and logic shifts as well as rotations. This shifter performs a single shift each cycle making its execution time dependent on the number of positions to shift.

The variable-cycle CIs require that, for a fixed *Oscar* configuration, the done control signal be extended until the next synchronization timepoint. The added logic called *done_wrapper* is shown in Fig. 12. The *start_wrapper* was added in order to synchronize the start signal in the special case where the processor would also be clocked by *Oscar* and a CI call is not aligned on a synchronization boundary. Although not strictly necessary, these synchronization wrappers make use of *Oscar* state information and are almost transparent in normal operation. Their asynchronous design introduces only combinatorial delay to the control signals path.

III. RESULTS

Validation of an ASIC design plays an important role in ensuring that design specifications be met before fabrication. Beside lengthy simulations at RTL or gate level, emulation is a key validation step. The process has been made popular by the wide adoption of FPGA platforms. Before presenting the results related to the use of *Oscar* clocking, we emphasize the validation of the system as a whole in the following section.

A. System Pre-Validation

The system architecture described in Fig. 5 is based on a Chip-FPGA codesign. Validation of the whole system was performed on a dual FPGA board shown in Fig. 13. The FPGA on the left containing the memory controller, the bus arbiter, a VGA controller, the BIOS, a timers module, and the USB interface used for software transfer and configuration. In the pre-validation phase, the second FPGA held the same code used for the chip except for the technology specific parts (memories, flip-flops, and *Oscar*). Later, this part was replaced with the test chip mounted on a mezzanine daughterboard (Fig. 14).

The bidirectional interface has been thoroughly tested between the two FPGAs. When moving to the *Oscar* chip daughterboard (Fig. 14), the differences in timing due to the high ca-

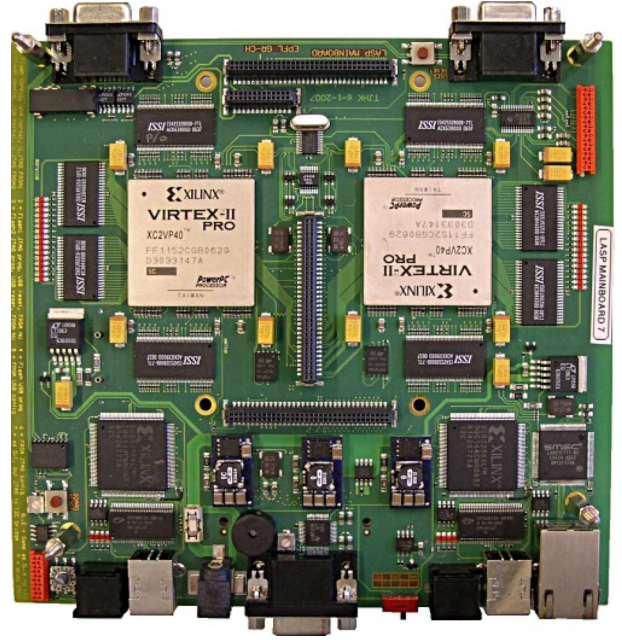


Fig. 13. The dual FPGA board used to validate the systems' architecture.

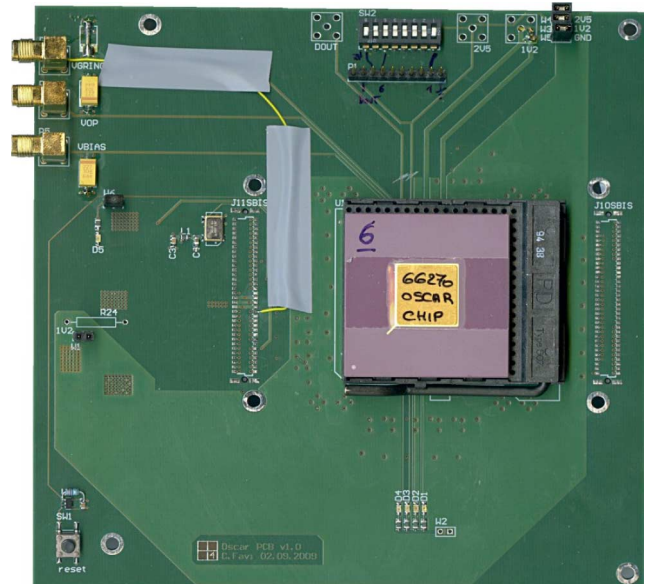


Fig. 14. Daughterboard holding the *Oscar* chip and interfacing to the FPGA board.

pacitative load of the connectors limit the maximum speed of the chip-to-mainboard communication to around 70 MHz. We finally ran the interface at 50 MHz to keep safety margins.

B. Test Setup and Methodology

Functional tests were first performed with external clocking as opposed to *Oscar* clocking. Correct functionality of the CPUs and Custom Instructions were validated. Especially the CIs were thoroughly tested with all combinations of input values, when possible.

To measure the frequencies of the oscillators, we use the CPU frequency counter that basically counts the clock cycles in a millisecond. In order to measure the oscillator frequency f_{osc} ,

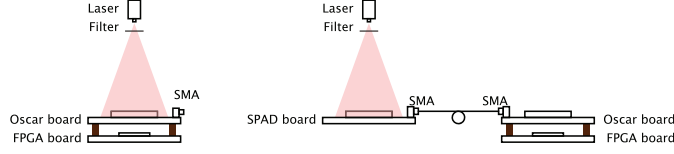


Fig. 15. Left, the original setup to test the *Oscar* chip with the internal SPADs. Right, the setup used for the tests with electrical connection between a $0.35\ \mu\text{m}$ SPAD chip and the *Oscar* chip.

we set *Oscar*'s sync at a frequency f_{sync} , successively increment the reload value RV , and record the reported frequency value. The maximum value approaches the real value and we see the following trend of reported frequencies:

$$f_{\text{sync}}, 2f_{\text{sync}}, \dots, Nf_{\text{sync}}, \frac{N+1}{2}f_{\text{sync}}, \dots, \frac{M}{2}f_{\text{sync}}, \frac{M+1}{3}f_{\text{sync}}, \dots$$

This is easily explained by the fact that whenever $RV \times f_{\text{sync}} > k \times f_{\text{osc}}$, we are crossing a synchronization time-point boundary and, therefore, the oscillator is stopped until the following sync arrives.

Power measurements were performed with a Tektronix TM502A current probe amplifier connected to a Picotech Picoscope 6403. A single measure is the average of 20 frames. A frame consists of 5 MSamples spanning 200 ms. The measurement precision, or reproducibility, was 1% of the absolute value. We only sampled the core voltage (1.2 V) thus leaving out I/O power (2.5 V). Whenever we measured dynamic power, unused parts were deactivated through clock gating.

The optical setup consists of a 637 nm laser diode. The nominal frequency of the diode is 40 MHz. However, the laser diode controller was also clocked externally with a function generator at lower frequencies. The uncollimated laser beam was directly pointed to the surface of a $0.35\ \mu\text{m}$ SPAD chip directly connect to the *Oscar* chip. The laser power was chosen to minimize pile-up effects. Fig. 15 shows the optical setup used.

All experiments were conducted at 20°C and 1.2 V core voltage. In all the tests, the influence of the bus clock, fixed at 50 MHz, has been minimized. For example, the test operations are performed on cached data values or processor registers to prevent bus accesses besides the initial mandatory fetches. This method maximizes power consumption and performance since a bus access would stall the processor for several cycles.

All chip control signals such as clocking muxes and *Oscar* parameters, were configured at run-time by the processor. The software was built with a customized toolchain based on GCC 3.4.4 in which custom instruction assembly opcodes were added. A JTAG-like interface is also available to set these parameters externally.

C. Measurements

The clock frequencies of the variable oscillator range from 114 MHz to 534 MHz while the fixed frequency oscillator runs at 502 MHz. These measures vary within 5% across different chips. The processor was validated to run up to 260 MHz. The difference between targeted frequencies and the reported frequencies is due to a simulation error in the oscillator.

TABLE I
POWER MEASUREMENTS INDEPENDENT
OF *Oscar* CLOCKING OPERATION

Static Power	42	μW
Dynamic Power	0.24	mW/MHz

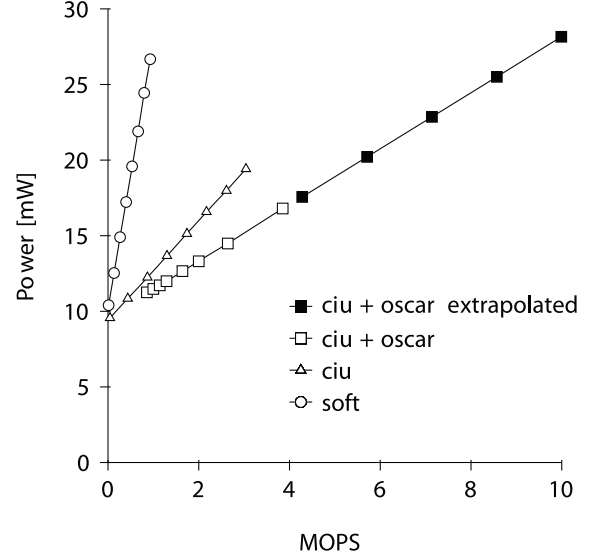


Fig. 16. Power measurements of the division operation. Note the expected linear trend of each set of measurements. The software division is here compared to the CIU only, our system lacking dedicated hardware division.

Power measurements independent of *Oscar* clocking operation are reported in Table I.

Static power includes the complete dual core system except IO power. Dynamic power for one CPU is measured with a tight loop of operations rearranging assembly code to prevent data dependency stalls as much as possible. The maximum value is selected. Note that only the 1.2 V core power is reported in this measurement.

Figs. 16–18 present power measurements versus million of operations per second (MOPS). For a given operation, a loop of 10000 iterations is run. The elapsed wall time for the loop execution is recorded in order to compute the MOPS figure. Three operations were tested: division, multiplication, and logic shift left. Depending on the operation, several ways of execution were tested. For the division, we used the software division available in the toolchain, the CIU clocked normally, and the CIU clocked with *Oscar*. For the multiplication, we ran the tests with the datapath's single-cycle multiplier, the CIU clocked normally, and the CIU clocked with *Oscar*. For the shift operation, we only compare the single-cycle datapath shifter with the CIU clocked with *Oscar*. In fact, for this operand-dependent variable-cycle instruction, the power is highly correlated with the operand value.

The offset of approximately 10 mW is the sum of static power and dynamic power due to the bus circuitry running at 50 MHz.

In order to compare the different implementations, we use the following well-known figure of merit:

$$\kappa = \text{performance} / (\text{power} \times \text{area}).$$

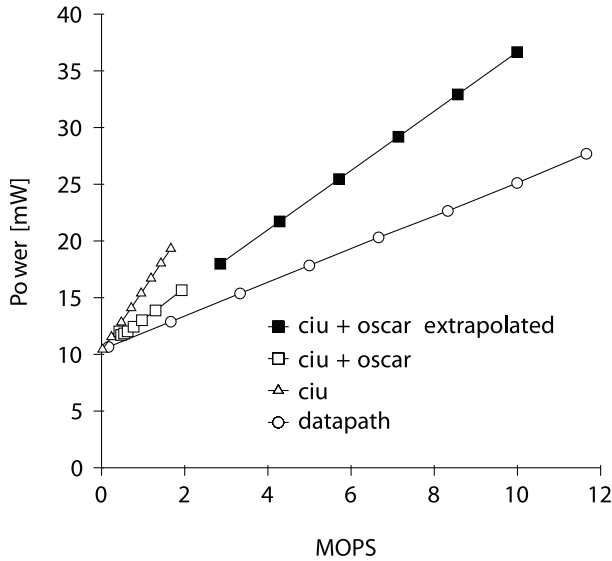


Fig. 17. Power measurements of the multiplication operation. Note the expected linear trend of each set of measurements. The datapath multiplier is compared to the multi-cycle CIU.

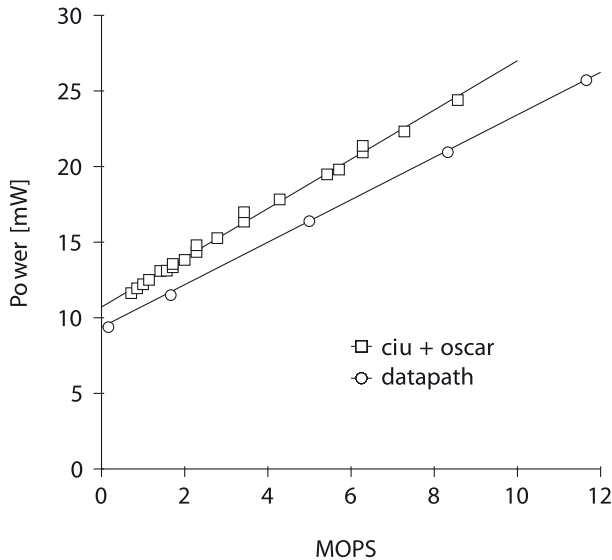


Fig. 18. Power measurements of the logic shift left operation. The data dependent variable-cycle CIU clocked with *Oscar* is run with a variety of input values, however the trend of the curve is still linear. The internal datapath shifter is also shown as reference.

Since our design is pad-limited, the area of the functional units, both CIUs and datapath, were recomputed separately. The constraints were 80% cell area usage, 500 MHz clock for datapath unit, and 2 GHz clock for the CIUs.

The figure of merit for the multiplication and shift operations are reported in Table II. Note that division is only represented with the CIU because of the lack of single-cycle datapath divider unit in our OpenRISC architecture. Also note that the area for custom instructions in this case is relatively small. However, we expect that in the near future custom instruction real estate will not only grow but overtake that of GP CPUs, by heavy use of parallelization. Thus our approach will have a significant impact on overall performance for given power dissipation.

TABLE II
FIGURES OF MERIT AND AREA OF CIUs AND DATAPATH UNITS

Instruction	Perf/Power [MOPS/mW]	Area [10^{-3} mm ²]	Figure of Merit [MOPS/(mW·mm ²)]
Mult DP	0.677	14.558	46.543
Mult CIU	0.184	4.573	40.303
Mult CIU+Oscar	0.382	4.573	83.631
Shift DP	0.711	2.077	342.816
Shift CIU+Oscar	0.636	1.598	398.326
Div CIU	0.304	2.874	105.775
Div CIU+Oscar	0.539	2.874	187.543

D. Discussion

The *Oscar* design shares some similarities with [59]. While applied to GALS designs, the digitally controlled clock multiplier of [59] also uses a gated ring-oscillator and counter. We differ from this design by being exclusively standard-cell based albeit being slightly less efficient in area, power and noise, because of the number of inverters and multiplexers used.

When compared to GALS designs, our clocking scheme is completely deterministic. In fact, our design is completely synchronous and all the known design verification rules still apply. For example, the synthesis constraints are set such that the CIU clocks frequencies are a multiple of the CPU clock. In this way, the synthesizer takes care of setup and hold time across time domains.

From the literature it is known that the use of photonics as means of clock distribution can only replace a small part of the total clock network power [4], [30]. The large capacitance to be switched is generally at the leaves of the clocking tree. The granularity of placing the optical-to-electrical converters is highly dependent of their area but also to the optical distribution means. For example, if the sensors were extremely small, one could have optical clock latches. The optics required to efficiently distribute the light could be based on holography. However, we place ourselves in a larger-grain approach by only clocking a few *Oscar* units. Optical distribution could be done with fiber optics although for sake of simplicity we beamed a sufficiently powered laser over the entire surface of the chip.

From the result section, multi-cycle CIUs can have better power efficiency with *Oscar* clocking. The difference of power between operations with and without *Oscar* is due to the CPU frequency difference. By design, when operating without *Oscar* the CIUs are underclocked, some energy is wasted while the stalled-cpu waits for the CIU's result, and they might be replaced by single-cycle designs at the cost of larger area.

When comparing datapath single-cycle operations (multiplication and shift) with the *Oscar* enabled CIUs, we note that the former are more power efficient. However this comes at the cost of larger on die area. The introduction of the figure of merit tries arbitrarily to balance these trade-offs. The benefits of *Oscar* may not be completely exploited in the area unconstrained case. However this discussion did not take into account leakage which become more and more important as technology scale and is dependent on the area. This omission was intentional, as we believe that leakage will equally impact a conventional approach and *Oscar*. Nevertheless, the simplicity of both

the clocking circuitry and CIs allow fast design development. Finally note that these power results are independent of the use of an optical clock. Throughout the paper we have intentionally omitted the power to run the optical clock from the overall power budget, since in principle it contributes in negligible way to the overall power budget of the chip and it can be considered as a separate, thermally independent block, perhaps shared among several receiving chips.

Although only the electric input of *Oscar* could be tested, the non-idealistic features of the SPADs are mitigated in this design in different ways. Reload time, after pulsing, and dark count are mitigated by the filtering inherent to the function of *Oscar*. The triggering window is purposely left small enough at the end of the clocking period so that spurious hits' impact is minimal. Any afterpulses are filtered by the SYNC flip-flop FF1 in Figs. 9 and 11. The jitter of the SPADs was not optimized in this design, however it could be reduced easily by employing several sensors and OR-wiring their outputs. The fixed frequency oscillator was meant to run at 1 GHz while variable oscillator would have selected frequencies between 200 MHz and 1.5 GHz. However a mistake in simulating the design yielded approximately half of these values. This should be fixed for a future design.

IV. OTHER APPLICATIONS

The *Oscar* clocking mechanism as presented in this paper shows some interesting applications. The simplicity of the design of *Oscar* itself and of the driven logic can be compelling in applications where power, performance, and area are serious constraints. Other applications could span from DSP to Network processors. In the security application context, *Oscar* could be used to generate a random-clock. In fact, when illuminated with non-coherent light, a SPAD produces uniformly spaced pulses (Poisson arrival times). This can be used to generate a truly random-clock with some simple filtering. Real-world applications of this clocking mechanism might include mobile phone chipsets, tablet computers, and embedded platforms such as commercial gaming platforms.

Also, the *Oscar* clocking scheme could be used in a 3D stack. Die thinning would be required for the optical clock to pass through to underlying dies. A discussion of die thinning can be found in [60]. Trade-offs between emitter power, die thinning and wavelength are required to be evaluated. *Oscar* clocking in this context could be achievable, however data-communication between dies should also be addressed.

V. CONCLUSIONS

In summary, the presented clocking scheme based on a globally optically synchronized local oscillator was applied to Instruction Set Extensions for an embedded processor using multi-cycle Custom Instruction Units. We presented an implementation of *Oscar* in 90 nm CMOS with the infrastructure around it. We discussed the power measurements results as a trade-off between performance, power and area. We briefly commented on similarities and differences compared to Globally Asynchronous Locally Synchronous systems. Finally, we presented some applications of this clocking scheme whether for security as truly random-clock generation or for 3D integration.

ACKNOWLEDGMENT

The authors would like to thank Zhong Zhong Ni at EPFL and Mohammad Karami and Mauro Scandiuozzo at TUDelft for their invaluable help during the design process.

REFERENCES

- [1] J. W. Goodman, F. J. Leonberger, S.-Y. Kung, and R. A. Athale, "Optical interconnections for VLSI systems," *Proc. IEEE*, vol. 72, no. 7, pp. 850–866, Jul. 1984.
- [2] A. Huang, C. Gunn, G.-L. Li, Y. Liang, S. Mirsaidi, A. Narashimha, and T. Pinguet, "A 10 Gb/s photonic modulator and WDM MUX/DEMUX integrated with electronics in 0.13 μm SOI CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2006, pp. 922–929.
- [3] C. Debaes, A. Bhatnagar, D. Agarwal, R. Chen, G. A. Keeler, N. C. Helman, H. Thienpont, and D. A. B. Miller, "Receiver-less optical clock injection for clock distribution networks," *IEEE J. Sel. Topics Quantum Electron.*, vol. 9, no. 2, pp. 400–409, Mar. 2003.
- [4] A. V. Mule, E. N. Glytsis, T. K. Gaylord, and J. D. Meindl, "Electrical and optical clock distribution networks for gigascale microprocessors," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 10, no. 5, pp. 582–594, Oct. 2002.
- [5] J. Fujikata, K. Nose, J. Ushida, K. Nishi, M. Kinoshita, T. Shimizu, T. Ueno, D. Okamoto, A. Gomyo, M. Mizuno, T. Tsuchizawa, T. Watanabe, K. Yamada, S. Itabashi, and K. Ohashi, "Waveguide-integrated Si nano-photodiode with surface-plasmon antenna and its application to on-chip optical clock distribution," *Appl. Phys. Expr.*, vol. 1, no. 2, p. 022001, 2008.
- [6] K. Ohashi, K. Nishi, T. Shimizu, M. Nakada, J. Fujikata, J. Ushida, S. Torii, K. Nose, M. Mizuno, H. Yukawa, M. Kinoshita, N. Suzuki, A. Gomyo, T. Ishi, D. Okamoto, K. Furue, T. Ueno, T. Tsuchizawa, T. Watanabe, K. Yamada, S.-I. Itabashi, and J. Aakedo, "On-chip optical interconnect," *Proc. IEEE*, vol. 97, no. 7, pp. 1186–1198, Jul. 2009.
- [7] S. J. Walker and J. Jahns, "Optical clock distribution using integrated free-space optics," *Opt. Commun.*, vol. 90, no. 4–6, pp. 359–371, 1992.
- [8] S. T. Tewksbury and L. A. Hornak, "Optical clock distribution in electronic systems," *J. VLSI Signal Process.*, vol. 16, no. 2, pp. 225–246, Jun. 1997.
- [9] L. C. Kimerling, "Silicon micropotonics," *Appl. Surface Sci.*, vol. 159–160, pp. 8–13, 2000.
- [10] P. J. Delfyett, D. H. Hartman, and S. Z. Ahmad, "Optical clock distribution using a mode-locked semiconductor laser diode system," *J. Lightw. Technol.*, vol. 9, no. 12, pp. 1646–1649, Dec. 1991.
- [11] S. Hasan, N. Bélanger, Y. Savaria, and M. Ahmad, "Crosstalk glitch propagation modeling for asynchronous interfaces in globally asynchronous locally synchronous systems," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 8, pp. 2020–2031, Aug. 2010.
- [12] M.-Y. Kim, D. Shin, H. Chae, and C. Kim, "A low-jitter open-loop all-digital clock generator with two-cycle lock-time," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 10, pp. 1461–1469, Oct. 2009.
- [13] K.-H. Cheng, Y.-C. Tsai, Y.-L. Lo, and J.-S. Huang, "A 0.5-V 0.4–2.24-GHz inductorless phase-locked loop in a system-on-chip," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. PP, no. 99, p. 1, 2010.
- [14] H. Kojima, S. Tanaka, and K. Sasaki, "Half-swing clocking scheme for 75% power saving in clocking circuitry," *IEEE J. Solid-State Circuits*, vol. 30, no. 4, pp. 432–435, Apr. 1995.
- [15] S. C. Chan, K. L. Shepard, and P. J. Restle, "Uniform-phase uniform-amplitude resonant-load global clock distributions," *IEEE J. Solid-State Circuits*, vol. 40, no. 1, pp. 102–109, Jan. 2005.
- [16] L. Zhang, B. Ciftcioglu, M. Huang, and H. Wu, "Injection-locked clocking: A new GHz clock distribution scheme," in *Proc. IEEE Custom Integrated Circuits Conf. (CICC)*, 2006, pp. 785–788.
- [17] L. Zhang, B. Ciftcioglu, and H. Wu, "A 1 V, 1 mW, 4 GHz injection-locked oscillator for high-performance clocking," in *Proc. IEEE Custom Integrated Circuits Conf. (CICC)*, 2007, pp. 309–312.
- [18] H. Lu, C. Su, and C.-N. J. Liu, "A tree-topology multiplexer for multiphase clock system," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 1, pp. 124–131, Jan. 2009.
- [19] T. Ragheb, A. Ricketts, M. Mondal, S. Kirolos, G. Links, V. Narayanan, and Y. Massoud, "Design of thermally robust clock trees using dynamically adaptive clock buffers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 2, pp. 374–383, Feb. 2009.

- [20] A. Chakraborty, K. Duraisami, P. Sithambaram, A. Macii, E. Macii, and M. Poncino, "Thermal-aware clock tree design to increase timing reliability of embedded socs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 10, pp. 2741–2752, Oct. 2010.
- [21] M. Alimadadi, S. Sheikhaei, G. Lemieux, S. Mirabbasi, W. Dunford, and P. Palmer, "A 4 GHz non-resonant clock driver with inductor-assisted energy return to power grid," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 8, pp. 2099–2108, Aug. 2010.
- [22] J. A. McNeill, "Jitter in ring oscillators," *IEEE J. Solid-State Circuits*, vol. 32, no. 6, pp. 870–879, Jun. 1997.
- [23] M. Combes, K. Dioury, and A. Greiner, "A portable clock multiplier generator using digital CMOS standard cells," *IEEE J. Solid-State Circuits*, vol. 31, no. 7, pp. 958–965, Jul. 1996.
- [24] M. Z. Straayer and M. H. Perrott, "A multi-path gated ring oscillator tdc with first-order noise shaping," *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1089–1098, Apr. 2009.
- [25] J. Borremans, J. Ryckaert, C. Desset, M. Kuijk, P. Wambacq, and J. Craninckx, "A low-complexity, low-phase-noise, low-voltage phase-aligned ring oscillator in 90 nm digital CMOS," *IEEE J. Solid-State Circuits*, vol. 44, no. 7, pp. 1942–1949, Jul. 2009.
- [26] X. Zhang and A. B. Apsel, "A low-power, process-and-temperature compensated ring oscillator with addition-based current source," *IEEE Trans. Circuits Syst. I, Reg. Papers*, p. 1, 2010, preprint.
- [27] M. A. Karami, M. Gersbach, and E. Charbon, "A new single-photon avalanche diode fabricated in 90 nm standard CMOS technology," in *SPIE Optics and Photonics*, 2010.
- [28] E. G. Friedman, "Clock distribution networks in synchronous digital integrated circuits," *Proc. IEEE*, vol. 89, no. 5, pp. 665–692, May 2001.
- [29] S. Dasgupta and A. Yakovlev, "Comparative analysis of GALS clocking schemes," *IET Computers Digital Techniques*, vol. 1, no. 2, pp. 59–69, Mar. 2007.
- [30] M. Krstic, E. Grass, F. K. Gurkaynak, and P. Vivet, "Globally asynchronous, locally synchronous circuits: Overview and outlook," *IEEE Design & Test of Computers*, vol. 24, no. 5, pp. 430–441, Sep. 2007.
- [31] J. Muttersbach, T. Villiger, and W. Fichtner, "Practical design of globally-asynchronous locally-synchronous systems," in *Proc. 6th Int. Symp. Advanced Research in Asynchronous Circuits and Systems*, 2000, pp. 52–59.
- [32] K. Y. Yun and R. P. Donohue, "Pausible clocking: A first step toward heterogeneous systems," in *Proc. IEEE Int. Conf. Computer Design: VLSI in Computers and Processors*, 1996, pp. 118–123.
- [33] X. Fan, M. Krstic, and E. Grass, "Analysis and optimization of pausable clocking based GALS design," in *Proc. IEEE Int. Conf. Computer Design*, 2009, pp. 358–365.
- [34] M. Heath and I. Harris, "A deterministic globally asynchronous locally synchronous microprocessor architecture," in *Proc. 4th Int. Workshop on Microprocessor Test and Verification: Common Challenges and Solutions*, 2003, pp. 119–124.
- [35] G. Wolrich, E. McLellan, L. Harada, J. Montanaro, and R. Yodlowski, "A high performance floating point coprocessor," *IEEE J. Solid-State Circuits*, vol. 19, no. 5, pp. 690–696, Oct. 1984.
- [36] W. Marwood and A. P. Clarke, "A coprocessor with supercomputer capabilities for personal computers," in *Proc. 1988 IEEE Int. Conf. Computer Design: VLSI in Computers and Processors*, 1988, pp. 468–471.
- [37] Y. Liu and S. Furber, "A low power embedded dataflow coprocessor," in *IEEE Computer Society Annual Symp. VLSI*, 2005, pp. 246–247.
- [38] A. Hodjat, D. Hwang, L. Batina, and I. Verbauwhede, "A hyperelliptic curve crypto coprocessor for an 8051 microcontroller," in *Proc. IEEE Workshop on Signal Processing Systems Design and Implementation*, 2005, pp. 93–98.
- [39] M. D. Galanis, G. Dimitroulakos, and C. E. Goutis, "Performance improvements in microprocessor systems utilizing a coprocessor datapath," in *Proc. Int. Conf. Embedded Computer Systems: Architectures, Modeling and Simulation*, 2006, pp. 85–92.
- [40] P. Li and H. Tang, "Design of a low-power coprocessor for mid-size vocabulary speech recognition systems," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. PP, no. 99, p. 1, 2010.
- [41] N. Clark, H. Zhong, and S. Mahlke, "Processor acceleration through automated instruction set customisation," in *Proc. 36th Annual Int. Symp. Microarchitecture*, San Diego, CA, Dec. 2003, pp. 129–40.
- [42] L. Pozzi and P. Ienne, "Exploiting pipelining to relax register-file port constraints of instruction-set extensions," in *Proc. Int. Conf. Compilers, Architectures, and Synthesis for Embedded Systems*, San Francisco, CA, Sep. 2005, pp. 2–10.
- [43] L. Pozzi, K. Atasu, and P. Ienne, "Exact and approximate algorithms for the extension of embedded processor instruction sets," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. CAD-25, no. 7, pp. 1209–29, Jul. 2006.
- [44] P. Nagaraju, K. Anshul, and P. Kolin, "Application specific datapath extension with distributed I/O functional units," in *Proc. 20th Int. Conf. VLSI Design*, Bangalore, India, Jan. 2007.
- [45] A. K. Verma, P. Brisk, and P. Ienne, "Rethinking custom isse identification: A new processor-agnostic method," in *Proc. Int. Conf. Compilers, Architectures, and Synthesis for Embedded Systems*, Salzburg, Austria, Sep. 2007, pp. 125–34.
- [46] A. K. Verma, P. Brisk, and P. Ienne, "Fast, quasi-optimal, and pipelined instruction-set extensions," in *Proc. Asia and South Pacific Design Automation Conf.*, Seoul, Korea, Jan. 2008, pp. 334–39.
- [47] K. Atasu, O. Mencer, W. Luk, C. Ozturan, and G. Dunda, "Fast custom instruction identification by convex subgraph enumeration," in *Proc. 19th Int. Conf. Application-Specific Systems, Architectures and Processors*, Leuven, Belgium, Jul. 2008, pp. 1–6.
- [48] L. Bauer, M. Shafique, S. Kramer, and J. Henkel, "Rispp: Rotating instruction set processing platform," in *Proc. 44th Annual Design Automation Conf., DAC '07*, New York, NY, 2007, pp. 791–796 [Online]. Available: <http://doi.acm.org/10.1145/1278480.1278678>
- [49] L. Bauer, M. Shafique, and J. Henkel, "Run-time instruction set selection in a transmutable embedded processor," in *Proc. 45th Annual Design Automation Conf. DAC '08*, New York, NY, 2008, pp. 56–61 [Online]. Available: <http://doi.acm.org/10.1145/1391469.1391486>
- [50] R. Lysecky and F. Vahid, "Design and implementation of a microblaze-based warp processor," in *ACM Trans. Embed. Comput. Syst.*, Apr. 2009, vol. 8, pp. 22:1–22:22.
- [51] A. Rochas, "Single photon avalanche diodes in CMOS technology," Ph.D. dissertation, EPFL, Lausanne, Switzerland, 2003.
- [52] C. Niclass, C. Favi, T. Kluter, M. Gersbach, and E. Charbon, "A 128 × 128 single-photon imager with on-chip column-level 10b time-to-digital converter array capable of 97 ps resolution," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2008, pp. 44–594.
- [53] M. Gersbach, J. Richardson, E. Mazaleyrat, S. Hardillier, C. Niclass, R. Henderson, L. Grant, and E. Charbon, "A low-noise single-photon detector implemented in a 130 nm CMOS imaging process," *Solid-State Electron.*, vol. 53, no. 7, pp. 803–808, 2009.
- [54] D. Lampret, "OpenRISC 1200 IP Core Specification," 2001 [Online]. Available: <http://www.opencores.org>
- [55] T. Kluter, "Architectural support for coherent architecturally visible storage in instruction set extensions," Ph.D. dissertation, EPFL, Lausanne, Switzerland, 2010.
- [56] T. Kluter, P. Brisk, P. Ienne, and E. Charbon, "Speculative DMA for architecturally visible storage in instruction set extensions," in *Proc. 6th IEEE/ACM/IFIP Int. Conf. Hardware/Software Codesign and System Synthesis (CODES+ISSS'08)*, New York, NY, 2008, pp. 243–248.
- [57] T. Kluter, P. Brisk, P. Ienne, and E. Charbon, "Way stealing: Cache-assisted automatic instruction set extensions," in *Proc. 46th Annual Design Automation Conf. (DAC'09)*, New York, NY, 2009, pp. 31–36.
- [58] Altera, Nios II Custom Instruction User Guide, Apr. 2010 [Online]. Available: http://www.altera.com/literature/ug/ug_nios2_custom_instruction.pdf
- [59] T. Olsson, P. Nilsson, T. Meincke, A. Hemam, and M. Torkelson, "A digitally controlled low-power clock multiplier for globally asynchronous locally synchronous designs," in *Proc. IEEE Int. Symp. Circuits and Systems*, 2000, vol. 3, pp. 13–16.
- [60] C. Favi and E. Charbon, "Techniques for fully integrated intra-/inter-chip optical communication," in *Proc. 45th ACM/IEEE Design Automation Conf.*, Jun. 2008, pp. 343–344.
- [61] Mobile Information and Communication Center. [Online]. Available: <http://www.mics.org/>
- [62] Xilinx University Program. [Online]. Available: <http://www.xilinx.com/univ/>



Claudio Favi received the Master degree in electrical engineering from Ecole Polytechnique Fédérale de Lausanne, Switzerland, in 2004. He worked as assistant project manager for m2 metro of Lausanne while teaching programming class to EPFL students from 2004 to 2005. In 2005, he joined EPFL where he received the Ph.D degree for his thesis entitled "Single-Photon Techniques for Standard CMOS Digital ICs" in May 2011.

He joined NagraVision SA in July 2010 where he currently works as hardware engineer in the field dig-

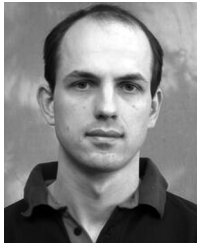
ital content protection. His research interests are optical/electrical communications, electronic design automation, reprogrammable computing and embedded systems.



Theo Kluter received the Master degree in electrical engineering from the Technische Universiteit Twente, Enschede, The Netherlands, in 1996. He worked as an R&D assistant in the Faculty of Computer Controlled Systems and Computer Techniques until 1997. From 1997 to 2002, he was a Design Engineer in the Design Center of Dedris Embedded Systems/Frontier Design/Adelante Technologies, Tiel, The Netherlands. In 2002, he joined Agere Systems as acting interim product development team leader for the Infotainment Group, Nieuwegein, The

Netherlands. In June 2003, he joined EPFL, where he received the Ph.D. degree in 2010.

Currently, he is a teacher at the Bern University of Applied Sciences (BFH) and at EPFL. His research interests include various aspects of embedded computer and processor architecture, embedded multiprocessor system-on-chip, design automation, and application-specific embedded system design.



Christian Mester received the Diploma degree in electrical engineering in 2006. From 2006 to 2009, he was a Marie Curie fellow at the European Organisation for Nuclear Research (CERN) in Geneva, Switzerland, where he designed full-custom high-speed high-precision timing integrated circuits. He received the Ph.D. degree from the University of Bonn, Germany, in 2009.

In spring 2009, he joined EPFL in Lausanne, Switzerland. His research interests include CMOS sensors, bio-photonics and low-power wireless

embedded systems.



Edoardo Charbon (SM'00) received the Diploma from ETH Zurich in 1988, the M.S. degree from the University of California at San Diego (UCSD) in 1991, and the Ph.D. degree from the University of California at Berkeley in 1995, all in electrical engineering and EECS.

From 1995 to 2000, he was with Cadence Design Systems, where he was the architect of the company's initiative on information hiding for intellectual property protection. In 2000, he joined Canesta Inc. as its Chief Architect, leading the development of wireless 3-D CMOS image sensors. Since November 2002, he has been a member of the Faculty of EPFL in Lausanne, Switzerland, working in the field of CMOS sensors, bio-photonics, and ultra low-power wireless embedded systems. In Fall 2008 he has joined the Faculty of TU Delft, as full professor in VLSI design, succeeding Patrick Dewilde.

Dr. Charbon has consulted for numerous organizations, including Bosch, Texas Instruments, Agilent, and the Carlyle Group. He has published over 200 articles in technical journals, conference proceedings, magazines, and two books; he holds 13 patents. His research interests include 3D imaging, advanced bio- and medical imaging, quantum integrated circuits, and space-based detection.

Dr. Charbon has served as Guest Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS and the IEEE JOURNAL OF SOLID STATE CIRCUITS and as member or chair of technical committees in ESSCIRC, ICECS, ISLPED, VLSI-SOC, and IEDM. He was recently appointed Distinguished Visiting Scholar by the W. M. Keck Institute for Space Studies at the California Institute of Technology in Pasadena, CA.