A Multiprocessor System-on-Chip Implementation of a Laser-based Transparency Meter on an FPGA

James Dykes, Paulman Chan, Glenn Chapman, and Lesley Shannon
School of Engineering Science, Simon Fraser University
Burnaby, Canada
{jdykes, pkchan}@sfu.ca, {glennc, lshannon}@ensc.sfu.ca

Abstract—**Modern FPGAs are large enough to implement Multi-Processor Systems-on-Chip (MPSoCs). Commercial FPGA companies also provide system design tools that abstract sufficient low-level system details to allow non-FPGA experts to design these systems for new applications. The application presented herein was designed by photomask researchers to implement a new technique for measuring the transparency of bimetallic grayscale masks using an FPGA platform.**

Production of the bimetallic grayscale masks requires a direct-write laser system. Previously, system calibration was determined by writing large rectangles of varying transparency on a mask and then measuring them using a spectrometer. The proposed technique uses the same mask-writing system but adds photodiode sensors connected to a multiprocessor computing system implemented on an FPGA. The added sensors combined with the laser beam's smaller focal point allows the calibration rectangles to be up to 5000 times smaller than those required by the spectrometer. This allows for direct mask verification on a µm-sized scale. Furthermore, the MPSoC design on the FPGA is easily scalable to support an increased number of photodiodes for the future addition of a feedback approach to the project.

Keywords – FPGAs; Laser Measurement Applications; Grayscale Photomasks; Transparency; Microlithography

I. Introduction

Conventional uses of FPGAs include prototyping, and hardware acceleration for data intensive applications such as image processing [1], encryption [2], and molecular dynamics [3]. Commercial FPGAs are large enough to implement complete Systems-on-Chip (SoCs) [4] and, more recently, even Networks-on-Chip (NoCs) [5] and Multiprocessor Systems-on-Chip (MPSoCs) [6]. Much of this work has been done by FPGA experts who are familiar with both the FPGA design platforms and CAD tool flow.

CAD tools and design methodologies have also been created to abstract low-level design issues and make FPGA design more accessible to non-FPGA experts. For example, AccelChip's DSP synthesis tool [7] can synthesize RTL from MATLAB [8] source and Simulink enables DSP developers to exploit a visual data flow to generate an FPGA design [9]. Also, both Mentor Graphics [10] and Celoxica [11] have tools that are able to generate FPGA designs from C-code. These high-level abstractions provide ease-of-use at a cost of slower runtime performance.

In this paper, we demonstrate the accessibility of modern commercial FPGA system-level CAD tools to non-FPGA experts. Moreover, we also describe a scalable MPSoC architecture implemented and applied to a new application area – the calibration of grayscale masks for microlithography.

To provide more information on our application area, Section 2 describes the process of microlithography and how bimetallic grayscale masks are created and currently calibrated. Section 3 then presents the proposed method and experimental platform for our laser-based transparency meter, while Section 4 presents the MPSoC design implemented on the FPGA. Section 5 then concludes the paper and discusses possible areas for future work.

II. Microlithography Using Bimetallic Masks

This section describes the application of grayscale masks, how grayscale masks are created, and how they are calibrated.

A. Grayscale Mask Preparation and Usage

Grayscale photomasks are similar to very high-resolution film negatives used to create 3D-structures, such as Micro-Electro-Mechanical Systems (MEMS) devices, using microlithography. Figure 1 illustrates the microlithographic process using a bimetallic grayscale photomask. The key to this process is that the thickness of developed resist is related to the light projected through the mask. Creating accurate 3D-objects requires photomasks to be precise both in the accuracy of their transparency, or gray levels, along with the positioning of the pattern itself.

There are different methods of creating grayscale masks. Our focus is on bimetallic grayscale masks, which are made of a Bismuth-on-Indium bimetallic film exposed by a focused Argon Laser (488nm wavelength) that converts the film into a transparent eutectic alloy oxide. The transparency of the oxide is dependent on the laser power used to expose the mask, such that a higher laser power results in a more complete oxidation and greater transparency [12].

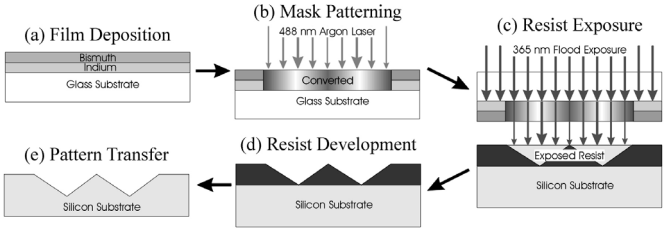

Figure 1. Creation and use of a bimetallic grayscale mask: (a) bimetallic film deposition, (b) laser exposure to create transparent oxide, (c) usage of the mask to limit photoresist exposure, (d) removal of exposed resist via development, and (e) resulting 3D-structure after pattern transference.

B. Grayscale Mask Calibration

In previous work, the bimetallic films for grayscale masks were calibrated before writing the desired mask by using a spectrometer to measure their transparency [13]. Large rectangles of varying transparency were written on a bimetallic coated glass slide with a layout similar to that shown in Figure 2. Due to the spectrometer's resolution, each rectangle was 1cm by 2cm and isolated by 2mm of unexposed bimetallic film. The size of each rectangle ensures that the spectrometer measures only the exposed film and does not include unexposed material in its measurement.

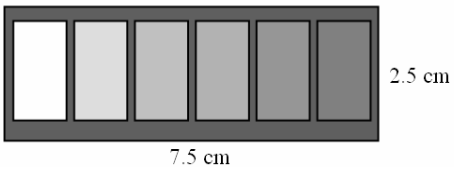

Figure 2. A patterned microscope slide for spectrometer measurements.

Properly calibrating a wide grayscale range (50+ gray levels) with a high degree of accuracy requires several of these slides. However, using several microscope slides is far from ideal since the bimetallic film properties can vary between slides. Another limitation of using the spectrometer is that it provides no direct method of verifying a mask's transparency at the micrometer scale of typical photomasks. A better method of measuring a photomask's transparency is thus needed.

III. Laser-Based Transparency Meter

The following section describes the direct-write laser setup, its operation and the capabilities of the existing platform.

A. Experimental Setup

The laser-based transparency meter setup, along with the original laser direct-write system, is depicted in Figure 3. Photodiode sensors, Analog to Digital Converters (ADCs), and an XUP Virtex II Pro Board (XUP Board) have been added to monitor and communicate with the original mask-writing system. The XUP Board currently has two independent connections for off-board communications to host Personal Computers (PCs). The first is a serial port to the laser table control computer, which provides the user interface and control to all the components of the direct-write laser system. The second host PC connects via a USB-JTAG connection and is temporarily used to verify and debug the operation of the laser-based transparency meter as new features are added.

The XUP Board's low speed expansion header provides a scalable interface that allows several ADCs to be connected to the board. Each ADC is connected to a sensor that is independently controlled from the board's FPGA. The ADC generates 24-bit voltage measurements from the sensor data at a maximum sampling rate of 12.19kHz. A function generator connected to the laser table control computer modulates the power of the argon laser via an electro-optic shutter. This function generator will be connected in the future to a Digital to Analog Converter (DAC) via the expansion header providing a closed loop feedback for the system.

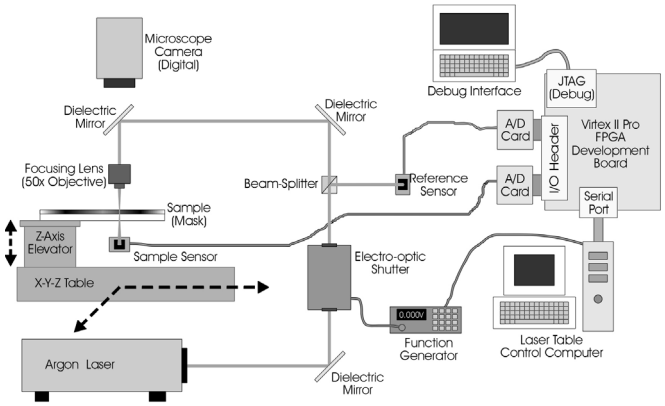

Figure 3. Laser-based Transparency Meter System.

B. System Operation

The original direct-write laser system consisted of an *argon laser*, an *electro-optic shutter*, a *focusing lens*, the *sample* (mask), a *microscope camera*, and an *X-Y-Z table*. To write a mask onto a sample, the power for the *argon laser* is modulated to achieve the targeted gray level for each X-Y location using the *electro-optic shutter*. The modulated laser is then reflected by *dielectric mirrors* towards the *X-Y-Z table* where the beam's diameter is reduced using the *focusing lens*. The *X-Y-Z table* holding the *sample* is able to move with a resolution of 0.15µm, allowing the user to create precise photomasks. To view the resulting pattern as the laser writes the mask, a *microscope camera* is required since the laser spot is only micrometers in size.

The laser-based transparency meter includes the addition of photodiode sensors to monitor the laser intensity before the beam passes through the *focusing lens* (reference sensor) and after it passes through the *sample* (sample sensor). The relationship between the initial reference beam and how the *sample* attenuates the beam's intensity is used to indicate the transparency of the *sample*. Calibration of the laser-based transparency meter is done by taking measurements for a set of known transparencies. A look-up table of transparency values indexed by the sensor measurements is then generated and stored, allowing subsequent measurements to use the stored table and interpolate the resulting transparency.

C. System Benefits and Limitations

The primary advantage of the laser-based transparency meter is that its minimum measurable area on the mask is equivalent to the area of the focused laser beam. In this system, the minimized area is approximately 2µm x 2µm (5000x smaller than the spectrometer) and is achieved by using a 50x objective lens as the *focusing lens*. Thus, the slide shown previously in Figure 2 could be shrunk down to 150µm x 25µm with the use of 20µm x 20µm squares. Moreover, the smaller size of the calibration segments means they can also be written on the *same* slide as the desired mask pattern. Using the same slide for both the calibration and the desired mask results in better accuracy since the film properties for

two separate slides would not be as similar.

One limitation of the proposed laser-based transparency meter is that the transparency profile of the grayscale mask is only being measured at the wavelength of the mask-writing laser (488nm) rather than the IC exposure system (typical 365nm). The relationship between the two wavelengths at all gray levels is being investigated.

Another challenge for the proposed platform is to ensure that the focused laser power is low enough when measuring such that it does not alter the mask. Currently, the laser's power intensity is verified using the reference and sample sensor prior to the addition of the sample mask. However, adding a third photodiode sensor to the system as an *input sensor* allows the beam to be measured prior to the electro-optic shutter. The resulting feedback provided by the three sensors could then be used to adjust the modulation performed by the shutter and account for both fluctuations in the incoming laser beam's power and also variations in the properties of the bimetallic film being patterned. Ideally, the resulting masks would be of significantly higher quality.

IV. FPGA Design

The following section describes the FPGA design's requirements, system architecture, and implementation results.

A. Design Requirements

The laser-based transparency meter application is the first phase of a multi-phase project, which will culminate in a feedback system geared toward improving the production of grayscale bimetallic photomasks. Thus, the computing system implemented on the FPGA must be easily scalable to allow for interfacing with other peripherals present in the system as well as additional sensors. The XUP Board has a Virtex2 Pro 30 and the necessary external connectors for this system, including: serial and Ethernet interfaces for uploading data to the host PC; low speed expansion headers that allow the connection of sufficient sensor inputs, via ADCs, to monitor the direct-write laser system; and the ability to implement a DAC to manipulate the electro-optical shutter for the future feedback addition to the system.

The FPGA design must include drivers for each of the peripherals currently in the system and facilitate the inclusion of additional sensors and ADCs as may be required in future phases of the project. For the current phase of the project, we require that the sampled sensor data be processed and recorded independently for each ADC; furthermore, the future feedback system will need to be able to simultaneously: process data, control the laser's power, and obtain new data.

B. System Architecture

Figure 4 illustrates the MPSoC design created for the laser-based transparency meter using Xilinx's [14] Virtex II Pro 30 on the XUP Board and their Embedded Development Kit (EDK) system CAD tools Version 8.2. Although, the Virtex II Pro includes two PowerPCs on-chip, only MicroBlaze [15] processors were used to create a base component that could be replicated to any number of processors without requiring the authors to learn about two different processor architectures and interfaces. In terms of operating systems, each MicroBlaze uses a standalone Board Support Package (BSP) to provide software modules that access processor specific functions. The entire MPSoC is comprised of two types of modules: the *Command Unit* and the *ADC Interface Units*.

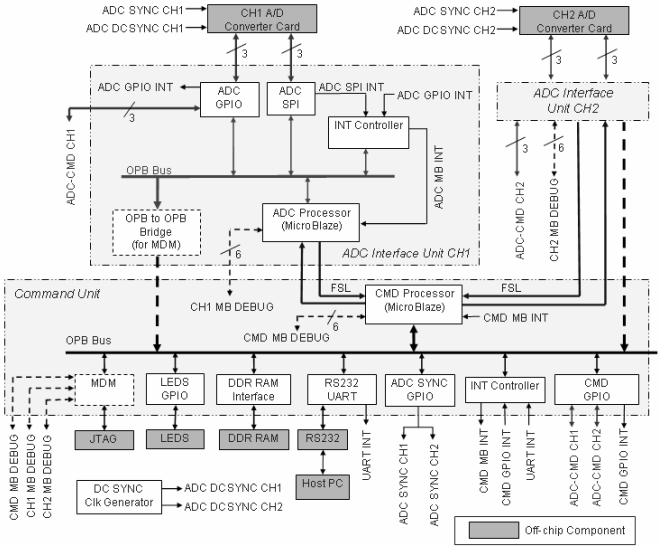

Figure 4. The laser-based transparency meter FPGA architecture.

Each sensor has its own interface unit, known as the *ADC Interface Unit*, used to read data from the *A/D Converter Card* as requested by the *Command Unit*. The *ADC Processor* receives instructions and parameters from the *CMD Processor* via Xilinx's Fast Simplex Links (FSLs), a FIFO-based structure. The *ADC Processor* uses these instructions to initialize the *ADC GPIO* to obtain measurements. Once the *ADC SYNC* signal synchronizes the sample measurements, the *A/D Converter Card* collects the data. The *ADC SPI* module generates an interrupt (*ADC SPI INT*) on the *INT Controller* when valid data is ready. The *ADC Processor*'s interrupt service routine reads this data from the *ADC SPI*, performs some initial processing and writes it to the *CMD Processor*'s FSL for final processing and storage. Each *A/D Converter Card* has its own *ADC Interface Unit* that operates independently, except for a global synchronization signal generated by the *ADC SYNC GPIO* in the *Command Unit*.

One *Command Unit* is used in this design. Its purpose is to: transfer and post-process data from the *ADC Interface Units* to store in *DDR RAM*, communicate with the Host PC, and track the status of the overall system. Each *ADC Processor* is connected to the *CMD Processor* via a set of FSLs. Although there are only two *ADC Interface Units* in the system shown, the MicroBlaze has eight Master/Slave FSL pairs, allowing the design to scale up to eight different *ADC Interface Units* provided there are sufficient GPIO pins and other resources.

The *CMD Processor* initializes the *ADC Interface Units*' to sample measurements based on user requests. These requests are transmitted via the *RS232 UART* and generate an interrupt request via the *INT Controller*. The *CMD Processor* then: 1) requests the appropriate *ADC Interface Units* obtains data samples, 2) updates the *LEDS GPIO* to indicate the status of

the system, and 3) sends a synchronization signal via the *ADC SYNC GPIO* to all the *A/D Converter Cards*. After the measurements are completed and stored in the *DDR RAM*, the *CMD Processor* updates the status LEDs and informs the Host PC that the measurements have been obtained.

For the purpose of debugging our design, we have included a *Microprocessor Debug Module (MDM),* which allows us to debug up to eight MicroBlazes at one time. All the debugging elements such as the *OPB-to-OPB Bridge* have been shown using dashed lines and will be removed upon completion of the project's final phase. Apart from the *OPB-to-OPB Bridge* which shares the *MDM* between all the processors, each processor has its own independent address space.

C. Implementation Results

The laser-based transparency meter runs at 100MHz and its resource usage on the Virtex II Pro 30 is given in Table I.

TABLE I. MPSOC DESIGN RESOURCE USAGE

Resource Type	Number of Resources Used	Percentage of FPGA Resources Used
Flipflops	6 434	23%
LUTs	9 675	35%
Multipliers	13	9%
BRAMs	82	60%

The *CMD Processor*'s MicroBlaze uses a floating point unit as well as the integer multiplier and divider units. These are needed to perform some of the final processing of the data before it is delivered to the user. However, the BRAM code space required for the *CMD Processor* is less than 16KB since space can be utilized in the *DDR RAM*.

The *ADC Processor*'s MicroBlaze only includes the integer multiplier and requires just over 32KB of BRAM for its current executable and data space. Future phases of the project will expand the functionality implemented by the *ADC Processor*; therefore, to ensure that there is sufficient memory to implement this increased functionality, we have allotted 40KB of BRAM memory for each processor.

As can be seen from Table I, there appears to be sufficient resources to scale the current MPSoC to include two additional MicroBlazes and the ethernet module specified for the final feedback system. We recognize that there are more resource efficient architectures that could have been chosen for this application. However, as non-hardware experts we were interested in obtaining a functioning system that would meet the operational requirements of the ADCs. Since their maximum sampling frequency is only 12.19kHz, a more optimized system design would not provide much improvement in performance. Lastly, our design currently only uses IP cores generated by the tools and no custom modules of our own design. The tools sufficiently abstract the low-level system design issues to make this MPSoC architecture possible for non-hardware experts to implement.

A possible improvement to the CAD tools that would have facilitated our design process is the ability to access a line or sub-segment of a bus. A custom core providing this functionality is currently needed to accomplish this task. More detailed documentation would also be helpful as the IP cores often had extra functionality that was not explicitly described, but had to be determined through experimentation.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we demonstrated that the system-level FPGA tools currently available are usable by even non-FPGA experts. They abstract sufficient low-level details of a system design, allowing us to successfully implement a scalable MPSoC to control our laser-based transparency meter. Since we have limited hardware design experience, the ability to update the functionality of the *ADC Interface Units* and the *Command Unit* in software greatly facilitates redesign. Moreover, as we wish to include more ADC units in the future, we benefit from the FPGA's reprogrammability.

The next phase of our work involves the addition of data buffering in the ADC Interface Units and the possible addition of an ethernet interface to the Host PC so that very large data sets (i.e. greater than 8 million samples) can be stored and uploaded to the user within a shorter time frame than possible over the RS232 serial interface. Moreover, methods for improving the accuracy of the laser-based transparency meter will also be investigated, including the development of the feedback portion of the system.

ACKNOWLEDGMENTS

The authors would like to thank Xilinx for providing a XUP Virtex II Pro Board and the CAD tools for this project.

REFERENCES

[1] X. Wang, and M. Leeser, "K-means Clustering for Multispectral Images using Floating-Point Divide," *Proc. IEEE FCCM Symp.*, Apr. '07.
[2] O. Nibouche, "Fast Architectures for FPGA-Based Implementation of RSA Encryption Algorithm," *Proc. FPT'04*, pp. 271-278, Dec. '04.
[3] Y. Gu, and M. Herbordt, " FPGA-Based Multigrid Computation for Molecular Dynamics Simulations," *Proc. IEEE FCCM Symp.*, Apr. '07.
[4] B. Fort, D. Capalija, Z. Vranesic, and S. Brown, "A Multithreaded Soft Processor for SoPC Area Reduction," *Proc. IEEE FCCM Symp.*, pp. 131-142, Apr. '06.
[5] P. Wolkotte, G. Smit, and J. Becker, "Energy-Efficient NoC for Best-Effort Communication," in *Proc. FPL'05*, pp. 197-202, Aug. '05.
[6] A. Patel, M. Saldana, C. Comis, P. Chow, "A Scalable FPGA-based Multiprocessor, " *Proc. IEEE FCCM Symp.*, pp. 111-120, Apr. '06.
[7] AccelChp DSP synthesis tool. Available: http://www.xilinx.com/ise/dsp_design_prod/acceldsp/index.htm
[8] MATLAB. Available: http://www.mathworks.com
[9] Simulink Model-based Design to FPGA Implementation. Available: http://www.mathworks.com/cmspro/req11162.html?eventid=31217
[10] Mentor Graphics Catapult C Synthesis. Available: http://www.mentor.com/products/c-based_design/news/catapult_asic-fpga.cfm
[11] Celoxica Homepage. Available http://www.celoxica.com:
[12] G. Chapman, Y. Tu, J. Dykes, M. Mio, and J. Peng, "Creating direct-write gray-scale photomasks with bimetallic thin film thermal resists," in *Proc. SPIE*, 5256:400-411, '03.
[13] D. Poon, G. Chapman, C. Choo, J. Wang, Y. Tu, and M. La Haye, "Expanding grayscale capability of direct-write photomasks by using modified Bi/In compositions," *Proc. SPIE BACUS Symp. Photomask Tech.*, v5992, pp.59920K1-K11, Sep. '05.
[14] Xilinx Homepage. Available: http://www.xilinx.com
[15] Xilinx's MicroBlaze Processor Reference Guide. Available: http://www.xilinx.com/ise/embedded/mb_ref_guide.pdf