

IR detection module with integrated real-time FIR filter implemented in FPGA

Krzysztof Achtenberg, Janusz Mikołajczyk, and Zbigniew Bielecki

Abstract—Infrared detectors are usually characterized by $1/f$ noise when operating with biasing. This type of noise significantly reduces detection capabilities for low-level and slow signals. There are a few methods to reduce the influence of $1/f$ noise, like filtering or chopper stabilization with lock-in. Using the first one, a simple 1st-order analog low-pass filter built-in amplifier usually cuts off $1/f$ noise fluctuations at low frequencies. In comparison, the stabilization technique modulates the signal transposing to a higher frequency with no $1/f$ noise and then demodulates it back (lock-in amplifiers). However, the flexible tuned device, which can work precisely at low frequencies, is especially desirable in some applications, e.g., optical spectroscopy or interferometry. The paper describes a proof-of-concept of an IR detection module with an adjustable digital filter taking advantage of finite impulse response type. It is based on the high-resolution analog-to-digital converter, field-programmable gate array, and digital-to-analog converter. A microcontroller with an implemented user interface ensures control of such a prepared filtering path. The module is a separate component with the possibility of customization and can be used in experiments or applications in which the reduction of noises and unexpected interferences is needed.

Keywords—IR detector; FIR; FPGA; DSP; noise; filtering

I. INTRODUCTION

IN recent years the rapid development of infrared technology is still observed. This applies to infrared detector production technology and its utilization in specific applications. Infrared IR sensors work in systems like fire detection, biometrics, spectroscopy, thermography, motion control, or security [1–4]. In most cases, they should have high sensitivity. However, the fundamental limit depends on both responsivity and noise levels determined by the detector’s construction and work conditions [5]. The shape of the noise spectrum can be divided into a low-frequency range where $1/f$ noise occurs (also named a flicker noise) and white noise at high frequencies. It can be seen that the fundamental way to minimize the noise influence is based on a selection of the operation frequency. Nevertheless, in many applications, there is no possibility to use high-frequency optical signals, e.g., optical spectroscopy, interferometry, crystal research and technology, infrared thermography, and laser wavelength stabilization [6].

That is why, some techniques are designed to reduce $1/f$ noise influence for low-frequency signals. For example, a stabilization technique with lock-in amplification can be used to

extract a signal from the noise, but the modulation frequency and phase must be known [7, 8]. Moreover, the lock-in instrumentation is complex and expensive [9]. Analog filtering is an effective method, but it is challenging to project flexibly tunable in a wide frequency range bandpass filter which can take various forms. The detector cooling also restricts the $1/f$ noise component and increases detectivity. However, a decrease in its temperature to 200 K (using a multistage thermoelectric cooler) is still insufficient to detect signals that cause millivolts changes at detection module output.

Our paper presented a novel concept of an IR detection module architecture equipped with an integrated digital filtering block. The use of digital filters has a few advantages in comparison with analog ones. First, the attenuation in the band-stop frequency range and the rate of the frequency-response characteristics decrease are much better with digital filters. Moreover, the digital filter offers a better compromise between the ringing effect, signal attenuation in the band-stop range (ex., in the case of the Chebyshev type), and a simple reconfiguration procedure. Additionally, high-order analog filters are more complex and require strictly defined tolerances of passive components.

In comparison, the main limitations of digital filters are due to hardware capabilities (processing speed) and coefficient precision [10]. For example, the number of available multiplication blocks in the use of field-programmable gate arrays (FPGA) determines the filtering limits.

Digital filters based on digital signal processing (DSP) techniques are widely used in many scenarios and applications like radio signal detection and ranging, e.g., in software-defined radio, cell phone communication systems, and satellite receivers [11–15]. Such filters can be implemented on various hardware like FPGA, microcontrollers (μ C), system-on-chip (SoCs), or application-specific integrated circuits (ASICs) [16–19].

Applying a finite impulse response filter (FIR) in the IR module defines its desired frequency response and cut-off frequencies by the values of its coefficients. The prepared procedures and components configuration ensures both the detection of “low-level” optical signals and the reconstruction of their electrical shape.

The module operation was simulated using calculated frequency responses and tested in a lab setup. Although generally available and low-cost components were used in the module prototype, the studies have shown its performance in detection of

This work was supported by Military University of Technology grant no. UGB 22-871.

All Authors are with Institute of Optoelectronics, Military University of Technology, Warsaw, Poland (e-mail: krzysztof.achtenberg@wat.edu.pl – corresponding author).



modulated signals in spectra range of $1/f$ noise with bandpass filtration (BPF). To our knowledge, no literature describes such a concept of an optical detection module with its possibilities and limitations.

II. MATERIALS AND METHODS

The main objective of this work was to define the performances of a “digitally filtered” IR detection module. Using this module, we read out the signal from the detector, sample it, provide the desired filtering operation, and send the result to the DAC block equipped with an output correction filter. The simplified schematic diagram of the module construction is shown in Fig. 1. In the photoreceiver, the mercury cadmium telluride (MCT) photoconductive detector optimized for 9 μm wavelength was used [20, 21]. An increase in detectivity was obtained using a four-stage thermoelectric cooler TEC and an immersion lens [22-24]. The intensity and specific corner frequency of $1/f$ noise depends on the applied bias and temperature. With a relatively low detector sheet resistance, the total photoreceiver noise is also contributed by designed read-out electronics.

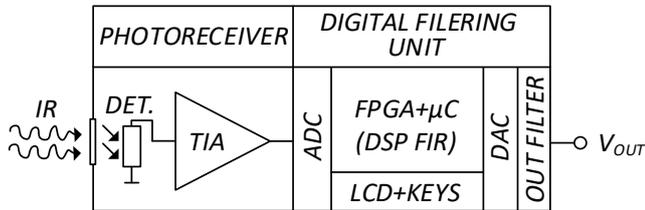


Fig. 1. Simplified block diagram of proposed IR detection module: TIA – transimpedance amplifier, DET – HgCdTe photoconductive infrared detector, μC – microcontroller

The two-stage electronics are built using low-noise bipolar-input AD797 opamps, with voltage noise of $0.9 \text{ nV}/\sqrt{\text{Hz}}$ at 1 kHz and about 110 dB of open-loop gain. It consists of a transimpedance amplifier TIA and voltage amplifier VA, respectively. An AC amplifiers' coupling (filter $R_A C_A$, $f_c=10 \text{ Hz}$) was used to eliminate the DC detector bias signal ensuring the sensor's dynamic range. The V_B bias voltage is filtered by a $R_1 C_1$ low-pass filter LPF ($f_c = 1 \text{ mHz}$). The gain G was set to 101 kV/A by TIA's feedback resistor and the VA's gain resistors (R_2 and R_3). All analog circuits use a linear power supply to avoid interferences from switching converters. The scheme of the read-out electronics is shown in Fig. 2.

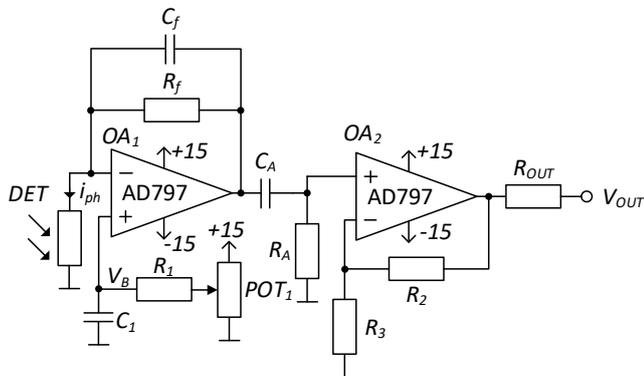


Fig. 2. Schematic diagram of the detector read-out circuit

The filtered output signal V_{OUT} is proportional to the infrared radiation power with the coefficient G :

$$V_{OUT} = i_{ph} G, \quad (1)$$

where i_{ph} is the registered photocurrent signal determined by beam power and detector responsivity.

Assuming that noise from the VA is negligible (comparing high transimpedance of TIA) and DC voltage from bias (V_B/R_{DET}) is removed by the AC filter, the output noise voltage can be expressed as:

$$V_{n_{out}} = R_f G \left(+ \frac{i_{n_{DET}} + i_{n_{OA1}} + i_{n_{Rf}}}{(R_{DET} || R_f)} + \frac{e_{n_{VB}}}{(R_{DET} || R_f)} \right), \quad (2)$$

where R_{DET} is detector resistance, $e_{n_{VB}}$ is voltage noise of bias source V_B , $i_{n_{DET}}$ is the detector noise current, $i_{n_{Rf}}$ is R_f resistor noise current, $i_{n_{OA1}}$ and $e_{n_{OA1}}$ are current and voltage noises of the opamp used in the TIA, respectively.

In the developed IR detection module, we used precision data converters (analog-to-digital ADC and digital-to-analog DAC). Using the prepared structure, we digitized the amplified signal from the photoreceiver, filtered the digital signal, and converted it into the analog one at the detection module output.

High-resolution 16-bit ADC (AD7606) samples the output signal with $\pm 5\text{V}$ set input range ($\text{LSB} \approx 153 \mu\text{V}$). This model is then characterized by 89 dB (typ.) of signal-to-noise ratio SNR and $\pm 0.5 \text{ LSB}$ of differential and integral nonlinearity (DNL and INL). It has also a built-in anti-aliasing filter (second-order Butterworth low-pass filter) with relatively flat response characteristics up to 10 kHz. When the input range is set to $\pm 5 \text{ V}$, this filter's -3 dB cut-off frequency is equal to 15 kHz [25]. Low-noise linear voltage regulator (model ADM7150) sets the supply and reference voltage for ADC. It has $1.6 \mu\text{V}$ RMS of the total noise integrated from 10 Hz to 100 kHz and $\pm 1\%$ voltage accuracy. The main core of the processing system is implemented in FPGA. Our prototype used a chip belonging to the ARTIX™7 family (XC7A35T in FTG256 package) from Xilinx. It has 33,280 logic cells, 5,200 slices, 600 kB of maximum distributed RAM, and 90 DSP slices, which are especially useful for implementing filtering functions [26]. The logic architecture of filtering, conversion, and microcontroller communications was written in VHDL. The synthesis, simulations, implementation, and programming were performed using dedicated Vivado software. The FPGA works with a 50 MHz clock frequency and a 100 kHz sampling frequency defined by the $10 \mu\text{s}$ - limit of DAC settling time. The communications (FPGA – data converters, FPGA – microcontroller) use 16-bit parallel interfaces. The schematic diagram of the described DSP signal trace for the detection module is shown in Fig. 3.

The FIR structure implemented in the FPGA ensures a flexible and stable digital filter configuration [10, 27-28]. Its operation algorithm is visualized in Fig 4. The 16-bit two-complement format samples $x(t)$ from the ADC are entering the 89-tap pipeline register in the FPGA. Each tap value is multiplied by a 16-bit fixed point b_x coefficient. All multiplications' products are summed, and the result $y(t)$ is sent to the DAC.

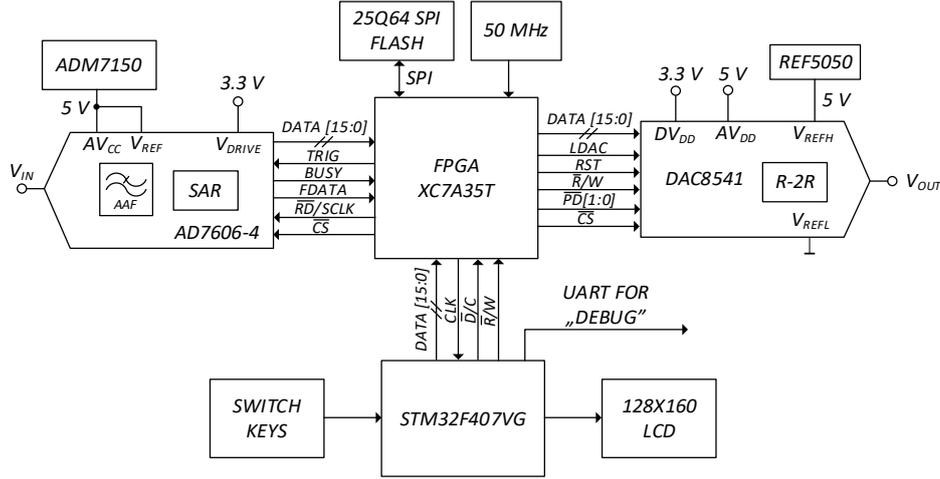


Fig. 3. Schematic diagram of the DSP signal trace for the detection module

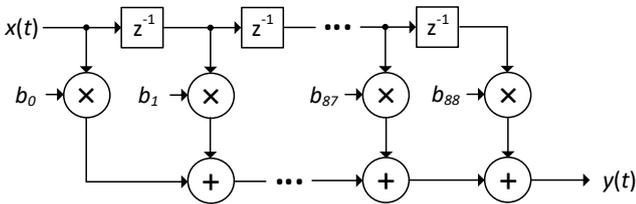


Fig. 4. Schematic diagram of the 89-tap FIR filter structure

In practice, the detection module can be equipped with a low-pass, high-pass, or bandpass filter. The selection of filter type directly defines the set of calculated coefficients. In our device, the coefficients are computed in 32-bit ARM® microcontroller STM32 (STM32F407VGT6) using DSP libraries (for \sin and \cos functions) and a built-in floating-point unit (FPU). The LPF discrete impulse response $h[i]$ (number of filter coefficient i) is calculated by a sinc -type Blackman windowed function from the formula [29]:

$$h[i] = \begin{cases} K \frac{\sin\left(\frac{2\pi f_c(i - \frac{M}{2})}{i - \frac{M}{2}}\right)}{i - \frac{M}{2}} [0.42 - 0.5 \cos\left(\frac{2\pi i}{M}\right) + \\ 0.08 \cos\left(\frac{4\pi i}{M}\right) \text{ for } i \neq \frac{M}{2}, \\ 2\pi f_c K \text{ for } i = \frac{M}{2}, \end{cases} \quad (3)$$

where: M is the filter length (number of coefficients), and K is the constant selected to obtain a gain of 1 at the zero frequency. In the case of sinc windowed filters, the cut-off frequency f_c remains the gain of 0.5 (instead of 0.707). For a high-pass filter, the impulse response is created using spectrum inversion of the low-pass impulse response. The bandpass filter is formed by combining low-pass and high-pass impulse responses (which constitute a band-stop filter) and converting using the spectrum inversion method. In practice, this inversion is created by reversing the sign of all coefficients and modifying the middle ($M/2$) coefficient. The calculated coefficients have a floating-point representation, and conversion is performed before sending to a 16-bit representation FPGA. Signal amplification is observed because the sum of the coefficients cannot be 1 for integer numbers. It is a significant effect for the LPF in which most integer type coefficients have high positive values, so the gain factor is also relatively high. We proposed two methods to

minimize this effect. The first is based on rescaling the coefficients using calculated gain at the desired frequency value placed at the flat range of filter response characteristics and the FPGA bit-shifting. This method decreases precision by dividing all coefficients by the gain value as a specific number with a fraction, e.g., dividing an integer number by non-integer floating point one. Owing to numbers rounding, the impulse response characteristic can be distorted. The second method calculates gain at the desired frequency and considers the correct value after the DAC output signal examination. These methods use a Discrete Fourier Transform (DFT) of the filter impulse response to define the gain magnitude at the desired frequency f [30]:

$$|H(f)| = \frac{1}{\sqrt{(\sum_{i=0}^{M-1} h[i] \cos(2\pi f i))^2 + (\sum_{i=0}^{M-1} h[i] \sin(2\pi f i))^2}}, \quad (4)$$

where f is expressed as a fractional sampling frequency. Then the coefficients are normalized to the frequency f (divided by the calculated $|H(f)|$):

$$h[i] \leftarrow \frac{h[i]}{|H(f)|}. \quad (5)$$

The filter settings, e.g., filter type, cut-off frequencies, and gain, are defined by the user and visualized on the LCD. The register-based table prepared in the FPGA contains calculated coefficients. The utilization of available XC7A35T FPGA resources used for this project implementation is shown in Fig. 5.

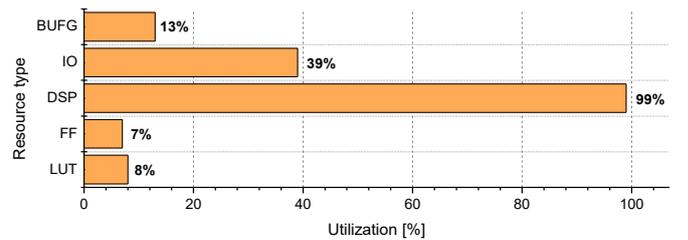


Fig. 5. The utilization of XC7A35T FPGA resources for implementation of 89-tap FIR filter with 16-bit fixed point coefficients. The maximum possible number of DSP blocks has been used (odd number for symmetric impulse response)

During the designed final procedure, the filtered digital signal is converted into the analog one using a 16-bit resolution DAC8541 and buffered output voltage stage [31]. The DAC reference voltage is supplied by a low noise ($3 \mu\text{V}_{\text{pp}}$ in 0.1 to 10 Hz range), very-low drift ($8 \text{ ppm}/^\circ\text{C}$), and a precision (0.2 % accuracy) source, model REF5050. The applied two-stage output signal conditioning unit (SCU) consists of a voltage amplifier, low-pass anti-aliasing, and a restoring filter. The amplifier compensates the difference in data converter ranges ($\pm 5 \text{ V}$ vs. $0\text{-}5 \text{ V}$ range) using JFET input opamp, model TL071ID (from Texas Instruments), in a non-inverting configuration. The filter is used to attenuate glitches and zero-order memory DAC phenomena [32]. It was built of OPA2140 opamps as the 4-th order Bessel type-LPF in the Sallen-Key configuration. Its -3dB cut-off frequency was set to 17 kHz with a stopband of -25 dB at 50 kHz. The schematic of the SCU is shown in Fig. 6.

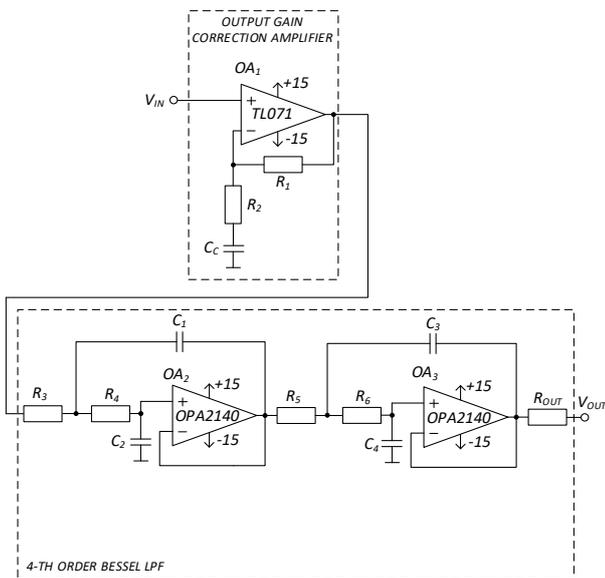


Fig. 6. Schematic diagram of the signal conditioning unit

The described detection module was tested by performing analyses of its components (photoreceiver and digitally-filtering unit) and examining the efficiency of signal filtering by reduction of the photoreceiver's noise influence in the developed IR detection module.

III. RESULTS

A. Photoreceiver

Analyzing the FIR filter influence on the detection module features requires measuring the noise power spectral density (PSD) of the photoreceiver. The detector was not irradiated during this test, cooled to 200 K, and 1 V-biased. Measurements were performed using a high-precision FFT spectrum analyzer. The registered characteristics are shown in Fig. 7. Low-frequency noise (up to about 10 Hz) was attenuated due to the applied AC coupling ($R_A C_A$ in Fig. 2). For higher frequencies, there is observed $1/f$ type noise. The integration of noise spectrum density can calculate the total noise signal:

$$v_{\text{rms}} = \sqrt{\int_{f_1}^{f_2} (e_n)^2 df}. \quad (6)$$

For example, we obtained a noise value of $7.3 \text{ mV}_{\text{rms}}$ from $f_1 = 1 \text{ Hz}$ to $f_2 = 100 \text{ kHz}$ (corresponding to about $17.5 \text{ mV}_{\text{p-p}}$).

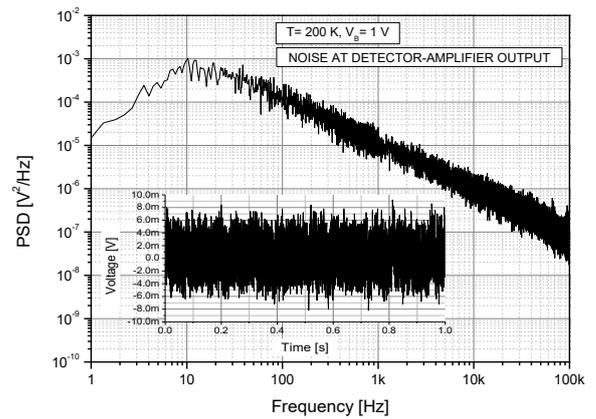


Fig. 7. Noise PSD measured at the output of the photoreceiver

B. FIR filter

The correct operation of the FIR filter was rated based on its coefficient calculations. It was performed by comparing results obtained from the STM32 μC with implemented equations (practical aspects) and MATLAB ones (theoretical aspects). All coefficients were simulated based on amplitude and phase responses using MATLAB ("custom debugging"). In the experimental part of the work, we measured the frequency and phase responses of some filters implemented in the FIR using the dedicated setup shown in Fig. 8. It consists of a tracking generator (Signal Hound TG44A) and a spectrum analyzer (Signal Hound SA44B).

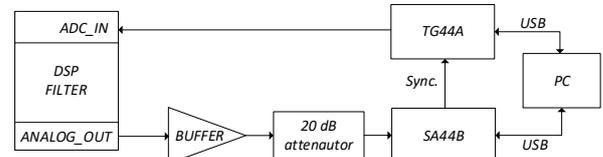


Fig. 8. The frequency response measurement set-up

Because of the spectral noise characteristics of the photoreceiver (high level of $1/f$ noise), the preliminary tests were performed for the LPF coefficients with the cut-off frequency (-6 dB) of 1 kHz [33]. This test's results for three coefficient sets (impulse response) and without the gain normalization procedure are shown in Fig. 9.

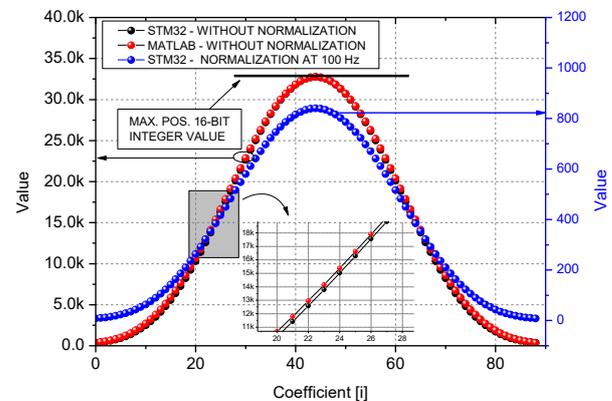


Fig. 9. Coefficient values (impulse response) calculated with MATLAB and the STM32 for the LPF cut-off frequency at 1 kHz

The sets correspond to coefficients calculated in MATLAB using floating point representation without the normalization (red circles), in microcontroller using the floating point extended to a 16-bit integer without the normalization (black circles), and in STM32 μ C using the floating point extended to a 16-bit integer with normalization (blue circles). A significant reduction in coefficient values is observed for the LPF with the normalization at 100 Hz. This procedure decreases precision by comparing results obtained by the maximum stretching to 16-bit representation. However, the input and output signal amplitudes should be equal at the normalization frequency, and the gain is calculated using the DFT algorithm implemented in the STM32 (eq. 3).

There is a high correlation between data obtained using both tools. A maximum relative difference is about 1.2% and results from the differentiation procedure in the range of rising edges of the *sinc* impulse response (Fig. 10).

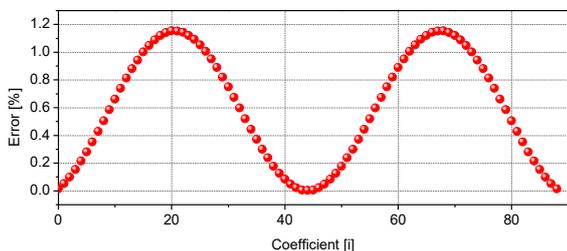


Fig. 10. Difference error between coefficients calculated in MATLAB and STM32 μ C

Using MATLAB and STM32 μ C tools, this procedure was performed with different number precision. Additionally, the *sin* and *cos* functions are more precise in MATLAB software. The microcontroller software uses standard DSP libraries (from STMicroelectronics) to calculate trigonometric functions with 32-bit float representation results (“*arm_sin_f32*; *arm_cos_f32*”).

Figure 11 visualizes the measured and simulated magnitude response of the designed LPF filter (a) and their phase characteristics (b). The simulations were provided with coefficients calculated using MATLAB and STM32 μ C and considering the influence of the normalization. The results of MATLAB simulation (without normalization) were calculated as the original frequency response curve (black line). Then the same response shape with coefficients extracted from the microcontroller memory was determined (red line). The obtained curves are similar indicating the correctness of the coefficient calculation process. They are characterized by -3 dB signal attenuation at 900 Hz and -6 dB at 1100 Hz. The small difference results from finite impulse response length, integer formats, and rounding limitations. In the case of the unnormalized impulse response, a 122 dB gain of the FIR is observed. This gain is determined by 89 operations demanding bit number standardization and by bit shifting in the FPGA (causing around 90.3 dB attenuation). Using the normalization, the nearly 0 dB gain is measured in the passband range (green curve). If the normalization is disabled, the resultant gain of this filter is about 32 dB (pink curve). Nevertheless, in many cases, there is no need to use auto normalization, and only information of gain value can be useful. In the designed module, the STM32 μ C continuously calculates gain at the desired frequency and

displays it. The phase response is linear up to about 3 kHz and fluctuates above this frequency. There are no significant differences between MATLAB simulation results and values calculated using the microcontroller’s coefficients. The measurement of the filter frequency response shows attenuation of about 55 dB in the band-stop range.

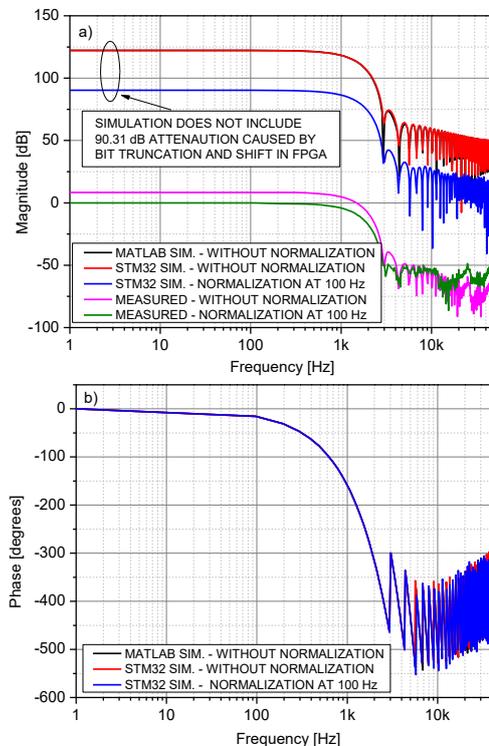


Fig. 11. The frequency (a) and phase (b) response of the designed LPF filter

C. BPF filter

The same analysis was performed for the BPF with impulse responses designed for high-pass and low-pass cut-off frequencies of 2 kHz and 5 kHz, respectively. The calculated coefficients are presented in Fig. 12.

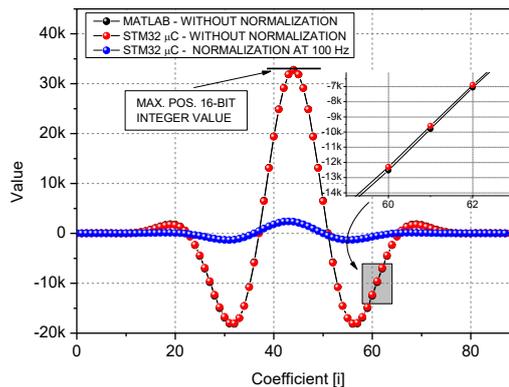


Fig. 12. Calculated coefficients (pulse response) with MATLAB and STM32 μ C for the BPF filter (2 kHz - 5 kHz)

Positive and negative values are observed with the *sinc* function contour. The frequency response is symmetrical around the middle ($i=M/2$) coefficient, and the normalization strongly reduces the values of the coefficients (blue triangles). The integral value of the normalized pulse response is close to 0. There were also determined relative differences between

impulse responses calculated in MATLAB and STM32 μC (Fig. 13). The difference changes are also defined by the differentiation procedure of the pulse response with a maximum value of about 0.7%.

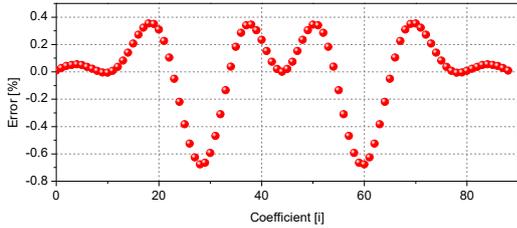


Fig. 13. Difference error between coefficients calculated in MATLAB and STM32 μC as a percent of maximum value

The calculated impulse response (coefficients) determines the magnitude and phase responses of the BPF filter (Fig. 14). Like the LPF filter, a significant normalization effect was also observed. The relatively high gain is obtained in MATLAB simulations without consideration of the attenuation by bit shifting and truncating.

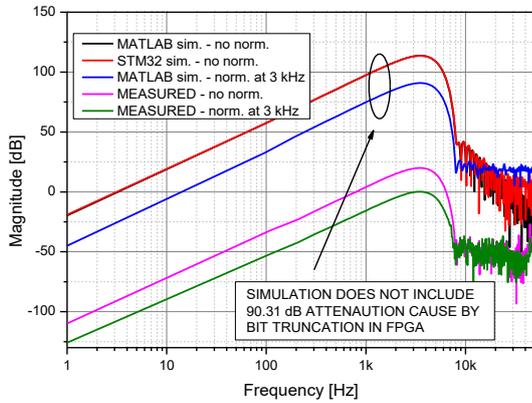


Fig. 14. The frequency response of the BPF filter

For BPF's cut-off frequencies, the top of the frequency characteristics is around 3 kHz. The measured filter response showed its attenuation around 50 dB in the low-frequency range. The high-pass frequency slope is 40 dB/dec., but distortion due to the rounding and sampling errors above 8 kHz and independent of the normalization are observed. In practice, it does not significantly affect the output signal. For phase response (Fig. 15) at high frequencies, the characteristics determined by the coefficient from STM32 μC differ from the MATLAB simulation one.

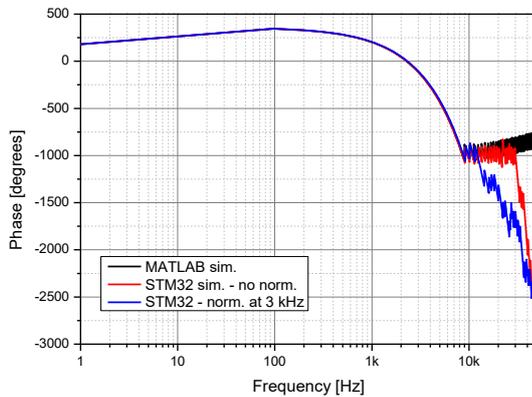


Fig. 15. The phase response of the BPF filter

D. LPF and BPF step response

The designed filter operation analyses were also performed for their step responses. In this procedure, we used the normalized coefficients. Figure 16 shows that the step response of the LPF has a rise time of about 400 μs (40 samples with a period of 10 μs). The HPF is a differentiation of step signal and takes about 400 μs . It means practically that the LPF can slow down the slope rise time for operation at high frequencies well above its f_c . On the other hand, the BPF differs signals with frequencies well below its f_c . To summarize, the analyzed FIR will work best with sinusoidal or similar signals.

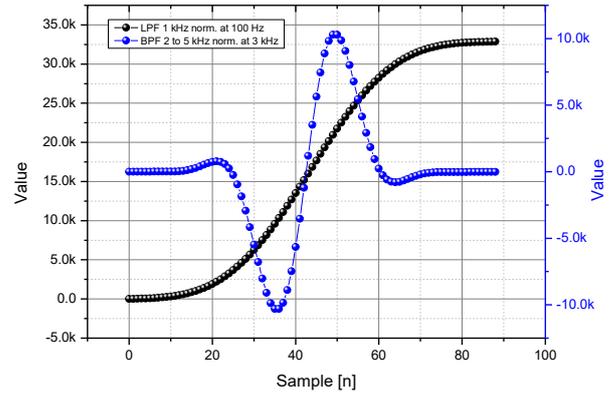


Fig. 16. Step responses of the implemented LPF and BPF

E. IR detection module

The main research task was to define the detection module filtering performances in the case of noise influence. In the experimental setup (Fig. 17), the photodetector was illuminated by a mechanically modulated IR beam from a thermal radiation source (Thorlabs, model SLS303).

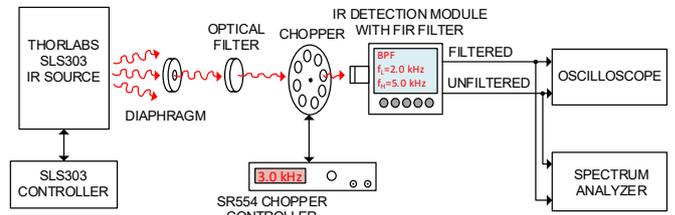


Fig. 17. Experimental setup for testing the described IR detection module

The radiation was shaped and power-tuned using a diaphragm and optical filters. The modulation frequency was set to ~ 3 kHz with a mechanical chopper (Stanford Research Systems, model SR554). The detection module output signal (unfiltered) and its digitally filtered format were registered and spectrally analyzed using FFT algorithm. During the experiment, the optical power was set to obtain a very low photoreceiver's output signal compared to its output noise. In the detection module, we set the gain normalization and the BPF configuration with the passband range from 2 kHz to 5 kHz. Figure 18a presents the oscillograms of both signals. The filtration procedure clears the "photocurrent signal" from others, e.g., noises, interferences, etc. We also analyzed the spectra of these signals (Fig. 18b) to compare noise. The measured signal attenuation was 40 dB at 100 Hz, corresponding to the simulation value (Fig. 14). The measured RMS voltage of the photoreceiver (unfiltered) and the detection module (digitally filtered) output signals in the frequency range 1 Hz - 2.9 kHz are 5.7 mV_{rms} and 0.6 mV_{rms}, respectively.

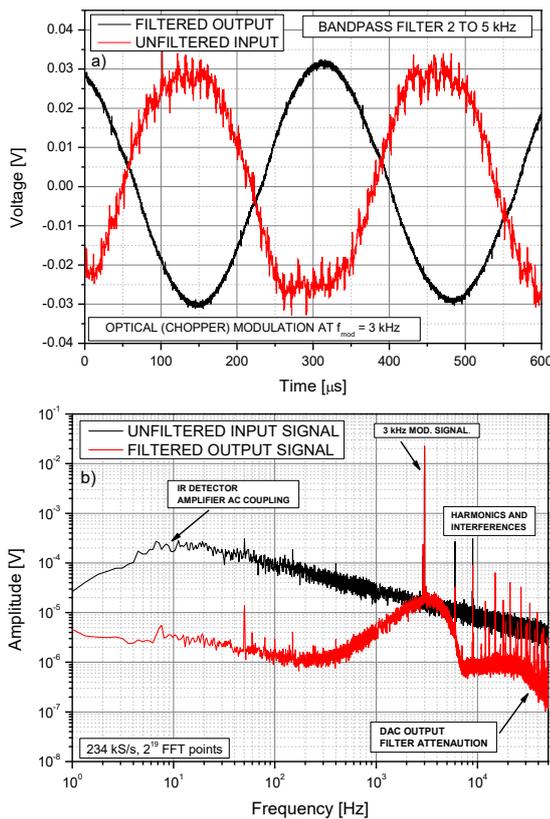


Fig. 18. Output signals from the IR photoreceiver and digitally filtered IR detection module: oscillograms (a) and spectra (b)

In this scenario, the influence of noise signals decreases tenfold resulting in the SNR's increase of about 19 dB. The registered change of the signal spectrum shape also confirms the correct operation of the designed digitally filtered IR detection module.

CONCLUSION

In this paper, we presented an advanced mixed-signal IR detection module. The novelty is a built-in analog-digital-analog conversion signal with an adjustable digital filter. The proposed architecture is a flexible solution to analyze signals in the time domain, and it allows the selection of filter type and operating frequency depending on the specific application. This functionality was implemented in the IR detection module, characterized by a high level of flicker noises. For low-frequency operation, it limits this type of noise and potentially interfering signals.

The main purpose of our work was to define the features of designing an analog IR detection module with an implemented fast (real-time) digital filtering procedure. However, the obtained results are not "the best" because the tested prototype of the "digitally filtered" detection module has several limitations resulting from hardware components that influence the precision of the obtained frequency response characteristics.

The first was the amount of DSP blocks (in FPGA - mainly for multiplications operation) that reduced the possibilities regarding attenuation and frequency ranges. It is critical for the HPF design, in which a long impulse response when the cut-off frequency is significantly lower than the sampling frequency. Moreover, the fixed-point integer coefficient representation complicates the filter gain normalization. The floating-point

fraction representation resolves this problem because it is much more flexible and easier to use.

The simulation and experimental results confirm the possibility of ensuring new functionality for IR photoreceivers equipped with a digital filtering option. They also defined some works based on the module architecture that extend its parameters, and add other functionalities by using, e.g., FPGA with greater potential dedicated to DSP purposes (multiplication blocks). In this way, a higher number of coefficients (more extended impulse response) allows stronger damping and a "faster" falling slope of the frequency characteristics, thus designing filters with narrower passbands. Applying the high-resolution ADC and DAC with higher sampling frequency rates may lead to obtaining a precision IR detection module operating in a range of MHz frequencies. In some cases, variable sampling frequency and time windowing can help obtain high performance in different applications by adapting to specific requirements.

Our paper proved that the digital filter built-in IR detection module can be used instead of the analog one. Such a solution has advantages like more flexibility, easily adjustable, and better attenuation at stopband frequencies. These virtues define new possibilities for universal IR detection modules, which can be easily adapted to many applications.

REFERENCES

- [1] Rogalski A (2005) HgCdTe infrared detector material: history, status and outlook. *Reports Prog Phys* 68:2267–2336. doi:10.1088/0034-4885/68/10/R01
- [2] Ahirwar S, Swarnkar R, Bhukya S, Namwade G (2019) Application of Drone in Agriculture. *Int J Curr Microbiol Appl Sci* 8:2500–2505. doi:10.20546/ijcmas.2019.801.264
- [3] Corsi C (2012) Infrared: A Key Technology for Security Systems. *Adv Opt Technol* 2012:1–15. doi:10.1155/2012/838752
- [4] Wojtas J, Mikolajczyk J, Nowakowski M, Rutecka B, Medrzycki R, Bielecki Z (2011) Applying CEAS Method to UV, VIS, and IR Spectroscopy Sensors. *Bull. Polish Acad. Sci. Tech. Sci.* 59: 415–418. doi:10.2478/v10175-011-0050-x
- [5] Rogalski A, Bielecki Z (2022) *Detection of Optical Signals*. CRC Press, New York.
- [6] Kloos G (2018) *Applications of Lock-in Amplifiers in Optics*. SPIE Press.
- [7] Zhang Q, Chang J, Wang Z, et al (2018) SNR Improvement of QEPAS System by Preamplifier Circuit Optimization and Frequency Locked Technique. *Photonic Sensors* 8:127–133. doi:10.1007/s13320-018-0468-y
- [8] Lv G, Chang J, Wang Q, et al (2014) Research progress of optical H₂O sensor with a DFB diode laser. *Photonic Sensors* 4:113–119. doi:10.1007/s13320-014-0151-x
- [9] Kishore K, Akbar SA (2020) Evolution of Lock-In Amplifier as Portable Sensor Interface Platform: A Review. *IEEE Sens J* 20:10345–10354. doi:10.1109/JSEN.2020.2993309
- [10] Winder S (2002) *Analog and digital filter design*. Elsevier.
- [11] Patel JJ, Parmar KR, Mewada HN (2016) Design of FIR filter for burst mode demodulator of satellite Receiver. In: 2016 International Conference on Communication and Signal Processing (ICCS). IEEE, pp 0686–0690 doi:10.1109/ICCS.2016.7754230
- [12] Cheng Xu, Su Yin, Yunchuan Qin, Hanzheng Zou (2013) A novel hardware efficient FIR filter for wireless sensor networks. In: 2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN). IEEE, pp 197–201 doi:10.1109/ICUFN.2013.6614811
- [13] Ojail M, Chevobbe S, David R, Demigny D (2008) A Frequency Domain FIR Filter Implementation Method for 3G Communication Systems. In: 2008 The Third International Conference on Digital Telecommunications (icdt 2008). IEEE, pp 1–5 doi:10.1109/ICDT.2008.9
- [14] Lavanya M, Kalaiselvi A (2016) High-speed FIR adaptive filter for RADAR applications. In: 2016 International Conference on Wireless

- Communications, Signal Processing and Networking (WiSPNET). IEEE, pp 2118–2122. doi:10.1109/WiSPNET.2016.7566516
- [15] Erdogan AT, Arslan T High throughput FIR filter design for low power SoC applications. In: Proceedings of 13th Annual IEEE International ASIC/SOC Conference (Cat. No.00TH8541). IEEE, pp 374–378. doi:10.1109/ASIC.2000.880767
- [16] Fridman A, Semenov S (2013) System-on-Chip *FPGA*-based GNSS receiver. In: East-West Design & Test Symposium (EWDTS 2013). IEEE, pp 1–7. doi:10.1109/EWDTS.2013.6673192
- [17] Xiaoyan Jiang, Yujun Bao (2010) FIR filter design based on *FPGA*. In: 2010 International Conference on Computer Application and System Modeling (ICCASM 2010). IEEE, pp V13-621-V13-624. doi:10.1109/ICCASM.2010.5622482
- [18] Pandey B, Das B, Kaur A, et al (2017) Performance Evaluation of FIR Filter After Implementation on Different FPGA and SOC and Its Utilization in Communication and Network. *Wirel Pers Commun* 95:375–389. doi:10.1007/s11277-016-3898-0
- [19] Park SY, Meher PK (2014) Efficient FPGA and ASIC Realizations of a DA-Based Reconfigurable FIR Digital Filter. *IEEE Trans Circuits Syst II Express Briefs* 61:511–515. doi:10.1109/TCSII.2014.2324418
- [20] Gawron, W., et al. "Recent progress in LWIR HOT photoconductors based on MOCVD grown (100) HgCdTe." *Semiconductor Science and Technology* 31.10 (2016): 105004. doi: 10.1088/0268-1242/31/10/105004
- [21] Grudzien M, Piotrowski J (1989) Monolithic optically immersed HgCdTe IR detectors. *Infrared Phys* 29:251–253. doi:10.1016/0020-0891(89)90058-4
- [22] Lei W, Antoszewski J, Faraone L (2015) Progress, challenges, and opportunities for HgCdTe infrared materials and detectors. *Appl Phys Rev* 2:041303. doi:10.1063/1.4936577
- [23] Piotrowski J (1993) Optical immersion of IR photodetectors as an effective way to reduce cooling requirements *Optica Applicata* 23.1: 85-90.
- [24] Slavek J, Randal H (1975) Optical immersion of HgCdTe photoconductive detectors. *Infrared Phys.* 15 339 – 340.
- [25] AD7606 datasheet. In: Analog Devices. https://www.analog.com/media/en/technical-documentation/datasheets/ad7606_7606-6_7606-4.pdf. Accessed 31 May 2023
- [26] 7-Series DSP48E1 Slice – User Guide. In: Xilinx. https://docs.xilinx.com/v/u/en-US/ug479_7Series_DSP48E1. Accessed 31 May 2023
- [27] Sudharsan RR (2019) Synthesis of FIR Filter using ADC-DAC: A FPGA Implementation. In: 2019 IEEE International Conference on Clean Energy and Energy Efficient Electronics Circuit for Sustainable Development (INCCES). IEEE, pp 1–3.
- [28] Meher PK, Chandrasekaran S, Amira A (2008) FPGA Realization of FIR Filters by Efficient and Flexible Systolization Using Distributed Arithmetic. *IEEE Trans Signal Process* 56:3009–3017. doi:10.1109/TSP.2007.914926
- [29] Chakraborty S (2013) Advantages of Blackman window over hamming window method for designing FIR filter. *Int J Comput Sci & Eng Technology (IJCSET)*, Vol. 4 No. 08 Aug 2013.
- [30] Sundararajan D (2001) The discrete Fourier transform: theory, algorithms and applications. World Scientific.
- [31] DAC8541 – datasheet. In: Texas Instruments. <https://www.ti.com/lit/ds/symlink/dac8541.pdf>. Accessed 31 May 2023.
- [32] Rapuano S, Balestrieri E, Daponte P, De Vito L (2012) Experimental investigation on DAC glitch measurement. In: XX IMEKO World Congress, Metrology for Green Growth, September. pp 9–14.
- [33] Create an FIR Filter Using Integer Coefficients. In: MathWorks® Help Cent. <https://www.mathworks.com/help/dsp/ug/create-an-fir-filter-using-integer-coefficients.html>. Accessed 31 May 2023