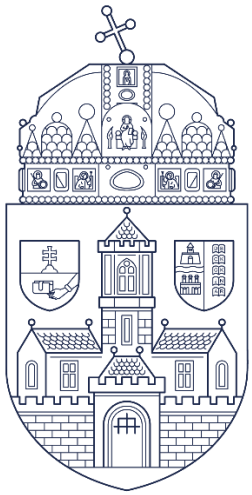
Óbuda University

PhD Thesis Summary



High Performance Image Signal Processing using Field Programmable Gate Arrays

Gábor Szedő Becker

Supervisor:

Róbert Lovas, PhD, habil.

Doctoral School of Applied Informatics and Applied Mathematics

Budapest, 2023.

1 Background

1.1 FPGAs

During the past 40 years Field Programmable Gate Arrays (FPGAs) were first transformed from a uniform sea-of-gates architecture [1] to a heterogenous multiprocessing System-on-a-Chip (SoC) platform. With relentless miniaturization and specialization, modern devices have the capacity and features to implement complex machine vision algorithms [2],[3].

In the first phase of this transition, 18kbit memories (Block RAMs) were added to the devices, followed by 18bit multipliers, and specialized clocking resources. Simple multipliers were expanded to 48-bit Digital Signal Processing (DSP) blocks. The devices were split to tiles, some having specialized functions. With Virtex2-Pro, hardened processors integrated into the Programmable Logic (PL) fabric were offered for the first time [3].

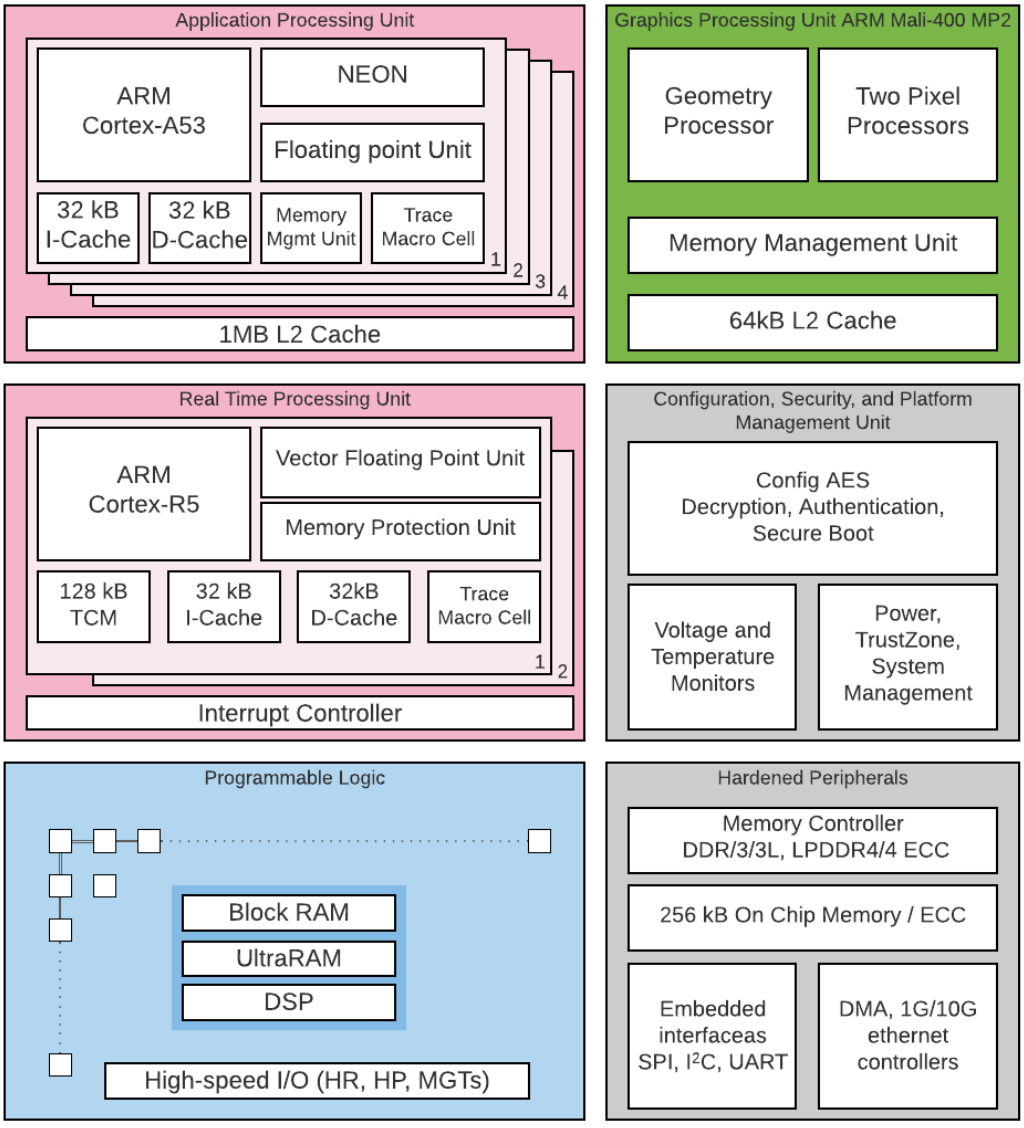


Figure 1: Example block diagram of a modern FPGA device, the Xilinx Zynq Ultrascale+ MPSoC[1]

[1] Source: adapted from https://www.xilinx.com/products/silicon-devices/soc/zynq-ultrascale-mpsoc.html

The latest generation of FPGA devices include multicore, heterogenous processing system (PS, the pink block on Figure 1), GPUs (Green area), and various hardened functions, such as Memory and Network controllers, peripherals, such as I2C, SPI controllers, DMA and watchdog timers. Thousands of DSP engines with single precision floating point units with integrated hierarchical cache memories can implement complex, high performance video processing solutions. Beyond implementing SIMD or MIMD architectures, fine grain configurability enables FPGAs to implement custom DSP and image processing blocks, allowing multiple trade-offs between area, performance, and numerical precision.

1.2 Image Signal Processing

Multi-core, general purpose processors may outperform FPGAs for complex image processing operations. However, for custom, high frame rate camera systems FPGAs remain the best design choice. FPGAs are ideal for implementing demanding image and signal processing algorithms due to the inherent parallelism of logic resources.
Recognition, registration, and reconstruction all need consistent, high-quality images [4]. The purpose of Image Signal Processing (ISP) is to condition images for higher level image and video processing functions. As opposed to other machine vision platforms, such as GPGPU SoCs, like the Nvidia Xavier AGX [5], specialized ASICs, like the Intel Movidius Myriad [6], or mobile platforms, like the Qualcomm 865 [7], FPGAs do not currently have hardened ISP units. While other embedded system processors have dedicated, proprietary ISP pipelines, FPGAs offer PL resources, tightly integrated with a system processor (PS).

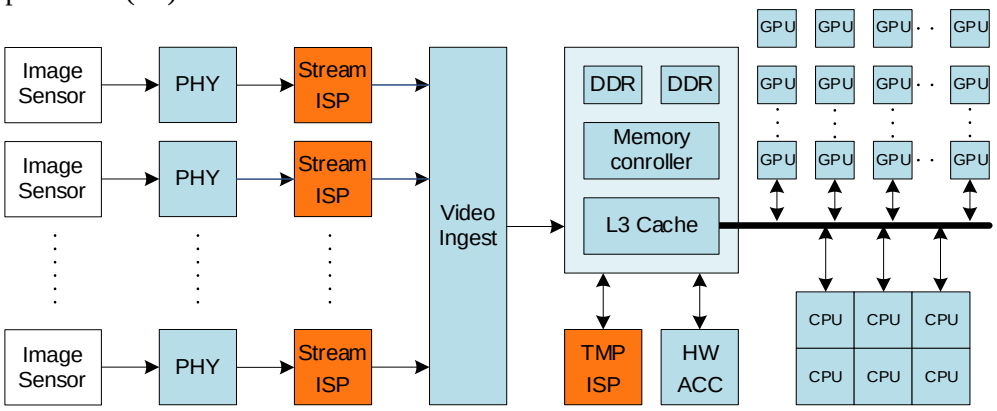


Figure 2: Example implementation of a machine vision platform

Figure 2 presents a typical machine vision platform, with multiple image sensors providing input to processing algorithms executed on an array of GPU and CPU cores. Image sensors typically connect to the platform via MIPI CSI2.0 DPHY or CPHY, sLVDS, or parallel CMOS interfaces. FPGAs provide high-speed I/O features to receive, de-serialize, and deinterleave streams (PHY block on Figure 2).

The spatial ISP functions can be performed directly on the image streams, without having access to previous frames. The Streaming ISP pipeline block can perform:

- Flat-field and pixel defect correction,
- Lens shading, and (limited) distortion correction,
- Bayer Color Filter Array (CFA) interpolation,
- Spatial edge enhancement and noise removal,
- Color correction,
- Tone mapping,
- Video format conversion (chroma resampling, video scaling)

The received frames are subsequently stored in video frame buffers, and are made available to a second set of ISP functions, performing functions, such as:

- Motion adaptive temporal noise reduction,
- Image composition and affine transformation e.g., rotation and/or translation
- Temporal resampling, e.g., frame rate changes or High-dynamic-range (HDR) processing

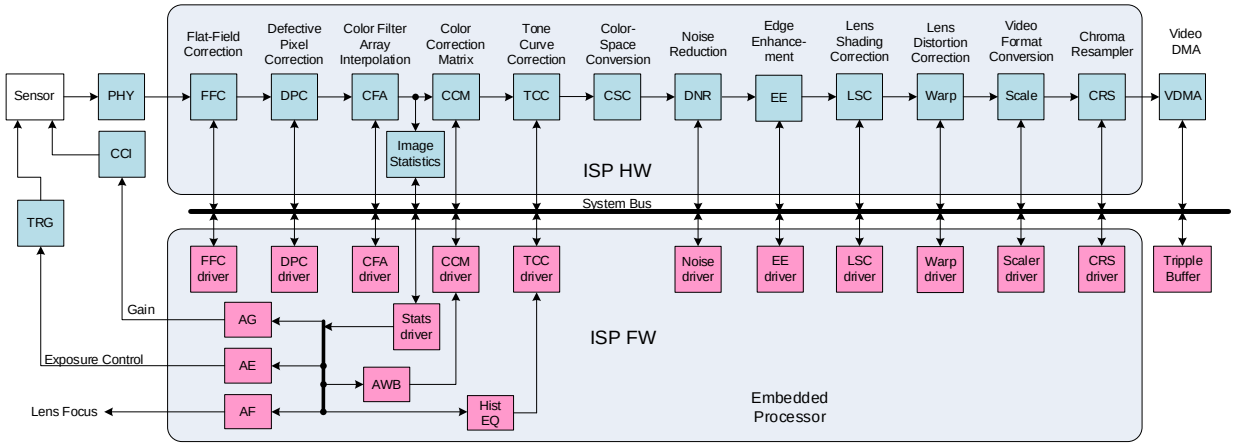


Figure 3: Streaming ISP pipeline HW and FW components

Figure 3 illustrates a typical streaming ISP pipeline, with blue functional units implemented in FPGA PL as custom accelerators, with corresponding drivers and low-level ISP functions (pink blocks) implemented as firmware executed in the FPGA PS.

1.3 Sensor Control

Besides conditioning image streams, ISPs initialize image sensors and adjust processing parameters, such as exposure time, to maintain optimal imaging conditions. Automatic Exposure Control (AE), Automatic White Balancing (AWB), and Automatic Focus (AF) adjustment, often referred to as "3A" [8] are functions the ISP need to perform with minimal latency after image acquisition. Namely, after each exposure, while the frame is being read out from the sensor, image statistics need to be gathered and evaluated. Based on the statistical data, the 3A processing engine needs to set new exposure parameters and control the voice coils of the lens assembly to maintain focus. Similar to the effects of latency introduced to a fed-back control loop, if frames were first committed to a frame buffer, then read back out and analyzed for statistics, the latency introduced would create a system difficult to control.

Image sensors typically contain programmable gain amplifiers, which can scale analog voltages read out from pixel capacitors, before analog to digital conversion (ADC). At low signal levels, a higher gain value may lift the signal out of quantization noise, however, higher gain also amplifies thermal and electric noise in the system. Based on priority (exposure or gain) settings of the camera, the AE algorithm controls exposure time and analog gain (AG) on a frame-by-frame basis. Gain values are programmed via the Camera Control Interface (CCI), which is typically an I2C or SPI register interface. Exposure times may be programmed via the register interface as well, but for stereo cameras performing visual odometry, where precise timing of exposure periods require synchronization of exposure pulse centers, dedicated logic (TRG) is necessary to generate exposure trigger pulses.

1.4 Flat-Field Correction

Flat-field Correction (FFC) aims to remove fixed-pattern image artifacts introduced by non-uniformity of the sensor or the optical system. Some machine vision applications, such as visual odometry and single-pixel airborne object tracking, are extremely sensitive to pixel-to-pixel sensitivity variations. FFC allows the correction of:

- Dark Signal Non-Uniformity (DSNU), pixel-to-pixel differences in dark current,
- Photo Response Non-Uniformity (PRNU), pixel to pixel sensitivity differences,
- vignetting, or lens shading,
- other optical defects, such as light scratches or contamination.

For correcting the DSNU, a set of reference images are collected with no exposure – often with the sensor covered [9]. To correct PRNU, a large set of reference images with uniform illumination are captured, and combined into a single frame, referred as the flat frame [10].
Due to the characteristic, row/column separable nature of DSNU and PRNU, for high-resolution cameras high-frequency pixel to pixel correction is often performed by row and column projections of calibration images [11]. In this case, lens-shading correction is performed by a separate ISP module based on parametric vignetting model of the sensor-lens system. For low resolution cameras where local RAM resources can store a compressed, or uncompressed calibration frame, full pixel to pixel compensation for low and high-frequency defects can be applied. The flat-field corrected image can be given formally as:

$$I_{FFC} = \frac{I - I_D}{I_F - I_D} m - c,, \tag{1}$$

where I is the input image to be corrected, I_D is the dark calibration image, I_F is the flat calibration image, and m and c are constants to adjust the output brightness and black level, respectively.

1.5 Defective Pixel Correction

An image sensor may have defects affecting individual pixels, or clusters. Affected pixels may have significantly altered dark currents (hot pixels), sensitivities (dead pixels), or excessive noise load (flickers) [12]. Sensors failing image quality checks are discarded by manufacturers. Most defects affecting machine vision systems arise during use by high energy particles impacting photodiodes. The purpose of defective pixel correction (DPC) is the identification and interpolation of defective pixels. In static solutions, locations of defects outside the range correctable by FFC are identified during FFC calibration and are stored in a Look-Up-Table. During regular operation sensor images are not analyzed for defective pixels, but defective pixel locations are interpolated on the fly.
Dynamic solutions analyze the sensor data stream and look for outlier pixels, which are apparently stuck or flicker independent of neighboring pixel values and local motion. While dynamic solutions are significantly more complex and often require access to frame buffers, they can keep up with the degradation of the sensor due to the detection of accumulating pixel defects.

1.6 Noise Reduction

1.6.1 Gaussian Noise Removal

Sources of imaging noise in order of significance are: shot-, electronic-, and thermal-noise are de-correlated and can be modelled as additive Gaussian noise[22]. Several effective methods have been adopted widely to denoise images and video streams:

- Spatial filtering techniques, such as bilateral filtering [13],
- Temporal (motion compensated, or motion adaptive) IIR filtering [14],
- Machine learning based methods [15],

While novel FPGAs can implement any of these approaches, for streaming ISPs without access to an external frame buffer, pure spatial filters are the most efficient option.

1.6.2 Rank Order Filtering

Rank order filtering is a non-linear filtering technique, which selects an element from an ordered list of n samples. Median filters, for example, are examples of rank order filters where the element in the middle of the ordered list, with index $\lfloor n/2 \rfloor$, is selected. Compared to linear filters, such as FIR or IIR, rank filters can effectively remove impulse-like noises while preserving high frequency content of the original stream. For image and video processing, rank order filters can be extended to two-dimensional (2D) filters, which remove "salt-and-pepper" noise while preserving the edges of the original image. This can be useful for removing transmission (bit-flip) artifacts, especially as pre-processing for edge detection.
2D filtering takes place on the contents of a rectangular window, which slides across the image. Every time the h pixel high by w pixel wide window is moved by one pixel, h obsolete pixels are discarded, and h new pixels are inserted to the kernel. The filter has to sort hw pixels to generate an output pixel. Due to predominance of comparison operators and subsequent branch misprediction median filtering is not efficiently carried out by GP processors or ALUs.

1.7 Color Processing and Image Enhancements

Color image sensors use a CFA overlay to enable differential spectral sensitivity between adjacent photodiodes in the sensor array. The Bayer CFA is the most commonly used arrangement of color mosaic filters [20], using Red, Green, and Blue (RGB) or Cyan, Magenta and Yellow (CMY) dyes. The mosaic pattern is laid out in repeating RGRG…RG, and GBGBG..GB lines (Figure 4).

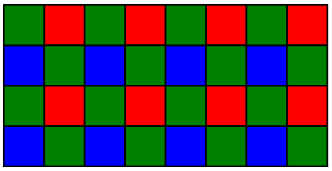


Figure 4. Example of the Bayer RGB Color Filter Array pattern

From one measured color component per pixel, the ISP needs to re-create all 3 color components by interpolating the two missing components using data from neighboring pixels. A critical input to the interpolation process is the phase alignment of the CFA: whether the top line of the image frame contains red or blue pixels, and if the top left pixel is green or not. If CFA interpolation is performed with incorrect assumptions on the color filter positions, the resulting images can be useless due to extreme color artifacts.

The measured color and intensity of light from a small, uniform surface element with no inherent light emission or opacity depend on three functions: the spectral power distribution of the illuminant, the spectral reflective properties of the surface material, the spectral sensitivities of the imager.
The trichromatic aggregate spectral power measured by image sensor pixels inherently mix the native color of objects with the spectral content of the illuminator. White balance estimation is based on heuristics about color content of the scene. Subsequent color correction aims to restore colors that appear correct regardless the illuminant (daylight, incandescent, fluorescent).

Enhancement of features, such as edges and contours, is a typical image pre-processing operation, improving not only the visual appearance of captured images but the accuracy and performance of downstream recognition and classification tasks [16]. Simple approaches like Canny and Sobel edge detectors produce predictable results, but are sensitive to noise, distort texture and unlikely to connect broken edges. Robust algorithms often rely on context from image pyramids or persistence of features across multiple image frames. However, these pre-processing features are seldom available for the streaming ISPs. The enhancement of features in the presence of noisy input continues to remain an area of active research.

1.8 Custom Image Processing Accelerators

Many image processing tasks, such classification, recognition, tracking, stabilization, and optical flow calculation depend on two dimensional searches in image regions. An extensible set of complex instructions, such as two-dimensional correlation or convolution, can be effectively accelerated by performing the search in the frequency domain [16]. In order to facilitate performing these operations on FPGAs, efficient implementation of Fast forward-, and inverse Fourier transforms was an active research area.

1.8.1 Fourier Transform modules

The Fast Fourier transform (FFT) is a computationally efficient algorithm for computing a discrete Fourier transform (DFT), defined as

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn} \ , \ x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn} \ , \tag{2}$$

1.8.2 Phase Factor generation for DFT and FFT

Silicon area in HW implementation of FFTs with more than 2048 points are dominated by memory [39]. Processing Elements (PE) in pipelined FFT processors have designated phase factor generators to provide periodic harmonic sequences. In a loop-engine structure, the single processing node needs all possible twiddle-factor combinations, W_N^{nk} (k>1, $0 \le n$<N/r). Twiddle factors are accessed in a fixed order, meaning the phase increment is constant. This allows recursive calculation of samples, which can lead to significant savings in logic complexity.

1.8.3 Direct Digital Synthesis

The Direct Digital Synthesis (DDS) block introduced by Tierney et.al. [18] consists of a phase accumulator and a phase to sinusoid converter, and generates samples of a harmonic function with frequency

$$f = f_{ref} \frac{\Delta\phi}{2^M}, \tag{3}$$

where $\Delta\varphi$ is the phase increment or frequency control word, M is the width of the phase accumulator and f_{ref} is the frequency of the reference clock. The original DDS does not lend itself well for high point-size, high precision FFTs, as memory size grows exponentially by the width of φ and linearly by the width of the output. Consequently, for low-noise and high period length DDS applications, where an exclusively memory-based solution would be prohibitively large, an under-sampled memory is used with refinement circuitry to interpolate between samples stored in memory.
The quality of the generated harmonic stream can be assessed by computing the Spurious Free Dynamic Range (SFDR),

$$SFDR = 20 \log \frac{A_{signal}}{A_{\max\_spur}}, \tag{4}$$

where A_{signal} is the amplitude of the signal, the strongest spectral component, and $A_{\max\_spur}$ is the amplitude the worst spur, which is the second strongest spectral component.

2 Research Goals

High framerate, high resolution video acquisition-, and pre-processing are areas at the crossroads of high-speed I/O interfaces, and custom, high-speed signal processing. The primary goal of my research was to study, improve, and adapt Image Signal Processing algorithms to FPGAs. Since the beginning of this work in 1996, FPGA platforms, the complexity of image processing algorithms, and performance expectations developed exponentially.
A secondary goal of the research activity was to reduce the barrier of entry using FPGAs as practical machine vision platforms. Back then, and still today, developers and researchers needed an Image Signal Processing pipeline to effectively convey images from an image sensor of their choice to the machine vision hardware accelerator of their design. Imaging SoC developers often keep details of their proprietary ISP subsystems and 3A algorithms trade secrets. To bridge this gap, compact, efficient modules have to be designed, implemented, documented and distributed. My continued research is focused on perfecting and publishing AE, AWB and denoising algorithms using the embedded HW components detailed in the thesis. My goal for AE and AWB was to provide methods which were at par or better than the performance of benchmark platforms at the time, in terms of stability, accuracy and convergence time.

In case of Flat-Field correction, the dependence of DSNU and PRNU of state-of-the-art CMOS image sensors over temperature and sensor gain was analyzed and modeled, to find a compact correction solution that lends itself well for ASIC or FPGA implementation.

For image de-noising median-, and rank-order filters were evaluated for FPGA implementation. While optimization of sorting algorithms was well understood at the time, efficient mapping of one-dimensional median filters to FPGAs were an area of research. We aimed to find a solution to generalize an architecture to 2D rectangular, and non-rectangular filter kernels, which could process streaming video, namely in a single cycle ingest multiple pixels and produce an output pixel.

To accelerate one-, and two-dimensional correlations and convolutions, often carried out in the frequency domain FFTs and IFFTs had to be mapped efficiently to FPGA PL resources. Specifically, scarce local memory resources had to be conserved. My goal on this field was to optimize memory accesses and data movements within parts to ensure compact, efficient architectures. I have evaluated scientific literature looking for alternatives to twiddle factor storage, such as variants of the CORDIC algorithm [19] and DDS [23][24], and pursued improvements to reduce the footprint of twiddle synthesis circuitry. My goal was to design a twiddle-factor generator which is optimal in the following terms:

- generate sin/cos with $N \geq 2^{16}$ samples period length
- SFDR >150 dB
- minimal slice, memory and multiplier count.
- compute both quadrature components simultaneously.
- support twiddle-factor access in reverse order, to support IFFT calculations.

3 Methods of Investigation

Each functional module of the ISP pipeline was thoroughly researched before the candidate implementation most suited for FGPA implementation was selected. Prior art from published conference and journal publications was analyzed for performance and applicability. An excellent example of this is Bayer CFA interpolation, for which numerous algorithmic approaches [20] were implemented using high level languages (Matlab or C++) to evaluate performance. Algorithmic improvements and simplifications were proposed and their effects on algorithmic complexity and image quality were evaluated using the Kodak Image Dataset [21]. The full precision reference model of the most promising prospective algorithm was selected for arithmetic optimization.

A set of optimization techniques were deployed to reduce the complexity of arithmetic operations. Quantization noise along arithmetic pipeline was evaluated to find the optimum resolution of operands while translating floating point to fixed-point operations. Performance of the fixed-point, bit-accurate reference model was evaluated against the full-precision model to quantify noise and image artifacts introduced.

The optimized, fixed-point architecture was implemented in either Verilog, or VHDL hardware Description Languages. Functional verification of the HDL model was performed using logic simulation. A Verilog, or SystemVerilog test-bench was developed, to consume stimuli and golden result files generated by the bit-accurate reference model, and to verify numerical performance. Code and functional coverage targets were established wherever necessary.

Xilinx (now AMD), Altera (now Intel), and Lattice FPGAs were used for the physical implementation of designs. Synthesis reports were used to compare resource usage of the improved ISP modules, and timing reports were consumed to verify operating performance.

To enhance ease-of-use and adaption, Graphical User Interfaces (GUI) were created for the modules. Performance metrics were collected and compiled into a comprehensive datasheet describing parameterization options, typical use scenarios and results.

4 New Scientific Results

Thesis group I: Contributions to ISP Image Quality improvements

Statement 1.1: *I performed in-depth analysis of temperature and analog gain dependence of fixed pattern noise of modern CMOS image sensors. I proposed FFC and ISP architectures, optimized for FPGA or ASIC implementation supporting different noise suppression performance and resource trades.*

Image Sensor nonuniformity and lens system characteristics were known to be temperature dependent. Numerous cameras, especially on the fields of infrared imaging and staring cameras, use multiple calibration images to correct for nonuniformities. My research characterizes the temperature and analog gain dependence of DSNU and PRNU of two contemporary global shutter CMOS image sensors for machine vision applications.

An optimized hardware architecture was proposed to compensate for nonuniformities, with optional parametric lens-shading correction. For both DSNU and PRNU, compensation with one or multiple calibration images, captured at different gain and temperature settings, as well as compensation with interpolated calibration images were evaluated. Based on model results, the effectiveness, external memory bandwidth, power consumption, implementation-, and calibration complexity of different nonuniformity correction approaches were compared.

Own publications pertinent to this statement: [25], [53], [26]

Statement 1.2: *I provided an algorithm and architecture for automatic detection of single-pixel and cluster defects on focal-plane array images, optimized for FPGA implementation.*

In comparison with existing methods, the proposed architecture requires no external frame-buffer access, therefore lends itself well for FPGA and VLSI implementation of streaming ISPs. In contrast with state-of-the-art algorithms, the proposed solution can effectively identify slowly varying temporarily defective pixels. With excellent convergence speed and accuracy, it also supports monochrome-, or color (Bayer CFA) sensors.

Own publications pertinent to this statement: [26], [48]

Thesis group II: Contributions to Color Correction of Image Sensor Streams

Statement 2: *I provided a method for color calibration, an algorithm suitable for low-power embedded processors and its efficient implementation on FPGAs for robust white balancing of image streams.*

The proposed algorithm leverages a hardware accelerator ISP module, implemented in FPGA logic fabric, which calculates focused, luminance weighted 2D Chrominance histograms. Illuminant estimation is performed by real-time classification of 2D chrominance histograms of image frames, comparing against illumination specific signature patterns established offline. Classifier output probabilities are filtered using temporal, adaptive IIR filters, which in turn control the mixing of color-calibration matrixes specific to illuminants. Besides efficient use of FPGA resources and low embedded processor load, one advantage of the proposed method is that correction errors due to illuminant misclassifications are bound, preventing gross distortion of image colors.

Own publications pertinent to this statement: [25], [47], [52], [52]

Thesis group III: Efficient Implementation of Rank Order Filters

Statement 3.1: *I demonstrated that FPGAs are ideal implementation platforms for 1D median filters, and I proposed an architecture that resulted in a compact, high-performance solution.*

The complexity of the ordering operation is strongly affected by the size of the kernel size *N*, with complexity scaling with $\theta(N\log(N))$. General purpose DSP processors exhibit sharply decreasing performance with increasing *N* due to inefficient pipelining and prediction misses caused by data dependent jumps in sorting algorithms. The proposed FPGA implementation can implement real-time median or rank order filters for high-speed video processing applications efficiently with dedicated ISP HW that scales with *N*, processing each input sample in a single clock cycle. Existing bit-serial approaches provide compact, low-footprint solutions, but does not support high dynamic range imaging. Word parallel sorting networks tend to use logic resources inefficiently. The proposed insert-delete architecture stores samples ordered by magnitude, resulting in compact footprint, and high performance even for high-dynamic range images.

Own publications pertinent to this statement: [31], [32], [33], [34]

Statement 3.2: *I proposed an efficient, generalized structure for 2D rank order filtering, capable of processing a configurable number of pixels in a single clock cycle.*

A word serial architecture is presented first, which stores samples in the order they were received. The novel architecture is extended to word-parallel processing multiple input samples per clock cycle, enabling efficient implementation of 2D rank order filters. A solution is presented to optimize the filter by balancing performance (filter clock frequency vs. video clock frequency) and parallelism (number of pixels processed per clock cycle). Use cases of the proposed filter for color image processing, and non-rectangular kernels are presented.

Own publications pertinent to this statement: [28], [29], [30], [40], [49]

Thesis group IV: Efficient Implementation of Fast Fourier Transforms

Statement 4.1: *I proposed methods and architectures to reduce the depth of memory buffers by 50% for in-place FFT implementations by eliminating double buffering.*

2D FFT and IFFT operations are important preprocessing steps for image classification. Local memories for sample and twiddle factor storage are critical resources to be optimized for FPGA FFT implementations. I have shown that with a large number of parallel multipliers and adjacent local memories, FPGAs can implement FFT/IFFT operations with configurable transform lengths efficiently, providing trade-offs between resource allocation and transform time. As point sizes (*N*) increase, HW implementations with $N > 4096$ are dominated by memory allocation. In previous implementations data frames were double buffered for in-place / loop-engine configurations to facilitate random access to samples without output samples overwriting input samples.

Own publications pertinent to this statement: [36], [37], [38], [59]

Statement 4.2: *I proposed a method to reorganize input data to an FFT from natural to digit-reversed order, or results of an FFT from digit-reversed order to natural order without double-buffering frames.*

Decimation In Time (DIT) processing structures expect the input data frame in bit/digit reversed order and generate an output frame in natural order. A Decimation In Frequency (DIF) FFT expects the input frame in natural order, but the output frame is in bit/digit reversed order. To access input and output samples in natural order, a reorganizer stage is necessary, which in previous implementations double-buffered the frame.

Own publications pertinent to this statement: [50], [58]

Statement 4.3: *I proposed a novel Direct Digital Synthesis architecture for sine / cosine synthesis optimized for twiddle-factor generation (W_N^{nk}) for FFTs, using a quadratic differentiator – integrator structure. I proposed a cascaded architecture using a digital resonator as the primary, and a continuous quadratic interpolating secondary stage as a high-quality, compact twiddle-factor source for large ($N>2^{17}$) FFTs.*

Twiddle factors (W_N^{nk}) for FFT computations are traditionally pre-computed and stored in a single LUT with coarse-rotation logic. For 36 bit coefficients, sufficient for most applications, this solution maps efficiently to FPGA BRAMs for $N \leq 2^{11}$. For $N > 2^{11}$, however, BRAM counts grow proportional to N. Targeting these applications, I adapted two existing twiddle factor generation QDDFS architectures for FFTs for optimal implementation in FPGAs. The quadratic differentiator-integrator based QDDFS can supply coefficients with SFDR>410 dB for $2^{12} \leq N \leq 2^{16}$, using only one BRAM. With configurable subsampling- and interpolation length, as well as configurable phase addressing of the BRAM based LUT, the solution can provide twiddle factors for in-place processing engines, dynamically configuring ranks for $N \leq 2^{16}$ FFTs. For larger point-sizes, a bit-serial digital resonator solution can replace the BRAM based LUT to provide samples to be interpolated by the quadratic interpolator stage. The resonator computes 63-bit complex twiddle factors in 128 CLK cycles. The solution demonstrated superior compression ratios and very low noise compared to other methods available, using slice-based logic only allocating less than 550 registers. The cascaded architecture demonstrated SFDR>370 dB and SNR>160 dB for $N \geq 2^{17}$ points. For comparable numerical performance, the CORDIC algorithm requires more than 2000 registers. The direct (LUT) + coarse rotation implementation for this performance level would require 72 BRAMs.

Own publications pertinent to this statement: [39], [56]

5 Practical Applicability of the Results

After an initial period of theoretical exploration of FPGA architectures, practical 2D FFT implementation on Altera Flex10k FPGAs developed in academia were performed for DARPA/ITO under contract DABT63-97-C-0020, at the time to be considered for US defense purposes [36][37].

A significant portion of the ISP module research activity was performed while employed by Xilinx Inc, now part of AMD. Working with inbound marketing on customer interest in FPGA modules to assist with image sensor interfaces, research had to be focused on market driven activities directly tied to specific lead customers. The result of these activities: the Color Correction Matrix[2], CFA interpolation[3], Chroma Re-sampler[4], Video Scaler[5], Tone Mapping[6], and Image Enhancement[7] modules have been assisting Xilinx customers, engineers, and researchers with free, high quality image processing blocks for FPGA image sensor interface implementations.

Optimized DSP modules, such as the complex multiplier [35] and the high speed FFTs were showcased as Technology demonstrators for Virtex devices and continue to exist in Xilinx Beamformer applications.

Reference designs and application notes, such as [8]XAPP953 for the Two-dimensional Rank Order Filter, or the RGB to YCrCb[9] and YCrCb to RGB[10] color space converters helped FPGA users to adopt using the ISP block-set.

My research directly contributed the 1080p60 Camera ISP design to development kits, such the Zynq Video and Imaging Kit[11], released in 2013, also announced by EE Times[12].

While commercial research and development is less conducive to open publication of results at public conferences and journals, during the 12 years at Xilinx I published 15 US patents [45]-[59].

Since 2014 my research contributed to avionics video product development. FPGA design modules were contributed to a multichannel, high-definition, high framerate, mixed mode, custom designed video interface card. The FPGA and corresponding PCB product has been part of ION[13], an avionics video-, audio-, and data recorder currently sold by Shotover Systems.

US Patent 10674063 [44] is a contribution to the synchronization of Time-of-Flight (ToF) cameras currently employed in Amazon Go stores[14].

My work on the field of Auto-Exposure, and Flat-Field correction of CMOS sensors was performed for Amazon Prime Air. The resulting video processing modules and corresponding firmware is being used in second generation delivery vehicles[15]

[2] https://www.xilinx.com/products/intellectual-property/ef-di-ccm.html
[3] https://www.xilinx.com/products/intellectual-property/v_demosaic.html
[4] https://www.xilinx.com/products/intellectual-property/ef-di-chrom-resamp.html
[5] https://www.xilinx.com/products/intellectual-property/1-1bkvlcw.html
[6] https://www.xilinx.com/products/intellectual-property/v_gamma_lut.html
[7] https://www.xilinx.com/products/intellectual-property/ef-di-img-enhance.html
[8] https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.169.2418&rep=rep1&type=pdf
[9] http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.169.1066&rep=rep1&type=pdf
[10] http://notes-application.abcelectronique.com/077/77-42794.pdf
[11] https://www.xilinx.com/support/documentation/application_notes/xapp794-1080p60-camera.pdf
[12] https://www.eetimes.com/image-sensor-color-calibration-using-zynq-7000-soc/
[13] https://shotover.com/products/ion
[14] https://ieeexplore.ieee.org/document/8597403
[15] https://www.aboutamazon.com/news/transportation/amazon-prime-air-delivery-drone-reveal-photos

6 References

[1] Manoel E.de Lima, David J.Kinniment: "Sea-of-gates architecture", *Microelectronics Journal*, Volume 26, Issue 5, July 1995, Pages 431-440. DOI: 10.1016/0026-2692(95)98945-N

[2] Allaoui, R., H. H. Mouane, Z. Asrih, S. Mars, and I. El Hajjouji. "FPGA-based implementation of optical flow algorithm." In *2017 International Conference on Electrical and Information Technologies (ICEIT)*, pp. 1-5. IEEE, 2017. DOI: 10.1109/EITech.2017.8255246

[3] Jammoussi, Ameni Yengui, Sameh Fakhfakh Ghribi, and Dorra Sallami Masmoudi. "Implementation of face recognition system in Virtex II Pro platform." In *2009 3rd International Conference on Signals, Circuits and Systems (SCS)*, pp. 1-6. IEEE, 2009. DOI: 10.1109/ICSCS.2009.5412313

[4] Menke, Jan. "Improving the image quality of contrast-enhanced MR angiography by automated image registration: A prospective study in peripheral arterial disease of the lower extremities." *European journal of radiology* 75.3 (2010): pp 1997-2009. DOI: 10.1109/TMI.2017.2725644

[5] Jetson AGX Xavier series specifications, Nvidia Corporation, https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-agx-xavier/

[6] Intel Movidius Myriad X Vision Processing Unit Specification, Intel Corporation, https://www.intel.com/content/www/us/en/products/sku/204770/intel-movidius-myriad-x-vision-processing-unit-0gb/specifications.html

[7] Qualcomm Corporation, Snapdragon 865 mobile platform product brief, https://www.qualcomm.com/media/documents/files/qualcomm-snapdragon-865-5g-mobile-platform-product-brief.pdf

[8] Jang, Wonwoo, et al. "An image signal processor for ultra-small HD video sensor with 3A in camera phones." *2009 Digest of Technical Papers International Conference on Consumer Electronics*. IEEE, 2009. DOI: 10.1109/ICCE.2009.5012325

[9] European Machine Vision Association, "EMVA Standard 1288 Standard for Characterization of Image Sensors and Cameras", https://www.emva.org/wp-content/uploads/EMVA1288-3.1a.pdf

[10] David Perry, Eustace Dereniak, "Linear theory of nonuniformity correction in infrared staring sensors", *SPIE Optical Engineering 32*(8), August 1993. DOI: 10.1117/12.145601

[11] Zhu, Yajun, et al. "A 160× 120 ROIC with Non-uniformity Calibration for Silicon Diode Uncooled IRFPA." *2019 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)*. IEEE, 2019. DOI: 10.1109/EDSSC.2019.8754184

[12] The International Standard on Ergonomic Image Quality Requirements, pp 2-3: https://standards.iteh.ai/catalog/standards/cen/346aee7a-22c1-466a-9acf-622f79ee1937/en-iso-13406-2-2001

[13] Tomasi, Carlo, and Roberto Manduchi. "Bilateral filtering for gray and color images." *Sixth international conference on computer vision* (IEEE Cat. No. 98CH36271). IEEE, 1998. DOI: 10.1109/ICCV.1998.710815

[14] Robert Szeliski, James Coughlan, "Spline-Based Image Registration." *International Journal of Computer Vision* 22, 199–218 (1997). DOI: /10.1023/A:1007996332012

[15] Juntao Guan, et al. "Fixed pattern noise reduction for infrared images based on cascade residual attention CNN." *Neurocomputing* 377 (2020): 301-313. DOI: 10.1016/j.neucom.2019.10.054

[16] Pablo Arbeláez, Michael Maire, Charless Fowlkes and Jitendra Malik, "Contour Detection and Hierarchical Image Segmentation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898-916, May 2011. DOI: 10.1109/TPAMI.2010.161

[17] Abhijit Mahalanobis, Vijaya Kumar, and S. Richard F. Sims. "Distance-classifier correlation filters for multiclass target recognition." *Applied Optics* 35.17 (1996): 3127-3133. DOI: 10.1364/AO.35.003127

[18] Joseph Tierney, Charles Rader, and Bernard Gold. "A digital frequency synthesizer." IEEE transactions on Audio and Electroacoustics 19.1 (1971): 48-57. DOI: 10.1109/TAU.1971.1162151

[19] Ray Andraka, "A survey of CORDIC algorithms for FPGA based computers." *Proceedings of the 1998 ACM/SIGDA sixth international symposium on Field programmable gate arrays*. 1998. DOI: 10.1145/275107.275139

[20] Daniele Menon, and Giancarlo Calvagno. "Color image demosaicking: An overview." *Signal Processing: Image Communication* 26.8-9 (2011): 518-533. DOI: i:10.1016/j.image.2011.04.003

[21] Mina Rafinazari and Eric Dubois "Demosaicking algorithm for the Kodak-RGBW color filter array", *Proceedings of the SPIE 2015 International Conference on Color Imaging, Displaying, Processing, and Applications*, 939503 (8 February 2015); DOI: 10.1117/12.2083667

[22] Igor Brouk, Amikam Nemirovsky, Kamal Alameh, Yael Nemirovsky: "Analysis of noise in CMOS image sensor based on a unified time-dependent approach", Solid-State Electronics, Volume 54, Issue 1, January 2010, Pages 28-36. DOI:10.1016/j.sse.2009.09.003

[23] Dengwei Fu, Willson, A.N. Jr., "A high-speed processor for digital sine/cosine generation and angle rotation", *Conference Record of the Thirty-Second Asilomar Conference on Signals, Systems & Computers*, 1-4 Nov. 1998, vol 1, pp:177 – 181. DOI: 10.1109/ACSSC.1998.750849

[24] J.M. Pierre Langlois, Dhamin Al-Khalili, "Novel Approach to the Design of Direct Frequency Synthesizers Based on Linear Interpolation", *IEEE Transactions on Circuits and Systems – II: Analog and Digital Signal Processing,* vol. 50, No 9, September, 2003. DOI: 10.1109/TCSII.2003.815020

6.1 Own publications related to the PhD thesis

[25] Gabor Szedo Becker, Róbert Lovas, "Uniformity Correction of CMOS Image Sensor Modules for Machine Vision Cameras", *Sensors* vol 22. no. 24: 9733. 2022. DOI: 10.3390/s22249733

[26] Gabor Szedo Becker, "Low-Cost Spatio-Temporal Algorithm for Defective Pixel Identification and Correction", in Proceedings of the *2022 IEEE 19th International Symposium on Intelligent Systems and Informatics (SISY)*, September 2022, accepted for publication

[27] Gabor Szedo, Steve Elzinga, Greg Jewett: "Image Sensor Color Calibration Using the Zynq-7000 SoC", *Xcell Magazine, Issue 81, Fourth quarter* 2012

[28] Gábor Szedő, Péter Szántó: "Cost-Effective Two-Dimensional Rank-Order Filters on FPGAs", *Xilinx DSP Magazine*, Apr 2007, pp 34-37

[29] Gábor Szedő, Péter Szántó: "Cost-Effective Two-Dimensional Rank-Order Filters on FPGAs", *DSP Design Online Magazine*, September 20, 2007, http://www.dspdesignline.com/showArticle.jhtml?printableArticle=true&articleId=201807345

[30] Péter Szántó, Gábor Szedő, Béla Fehér: "High Performance Timing-Driven Rank Filter", *ICECS 2006, 13th IEEE International Conference on Electronics, Circuits and Systems. Special Issue, IEEE, 2006.*pp 752-755, DOI: 10.1109/ICECS.2006.379898

[31] Gábor Szedő, Béla Fehér: "High Speed FPGA implementation of Median Filters" *NDES'98 Nonlinear Dynamics of Electronic Systems*, Technical University of Budapest, Hungary 16-18. July 1998 Proceedings, pp. 191-193.

[32] Gábor Szedő, Béla Fehér: "Noise Filter Applications on Reconfigurable Hardware", *PACT '98. International Conference on Parallel Architectures and Compilation Techniques, Workshop on Reconfigurable Computing*, Paris, France October 13. 1998. Proceedings, pp. 95-100

[33] Gábor Szedő: ”Noise Filter Applications on Reconfigurable Hardware”, *XI. Conference on Application of Microprocessors in Automatic Control and Measurement*, Warsaw, Poland 13-14. October, 1998. Proceedings,Vol 1. pp. 3-11

[34] Béla Fehér, Gábor Szedő: ”Nonlinear Filters on Reconfigurable Processors”, *Tempus Symposium on Intelligent Systems in Control and Measurement*, Miskolc, Hungary, 21-27. November, 1998. Proceedings, pp. 162-167

[35] Béla Fehér, Gábor Szedő: “Cost effective 2x2 inner product processors”, *Field Programmable Logic and Applications*, 8th International Workshop, FPL’98, Tallin, Estonia, 3-5 September 1998. Proceedings, pp. 348-355.

[36] Gábor Szedő, Ted Bapty, Sandeep Neema, Jason Scott: “Reconfigurable Target Recognition System” *Eighth ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, FPGA 2000, February 10-11, Monterey, CA, Proceedings pp 221.

[37] Sandeep Neema, Jason Scott, Theodore Bapty, Gabor Szedo, Bela Feher, "Real Time Reconfigurable Image Recognition System", *18th IEEE International Conference on Instrumentation and Measurement Technology*, May 21-23, 2001, Proceedings pp. 351-355

[38] Gabor Szedo; Vanessa Yang; Chris Dick: “High-performance FFT processing using reconfigurable logic”, *2001. Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers*, 4-7 Nov. 2001, Proceedings, pp:1353 - 1356 vol.2

[39] Gábor Szedő: “A Compact High Resolution Phase-Factor Generator for Large Point-Size FFTs” *Global Signal Processing Conference on Pervasive Signal Processing, GSPx 2005* Oct 24-27 2005, Santa Clara, CA USA, Proceedings on CD ROM, track: FPGA based DSP

[40] Péter Szántó, Gábor Szedő, Béla Fehér: “Implementing 2D Median Filters in FPGAs”, *Transactions of the University of Mining and Metallurgy of Ostrava*. (1210-0471 1804-0993): 1 pp 179-184 (2006) ICCC 2006

[41] US Patent 11321873, “Calibrating and detecting vibration of stereo ranging systems”, Filed: February 4, 2019, Granted May 3, 2022, Inventors: Dr. Joachim Bauer, Gabor Szedo Becker

[42] US Patent 11076085, “Detecting interference between time-of-flight cameras using modified image sensor arrays”, Filed: June 20, 2018, Granted: July 27, 2021, Inventor: Gabor Szedo Becker

[43] US Patent 10812777, “Rolling shutter motion and depth sensing”, Filed: Oct 31, 2018, Granted: Oct 20, 2020, Inventors: Gabor Szedo Becker, Chengwu Luke Cui, Anirudth Nambirajan

[44] US Patent 10674063, “Synchronizing time-of-flight cameras”, Filed: Jun 20, 2018, Granted: Jun 2, 2020. Inventor: Gabor Szedo Becker

[45] US Patent 9305362, “Image Stabilization”, Filed: Feb 28, 2014, Granted: Apr 5, 2016: Inventors: Gabor Szedo, Christopher J. Martin, Ted N. Booth

[46] US Patent 9237257, “Circuits for and methods of generating a digital image”, Filed: June 14, 2013, Granted Jan 12, 2016: Inventors: Gabor Szedo, Vanessa Y. Chou

[47] US Patent 9013611, “Method and device for generating a digital image based upon a selected set of chrominance groups”, Filed Sept 6, 2012: Granted: Apr 21, 2015, Inventors: Gabor Szedo, Steven P. Elzinga, Jose R. Alvarez

[48] US Patent 8774544, “Determining outlier pixels of successive frames”, Filed: Jul 23, 2010, Granted: Jul 8, 2014, Inventors: Gabor Szedo, Jeffrey D. Stroomer, Jose R. Alvarez

[49] US Patent 8713082, “Scalable Architecture for Rank Order Filtering”, Filed on Mar 2, 2007, Granted Apr 29, 2014, Inventors: Peter Szanto, Bela Feher, Gabor Szedo

[50] US Patent 8572148, “Data reorganizer for Fourier transformation of parallel data streams”, Filed: Feb 23, 2009, Granted Oct 29, 2013: Inventors: Gabor Szedo, Hemang Parekh

[51] US Patent 8484267, “Weight normalization in hardware without a division operator”, Filed: Nov 19, 2009, Granted: July 9, 2013, Inventor: Gabor Szedo

[52] US Patent 8441562, “Color filter array alignment detection”, Filed: Sept 2, 2010, Granted: May 14, 2013, Inventors: Gabor Szedo, Jose R. Alvarez

[53] US Patent 8400533, “Methods of reducing aberrations in a digital image”, Filed: Feb 23, 2010, Granted: Mar 19, 2013, Inventors: Gabor Szedo, Jose R. Alvarez

[54] US Patent 8181149, “Interface for managing multiple implementations of a functional block of a circuit design”, Filed: Sept 3, 2009, Granted: May 15, 2012, Inventors: Sean A. Kelly, Gabor Szedo

[55] US Patent 8005881, “Scalable architecture for rank order filtering”, Filed: Mar 2, 2007, Granted: Aug 23, 2011, Inventors: Peter Szanto, Gabor Szedo, Bela Feher, Wilson C. Chung

[56] US Patent 7984091, “Quadratic Approximation for Fast Fourier Transform”, Filed: Oct 14, 2005, Granted: July 19, 2011, Inventors: Gabor Szedo

[57] US Patent 7684968, “Generation of a high-level simulation model of an electronic system by combining an HDL control function translated to a high-level language and a separate high-level data path function”, Filed: Dec 9, 2004, Granted: Mar 23, 2010, Inventors: Gabor Szedo, Singh Vinay Jitendra, L. James Hwang

[58] US Patent 7669017, “Method of and Circuit for Buffering Data”, Filed Sept 27, 2006, Granted: Feb 23, 2010, Inventors: Hemang Parekh, Helen Tarn, Gabor Szedo, Vanessa Chou, Jeffrey Graham, Beth Cowie

[59] US Patent 7395293, “Memory segmentation for Fast Fourier Transform”, Filed: July 23, 2004 Granted: Jul 1, 2008, Inventors: Gabor Szedo, Helen Hai-Jo Tarn

[60] Gábor Szedő: “FPGA Based Reconfigurable Logics” *Minisymposium, Techical University of Budapest Department of Measurement and Instrumentation*, Feb. 1997, Proceedings, pp. 38-39.

[61] Gábor Szedő: “Potentials of Dynamic Reconfiguration”, *Conference on the latest results of information technology*, Technical University of Budapest department of Process Control, 1997 Proceedings pp. 14-15.

[62] Szedő Gábor, Fehér Béla: “Nagy sebességű medián szűrők megvalósítása FPGA áramkörök segítségével”, Mérés-Automatizálás'98 Konferencia az Alkalmazott Informatika Jegyében 1998. November 3-5.Proceedings, pp. 39-47

[63] Gábor Szedő: “An Application Example of Reconfigurable Logics”, *Minisymposium, Technical University of Budapest department of Measurement and Information Systems*, February 1998 Proceedings pp. 38-39.

[64] Gábor Szedő: “An Application of Reconfigurable Hardware: the Finite Element Model” *microCAD'98 International Computer Science Conference*, February 25-26, 1998. Proceedings, Section E. pp. 43-45.