KTH ROYAL INSTITUTE
OF TECHNOLOGY

Degree Project in Numerical Analysis

First cycle, 15 credits

Porting of Passive Radar Software to New Hardware

In collaboration with Saab

AXEL BERGENGREN

Stockholm, Sweden 2024

Abstract

This thesis explores the process of porting a passive radar system from one SDR to another. Passive radar makes use of existing electromagnetic signals from sources such as TV and mobile phone towers to detect various objects. By leveraging consumer-grade SDRs, which have become increasingly accessible and powerful in recent years, new types of passive radar systems can be created.

The project involves adapting a passive radar system originally implemented on a KrakenSDR, to an AntSDR, which supports significantly higher bandwidth. This transition necessitates modifications to both the hardware setup and the software to accommodate differences in firmware and sampling capabilities between the two SDRs. Through theoretical analysis and practical implementation, I detail required considerations for the passive radar systems' sampling, signal processing, and display.

The performance of the two systems in real-world scenarios is compared, focusing on their ability to detect and track aircraft in the vicinity of the Saab Järfälla site using a digital TV tower as the transmitter. Results demonstrate that the AntSDR system offers improved precision and detection capabilities due to its higher sampling rate, though challenges such as processing time and signal noise persist. This thesis underscores the feasibility and benefits of using modern SDRs for passive radar applications, while highlighting challenges and potential areas for further development.

Contents

1 Introduction

In recent years, interest in passive radar has increased considerably. Passive radar is a radar technology which differs from active radar by the system not emitting any signals of its own, instead making use of transmitters such as TV towers and mobile phone towers. By observing reflections of signals sent by other transmitters, passive radar establishes the position and velocity of objects. Not transmitting any signals of their own gives passive radars a number of advantages to active radar, resulting in several applications. Important advantages include hardware costs, since no transmitter dish is needed, flexibility of use, since a passive radar system is more mobile, and stealth, since the system is not constantly showing its location [1]. Additionally, since passive radar systems do not require their own reserved frequency band, they are more regulatorily simple and less susceptible to jamming by other actors.

Meanwhile, software defined radios (SDRs) are technologies which have grown immensely in popularity and sophistication over the last decade, especially at the consumer and hobbyist level. SDRs are radio communication systems that sample electromagnetic waves and pass the samples to digital devices, such as PCs [2]. This allows for signals to be processed through digital rather than traditional analog means, which in turn offers many possibilities for advanced signal processing, such as passive radar. Due to their greatly reduced costs, recent commercial grade SDRs have opened up possibilities for advanced signal processing to the general public in ways that would have been difficult a decade ago.

For those interested in passive radar, the newer type of commercial grade SDRs offer a potential new approach to the technology. Saab, as one such interested business, has an interest in investigating and testing the use of these SDRs for this purpose. Previously, Saab had installed a complete passive radar system at their Järfälla site using the KrakenSDR, a SDR using Heimdall DAQ firmware. This system can detect aircraft travelling to and from Bromma Airport, as well as other aircraft passing through northwestern Stockholm. A drawback of using the KrakenSDR for passive radar is its relative unprecision compared to many other, newer SDRs. This is due to low bandwidth, with the KrakenSDR offering a maximum of 2.6 MHz. For improved performance, Saab therefore had an interest in adapting the passive radar system to a different SDR. For this, the AntSDR was chosen, a newer SDR with bandwidth of up to 56 MHz, which also uses different firmware, instead supporting PlutoSDR and UHD USRP firmware.

The objective of this project is thus to port the passive radar system of the existing KrakenSDR to an AntSDR, and to evaluate the performance of the new AntSDR passive radar system.

2 Theory

2.1 Passive radar geometry

The transmitter used by a passive radar system is known as the illuminator. Examples of useful illuminators are radio, TV and mobile phone towers, as these consistently transmit strong signals at known frequencies, allowing constant use by passive radar. The radar looks for signs of reflection in the received signal and presents results on the position and velocity of objects based on observed reflections [3].

Because of the geometry of a passive radar setup, the only positional measurement which can be obtained by a simple passive radar is the bistatic distance of an object in relation to the receiver and illuminator [3]. This is because bistatic distance is the single parameter which affects total travel time, and thus observable delay, in the reflected signal. Bistatic distance is defined as

$$R_{bistatic} = R_{tx} + R_{rx} - L$$

where R_{tx} is the distance between target and illuminator, R_{rx} the distance between target reciever and L distance between illuminator and receiver. An implication of this is that multiple objects can be located at vastly different positions and still appear identical to the passive radar in terms of bistatic distance. An illustration of the passive radar geometry is seen in figure 1. Likewise, velocity, which is obtained by the radar using doppler effects, can only be measured as rate of change of the bistatic distance. The combination of bistatic distance and bistatic velocity measurements makes up the information obtained by a passive radar.

In practical terms, to determine the bistatic distance and velocity of objects, passive radars need two separate receiving channels [4]. One receiver channel is referred to as the reference channel and is directed towards the illuminator. It is assumed to be the pure signal transmitted by the illuminator without reflections. The other is referred to as the surveillance channel and assumed to hold the signal reflected off an object. The surveillance channel is often installed in a manner which blocks direct view of the illuminator signal, while remaining more open to reflected signals, easing detection of weak reflections. The system mathematically compares these two channels to find possible patterns of reflection.

2.2 Discrete sampling

Though the signal transmitted by the illuminator and reflected of the target is continuous, one must for a real-world receiver sample it discretely, preferably as often as possible [5]. For a software defined radio (SDR), samples are typically in the form of in-phase quadrature components (IQ-samples), which describe both the amplitude and phase of the sampled signal.

A central concept of discrete signal sampling is the Nyquist theorem and how it relates to bandwidth. The Nyquist theorem states that in order to uniquely determine a signal of frequencies no higher than B, the signal must be sampled at at least double that frequency, $fs = 2B$ [5]. The reason for this requirement, referred to as the Nyquist criterion, is that many signals with differing frequencies will appear by their samples to have identical frequencies. If sampling of a frequency f_0 is performed at frequency f_s, the samples recorded will be identical to those sampled of signal $f_0 + f_s$. An example illustrating this phenomenon is shown in figure 2.

The same relationship holds for all input frequencies differentiated by frequency f_s, e.g. samples from a signal of $0.7f_s$ will be impossible to differentiate from samples of a signal of $-0.3f_s$. More generally, this means a signal of frequency f_0 can, by a receiver sampling at f_s, be equally interpreted as $f_0, f_0 + f_s, f_0 - fs, f_0 + 2f_s, f_0 - 2f_s \ldots$. Therefore, regardless of the transmitters original frequency, any signal can be interpreted as belonging an infinite number of frequencies ranging $(-\infty, \infty)$. It follows that while the Nyquist criterion must be fulfilled in order to uniquely sample a signal, a signal outside the band of uniquely determinable frequencies $[-f_s/2, f_s/2]$ can still be recorded by moving the band of observed frequencies to a different part of the spectrum. This is achieved by choosing to interpret all received signals as being within that band, in effect moving them there, regardless of their originally transmitted continuous frequency. In practice, SDRs and other modern receivers have built in technologies known as anti-aliasing which significantly reduce the strength of signals at frequencies outside the observed frequency band, minimizing the effect of frequencies outside the observed band wrongfully being interpreted as inside it.

Because the band of measurable frequencies can be moved, the width of this band serves as an important parameter. This is referred to as bandwidth and is especially important for radars, where

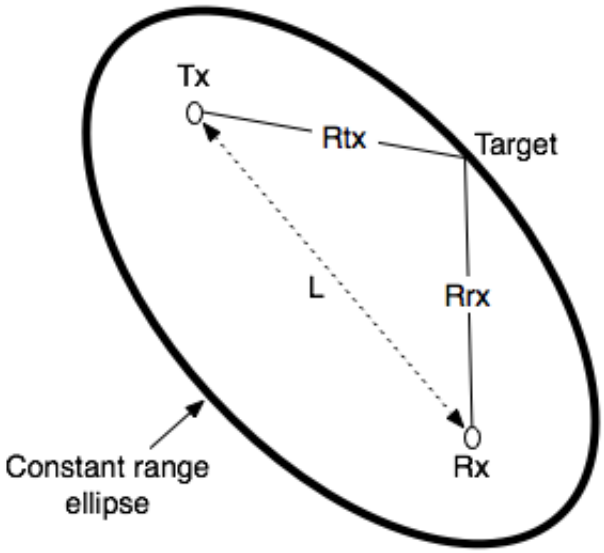

Figure 1: Illustration of bistatic geometry showing relevant distances of the passive radar. The "Constant range ellipse" refers to the ellipse of possible points at the same bistatic distance as the target, which are therefore indistinguishable from the target by the system.

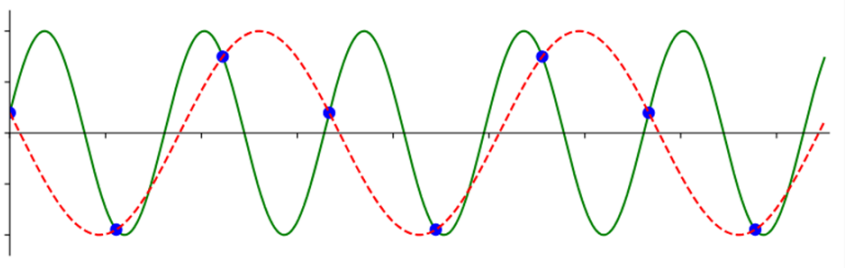

Figure 2: Example demonstrating importance of Nyquist theorem. The figure shows a signal with frequency f, sampled at $1.5f$, yet being interpreted as frequency $-0.5f$.

it is closely tied to precision. Bandwidth is very closely linked to sampling frequency through the Nyquist theorem and the two are sometimes used interchangeably. Another important parameter created by this arrangement is the center frequency f_c, which specifies where the middle of the band of observable frequencies is located.

Another central concept in discrete signal processing is the Discrete Fourier Transform (DFT), which for n samples x_m of a signal is defined as

$$X_k = \sum_{m=0}^{n-1} x_m e^{-i2\pi km/n}, \text{ for } k = 0, 1, 2, ..., n-1[5]$$

with each value of k corresponding to a particular frequency. The DFT functions much like its continuous counterpart, providing a frequency representation of a signal. A notable difference is that the range of frequencies is not infinite in the DFT, since the span of observable frequencies is limited by bandwidth. The FFT algorithm allows for efficient calculation of the DFT with $O(NlogN)$ time complexity.

2.3 Processing passive radar

Passive radars can make use of a variety of methods to process sampled data and identify objects. The central measurement used in this project is the **cross-ambiguity-function**, which is calculated for sampled data after N samples have been collected. The function is written as

$$\psi(m, k) = \sum_{n=0}^{N-1} x_e[n] x_r^*[n-m] e^{-j2\pi kn/N}$$

where x_r is the reference channel and x_e is the surveillance channel, each with N samples [4]. This discrete function, referred to as CAF, gives the strength of measurement of each bistatic distance – Doppler combination. The value of the CAF for each m and k corresponds to the confidence of finding an object at the relevant bistatic combination. The link between m and k, which are integers (m must be non-negative), and actual bistatic doppler measurements is given by the radar system's precision. A more accurate system corresponds to finer detail and therefore smaller differences in bistatic distance and Doppler shift between m and $m+1$, as well as k and $k+1$, respectively.

The function can be understood as analyzing the likeness of the surveillance channel to a theoretically shifted reference channel [4]. The shift in reference channel, which is twofold, in phase as determined by m and in frequency as determined by k, represents the theoretically expected form of a signal which has reflected of an object at the m and k combination in question. If the theoretically reflected reference channel matches up well with the observed, practically reflected surveillance channel, the function will return a high value, showing confidence in an object at that bistatic distance - Doppler combination. Importantly, the CAF only gives the confidence of a reflected signal for a given m and k combination. Thus, to comprehensively process sampled data, the CAF must be calculated for all relevant values of m and k, significantly impacting processing time.

2.4 Batch Algorithm

The most important concern for algorithms making use of the CAF is therefore how to minimize processing time for all CAF calculations, which if performed directly would have time complexity $O(MKN)$. In this project, a method sometimes referred to as the "batch algorithm" is used [4]. The Batch Algorithm begins by splitting N into Q segments P in length by setting $n = (qP + p)$, thereby splitting the summation into a nested summation, as in

$$\psi(m, k) = \sum_{q=0}^{Q-1} \sum_{p=0}^{P-1} x_e[qP+p] x_r^*[qP+p-m] e^{-j2\pi k \ (qP+p)/N}$$

. As the maximum values for m and $|k|$ arbitrary, the algorithm chooses $M = P$ and $K = Q$. Next, an assumption is made that within each P sized segment, the change in phase of the exponential term is insignificant, as in $e^{-j2\pi k(qP+p)/N} \approx e^{-j2\pi k(qP)/N}$, thus the exponential can be factorized out to

$$\psi(m,k) = \sum_{q=0}^{Q-1} e^{-j2\pi k\ (qP)/N} (\sum_{p=0}^{P-1} x_e[qP+p] x_r^*[qP+p-m])$$

The outer summation, including the exponential term, now matches the definition of the DFT with a summation across q. By using FFT to calculate the CAF value for all k at once, the expression again simplifies to

$$\psi(m,k) = [FFT_q(\sum_{p=0}^{P-1} x_e[qP+p] x_r^*[qP+p-m])]_k$$

Since the FFT algorithm allows calculation of DFTs with time complexity $O(QlogQ)$, this is a useful simplification. Note that this expression now calculates the CAF's value for all k at once. The inner expression may also be performed in a more efficient manner, as it is a case of cross correlation, an operation similar to convolution, defined as

$$(f * g) = \sum_{m=0}^{N-1} f[m-n]^* g[m]$$

Similarly to convolutions, cross correlations are calculated very simply in the frequency space on Fourier-transformed functions by simply multiplying elementwise, an $O(P)$ operation. Combining this with the FFT and its inverse, the IFFT, each inner sum of P values may be calculated for all m by mere $O(PlogP)$ time complexity, totalling to $O(QPlogP)$ for all segments. This rewrites the entire algorithm as the compact expression

$$\psi(m,k) = \Big[FFT[IFFT(FFT[x_e] \odot FFT[x_r]^*)]\Big]_{m,k}$$

which has the total time complexity of $O(PQlogQ + QPlogP) \leq O(NlogN)$. Utilizing the Batch Algorithm on sampled reference and surveillance channel data will return a two-dimensional bistatic distance - Doppler matrix of results for each m and k.

2.5 CAF precsion

The fineness of the CAF, as in the bistatic distance and Doppler shift corresponding to each change in m and k, is determined by sampling frequency and number of samples N. Between each sampling, the signal travels a distance of c/fs, which therefore is the difference in bistatic distance between each m [4]. Conversely, it follows from the DFT that the size of each Doppler frequency step of k is f_s/N, in turn corresponding to a bistatic velocity step of $-\frac{cf_s}{f_cN}$.

Another inportant accuracy consideration is that despite using a high sampling frequency, bistatic distance will be hard to determine for fast moving objects when using long sampling times, as the distance of an object will vary over the course of sampling, which the CAF cannot account for.

2.6 Filtering

Another algorithmic consideration of a passive radar is filtering. The passive radar system in this project uses a Wiener filter on the surveillance channel, which is a filter assuming time invariant stochastic noise applied in the sampled signal. The original, clean signal is estimated through minimum square error calculations. The time complexity of this filter is $O(NlogN)$, once again making use of the FFT algorithm.

3 KrakenSDR passive radar system

3.1 Physical setup

The KrakenSDR is a SDR using a newer form of the older Heimdall DAQ firmware. The SDR is capable of receiving on up to five channels at once, though it has no transmitting capabilities. In terms of specifications, a notable outlier is the somewhat lackluster maximum sampling frequency of 2.6 MHz, which is significantly less than most recent SDRs.

The installed KrakenSDR system at Saab uses the Nackamasten digital TV tower as an illuminator and is located on the Saab Järfälla roof facing Bromma Airport. The system is designed for detecting aircraft in Stockholm, especially northwestern Stockholm, near Bromma airport. It consists of two parabolic antennas connected to a KrakenSDR with SMA cables, the SDR in turn being linked by USB to a Linux PC running relevant software. The reference channel antenna is installed directly facing Nackamasten, while the surveillance channel antenna is facing the opposite direction, reducing direct exposure from Nackamasten while still observing reflections of some airplanes. Figure 3 shows a map of the points around Stockholm which make up the passive radar system.

Signal sampling is performed by the KrakenSDR using the two antennas with samples sent digitally to the PC where all processing is handled. A GUI on the PC is used to control the system and display passive radar results. When the system is running, packages of signal samples, each corresponding to approximately 1-second-long intervals are processed into bistatic distance - Doppler matrixes and displayed in the GUI as a heatmap. To have a useful reference point to validate the detection of planes by the radar, the Linux PC is also connected to a small RTL-SDR receiver. This receiver obtains the correct location of all nearby aircraft using their own transmissions, which works because all aircraft are required to report their position publicly on radio in a system known as ADSB. The correct bistatic distances of nearby aircraft detected by ADSB are shown in the GUI next to passive radar results, allowing the two to be compared.

3.2 Software

The software for the passive radar system is purpose built for the specific firmware and API of the KrakenSDR. The entire signal processing is done in Python 3.9, using many modules which are by now depreciated, making updating the system to a newer Python version difficult without rewriting major parts. A figure illustrating the main programs and their routing order is shown in 5.

The software of the system can be divided into three main processes running at once. The firmware, which operates the SDR and collects samples, the signal processor and receiver, which reads and processes data, and the web interface, which runs an app that displays results and allows the user to control the system. A .sh file is used to start the simultaneous running of the entire system

Heimdall DAQ is software made by the manufacturer, controlling the KrakenSDR and sending data out in packages to a fifo (first in first out, pipe) file which the receiver reads. The SDR records samples in IQ-format from both channels according to parameters set when the receiving is ordered to start. Important parameters that the user might want to change are center frequency, gain of the two receiving channels and sampling rate. When a specified number of samples have been received by the SDR, the firmware sends the samples as a package to the fifo file along with a set of tags that specify information about the samples which might be useful to view. This information includes time of sampling, parameters of sampling and whether the SDR is working as intended. All these processes take place in a set of programs developed for the SDR, running in completely separate instances to the signal processing software, only sending data along to processing through the fifo file.

The signal processor is a special Python object in the main web interface script, running a looping method on a separate thread. In each loop, a receiver attribute to signal processor checks for a new data package in the fifo file. If one is found, the IQ-samples from the fifo file package are turned into a two-row numpy-array of complex numbers (one row for each channel) and the tags of information are set as attributes of an object known as "iq_header". The signal processor handles the unprocessed IQ-data by running two major functions. First, a Wiener filter is applied to the surveillance channel, returning filtered samples from which airplanes are more easily detectable. Second, passive radar-calculations are performed on the data to return a matrix of possible detected

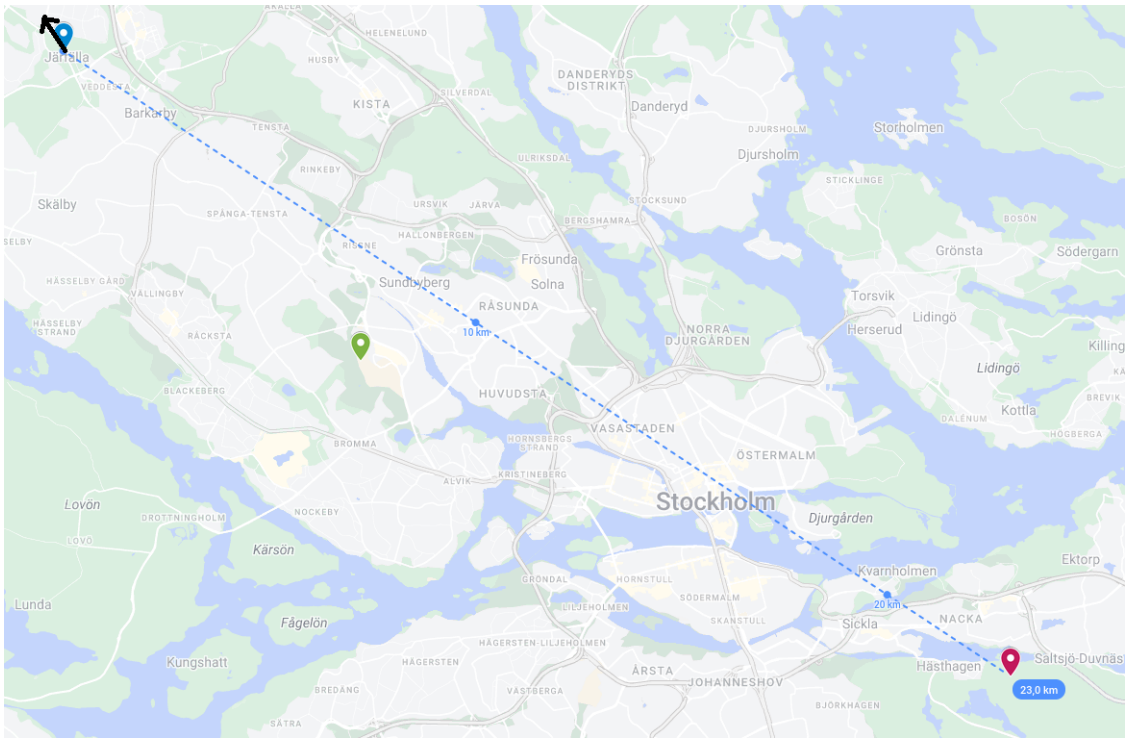

Figure 3: Map of Stockholm showing locations of the passive radar's components. Blue=Saab/receiver, Red=Nackamasten/illuminator, Green=Bromma airport/main target. The arrow from the receiver shows the direction of the surveillance channel antenna

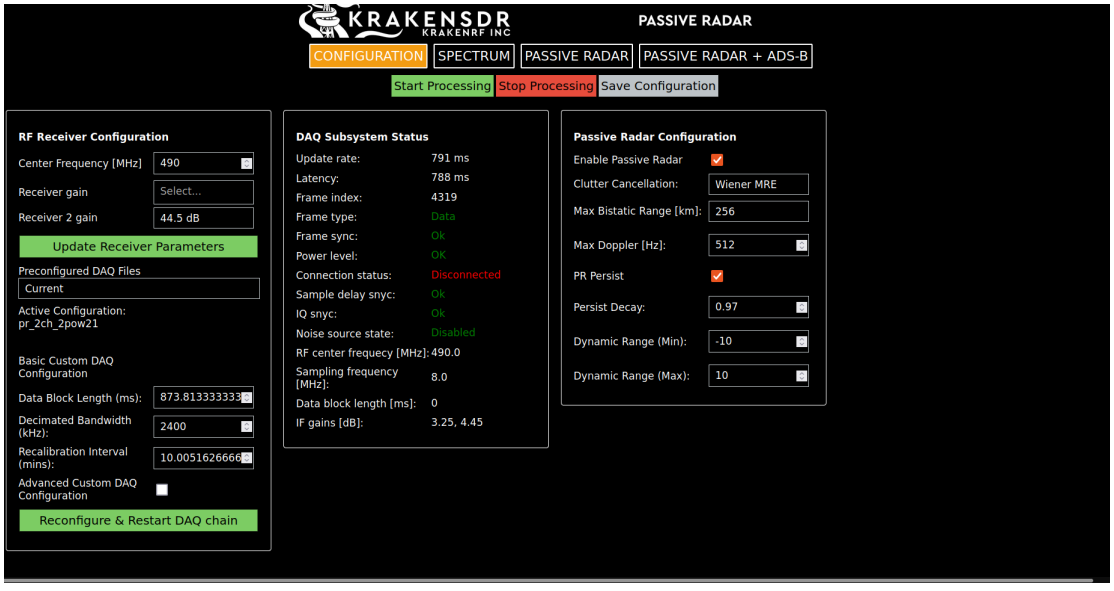

Figure 4: Screenshot of main GUI menu

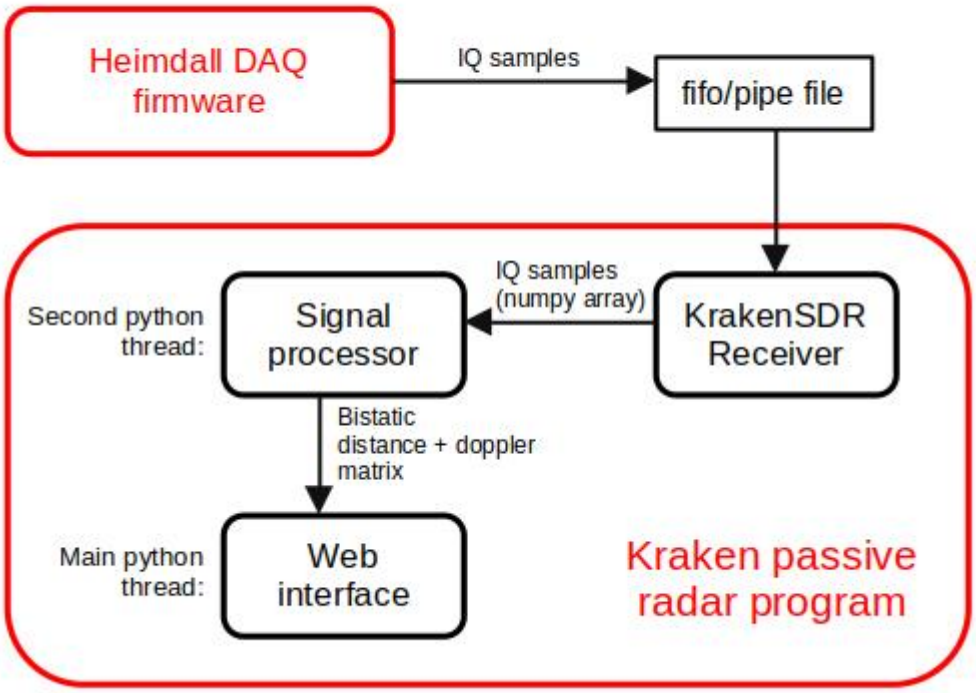

Figure 5: Figure illustrating information routing in the KrakenSDR passive radar system software

airplanes. The resulting bistatic distance - Doppler matrix is passed on to the web interface using a Python queue object along with information in "iq_header".

When running, a loop in the web interface program checks for new entries in the queue object. When a processed package is found in the queue, web interface reads the package and, in the GUI, displays relevant information from "iq_header" and plots the matrix in the form of a decaying heatmap. The heatmap shows the confidence of finding an object in each bistatic distance-doppler combination. After plotting the matrix from one package, results slowly fade out as the next displayed matrixes are plotted. As a result, when an aircraft travels through the radar's range, it will appear as a slowly decaying path of illuminated pixels in the plot, figure 7 shows an example of this.

3.3 Operation of KrakenSDR system

The KrakenSDR system, in its current form uses a sampling rate of 2.4 MHz at a center frequency of 490 MHz, corresponding to a digital TV channel broadcast by Nackamasten. Samples are sent to processing in packages of size 2620000 for each channel, meaning slightly more than one second between each processed package and thereby update to the displayed heat map. From these parameters, the passive radars precision can be theoretically calculated. Each step in the x-axis of the heat map (each m in the CAF) corresponds to $\Delta R_{bistatic} = c/f_s \approx 124$ meters bistatic distance, and each step in the y-axis (each k in the CAF) corresponds to $\Delta f = f_s/N \approx 0.9$ Hz Doppler shift, in turn corresponding to about $\Delta v = -c\Delta f/f_c \approx -0.55$ m/s bistatic velocity for each step. Note that here $v = \frac{d}{dt}R_{bistatic}$, meaning positive values of k correspond to negative bistatic velocities.

In the absence of detected airplanes, the heat map displayed by the passive radar system typically appears as in figure 6. The system always detects objects at low distances with negligible speeds, which is expected as there is simply ground at all distances for the signal to reflect off. These low speed detections only stop at about $m \approx 30$ because the reflected signal is too weak to detect. Additionally, the system detects many other "fake" objects for low distances. This is due to noise, which low distance detections are especially vulnerable to as these correspond to mathematically very small differences between the surveillance and reference channel. Due to the

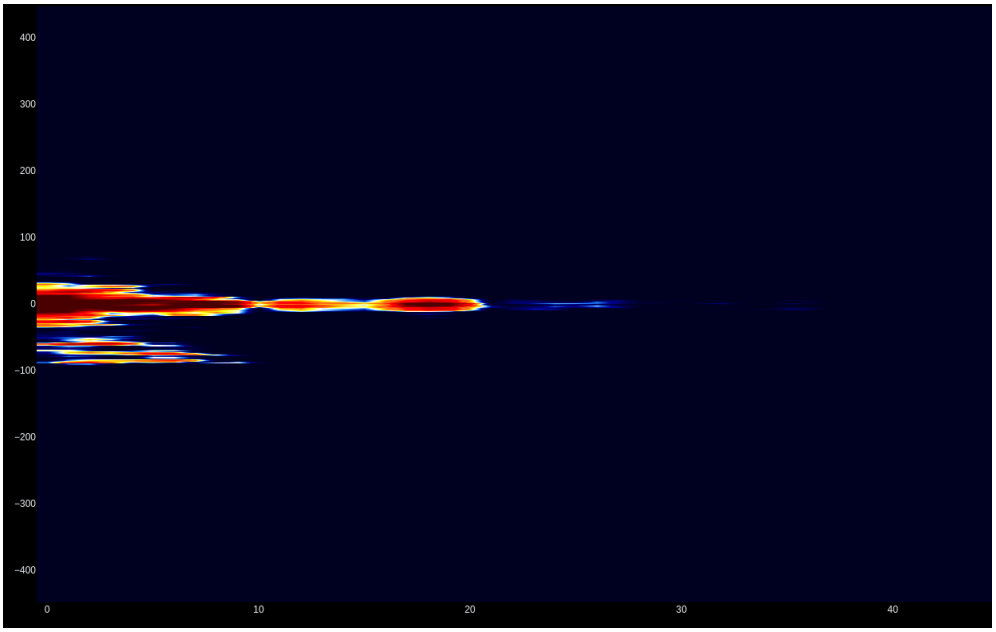

Figure 6: Example of typical heatmap displayed by passive radar system in the absense of detected aircraft. The x- and y-axis represent m and k in the CAF, respectively.

constant noise at low speeds, detected aircraft are only really identifiable at significant bistatic speeds and moderate bistatic distances.

In practice, the only aircraft the current passive radar system is consistently able to detect are those travelling past the receiver to and from Bromma airport. Figure 7 shows a typical example of this, in this case an aircraft travelling from Bromma. The successive detections of the aircraft form a clear line in the plot. Each detection appears wide in the heat map, but only stretch across one or two m, the width is due to the systems low bistatic distance precision. Note, that due to the aircraft's position changing significantly over the course of each one second sampling, some uncertainty of bistatic distance is expected. Overall, the observed results for aircraft detectable by the system matches theoretical results well.

The reason for the limited direction in which aircraft are detectable is mainly the surveillance channel only facing northwest. Therefrore, the system has very limited ability to detect airplanes passing Järfälla on any other side. Addtionally, even in the detectable direction, the systems range is limited, most ikely by antenna size. The example in 7 shows a maximum range for the airplane detection of approximately $m = 13$ or $13 \cdot \Delta R_{bistatic} \approx 1600$ meters. Detection range might also be limited by the detected object's speed.

Another important consideration for operating the passive radar system is processing time required for each sample package. Currently, the SDR sends a package approximately every 1100 milliseconds and the signal processor needs about 950 milliseconds to fully process the data. Most of this processing time consists of two main operations, Wiener filtering the surveillance channel, and calculating the CAF matrix of bistatic distance - Doppler results. As previously discussed, these processes have time complexity $O(NlogN)$. Wiener filtering is responsible for about 40% of processing time of each package, with the CAF calculations responsible for most of the other 60%. This has the implication of the program in its current form almost being at its limit in terms of sample processing rate. Therefore, any increase of sampling rate would have to be coupled with either ignoring some sampled data or altering processing systems.

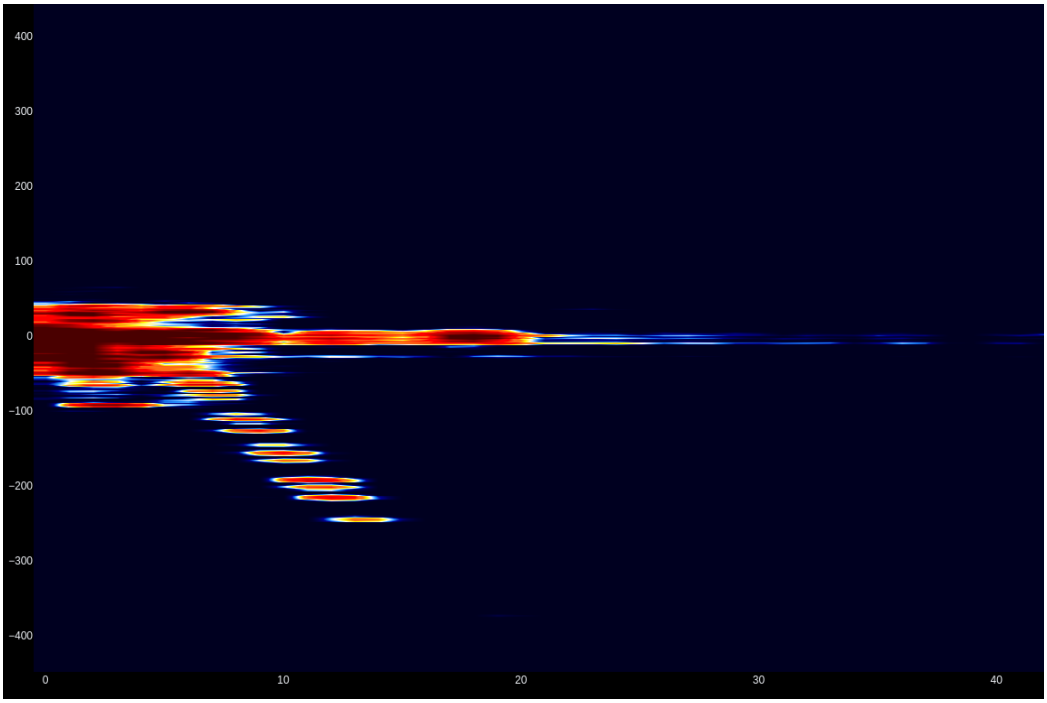

Figure 7: Screenshot of displayed heatmap in KrakenSDR system showing the bistatic distance - Doppler path of an aircraft travelling northwest from Bromma airport. The x- and y-axis represent m and k in the CAF, respectively.

4 AntSDR passive radar system

4.1 Firmware differences

The AntSDR is a SDR developed by small Chinese firm Microphase and released in 2023 following a successful Kickstarter. It offers two receiver channels and two transmitter channels, with better sampling specifications compared to the KrakenSDR, including a 56 MHz bandwidth. On the software side, the AntSDR offers support for two different SDR firmwares developed by other companies, allowing some cross compatibility of software with other SDRs. For this project, the open source UHD USRP firmware, developed by Ettus Research, was used.

SDRs vary lots in terms of controls, APIs, cable connections and software, despite the same overall goal of digitally sampling signals. Much of this is due to the rapidly developing nature and very varied market of SDRs, preventing the emergence of widespread industry standards. The UHD USRP firmware used here has quite a different approach to APIs and controls compared to KrakenSDR's Heimdall DAQ firmware. In general, Heimdall DAQ does not require much installed software on the operating computer, and when the system is in use, it is treated more as a separate, closed process which the user only is to interact with indirectly. The collecting of samples is started through standard .sh files, and the samples themselves are sent out into a fifo file for signal processing programs to utilize.

On the other hand, UHD USRP firmware instead relies more on installed software running in the background of the computer controlling the SDR. The user can communicate with and use the SDR using high level commands, either in console or through APIs in various programming languages. As a result of this software design, for the relatively simple case of consistent sampling of two channels, only a small set of Python methods from a specific UHD module are needed to start sampling, specify parameters, collect samples, and end sampling. The samples are then already in Python as complex numbers, easily sent along to processing. This reduces the amount of visible code needed to for the whole passive radar system significantly, by moving many processes to installed systems operating in the background.

4.2 Ported software

When developing the new passive radar software for the AntSDR, the general principle was, since this is a port and only small project, as much as possible of the original code should be preserved. Despite large differences between the respective SDRs firmware, this was achieved by completely removing the separate SDR controlling section of the systems software and handling all sampling directly in the receiver program. The previous KrakenSDRReceiver object was treated as part of an abstract class, replacing it with AntSDRReceiver using the same name for methods and attributes. Web interface and signal processing programs are mostly unchanged aside from small changes, but the receiver is put on a seperate thread to handle sampling. The signal processor calls upon the receiver to receive a copy of the latest package of samples along with information tags, utilizing the same formatting as before. An illustration of data routing in the new system is shown in 8.

This setup preserves the users experience of the passive radar system, despite making the entire system a single Python program, with everything now initiated by the web interface script. If working as intended, the new software should be compatible with any SDR using UHD USRP firmware, though this has not been tested.

4.3 Operation of AntSDR system

The physical setup of the passive radar system was not meaningfully altered when switching to the AntSDR. The system still uses the same antennas connected to the SDR using the same SMA cables and performs signal processing on the same Linux PC. The only physical difference is the AntSDR using ethernet to connect to the PC instead of USB, once again demonstrating the large variation in choices of SDR manufacturers.

The AntSDR has capabilities of up to 60 MHz sampling frequency, but for the purposes of this project the new systems sampling rate was chosen to 8 MHz. Reasons for this choice were the limitations in signal processing capacity as well as the limited bandwidth of each digital TV channel. Since each broadcasted channel only has a bandwidth of about 8 MHz, letting the sampling frequency exceed this might require extra considerations which are beyond the scope of this project. In combination with a sampling duration of 0.5 seconds per package of samples, also chosen in regards to signal processing capacity, the theoretical precision of the passive radar system becomes $\Delta R_{bistatic} = c/f_s \approx 37$ meters and $\Delta v = -c\Delta f/f_c \approx -1.2$ m/s for each respective change of m and k in the CAF. Note that $|\Delta v|$ increased compared to the previous system due to shorter sampling duration here.

With the new SDR, the displayed pattern in the resulting heatmap is still very similar to that of the KrakenSDR system. The system still "detects" objects at all low distances and speeds, with the main difference being the scale of each cell in the heatmap as discussed previously. For some unknown reason the AntSDR system also returns many more detections at high speeds but very low bistatic distances. Regardless, it simply means another layer of noisy data which must be ignored, and does not meaningfully affect aircraft detection capabilites. The maximum range of static detections also appear similar, now at about $m \approx 90$, roughly corresponding to the previous $m \approx 30$ of the KrakenSDR system.

The new system using the same antennas also appears to have the same limitations on direction of detection, as only aircraft travelling to and from Bromma airport past the receiver are consistently detected. A screenshot of the heatmap when detecting one such plane, in this case travelling to the airport, is shown in figure 9. Contrary to the previous system's example shown in 7, this detection appears in the top half of the heatmap because the aircraft is approaching, not leaving, the receiver. A major difference in results is the appearance of additional "echoed" paths of aircraft detection, which are weaker and appear at regular additional Doppler shifts. This means the system is "detecting" objects at multiple frequency shifts at the same bistatic distance, simultaneously "echoed" paths were determined to be due to side lobes in the surveillance channel antenna. Side lobes are directions on the side of a parabolic antenna in which the antenna will, due to interference, receive an altered form of the signal of the main observed direction. The side lobe detections are easy to identify and filter out as they appear at regular Doppler shifts, but they must be accounted for in order to not make incorrect conclusions about objects' bistatic velocities. The reason for side lobe detections only appearing in the AntSDR system is related to the greatly increased sampling frequency, though the exact causes are considered outside the scope of this project.

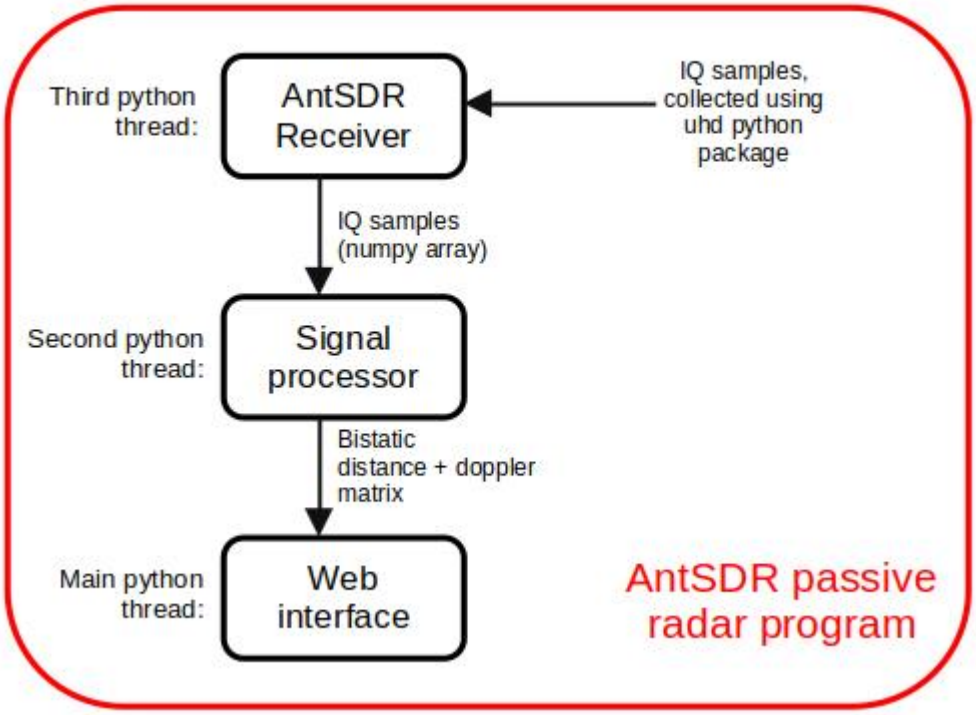

Figure 8: Figure illustrating information routing in the AntSDR passive radar system software

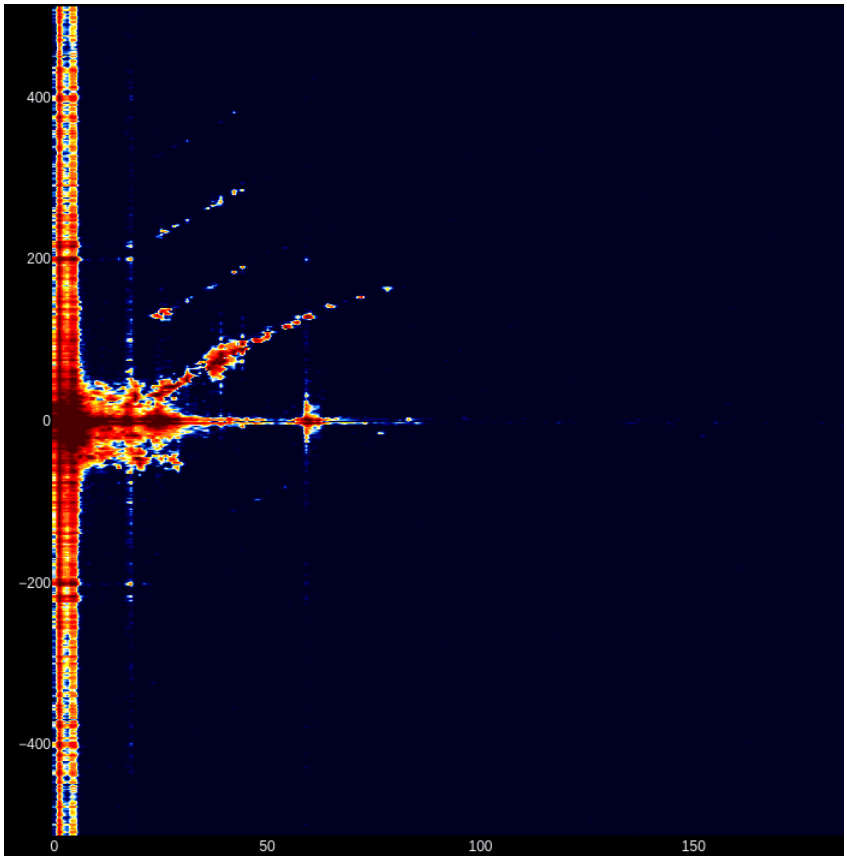

Figure 9: Screenshot of displayed heatmap in AntSDR system showing the bistatic distance - Doppler path of an aircraft travelling towards the receiver from northwest. The x- and y-axis represent m and k in the CAF, respectively.

In general, detections of aircraft in the AntSDR system typically stretch across more cells than with the KrakenSDR. Despite this, the overall practical precision of the AntSDR system is roughly in line with expectations from theory. With much a smaller value of Δm, detections in the heatmap being spread out across more cells is expected. This is partially due to fast moving aircraft varying across more values of m across the duration of sampling than in the KrakenSDR system. It is also due to the mathematical difference between the signal at neighbouring samples being smaller, increasing the vulnerability to noise. However, even when effects produce detections in the heatmap spread out across multiple cells as in some parts of 9, it is still possible to ascertain the most likely true bistatic distance - Doppler combination by looking for the strongest detected cells in the CAF, corresponding to a darker red color in the heatmap. This is a core advantage of the using the CAF for detecting reflections, as it measures every m and k, so even in cases of detections spread out across multiple cells, conclusions may still be drawn from relative strength of detection. By utilizing this in combination with curve fitting, the most likely travel path of the aircraft across the heatmap may be marked out.

In terms of signal processing time, the problem of insufficient capacity to process all data sampled at 8 MHz is handled by choosing not to process most packages. Despite collecting a sample package every half-second, the system only makes use of the those packages it has time to process. Because fully processing a sample package takes about 1.35 seconds, roughly 63% of sample packages end up discarded by the system.

5 Discussion

5.1 Evaluation of passive radar systems

Results from both systems are broadly in line with expectations from theory. Both systems are capable, within the antennas' ranges, of identifying aircrafts' bistatic distances and velocities as within certain intervals, each specified by their respective sampling parameters. Noise in the signal, and the aircrafts' observables varying across sampling duration somewhat obfuscates exactly which k and m in the CAF an aircraft corresponds to, but a most likely estimate of the aircraft's path may still be obtained from curve fitting.

In terms of practically implementing passive radar systems with SDRs, this project also demonstrates the difficulties of making generalized software. As shown, SDRs often have very different firmwares and APIs, making cross compatability of passive radar software diffcult. This is an unfortunate problem undermining the otherwise great flexibility of SDR passive radars. As the market for SDRs matures, a clearer industry standard might emerge, but for now the main way of preserving cross compatability is keeping to SDRs of a certain firmware, such as UHD USRP.

5.2 Improvements and further study

Despite the vast differences in sampling rate between the two systems, each had about the same detection range, about 4000 bistatic meters, and only in the northwestern direction. This demonstrates that in order to identify more aircraft, antennas must be upgraded. Larger antennas would make weaker signals more clearly discernable from noise, increasing maximum distance. In order to widen the angle of detection of the system, a second or wider antenna could be used. On the other hand, this would cause more of the direct unreflected signal to be picked up by surveillance channel, which is undesirable as the surveilllance channel is mathematically assumed to be the purely reflected signal. How much this would practically impact passive radar performance will have to studied in further projects.

Another large limitation of the current passive radar systems is velocity and distance is only measured in a single dimension, namely bistatically. For both location velocity, this can be imagined as identifying each aircraft to either a 2D ellipse or 3D ellipsoid, depending on whether one is interested in altitude. In order to determine the position/velocity more specifically, a second parallel receiver can be used. There are two ways of implementing this, either using the same

illuminator but a receiver at a second location, or a different illuminator, with the receiver at the same location. Each detected object may then be determined to two different ellipses or ellipsoids, the intersection of which gives a two-dimensional measurement of both position and velocity for the object.

For the passive radar system of this project, computational capcaity for signal processing was an issue. If the sampling rate is to be increased further, more advanced solutions than a simple PC processor should be considered. A possible tool for this is GPUs, which, on top of being widely available, are highly efficient for the sort of processing used in this project that involves many FFTs.

5.3 Conclusions

This project demonstrates the potential of SDRs for passive radar applications, as results show that highly accurate systems may be constructed using only relatively inexpensive, consumer grade products. It is notable that the full cost of all components for the system is most likely only about 20 thousand SEK. In comparison, professional passive radar systems used by modern militaries may cost many millions, though naturally, such systems have capabilities beyond those of the simple system of this project. Nonetheless, these results still demonstrate that consumer grade SDRs are changing the cost and accessibility of some passive radar systems, which, especially with further developments, may require governments, militaries and companies to alter their thinking about passive radars.

References

[1] "Passive radars as sources of information for air defence systems". (), [Online]. Available: https://www.sto.nato.int/publications/STO%20Meeting%20Proceedings/STO-MP-SET-187/MP-SET-187-10.pdf. (accessed: 15.05.2024).

[2] "Software defined radio". (), [Online]. Available: https://www.sciencedirect.com/topics/engineering/software-defined-radio. (accessed: 15.05.2024).

[3] krakenrf. "Krakensdr passive radar homepage". (), [Online]. Available: https://web.archive.org/web/20221004071929/https://github.com/krakenrf/krakensdr_docs/wiki/08.-Passive-Radar. (accessed: 28.04.2024).

[4] M. Malanowski, *Signal Processing for Passive Bistatic Radar*. Arctech House, 2019.

[5] B.P.Lathi, *Signal Processing and Linear Systems*. Berkeley Cambridge press, 2000.