IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 September 2022, accepted 13 October 2022, date of publication 19 October 2022, date of current version 25 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3215800

APPLIED RESEARCH

Hardware-Accelerated Real-Time Spectrum Analyzer With a Broadband Fast Sweep Feature Based on the Cost-Effective SDR Platform

PRZEMYSŁAW FLAK
Department of Automatic Control and Robotics, Faculty of Automatic Control, Electronics and Computer Science, Ph.D. School, Silesian University of Technology, 44-100 Gliwice, Poland

e-mail: przemyslaw.flak@polsl.pl

This work was supported by the Ministry of Education and Science of Poland under Grant DWD/4/21/2020-00375/003.

ABSTRACT Radio Frequency (RF) spectrum monitoring and broadband signal analysis have multiple application areas, especially in the era of a constantly growing number of wireless devices. One of the essential challenges for a spectrum sensor is to achieve an adequate measurement rate over a wide bandwidth to detect signals of short duration so that a low latency response can be provided. In procedures that require field measurements, and some compromise in accuracy is acceptable, low-cost Software Defined Radio (SDR) devices can be used instead of expensive and bulky professional spectrum analyzers. This paper introduces a real-time swept spectrum sensor based on LimeSDR-USB with custom embedded Field Programmable Gate Array (FPGA) firmware, designed to outperform similar software implementations. The Welch's spectral density estimation is implemented in hardware to minimise the USB transfer rate and offload the host PC signal processing. Furthermore, the frequency tuning state machine and cache calibration memory are also managed by the FPGA to reduce the blind time during broadband sweep. The performance of the proposed solution indicates up to 96 MHz of real-time bandwidth along with a capability of less than millisecond cumulative sweep time per gigahertz. The characteristics of various design elements are investigated and refined during simulation and laboratory measurements, whereas the final prototype implementation is verified in real-world scenarios. The results demonstrate the effectiveness of the proposed device as a sensor for propagation studies, multiband spectrum utilisation monitoring, and spectral white spaces detection.

INDEX TERMS Electromagnetic analysis, field programmable gate array, radio frequency, software defined radio, surveillance.

I. INTRODUCTION

The Radio Frequency (RF) spectrum is a finite and limited natural resource that is used for a broad range of essential activities in both the military and civil domains. This technology is used for mobile communications, radio astronomy, and broad surveillance. Spectrum scarcity and congestion are persistent problems for the wireless industry, especially in the age of an ever-increasing demand for wireless data transfer [1]. Whereas numerous studies [2], [3], [4] report that spectrum allocation efficiency is extremely low, there is a requirement to exploit more sophisticated management methods to address the underutilisation issue.

On the other hand, monitoring RF spectrum activity is an important tool to ensure the security of industrial facilities, airports, and other objects of strategic importance. Long-term broadband analysis can reveal information about spectrum violations by recognising transient aberrant usage patterns that might go undetected in coarse occupancy scans [5]. In this area, a novel threat has emerged in the last few years, caused by the unauthorised use of commercially available drones with wireless connectivity. The problem is so critical

The associate editor coordinating the review of this manuscript and approving it for publication was Ilaria De Munari.

that the rapid evolution of drone countermeasure technologies based on RF analysis has recently been observed [6].

An urban environment with a high concentration of devices connected to wireless networks poses a serious challenge for a standalone central sensing element [7]. In this scenario, building the electromagnetic situation awareness with a distributed spectrum monitoring system can provide the best area coverage [8]. The infrastructure could potentially be composed of an endless number of sensing nodes with a fusion data centre capable of processing both fragmentary and unambiguous input data [9]. For such a concept to be feasible, the price of the individual node must be minimised. Therefore, the focus of this study is to develop a low-cost sensor based on off-the-shelf Software Defined Radio (SDR), that can be used in place of expensive and bulky professional spectrum analyzers in procedures where some accuracy trade-off is acceptable.

The current research extends the functionality of a device described by the author in a prior study for drone detection to other general-purpose applications including broadband monitoring [10]. The main focus is on migrating baseband processing tasks to the Field Programmable Gate Array (FPGA), as suggested in [11], in order to improve performance compared to the common SDR solutions. Additionally, the Welch's spectral density estimation and wideband scan controller are implemented in hardware to reduce the USB transfer rate and offload the host PC for further signal processing in software.

The structure of this paper is organised as follows: the SDR fundamentals, including the problem statement, are introduced in Section 2, and the related work is discussed in Section 3. The spectrum sensor with implementation details is presented in Section 4. Finally, an experimental setup with test results is provided in Section 5, whereas conclusions and ideas for future development are outlined in Section 6.

II. SDR FUNDAMENTALS

The SDR paradigm and its architectural principles without implementation details were defined in 1991 by Mitola [12]. In this concept, the physical components include only an antenna with an Analog Digital Converter (ADC) on the receiver side and a Digital Analog Converter (DAC) on the transmitter path. The remaining functions are handled by reprogrammable processors. Thanks to technological developments, this idea has evolved over time, and the solution has now been commercialised. An interesting review on SDR in function of spectrum sensor can be found in [13].

A contemporary, inexpensive SDR hardware architecture is presented in Fig. 1. For clarity, further considerations are limited to the receiver (RX) path. The RF front-end integrates analogue and digital signal chain elements like mixers, filters, frequency synthesizers, and ADCs. The time-domain representation of the baseband signal, in the form of In-phase and Quadrature components (IQ) [14], is passed through the FPGA and sent to the host PC via a USB controller. The hardware contribution to this process is limited to

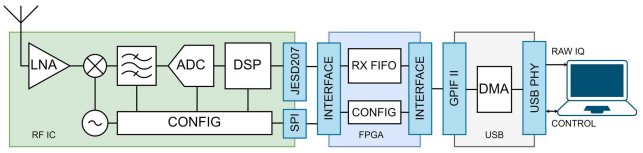


FIGURE 1. Block diagram of the classic SDR architecture, with individual on-board chips colour scheme. In this framework, raw IQ baseband signal representation is transferred to the host PC and further processed in the software to obtain a spectral estimator.

interface translation, low-level host-controlled configuration, and data buffering.

Modern mid-range SDR platforms, such as the Universal Software Radio Peripheral - USRP B210 [15], HackRF [16], and LimeSDR [17], include an RF front-end with sufficient tuning range and ADC parameters to implement a spectrum analyzer [18]. All of these devices has a USB 3.0 interface to transfer baseband time-domain samples to the host PC and suffer from its limitations [10], [19]. As a result, despite the greater capabilities of the analogue front-end, the instantaneous bandwidth of LimeSDR is limited to 61.44 MHz.

On the other hand, high-performance SDR devices that offer more than 120 MHz of instantaneous bandwidth and a 10 GigE interface cannot be considered cost-effective, which is one of the main objectives of this work [20]. Additionally, the broadband scan is not a default function to be performed by any of the listed devices, and thus its efficient implementation requires additional effort.

There are two basic modes of operation for wideband spectrum sensing instruments: sweep mode and Fast Fourier Transform (FFT) mode [21]. The classic spectrum sensing approach is the swept mode, in which the centre frequency is rapidly incremented by a small step. A Resolution Bandwidth Filter (RBW) is applied to the signal obtained at each step, and the amplitude is estimated by a detector. The sweep time is the amount of time it takes for the front-end to scan the desired frequency range. The spectrum can only be measured at one frequency point at a time using a swept analyzer, which is a severe drawback. FFT-based spectrum analyzers do not require RF front-end sweeping. Alternatively, FFT is used to transform from the time-domain to the frequency-domain. The sampling frequency, and hence the instantaneous bandwidth, define the frequency range of the FFT-based analyzer. The sweep time describes the period between two successive FFT outputs in this operating mechanism. Furthermore, both operating modes can be combined. The end product is then a composite of multiple FFT images captured at various centre frequencies. This mode is referred to as swept FFT, and it is the focus of this paper.

For an SDR device to function as a broadband spectrum analyzer, it must be tuned repeatedly during the sweep. With the samples collected while being at each centre frequency, an FFT is performed, and a part of a spectral estimate that equals the available instantaneous bandwidth is obtained. Typically, successive scans are overlapped by 10% to eliminate filter roll-off at both edges of the observed band.

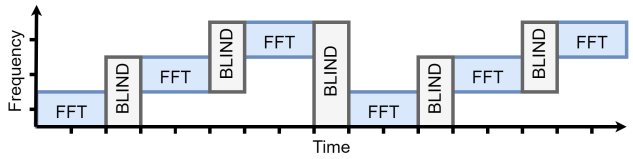


FIGURE 2. An introduction to the concept of frequency sweeping. The stages of data collection for FFT calculation are interspersed with blind times.

The interval between consecutive sampling phases is known as blind time. This principle is presented in Fig. 2. The rapid retuning procedure of the RF front-end raises several implementation concerns, and all of them have an impact on the final sweeping speed. The first issue is the time needed to send the configuration command from the host PC via USB to the FPGA, and next, using the onboard Serial Peripheral Interface (SPI) to the RF front-end chip. The second factor is frequency synthesizer lock time, which varies depending on the device, requested settings, and the distance of the consecutive frequency jump. Finally, because automatic corrector blocks require additional time to settle immediately after tuning, the initial batch of streamed IQ samples containing significant DC errors is often discarded. Each of these characteristics is addressed separately in the current work to resolve these vulnerabilities.

III. RELATED WORK

This section highlights some of the recent spectrum sensing systems, including both commercially available portable analyzers and those identified during the literature review. High-end spectrum analyzers intended for laboratory application are excluded from this discussion since they are exceedingly expensive and are not designed to be battery operated in field measurements. Alternatively, related solutions based on SDR are thoroughly discussed for parameter comparison with the proposed approach. The emphasis is on the instantaneous bandwidth parameter and the wideband sweep functionality implementation details.

A. PORTABLE SPECTRUM ANALYZER

Tektronix's entry-level RSA503A [22] is a small, lightweight, battery-powered instrument that covers a 9 kHz to 3 GHz spectrum range. It features a 40 MHz real-time bandwidth and a full-span sweep capability of 70 GHz per second. It connects to a PC tablet to form a comprehensive spectrum analyzer with sophisticated measurement features aided by host programs. By eliminating the embedded display, it is possible to overcome the limitations of previous portable designs in which the signal processing path was tailored to the capabilities of the screen [23]. Extended recording times for long-term analysis are thus possible with this method. However, a new problem with the USB 3.0 transfer limit has emerged. Even though this is only a basic version of the device, the SDR chosen for the current design is ten times less expensive.

B. SDR-BASED SOLUTIONS

The authors in [23] provided an in-depth comparative analysis of SDR devices with laboratory spectrum analyzers. Moreover, a detailed theoretical background for wideband sensing is presented. The main objective of this work is to implement the sensing engine software that relies on multi-threading FFT to achieve parallel processing. The target 100 MHz scan, made with overlapping portions of the 25 MHz band, is accomplished in one second. However, whereas it is not possible to achieve better parameters than with the laboratory analyzer, the performance is improved compared to similar SDR sensor solutions. In conclusion, it is found that the proposed pipeline architecture can be transferred to FPGA in the future to reduce the computational power required by the embedded system.

The multi-band spectrum sensing technique proposed in [24] is based on the idea of linking several affordable SDRs to operate in parallel on different frequency segments. In the paper, a sensor for spectrum occupancy detection is proposed, enhanced by advanced software-implemented signal processing techniques. The results in terms of detection probability and band occupancy measurements are promising. Unfortunately, the update period is indeed 100 ms.

Another pure software solution based on USRP and open-source GNU Radio is found in [25]. The problem with dynamic tuning of a centre frequency during wideband sweep is observed by the authors and adjusted by using a custom Python code block. Because the design is highly reconfigurable, the achieved parameters are not mentioned explicitly. A similar approach in [19] additionally highlights the presence of a significant DC offset contribution in the first batch of USRP data following frequency retuning. For a 100 MHz bandwidth, the attainable sweep rate is 0.875 per second.

Resolving the problem of latency in the frequency tuning process via a host PC USB command is proposed in [26]. Minimising blind time by incorporating sweep control into HackRF FPGA firmware yields a substantial 8 GHz per second scan rate. Furthermore, the findings mention the possibility of additional improvement, although no major advancements are expected. As a consequence, for simplicity, the authors decided to include a fixed delay of 820 μs after each retuning to let the analogue front-end settle. Aside from that, even though the FFT is hardware implemented in FPGA, the time-domain form is reconstructed after spectrum stitching to preserve compatibility with current software visualisers, which is the method's fundamental limitation.

Apart from hardware enhancements, spectrum observation efficiency may also be improved alternatively by sweeping in an intelligent manner. The inherent contradiction between the necessity to scan a wide spectrum fast and to obtain detailed sub-band information is discussed in [5]. Essential elements include a learned database of signal patterns as well as a novel scheduling algorithm that leverages these patterns to determine when to sample each band to increase the possibility of detection. Implementing this in real-time is challenging since it requires processing over a Gbit per second data stream.

All key operations are implemented using Intel's streaming extension, offering instruction-level parallelization and an appropriate software library to accommodate such high data rates.

According to [27], the minor hardware modification of the USRP delivers an impressive scan result of 5 GHz in 5 ms. High temporal resolution is achieved by supplying a chirp signal to the mechanism that sets a front-end to a specific frequency and sweeps over the spectrum. Although this approach causes signal distortion, it is corrected using self-generated calibration data afterwards. Interesting results are provided in the experiments with the classification of some standard protocols in the 2.4 GHz band at various scan speeds. Due to the limited number of samples that lead to larger distortions, significant accuracy loss is observed at the fastest sweep rate of 100 MHz per 125 μs.

IV. PROPOSED METHOD

LimeSDR is chosen as the foundation for building the SDR-based spectrum analyzer. A broadband sweep is represented by a collection of consecutive spectrum snapshots obtained at various centre frequencies. Since the 96 MHz instantaneous bandwidth capability of the RF front-end exceeds other mid-range SDRs, choosing this platform is critical in developing a sensor for a faster swept FFT mode. To fully exploit this advantage, significant changes to the manufacturer's firmware are required. The proposed approach and the degree of hardware modifications increasing FPGA's contribution to the process are demonstrated in Fig. 3.

Considering only the receiving part, in the original firmware, the FPGA is responsible for buffering and interface translation between the analogue front-end and the USB driver. There is no signal processing inside the FPGA, and the raw IQ samples are sent forward. An integrated NIOS soft-core processor receives control data to configure the front-end according to host PC commands. The latency in this process allows the front-end to be tuned only a few times per second. In this case, the bandwidth of the off-the-shelf device is limited to 61.44 MHz due to a USB 3.0 transfer rate constraint.

In the extended implementation, the available transfer rate is sufficient to achieve the full instantaneous bandwidth provided by the analogue front-end. Through the utilisation of the time-frequency transform and decibel-scale PSD formatting inside FPGA, the USB load is already reduced by half. To avoid the problem of control latency, the original host-controlled tuning scheme is therefore replaced by a state machine and cache inside the FPGA. The additional techniques introduced for the purpose of this paper to accelerate the sweep and calibration processes will be discussed in greater detail in the following subsections.

A. SPECTRAL ESTIMATE

Acquiring the entire available instantaneous bandwidth in the form of a 12-bit time-domain IQ stream requires a 288 MB/s

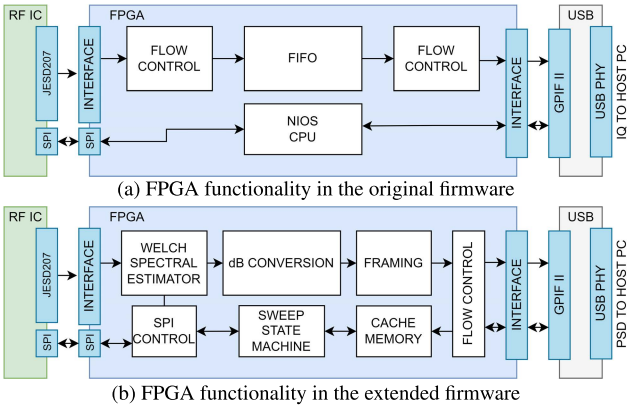


FIGURE 3. Comparison of the FPGA's contribution to signal processing in the original and extended firmware versions. The proposed sensor architecture based on classic SDR includes signal processing chain for Welch's spectral estimator and sweep controller implemented in the FPGA fabric to minimise the USB transfer rate and analyzer blind time.

USB transfer rate. This is a considerable amount of data for the portable computer to post-process in real-time. In addition, performing the frequency-domain transform is only the first step of a more complex signal analysis. As a solution, Welch's periodogram algorithm is implemented in hardware to estimate Power Spectral Destiny (PSD) [28]. The advantage of using a periodogram rather than performing FFT directly is the ability to control the trade-off between time and frequency resolution.

To provide a theoretical basis for Welch's method, suppose the received signal is represented as follows:

$$x(n) = s(n) + z(n), 0 \leq n \leq N-1, \tag{1}$$

where $s(n)$ is signal of interest, $z(n)$ stands for noise, n denote sample number, and N is the total number of samples. Next, divide the signal $x(n)$ into K segments of length M with an overlap of D, and write the signal in the segment k as:

$$x_k = x(m + kD), m = 0, \ldots, M-1, k = 0, \ldots, K-1. \tag{2}$$

The Fourier transform $X_k(\omega)$ of a data segment, with the window function $w(m)$ applied, is given as:

$$X_k(\omega) = \sum_{m=0}^{M-1} w(m) x_k(m) e^{-jm\omega}. \tag{3}$$

Then, the periodogram of the segment k is defined as:

$$I_k(\omega) = \frac{1}{MU} |X_k(\omega)|^2, \tag{4}$$

where U identifies the power of the window, expressed as:

$$U = \frac{1}{M} \sum_{m=0}^{M-1} w^2(m). \tag{5}$$

A final estimate of the PSD is produced using the Welch's method by averaging the periodograms of all the segments,

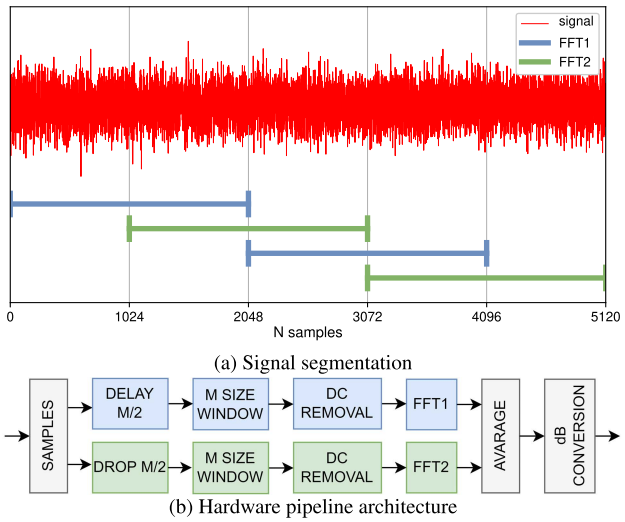


FIGURE 4. Illustration of Welch's method while segmenting 5120 samples. Dual independent FFT modules implemented together with the signal processing chain in an FPGA for parallel processing. Color-coded segments of time-domain data are processed by the corresponding FFT module.

as follows:

$$P_{Welch}(\omega) = \frac{1}{K}\sum_{k=0}^{K-1} I_k(\omega). \tag{6}$$

In accordance with the preceding, a fixed number of IQ samples at a specific centre frequency, denoted by N, are collected. The N samples are then divided into M segments. Each data block overlaps the adjacent one by 50%. An FFT is performed on each windowed chunk, and the result is averaged to produce a final PSD. The process of segmentation and overlap is illustrated in Fig. 4a. The procedure is repeated on every new centre frequency.

The hardware-specific characteristics should be considered while selecting the most appropriate N and M values. The LimeSDR RF front-end includes an automatic DC offset corrector block with a minimum observation window length of 4096 samples. This indicates that after a defined sample count, the new adjusted signal appears. It is explored in greater detail later, but for now, the value is considered the bare minimum to be acquired after tuning. The sizes of 2048 for M and 5120 for N are chosen respectively to maintain a balance between time and frequency resolution. This yields an average factor of four in the final PSD calculation stage. Consequently, each 96 MHz spectrum fragment takes roughly 53 μs to analyse, with a resolution of 46 kHz.

The open-source code from the LimeSDR-USB vendor created with Quartus II software is used as a starting point for the FPGA structure modification [29]. Since FFT overlap was not a goal of the author's previous study, pipelined processing with a single FFT module has been implemented before. As the current study applies the overlapping methodology required to carry out Welch's method, additional effort is needed to fit the signal processing chain within an FPGA.

The original firmware provided by the vendor requires over 74% of FPGA combinational logic resources and 72% of integrated memory blocks. Fitting the modules necessary for this study into this form is not achievable. Therefore, the first step is to identify and eliminate all of the design components responsible for the transmit path that are irrelevant in the spectrum analyzer application. Furthermore, because the original architecture includes an NIOS soft-core processor, it can also be optimised in terms of internal memory utilisation. Some C/C++ code changes enables the release of these resources due to the decreased number of executable programme lines needed for receiving only applications. For now, this action should be carefully maintained in order to retain compatibility with host-PC configuration software. In the future, NIOS will be removed in order to further enhance the signal processing chain for a specific sensor application.

It is sometimes possible to reuse calculation modules and conserve resources when developing with high-speed FPGAs by increasing the clock speed in comparison to the front-end ADC, while still maintaining the necessary pipeline latency. This method cannot be used when 96 MHz sampling is desired since the maximum clock speed estimated for the device with FFT is around 120 MHz. Another choice is to buffer the samples and postprocess the data during the subsequent retuning process. However, because the complete transform computation of four data segments takes longer than the standard blind time, any additional delay may have an adverse effect on overall performance. As a consequence, in exchange for increased resource utilisation, it was decided to employ a solution based on two parallel FFT modules.

The signal processing path implemented in FPGA is presented in Fig. 4b. The pipeline and parallel processing paradigm are used to achieve real-time performance. The FFT2 module begins the computation by skipping the first $M/2$ samples in order to process the data from the second segment. FFT1 can begin analysing the first segment right away, but in this case the output will be $M/2$ ahead of FFT2. This result can be further delayed to compensate, but when considering practical issues of implementation, it is more convenient to delay the input. In this approach, both FFTs simultaneously generate a result that can be directly summed in an averaging module. Since the input bit width is set at the IQ 12-bit resolution of the RF front-end, the memory size may be determined in advance. Furthermore, the memory capacity is smaller because the subsequent modules involving DC removal, twiddle multiplication, and butterfly calculations provide significant bit growth.

Implementation details of other blocks related to the signal processing chain are provided in the source publication [10]. The final PSD is transferred to the host PC in a 12-bit formatted dB scale, which results in a USB throughput of around 58 MB/s. This amount is eventually lowered due to blind time inclusion and finally remains even under the USB 2.0 limit.

B. FREQUENCY TUNING CONTROL

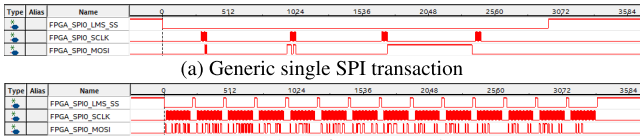
Migrating the frequency tuning mechanism to the FPGA firmware is the essential factor for sweep speed acceleration. It is difficult to determine the exact delay of the single command realisation via software as it is composed of many factors. However, it could be predicted based on the wide bandwidth sweep in comparable systems. Nevertheless, the SPI configuration time itself can be precisely defined. The quickest way to perform frequency retuning during a sweep is to restore the previously determined tune register content. A set of six 16-bit registers is generally used for centre frequency management. An original method employs the NIOS soft-core processor inside the FPGA, which is responsible for receiving commands from the host PC and a generic SPI module to configure the RF front-end.

The SignalTap II Logic Analyzer is a tool for real-time and high-speed design debugging incorporated into the Quartus II software. The exploitation of the FPGA's integrated memory allows the observation of signals inside the structure. It is used in the initial design phase for system vulnerability identification and accurate timing measurements. Figure 5 shows a comparison of the generic and optimised SPI transaction processes captured in hardware. It can be observed that the time of a single SPI transaction using the original project is comparable to the complete sequence of fully hardware-controlled transfers after modification.

In the original firmware, the single 32-bit SPI transaction required to set one configuration register is separated into four 8-bit chunks over time. The generic SPI peripheral, which is part of the NIOS CPU, is configured in this manner by default. Moreover, due to USB and host PC command latency, the time interval between consecutive 32-bit SPI writes is more than tens of microseconds. The redesigned SPI module can conduct all the fourteen read/write transactions in burst mode, required for a complete retune cycle. As a result of this improvement, the setup procedure is significantly faster, which influences the reduction of blind time. Furthermore, a mechanism for dynamically determining the frequency synthesiser lock moment is now available. After setup, an RF front-end register that indicates lock is continually read instead of having a fixed delay like in previous studies. Because the lock delay varies in different operating scenarios, this is the most versatile solution. As an outcome, the worst-case delay does not need to be predetermined.

C. ERROR CORRECTION AND CALIBRATION

There are three sources of the DC error at the RX output [17]. The most significant is the second-order distortion component, which varies with the RX input level at the current centre frequency. A real-time compensation loop inside the RF front-end is employed to track and cancel any variations in the RX DC caused by signal level changes or temperature effects. This is an analogue correction that depends on controlling the current injected to bias the input RF amplifier. This process is also important to ensure the full dynamic range operation of the input ADC and prevent saturation.



(a) Generic single SPI transaction

(b) Improved set of SPI transactions

FIGURE 5. Comparison of a single 32-bit transaction via the generic SPI interface with the entire set of fourteen 32-bit transactions required for the retune procedure applying the proposed method. Both analyses are displayed on a common time scale.

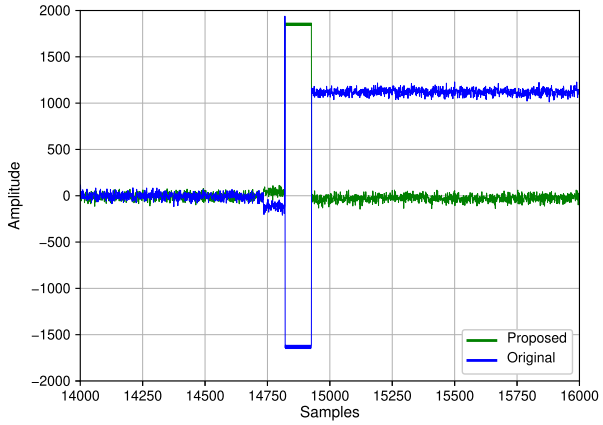
The only programmable parameter in the front-end calibration loop is the averaging window length, and there is no mechanism to synchronise the observation space with the centre frequency tune process. However, the exact values elaborated by the corrector can be read and written. This is a time-consuming operation for the host PC controlled approach. Because the access is edge-triggered in relation to a specific bit in the SPI frame, the modification procedure implies a sequence of transactions. Moreover, there are distinct registers for I and Q components.

The reaction of the RF front-end to a rapid switchover to a new frequency is presented in Fig. 6. Due to the asynchronous nature of the calibration window, the corrector's response is completed in two steps to properly compensate for the DC offset. In this case, a number of samples taken prior or even during the retuning process influence the corrector outcome. As a result, in other designs, sweeping speed performance decreases because a large number of samples are dropped while waiting for the automatic corrector.

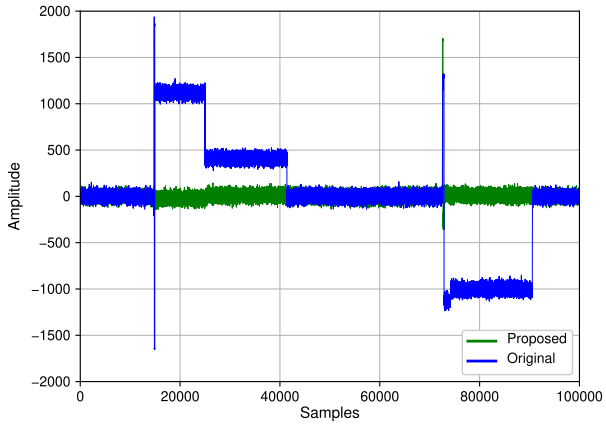
During initial experiments, the short-term repeatability of DC corrector values related to a specific frequency was observed. The current idea is to let the corrector run in the background and read the calculated registers after each data collection stage. The values are saved in FPGA cache memory and restored while returning to the same centre frequency in a subsequent sweep. This solution enables a considerably smaller number of initial samples to be dropped. This idea will be ineffective without SPI acceleration and the FPGA firmware adjustment because the original interface latency is longer than the corrector response time. Furthermore, since the corrector and the IQ samples source have independent access interfaces, this activity can be carried out in the background. Because of the USB that aggregates these two interfaces, true parallel reading is not possible when the host PC is performing the same task. Therefore, this approach has never been introduced before.

In addition to analogue correction, each signal path has a mathematical DC component equaliser calculated over the exact FFT observation window. The combination of analogue and digital correction leads to a significant reduction in the DC component visibility in future analysis.

The phase offset and gain mismatch between the I and Q components, unlike the preceding, cannot be tracked and automatically eliminated in the background. Tuning these parameters is an iterative process requiring an internal signal as a reference. Moreover, the procedure is poorly



(a) Single retune procedure shown in detail.



(b) Wide view including two retune procedures.

FIGURE 6. Comparison of a signal response to a rapid centre frequency change. The automatic corrector latency provides considerable DC offset and limitation of the ADC dynamic range after retuning in the original firmware. This is significantly reduced in the proposed approach.

documented. When building a demodulator for an advanced modulation scheme, calibration of these values is critical. It should not be neglected for spectrum analysis, but it is less significant than the real-time calibration of the DC offset. This is the reason for establishing the calibration stage once in the first step of the sweeping procedure. When an operation is completed during a preliminary sweep, the calibration data is stored in the FPGA cache memory section associated with the relevant centre frequency.

D. SWEEPING METHOD

There is no specific criterion to determine the level of hardware reconfigurability required to fit into the blurry definition of SDR. The flexibility of the final solution is usually one of the main benefits of fully software realisations over FPGA-aided alternatives. To tackle this concern, the sweeping pattern is not predefined and can be uploaded to the sensor via the Lime API software provided by the manufacturer. The form of a hopping table is saved in an FPGA internal memory shared with the calibration data cache. Figure 7 illustrates the structure of a single configuration block.

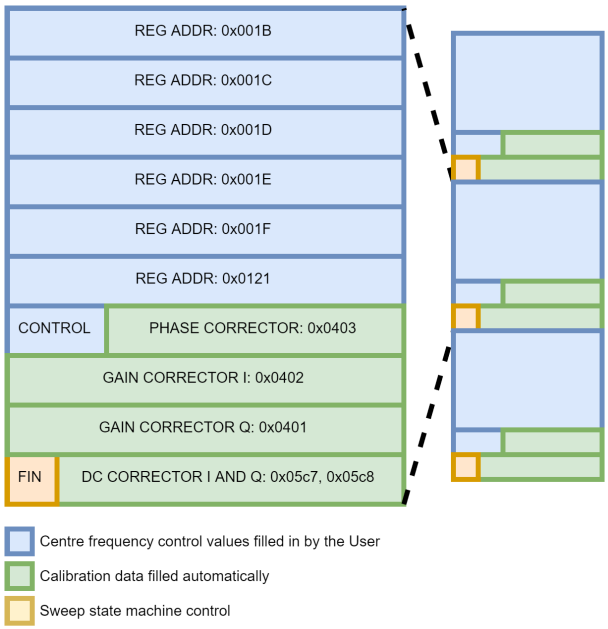


FIGURE 7. Cache memory structure inside the FPGA fabric. A single centre frequency configuration block with calibration values that consists of ten 16-bit registers is shown in detail. Internal register addresses are given according to RF front-end documentation. A collection of that blocks creates a hopping table.

The first six registers that define the centre frequency are filled in by the user. The Lime API package includes a function that assists in generating their content. Since the phase corrector only requires 12 out of 16 available bits, the rest of the seventh register is intended for additional configuration. The remaining bits identify the number of consecutive Welch's periodograms to be obtained at the current centre frequency. This method enables more intelligent sweeping like that presented in [5], or a fast full band scan when the value is set to zero. Furthermore, this approach supports selective band sweep, so scan patterns do not have to be continuous. This is impossible in HackRF modification [26] which only specifies the start and stop frequencies.

The DC equaliser requires only 14 bits, so the remaining 2 bits are used to determine whether the state machine should return to the beginning of the cache and restart the sweep. This method provides full control over the sweep process by editing the internal memory contents without making any changes to the FPGA code. The sweep control state machine diagram is shown in Fig. 8.

For this idea to work properly, the framing concept is implemented for the data transport layer, instead of delivering raw PSD samples to the host PC. Each frame can be distinguished from the others using this method and easily stitched to form the final wideband periodogram. Furthermore, some losses resulting from probable USB overflows or errors could be resolved in contrast to a constant basic data stream. The detailed frame structure based on the preamble, payload, and checksum is provided in Fig. 9. Additionally, the centre frequency information is included after the preamble.

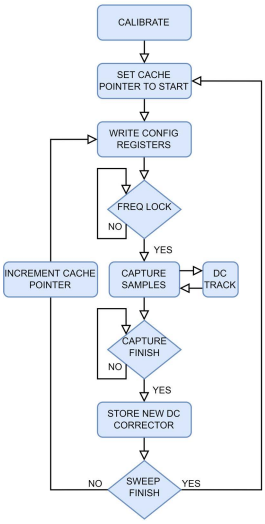


FIGURE 8. Flow chart for the sweeping controller state machine.

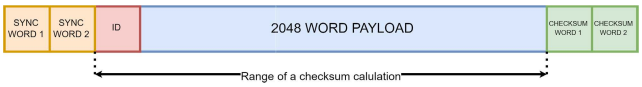


FIGURE 9. Data link layer frame structure with the range of the checksum calculation indication.

For a relatively simple method of validating data integrity, a Fletcher's algorithm is used as the checksum scheme [30].

V. EXPERIMENTAL SETUP AND RESULTS

The implementation outlined in this article is based on an Open-Source project [29] by LimeSDR-USB manufacturer created with the VHSIC hardware description language (VHDL). It has been synthesised in Quartus Prime 20.1 and simulated in the integrated ModelSim software provided by Intel. The synthesis results for the Intel CYCLONE IV E: EP4CE40F23C8 FPGA located on the LimeSDR-USB board version 1.4s indicate: 120 MHz maximum system frequency, 82% of overall logic resource utilisation with all M9K blocks allocation. The periodogram module is designed around the dual independent fixed-point pipeline FFT IP cores integrated into the framework. The evaluation of the proposed approach based on custom FPGA firmware is discussed in the following subsections, considering both real-life scenarios and some laboratory measurements.

A. LABORATORY MEASUREMENTS

Numerous applications require not only spectrum imaging but also accurate power readings. If the error sources are correctly identified, the FFT analysis can yield precise signal measurements. These factors and how they can be mitigated or compensated for are highlighted in [31]. The following procedures cover the suggested aspects of evaluating sinusoidal and narrowband signals.

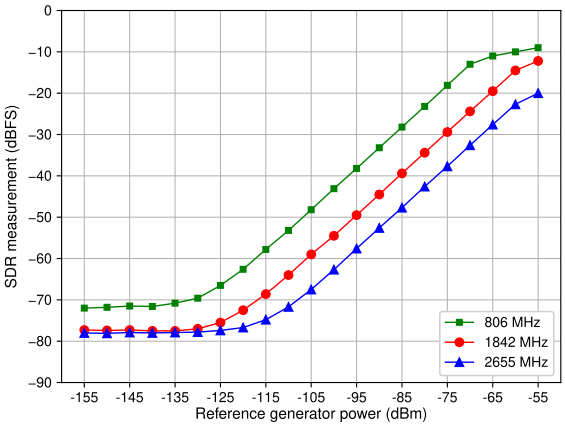
Decibel level with reference to full scale (dBFS) is a straightforward approach to formatting the PSD output in FPGA implementation because output bit-width is constant and strictly defined in the design. Furthermore, the FPGA module does not require any information regarding bandwidth or gain, which is required for the reference level calculation. Therefore, comparative characterisation with absolute reference is applied to determine the real PSD offset in relation to one milliwatt (dBm).

The laboratory setup was arranged to calibrate the SDR readout with the E4432B Agilent Digital Signal Generator in terms of signal power. In this experiment, the low-loss coaxial cable connection between the reference generator and SDR was established. A Constant Wave (CW) signal was employed to identify the linear operation region and dBFS offset level in relation to dBm. For narrowband analysis, measurements were taken at 7.68 MHz bandwidth with a maximum front-end RF gain of 73 dB to define the sensor's highest sensitivity area. Whereas, to observe strong signals in further wideband imaging experiments, the maximum sampling of 96 MHz was applied, and the gain was adjusted to 60 dB.

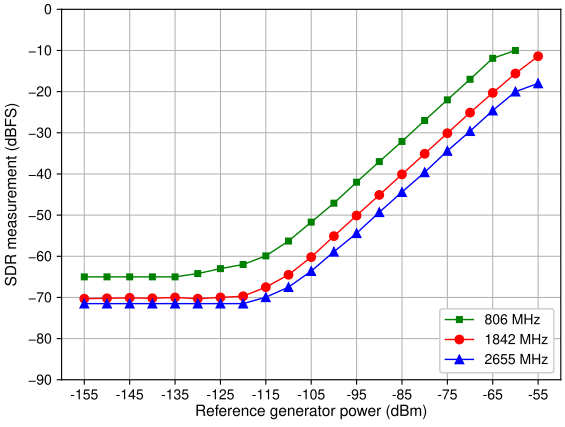
All instruments were turned on for 2 hours beforehand to reach thermal equilibrium as postulated in [32]. To verify the prepared measurement setup, the Rohde&Schwarz FSV Signal Analyzer was connected instead of the SDR to determine the real power at the signal entry point. Following that, with the reference signal generator turned off, 1000 trials of the test statistic were captured, collecting 2048 PSD samples containing only noise. Then, with the signal generator switched on, a further 1000 trials were carried out with the signal generator power ranging from -155 dBm to -55 dBm in steps of 5 dBm. The signal power in each trial was determined, and the final reading from all detection trials was obtained using maximum likelihood estimation. The measurement outcomes for various frequency ranges and bandwidths are compared in Fig. 10.

A directly proportional relationship between signal generator power and proposed sensor indications at particular input ranges can be observed in the graphs. However, at distinct frequencies, the device yields different readings for the same input power. This is related to the significant decrease in LNA gain as frequency increases. The difference between 1 GHz and 2 GHz is 10 dB, and between 2 GHz and 2.5 GHz is 2.5 dB. The insertion losses of the analogue switch and the matching transformer produce additional offsets that grow with higher frequency. This is consistent with the measurements provided in [33]. Therefore, a calibration factor is applied separately for each frequency to produce accurate results. The correction is conducted on the software side of the system to achieve valid readings of received power and to enable flexibility for future enhancements. The impact of the calibration procedure in terms of total error is presented in Fig. 11. The criterion to evaluate measurement validity is the Mean Absolute Error (MAE), which is defined as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{t}_i - t|, \tag{7}$$



(a) Results of the analysis at 7.68 MHz bandwidth



(b) Results of the analysis at 96 MHz bandwidth

FIGURE 10. SDR sensor measurement results for various frequency sets and bandwidths when loading a CW signal from a reference generator. The noticeable linear operation region can be used for accurate measurements.

where $\hat{t}_i$ is measured energy, t is the reference value and N is the trial length. The final measurements demonstrate that the SDR sensor is accurate to 0.5 dBm over the linear operation area, with MAE of 0.29 dBm.

B. RF PROPAGATION STUDIES

The project presented in [34] employs a portable, low-cost SDR for measuring RF propagation in urban areas. The experiments were performed with 2.048 MHz bandwidth settings for 71 MHz and 869.525 MHz centre frequencies simultaneously. Since the data was collected in motion, the instrument should, according to the article, enable a rapid transition between two bands, allowing both readings to be obtained at the same time and location. The measurement cycle to provide the desired spatial resolution is estimated to be two seconds, which corresponds to the typical human walking speed. Shortening the data collection period is also important for concentrating more readings in one location.

The data gathering stage for obtaining 5000 samples 100 times took 0.24 s and was prolonged to 0.36 s when combined with the FFT processing and averaging. The entire acquisition period for both frequencies was 2.25 s,

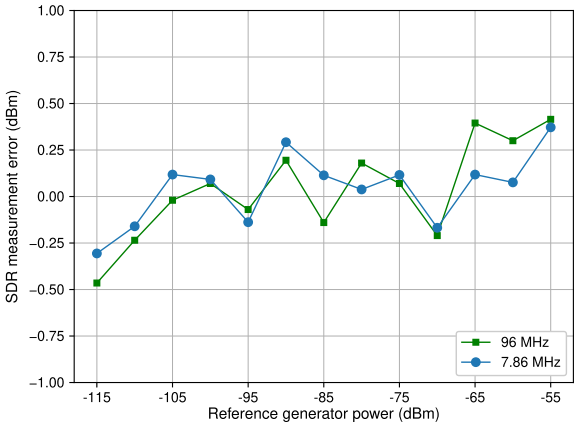


FIGURE 11. The difference between the signal generator's input power and the SDR instrument reading after calibration.

including 0.5 s for each antenna changeover via analogue switch. This demonstrated that frequency retuning takes approximately 0.265 s to complete.

The process of data aggregation may be substantially enhanced using the sensor proposed in this work. It is worth noting that, despite considerable alteration of the FPGA firmware, the user retains complete control over LimeSDR-USB's signal processing elements. Therefore, combining oversampling with the digital filtering provided by the DSP block inside the analogue front-end can reduce the data collection period by up to 5.3 ms for a single frequency. The additional time spent on FFT calculation is also unneeded in this case since improved firmware computes it in parallel. Furthermore, the time required for frequency switchover may be altered to approximately 35 μs with a configuration set for dual centre frequency sweep with in-band repetitions. LimeSDR-USB also gives the advantage of two independent RX channels that operate in parallel but share the same frequency synthesizer. This feature, as detailed in the source paper, might be advantageous in dual antenna arrangements. Unfortunately, the throughput of the interconnection between the FPGA and the RF front-end limits data transfer at the maximum ADC rate to only one channel. However, parallel sampling is possible and channel selection can be done by changing the data source for the digital front-end interface. In comparison to retuning time, the latency of this procedure is negligible.

These analyses, combined with laboratory measurements, lead to the conclusion that a proposed sensor can be mounted on a fast-moving object to achieve greater coverage in less time. It is also possible to reduce the dislocation between successive measurements. Nevertheless, it should be noted that the proposed solution is slightly more expensive but may still be considered cost-effective.

C. LTE COVERAGE AND UTILISATION ANALYSIS

Researchers around the world are constantly interested in tracking Long Term Evolution (LTE) signals, both for coverage studies [35], [36] and for estimation of channel

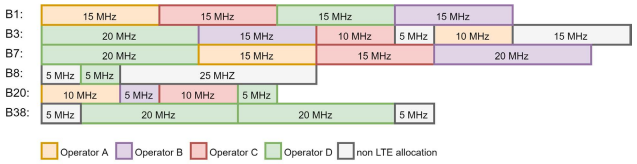


FIGURE 12. Characteristics of the LTE band division with operator allocation details.

TABLE 1. Overview of the LTE downlink bands operated in poland with channel parameters.

Band	Range (MHz)	Centre (MHz)	Bandwidth (MHz)
B1	2110-2170	2140	60
B3	1805-1880	1842.5	75
B7	2620-2690	2655	70
B8	925-960	942.5	35
B20	791-821	806	30
B38	2570-2620	2595	50

utilisation [37], [38]. LTE bands B1 to B71 are distributed throughout a wide and discontinuous range of spectrum frequencies, occupying up to a 90 MHz bandwidth [39]. The proposed sensor is ideally suited to monitoring these types of transmissions due to its appropriate real-time bandwidth. Therefore, each band in the mentioned range can be inspected to its full extent in one scan without the need to retune. The Polish mobile market is managed by four providers who operate on the common bands in exclusively allocated channels. Table 1 outlines the LTE bands that are currently in use, according to the list of references gathered at [40]. Additionally, utilisation details with concerning operator channel assignment are presented in Fig. 12.

During an experiment performed with the sensor proposed in [27], the 1.9 GHz to 2.1 GHz spectrum was swept in three 80 MHz steps, at a rate of 375 μs per 100 MHz. An average power level recorded in the individual channels was used to illustrate the utilisation level since the LTE protocol only allocates energy to subcarriers when downlink packets are sent. The granularity achieved was 900 μs, which is less than the LTE scheduling interval.

In a related experiment, a multiband omnidirectional LTE antenna (AO-ALTE-G016LS) was connected to the sensor and installed in a window on the fourth level of a multi-story residential building in the highly urbanised area. The proposed sensor was configured for 96 MHz real-time bandwidth with 50 dB gain, and the cache memory was filled to support jumps over all listed LTE centre frequencies. To achieve the fastest possible sweep speed, a single PSD estimation was obtained for each band. The entire measurement cycle, which consisted of 53 μs of data acquisition followed by 35 μs of blind time, led to a total scan time of around 528 μs for 6 disjoint bands. This period is still lower than the LTE single sub-frame time of 1 ms. As a result, the readings can be refreshed at a rate of more than 1890 times per second. The accumulated average channel energy across all bands for an operator with the highest bandwidth allocation is compared. The findings normalised to the highest power in each channel are demonstrated in Fig. 13. Furthermore, the spectrogram for the band B1 is extracted and provided as a reference in Fig. 14.

The daily variation in mobile traffic volume is primarily influenced by user actions. During the day, the average utilisation across the bands was found to be high and fairly stable. This changes significantly at night when energy spikes and uneven allocation patterns can be seen. Furthermore, it was discovered from the spectrogram perspective that operator D does not use the B1 band for wideband LTE, and a spectral gap in the measurement zone can be observed.

D. RADIO ENVIRONMENT AWARENESS FOR DYNAMIC SPECTRUM ACCESS

The demand for access to the radio frequency spectrum is constantly increasing as a result of the proliferation of radio communication tools and services. A new management paradigm is necessary to resolve the problem of the

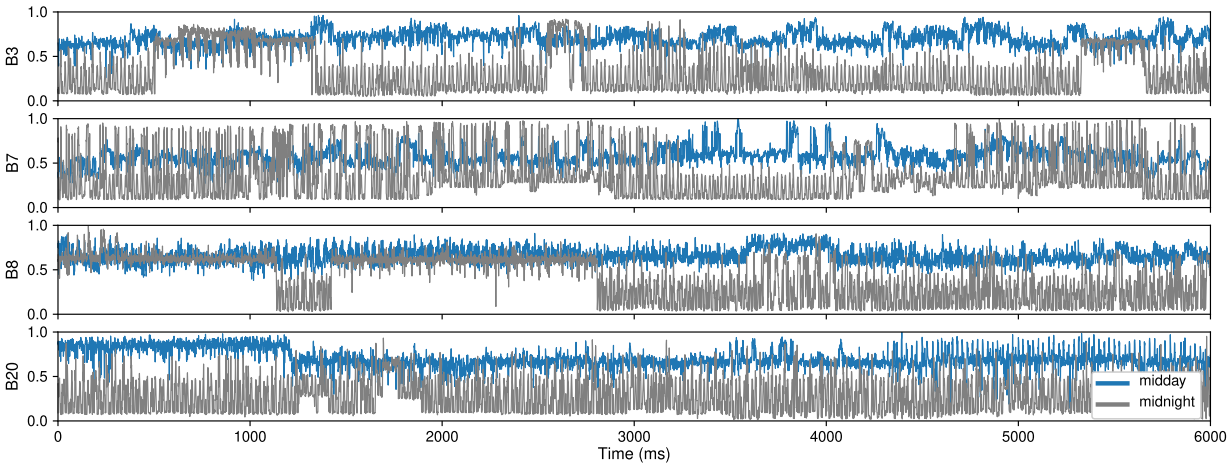


FIGURE 13. Temporal traffic dynamics in mobile network for operator D allocation outlined in Fig. 12, obtained by channel energy estimation during broadband sweep in urbanised area.

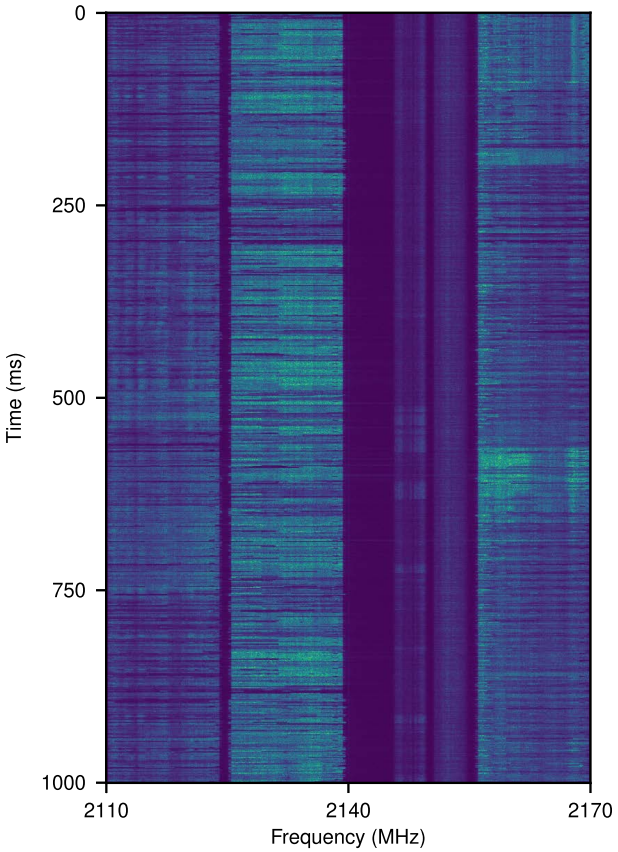


FIGURE 14. B1 band spectrogram derived from a simultaneous sweep over all LTE channels used in Poland. The spectral utilisation gap can be observed in the area of operator D allocation.

present frequency licensing regime, which results in the vast majority of the spectrum being underutilised. The problem of spectrum scarcity is now substantially addressed by dynamic management, which uses Cognitive Radio (CR) technology for opportunistic medium access. This idea allows the potential utilisation of unused spectrum, provided that Secondary

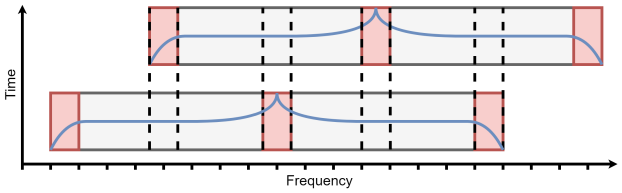


FIGURE 15. A method for merging adjacent parts of the spectrum that eliminates signal distortion caused by edge filter roll-off and DC spike inducted by FFT calculation.

Users (SU) do not interact negatively with Primary Users (PU). This activity often has a local character, and therefore the principle of Radio Environment Mapping (REM) aims to provide electromagnetic situational awareness for CR networks. The methods for creating maps based on spatial statistics with various sensor counts are discussed in [41].

The good propagation properties for different types of communication achievable in the TV White Spaces (TVWS) have attracted many researchers worldwide [42], [43], [44]. The TVWS are frequencies that are unoccupied, underused, or interleaved between broadcast TV channels, usually between 470 MHz and 694 MHz. The capabilities of the proposed sensor for white space blind detection and thus for creating dynamic REMs that represent the local state of the radio spectrum will be investigated.

The configuration of the sensor in this scenario differs from that for monitoring LTE signals, where the sub-bands are precisely allocated along with the guard bands among them. Moreover, the device centre frequency may be placed in the guard band space while the real-time observation region is sufficient to image the full sub-band. The case is changing for blind scans above real-time bandwidth. Despite advanced DC filtering, some distortion in the area of the centre FFT bin, that may affect the observed signal, will always be present. For that reason, an internal overlap stitching method is applied in the white space detection scenario. This process is demonstrated in Fig. 15.

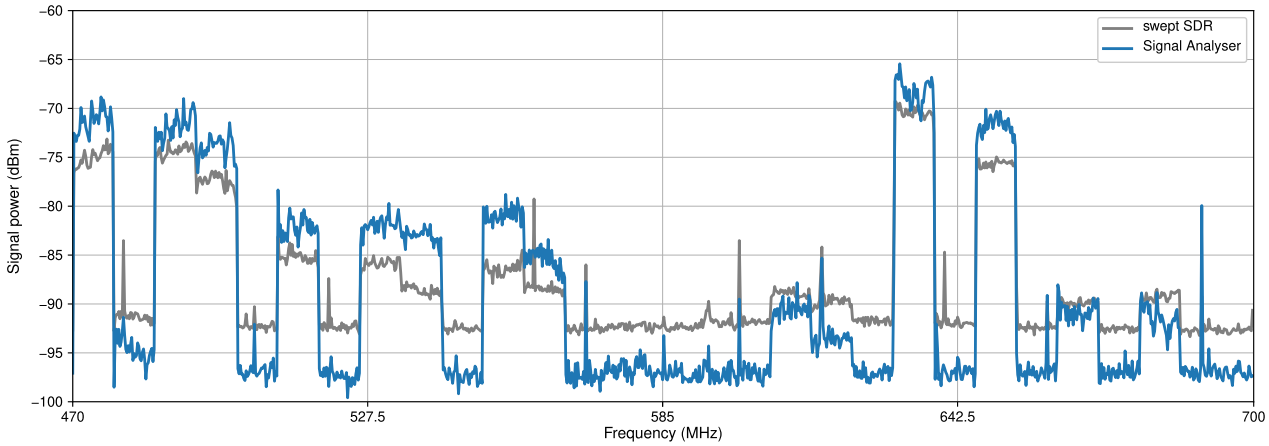


FIGURE 16. Comparison of proposed sensor indication with FieldFox Spectrum Analyzer. Some amplitude differences between signals are caused primarily by the inability to provide perfect trigger synchronisation and different sweep approach.

To ensure an identical input signal for the proposed sensor and the Agilent FieldFox N9912A Spectrum Analyzer (SA), an experimental set with a single antenna (Hama, DVB-T/DVB-T2 Rod Antenna) and RF splitter (Fairview, MPR18-2) was applied. In both cases, the span range was set at 470 MHz-700 MHz, and the RBW of SA was adjusted to 3 kHz with a preamplifier to achieve a noise floor of around −98 dBm. The proposed sensor used a maximum real-time bandwidth of 96 MHz, 60 dB front-end gain, and the centre frequency step was set as 24 MHz to support overlap stitching in post-processing. To cover the selected frequency span in that manner, eight sweep steps were necessary. As a result, a complete scan for SDR took 704 μs, whereas for SA it was 4.5 s. The measured signals are shown in Fig. 16.

The inability to fully synchronise the trigger time to store the data causes some slight variations between the two measurements. The SA is a portable device with integrated display, and the capture process is initiated by pushing a button. Furthermore, SA does not use FFT but retunes the RBW filter through the span range as in the classic sweep technique, which would result in a better noise level in exchange for scan time. The occupancy information of the selected sub-bands is substantially similar, despite the fact that the two signals are not always identical in amplitude. This is also an effect of different sweep and average approaches. An almost regular distribution of TV signals and guard bands can be observed over the entire scan range, apart from the central area near 585 MHz, where a noticeable white space is present.

VI. CONCLUSION AND FUTURE WORK

This paper presents a hardware-accelerated realisation of a broadband RF spectrum sensor based on a cost-effective SDR platform. A brief introduction to SDR technology is given initially as a foundation for further discussions. An overview of different spectrum sensing solutions was then outlined, along with an investigation of their potential for broadband analysis. Following that, implementation details and techniques for sweeping process acceleration were presented.

In the proposed approach, the FPGA contribution was extended to send a calculated Welch's spectrum estimator over the USB interface instead of raw IQ data. As a result, without the requirement to pre-evaluate any time-frequency domain transform, the entire computing power of the host PC may be used for further calculations. In addition, the sweeping control engine with cache calibration memory was also hardware arranged to reduce retuning latency and blind time. The validation results demonstrate that the sensor may be applied for power measurements with a 0.5 dB accuracy after calibration. The operating frequency range from 100 kHz to 3.6 GHz enables the instrument to be applied in a number of scenarios. However, the inability to monitor 5 GHz WiFi signals without additional accessories may be considered a fundamental limitation of the proposed device.

Further optimisation of the FPGA code will be the subject of future work to provide the resources required for additional signal processing. The goal will be to replace two independent FFT calculation modules with a single dual-channel that shares constant twiddle coefficients. Currently, with the IP core provided in the framework for fixed-point operations, it is not possible. Alternatively, the low-complexity Welch's algorithm modification [45], which reduces the size of FFT to M/2 at the price of some accuracy loss, is promising and will be investigated. In addition, characterisation for a wider range of bandwidth and gain combinations will be arranged. In order to construct a broader calibration matrix, an effort will be made to automate the reference generator control and data gathering operations. Based on the studies discussed, the sensor appears to be suitable for a variety of experiments in mobile applications. Thus, in future development, a GPS receiver will be directly connected to the FPGA, allowing the hardware to incorporate time and position information into each data frame, increasing the sensor's autonomy.

REFERENCES

[1] *Mobile Networks and Spectrum—Meeting Future Demand for Mobile Data*, OFCOM, London, U.K., Feb. 2022. [Online]. Available: https://www.ofcom.org.uk/_data/assets/pdf_file/0017/232082/mobile-spectrum-demand-discussion-paper.pdf

[2] A. Al-Hourani, V. Trajkovic, S. Chandrasekharan, and S. Kandeepan, "Spectrum occupancy measurements for different urban environments," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2015, pp. 97–102.

[3] A. Ayeni, N. Faruk, O. Bello, O. Sowande, S. Onidare, and M. Muhammad, "Spectrum occupancy measurements and analysis in the 2.4–2.7 GHz band in urban and rural environments," *Int. J. Future Comput. Commun.*, vol. 5, p. 147, May 2016.

[4] B. K. Engiz and Y. A. Rajab, "Investigation of spectrum occupancy in GSM band in Samsun, Turkey," in *Proc. 6th Int. Conf. Electr. Electron. Eng. (ICEEE)*, Apr. 2019, pp. 158–161.

[5] L. Shi, P. Bahl, and D. Katabi, "Beyond sensing: Multi-GHz realtime spectrum analytics," in *Proc. 12th USENIX Symp. Networked Syst. Design Implement.* Oakland, CA, USA: USENIX Assoc., May 2015, pp. 159–172.

[6] F.-L. Chiper, A. Martian, C. Vladeanu, I. Marghescu, R. Craciunescu, and O. Fratu, "Drone detection and defense systems: Survey and a software-defined radio-based solution," *Sensors*, vol. 22, no. 4, p. 1453, Feb. 2022.

[7] J. J. Yang, D. Chen, H. Tang, J. Yu, and M. Huang, "Cooperative compressed spectrum sensing model for regional radio monitoring," in *Proc. 31st URSI Gen. Assem. Sci. Symp. (URSI GASS)*, Aug. 2014, pp. 1–4.

[8] J. Schuette, B. Fell, J. Chapin, S. Jones, J. Stutler, M. Birchler, and D. Roberson, "Performance of RF mapping using opportunistic distributed devices," in *Proc. IEEE Mil. Commun. Conf.*, Oct. 2015, pp. 1624–1629.

[9] P. Skokowski, K. Malon, and J. Łopatka, "Building the electromagnetic situation awareness in MANET cognitive radio networks for urban areas," *Sensors*, vol. 22, no. 3, p. 716, Jan. 2022.

[10] P. Flak, "Drone detection sensor with continuous 2.4 GHz ISM band coverage based on cost-effective SDR platform," *IEEE Access*, vol. 9, pp. 114574–114586, 2021.

[11] A. Martian, "Real-time spectrum sensing using software defined radio platforms," *Telecommun. Syst.*, vol. 64, no. 4, pp. 749–761, Apr. 2017.

[12] J. Mitola, "Software radios: Survey, critical evaluation and future directions," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 8, no. 4, pp. 25–36, Apr. 1993.

[13] N. Kassri, A. Ennouaary, S. Bah, and H. Baghdadi, "A review on SDR, spectrum sensing, and CR-based IoT in cognitive radio networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 1–22, 2021.

[14] A. Wolke. *What's Your IQ—About Quadrature Signals*. Accessed: May 30, 2022. [Online]. Available: https://www.tek.com/en/blog/quadrature-iq-signals-explained

[15] Ettus Research. *The USRP B210*. Accessed: May 30, 2022. [Online]. Available: https://www.ettus.com/all-products/ub210-kit/

[16] Great Scott Gadgets. *The HackRF*. Accessed: May 30, 2022. [Online]. Available: https://greatscottgadgets.com/hackrf/

[17] Lime Microsystems. *The LimeSDR*. Accessed: May 30, 2022. [Online]. Available: https://limemicro.com/products/boards/limesdr/

[18] D. M. Molla, H. Badis, L. George, and M. Berbineau, "Software defined radio platforms for wireless technologies," *IEEE Access*, vol. 10, pp. 26203–26229, 2022.
[19] N. Bello and K. O. Ogbeide, "Designing a real-time swept spectrum analyser with USRP B210," *Nigerian J. Environ. Sci. Technol.*, vol. 5, no. 2, pp. 329–339, Oct. 2021.
[20] Ettus Research. *The USRP X310*. Accessed: May 30, 2022. [Online]. Available: https://www.ettus.com/all-products/x310-kit/
[21] Tektronix. *Fundamentals of Real-Time Spectrum Analysis*. Accessed: May 30, 2022. [Online]. Available: https://download.tek.com/document/37W_17249_5_HR_Letter.pdf
[22] Tektronix. *RSA500 Series Real Time Spectrum Analyzers*. Accessed: May 30, 2022. [Online]. Available: https://www.tek.com/en/products/spectrum-analyzers/rsa500
[23] W. Liu, D. Pareit, E. D. Poorter, and I. Moerman, "Advanced spectrum sensing with parallel processing based on software-defined radio," *EURASIP J. Wireless Commun. Netw.*, vol. 2013, no. 1, p. 228, Dec. 2013.
[24] Y. Molina-Tenorio, A. Prieto-Guerrero, and R. Aguilar-Gonzalez, "Real-time implementation of multiband spectrum sensing using SDR technology," *Sensors*, vol. 21, no. 10, p. 3506, May 2021.
[25] M. B. Perotoni and K. M. G. D. Santos, "SDR-based spectrum analyzer based in open-source GNU radio," *J. Microw., Optoelectron. Electromagn. Appl.*, vol. 20, no. 3, pp. 542–555, Sep. 2021.
[26] M. Ossmann and D. Spill. (2017). *What's on the Wireless? Automating RF Signal Identification*. Accessed: May 30, 2022. [Online]. Available: https://www.blackhat.com/docs/us-17/wednesday/us-17-Ossmann-Whats-On-The-Wireless-Automating-RF-Signal-Identification-wp.pdf
[27] Y. Guddeti, R. Subbaraman, M. Khazraee, A. Schulman, and D. Bharadia, "SweepSense: Sensing 5 GHz in 5 milliseconds with low-cost radios," in *Proc. NSDI*, 2019, pp. 317–330.
[28] P. D. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. AE-15, no. 2, pp. 70–73, Jun. 1967.
[29] Myriad-RF. (2020). *LimeSDR-USB*. [Online]. Available: https://github.com/myriadrf/LimeSDR-USB
[30] J. Fletcher, "An arithmetic checksum for serial transmissions," *IEEE Trans. Commun.*, vol. COM-30, no. 1, pp. 247–252, Jan. 1982.
[31] S. Scholl. (2016). *Exact Signal Measurements using FFT Analysis*. [Online]. Available: http://nbn-resolving.de/urn:nbn:de:hbz:386-kluedo-42930
[32] T. Šolc, M. Mohorčič, and C. Fortuna, "A methodology for experimental evaluation of signal detection methods in spectrum sensing," *PLoS ONE*, vol. 13, no. 6, Jun. 2018, Art. no. e0199550.
[33] Lime Microsystems. *RF and Analog Measurement Results*. Accessed: May 30, 2022. [Online]. Available: https://limemicro.com/app/uploads/2015/08/LMS7002M_Measurements-v1_05.pdf
[34] D. P. Wright and E. A. Ball, "Highly portable, low-cost SDR instrument for RF propagation studies," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 8, pp. 5446–5457, Aug. 2020.
[35] M. Patlayenko, O. Osharovska, and V. Solodka, "Comparison of LTE coverage areas in three frequency bands," in *Proc. IEEE 4th Int. Conf. Adv. Inf. Commun. Technol. (AICT)*, Sep. 2021, pp. 212–215.
[36] I. Surahmat and U. L. Hakim, "Mobile scanning of LTE frequency with SDR technology," in *Proc. 1st Int. Conf. Electron. Electr. Eng. Intell. Syst. (ICE3IS)*, Oct. 2021, pp. 76–79.
[37] J. Yuan, A. Huang, and H. Shan, "Resource management of LTE-U systems for channel utilization and user satisfaction," *IEEE Access*, vol. 7, pp. 107473–107490, 2019.
[38] S. Park, M. Agiwal, H. Kwon, and H. Jin, "An evaluation methodology for spectrum usage in LTE-A networks: Traffic volume and resource utilization perspective," *IEEE Access*, vol. 7, pp. 67863–67873, 2019.
[39] Electronics Notes. (2021). *LTE Frequency Bands, Spectrum & Channels*. Accessed: May 30, 2022. [Online]. Available: https://www.electronics-notes.com/articles/connectivity/4g-lte-long-term-evolution/frequency-bands-channels-spectrum.php
[40] Wikipedia Contributors. (2022). *List of LTE Networks in Europe*. Accessed: May 30, 2022. [Online]. Available: https://en.wikipedia.org/wiki/List_of_LTE_networks_in_Europe#Commercial_deployments
[41] P. Kaniewski, J. Romanik, E. Golan, and K. Zubel, "Spectrum awareness for cognitive radios supported by radio environment maps: Zonal approach," *Appl. Sci.*, vol. 11, no. 7, p. 2910, Mar. 2021.
[42] E. Mureu, P. Kihato, and P. Langat, "The authorization of the use of TV white spaces: The Kenyan scenario," in *Proc. Sustain. Res. Innov. Conf.*, 2022, pp. 102–106.
[43] T. Chakraborty, H. Shi, Z. Kapetanovic, B. Priyantha, D. Vasisht, B. Vu, P. Pandit, P. Pillai, Y. Chabria, A. Nelson, M. Daum, and R. Chandra, "Whisper: IoT in the TV white space spectrum," in *Proc. 19th USENIX Symp. Networked Syst. Design Implement. (NSDI)*. Renton, WA, USA: USENIX Assoc., Apr. 2022, pp. 401–418.
[44] E. Orumwense and K. Abo-Al-Ez, "Exploiting TV white spaces for smart grid communications," *J. Commun.*, vol. 15, p. 613, Jul. 2020.
[45] K. K. Parhi and M. Ayinala, "Low-complexity Welch power spectral density computation," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 1, pp. 172–182, Jan. 2014.

PRZEMYSŁAW FLAK was born in Katowice, Poland. He received the B.S. and M.S. degrees in electronic engineering from the Silesian University of Technology, Gliwice, Poland, in 2010, where he is currently pursuing the Ph.D. degree. Since 2009, he has been with Flytronic SA, WB Group, Poland, where he is working on research and development topics related to drones, programmable systems, and radio telecommunication. His current research interests include unmanned aerial systems (UAS), counter-UAS systems (C-UAS), software defined radio (SDR), and field programmable gate arrays (FPGA). He was a recipient of the 5th Annual Digilent Design Contest Award, in 2009, organized within the Technical University of Cluj-Napoca, Romania, sponsored by Digilent and Xilinx.