Research Article

Optimizations for FPGA-Based Ultrasound Multiple-Access Spread Spectrum Ranging

Laurent Segers, An Braeken, and Abdellah Touhafi

Department of Engineering Sciences and Technology (INDI), Vrije Universiteit Brussel, Pleinlaan 2, Elsene 1050, Belgium

Correspondence should be addressed to Laurent Segers; laurent.segers@vub.be

Received 17 November 2019; Accepted 20 April 2020; Published 4 June 2020

Abstract

Indoor localization based on ultrasound signals has been carried out by several research groups. Most of the techniques rely on a single ultrasound pulse ranging, where the Time of Flight between the ultrasound emitters and a receiver is computed. Ultrasound orthogonal modulation techniques have also been investigated and allow to compute the range between the receiver and multiple simultaneous emitters with increased accuracy. However, no comparative investigation on the possibilities of each of the modulation techniques, comprising Direct Sequence Spread Spectrum, Frequency Hopping Spread Spectrum, and Chirp Spread Spectrum, could be found. Also, common optimized demodulation and correlation techniques for FPGA ready implementations are not widely available. Moreover, the hardware requirements for capturing modulated ultrasound signals could not be found for all the techniques. In this work, the different modulation techniques are optimized and implemented on an FPGA. A dedicated custom ultrasound MEMS-based receiver hardware for broadband ultrasound signal capturing is developed. Several modulation parameters are developed and applied for optimized signal processing. The FPGA resource consumptions are evaluated for the implemented methods. All methods are compared against the regular pulse ranging method, in both single-access and multiple-access ranging mode. Results show that, on average, up to 8 ultrasound-modulated emitters with an orthogonal sequence of length 63 can be demodulated on a Zynq7020 FPGA. In most cases, ranging up to 8 m is demonstrated in both single- and multiple-access mode, with accuracies generally remaining at a centimeter level. The requirements and capabilities for each of the modulation schemes are highlighted in the conclusions.

1. Introduction

Indoor localization of devices and objects becomes a more actual phenomenon. One might think of indoor cleaning robots, patient tracking in hospitals, autonomous navigation, obstacle detection, etc. The Global Positioning System (GPS) already assists people in outdoor navigation with an accuracy between 0.5 m and 10 m [1–3]. However, for indoor situations, GPS does not work or performs badly. Many researchers have developed technologies to overcome this shortcoming. These can be categorized in families depending on the physical properties [4]: optical localization, radio communication, localization based on movement detection, and acoustic and ultrasound ranging.

Many of the above technologies are employed with different localization strategies. Some of the well-known principles comprise Time of Arrival (ToA), Time Difference of Arrival (TDoA), Angle of Arrival (AoA), and Received Signal Strength Indicator (RSSI). In all cases, an active emitter and a receiver are utilized to estimate the range between the objects, which in some cases can be housed in the same device. In the ToA method, the distance is estimated by the travel time of a given signal between an emitter and a receiver. Both the emitter and the receiver need to be synchronized in case of two physically separated devices. In AoA, the angle between the signal and the receiver is estimated and generally requires an array of receivers. The position of the emitter can be computed by triangulating the results of several array receivers. The position of the emitter can also be computed via the RSSI values received at each receiver since the RSSI scales with the inverse square distance. At last, the TDoA method allows to compute the distance by difference in travel between two emitted signals from the same emitter. Many of these techniques are applied...
within the context of a Wireless Sensor Network (WSN) to create a context-aware sensing [4, 5]. Localization of wearable devices could also help to locate elderly people willing to remain independent even with the augmented possibility of residential accidents [6].

Ultrasound ranging has been reported in several research papers. Several implementations have been reported, but no comparative study of the existing methods for ultrasound ranging on Field Programmable Gate Arrays (FPGAs) could be found. Therefore, the existing technologies and implementations are detailed in Section 2, while a focus on ultrasound modulation and demodulation schemes reported in literature is detailed in Section 3. In this chapter, the ultrasound-modulated signals are investigated and further optimized by utilizing orthogonal code sequences (Section 4). The proposed techniques are adapted for implementation on embedded FPGA devices. Optimizations leading to a lower resource demand from the FPGA are highlighted in Section 6. The resulting FPGA resource consumption of each method is computed and presented in tables in Section 7. In order to receive the ultrasound signals, a dedicated ultrasound receiver hardware, which allows to capture the signals for the FPGA, is developed. This hardware allows to compensate the nonflat ideal responses of the chosen emitters so that a broad frequency range can be captured (Section 8). All the modulation and demodulation methods are evaluated, and the results are provided in Section 9, while the conclusions are drawn in Section 10.

2. Indoor Localization and Ranging Techniques

2.1. Optical Localization. Indoor localization based on optical information is generally carried out by means of infrared signals. Kirschner and Furukawa [7] describe a method for indoor drones to locate and position themselves based on infrared signals. The drone emits an infrared signal which is picked up by three nearby located directional receivers. Each of the receivers calculates the AoA of the received signal; after which, the locations of the drone are calculated via trigonometry functions. Depending on the distances between the emitting drone and the receiver beacons, the accuracy of the localization varies between 1.6 cm and 5 cm for distances between 0.5 m and 2.5 m. Li et al. [8] also use infrared on their drone. Their technique, however, is based on the combination of Light Detection And Ranging of Laser Imaging Detection And Ranging (LIDAR) and MicroElectro-Mechanical System- (MEMS-) based Inertial Movement Unit (IMU), where the LIDAR allows to obtain a map around the drone while the IMU enables to calculate the current kinematics of the drone. The LIDAR method consists in scanning the environment by emitting a laser pulse in one direction and by measuring the back-scattered signals. Since LIDAR is very directive, the device rotates at high speed to scan the environment with a discrete number of orientations. The obtained results are combined into a cloud of points. With the aid of the IMU sensor, the planar angle of the drone is computed to better locate the points. The point cloud is then mapped so that the environmental situation of the drone can be estimated and is also known as Simultaneous Localization And Mapping (SLAM). Another less common application of the LIDAR is the mapping of cities of old civilizations. With the help of this technique, Fisher et al. [9] are able to map a certain area of the Amazonian forest in a very fast way and to investigate the archaeological remains. Other methods include the use of visual cameras to locate objects. Barry et al. [10] describe a method based on camera stereo vision. A small Unmanned Aircraft Vehicle (UAV) is equipped with two cameras mounted on each wing. There are two processing methods, one based on an FPGA device and the other one on an ARM processor. The choice of these platforms is motivated by the low weight of the overall system compared to the whole UAV. They report object detection as long as the cameras detect a well textured image so that stereo vision can be well applied. Although their system is designed to operate in outdoor environments, indoor vision navigation has been reported by Scaramuzza et al. [11]. Other applications utilizing infrared localization include the first generations of virtual reality devices [12], self-driving cars avoiding obstacles based on camera and LIDAR vision [13].

2.2. Localization Based on Radio Communication. One of the categories of indoor localization is based on radio ranging. A few methodologies are applied, and most of these implementations can be categorized in three families:

(i) radio ranging based on signal strength

(ii) localization estimation based on radio fingerprinting

(iii) radio ranging based on the Ultra Wide Band (UWB) IEEE 802.15.4a standard

The first method is more commonly known as the RSSI method and is used in WSN applications when the IEEE 802.15.4a standard is absent. The RSSI is used to estimate the distance between an emitter and a receiver following the inverse square law [14] in ideal circumstances. However, it is reported that effects such as multipath, reflections, diffrations, and scattering increase the possibility to accurately locate a given device [14]. Moreover, the alignment between the emitting and the receiving antenna also influences the quality of the RSSI values.

Huang et al. [15] propose the RSSI ranging scope which allows to improve the accuracy of ranging to approximately 1 m. The RSSI principle has also been implemented on top of the ZigBee protocol by Kaemarungsi et al. with a reported error margin of 0.77 m [16].

Another emerging method of localization is based on the position of neighboring beacons. Several implementations such as the WiFi and Radio Frequency Identifying Device (RFID) have been developed. In case of WiFi, a fingerprinting approach is used to recognize and estimate the distance to a given access point [17]. This process can be time consuming and requires collecting a lot of data making it impractical. An alternative can be provided by using the Bluetooth Low Energy (BLE) stack, which is available on most modern smartphones [18]. A method used in the industry relies on locating items via RFID tags. This method allows to retrieve items in a range of approximately 1 m and
operates on beacons which actively emit a radio pulse towards the tags. The tags are powered by this radio pulse and transmit an identification code back to the beacon. The major drawback of this method is the limited range of operation of the tags which generally does not exceed 2 m [19, 20].

The last method adopted is based on the IEEE 802.15.4a radio ranging method. Here, the ranging is embedded in the Medium Access Control (MAC) layer of the radio communication and offers high accuracy localization for devices supporting this protocol. This protocol is implemented within the context of UWB localization [21]. Startups like Pozyx [22] use UWB in their localization products with reported accuracy levels of less than 10 cm.

2.3. Localization Based on Movement Sensors. Accelerometers, gyroscopes, and magnetometers are also widely used in combination with other technologies to enhance the accuracy of self-localization [7, 23]. The combination of accelerometers, gyroscopes, and magnetometers is also referred to as an IMU. Nowadays, IMUs are widely adopted in smartphones and allow to accurately fine tune the location of the device when combined with GPS [24]. The fusion of both sensory data allows to obtain the position of the device together with the heading. This principle is also used in GPS devices built in cars allowing to retrieve its current position in tunnels and underground parking lots for a short amount of time where GPS signals cannot be processed [25]. The instantaneous errors of IMUs are generally small. However, since IMUs only collect relative data, these errors tend to accumulate over time resulting in errors of several meters when not adequately corrected with other sensory data collected from absolute localization. Researchers try to alleviate this phenomenon by applying statistical enhancements to a certain extent [26].

2.4. Acoustic and Ultrasound Ranging. The last category of localization techniques is based on acoustic and ultrasound ranging. The technologies for locating a given sound source, being it an active or passive sound source, have started in the early years of the 20th century. This technique gained more interest during the Second World War when German U-Boats located enemy naval vessels to sink them [27]. This technique allows accurate localization thanks to the low speed of acoustic waves in gasses, fluids, and solid materials. Since the upcoming of low-cost transducers such as the piezoelectric crystals and MEMS microphones, a wide variety of implementations and applications is emerging.

Priyantha developed the Cricket system [28] that works on the principle of TDoA where a radio synchronization packet is sent out to the other beacons prior to an ultrasound pulse. Each of the receiving nodes estimates the distance to the emitter by computing the travel time between the radio packet and the ultrasound pulse. The nodes are grouped into the active ultrasound nodes and the passive listening nodes. The position of the fixed nodes is known beforehand and sent to all other nodes of interest. This latter allows to obtain a decentralized system. Since only a few nodes are allowed to emit a pulse, the system proves to be scalable regarding the number of passive listening nodes.

The second implementation is the centralized Active Bat system [29]. This method uses a grid of ceiling-mounted acoustic receivers from which the location of an active emitting device can be computed via triangulation. The Time of Flight (ToF) approach is used, where the emitting beacon emits a pulse after it receives a radio packet from the system. Since a high number of receivers is used in the implementation, a high degree of accuracy of less than 10 cm can be obtained. A major shortcoming, however, is this high number of receivers which need to be precisely placed at carefully chosen locations. In the Dolphin [30] system, the shortcoming of the precise placement of the receivers is alleviated by reducing the number to three starting nodes with a precisely determined position. The position of the other nodes, both active and passive, is computed based on the location of the first nodes, which considerably reduces the deployment cost. The system also claims recovery capabilities in case of failures. Since a few beacons have a known position and the position of the other nodes is computed iteratively, the total error margin accumulates. The reported localization accuracies, however, are still acceptable with approximately 15 cm.

The technique of acoustic and ultrasound ranging has recently also been deployed on smartphones [31, 32]. In the latter implementations, the ranging operates in frequencies between 18 kHz and 22 kHz. Higher frequencies are attenuated by built in low pass filters whereas lower frequencies can be observed by a large amount of people resulting in a less convenient solution.

Aside of the previously mentioned implementations, some researches try to improve the ultrasound ranging and localization by modulating the emitted signals. The modulation techniques proposed are well known in the field of radio communication and are applied with the use of orthogonal sequences. These techniques offer a better resistance to acoustic noise and allow several emitters to overlap in time resulting in a multiple-access ranging, with an improved accuracy. Moreover, most implementations found demodulate the signals on a general purpose computer instead of using an embedded device such as an FPGA. Therefore, the existing modulation techniques, along with the respective demodulation schemes, are detailed and adapted for implementation on an FPGA in the following sections.

3. Ultrasound Modulation and Demodulation

Several modulation techniques have been reported and can be categorized into four families:

(i) Pulse range modulation

(ii) Ranging based on Direct Sequence Spread Spectrum (DSSS)

(iii) Ranging based on Frequency Hopping Spread Spectrum (FHSS)

(iv) Chirp Spread Spectrum (CSS) ranging

These techniques are further detailed in the paragraphs below and are adapted towards FPGA-enabled implementation.
3.1. Pulse Range Modulation. The first and most simple method is based on an ultrasound pulse of a given frequency emitted during a short amount of time. In most cases, an envelope detector followed by a threshold comparator at the receiver’s side allows to detect the pulse. The time difference between the reception of the pulse and a synchronization signal allows to estimate the ToF of the ultrasound pulse between the emitter and the receiver. This method has been applied in many ranging systems, the Cricket [28], the Active Bat [29], and the Dolphin [30] systems. The major shortcoming of the pulse range technique is the limited amount of simultaneous emitting nodes in a given area. Since all emitters emit the same ultrasound pattern, no distinction between several simultaneous emitting nodes can be made at the receiver’s side. To avoid signal ambiguity, all emitters in a given area emit their ultrasound pulse following a Time-Division Multiple-Access (TDMA) pattern. Each emitter is allowed to emit during a short amount of time. In most cases, an envelope detector followed by a threshold comparator at the receiver could be made at the receiver’s side. To avoid signal ambiguity, all emitters in a given area emit their ultrasound pulse following a Time-Division Multiple-Access (TDMA) pattern. Each emitter is allowed to emit during given time slots whereas the other emitters remain silent. Given the speed of sound \( c \) in air and with the assumption of a \( 20 \times 20 \text{m}^2 \) office room, the acoustic wave needs approximately 83 ms to travel along the diagonal line of the room. Moreover, walls and objects inside a regular office cannot be considered ideal acoustic absorbers and may thus reflect a given amount of the ultrasound-emitted signal. The principle of acoustic reverberation in a given room is referred as Sabine’s law [33]. For large numbers of emitters, this could lead to lower refresh rates of node localizations. The pulse-based system may also suffer from noise, reducing the efficiency in noisy environments. Reported localization accuracies range between 5 cm and 15 cm [28–30].

3.2. Direct Sequence Spread Spectrum Modulation. To relax these constraints of the pulse-based ranging, researchers developed other approaches based on orthogonal coding schemes. Hazas and Ward [34] developed a broadband ultrasonic localization scheme based on the Direct Sequence Spread Spectrum (DSSS) modulation scheme. A single carrier is used onto which data is modulated with Binary Phase Shift Keying (BPSK). The signal is demodulated at the receiver’s side by multiplying the signal with an in-phase (I) and quarter-phase-offset (Q) carrier. The signal is then correlated with the original data from which the exact time difference between ultrasound emitting and ultrasound receiving can be obtained. The whole process of signal demodulation can be found in Figure 1. Several implementations have reported localization with this modulation scheme while using orthogonal codes as the data payload. Orthogonal codes allow to unambiguously retrieve the signal of a given emitter from a captured signal. This signal can be composed of several emitted ultrasound-coded signals together with noise. The Gold code is one of the many existing orthogonal coding schemes and is adopted in the GPS and telecommunications due to the optimal cross-correlation and autocorrelation properties [35]. It is therefore used in several DSSS-based localization schemes [34, 36, 37]. Reported localization accuracies range between 0.1 mm and 10 cm.

3.3. Frequency Hopping Spread Spectrum Modulation. The DSSS method described in the previous paragraph is derived from the field of the telecommunications. Other modulation schemes such as the Frequency Hopping Spread Spectrum (FHSS) have also been proposed [38–40]. FHSS ranging with radio communication has been simulated by Cheng and Chang [41] with an accuracy of approximately 70 cm on a distance of 1500 m. Gonzalez and Bleakley [42] implemented the FHSS modulation scheme for ultrasound localization where both the range and the AoA between the emitter and the receiver could be found accurately. This same principle has also been proposed by Saad et al. [43]. Obtaining the FHSS signals is typically done by demodulating the data on a given set of carriers with different frequencies, where switching from carrier is applied on regular time basis. Demodulating the FHSS-modulated stream at the receiver’s side is typically done by demodulating each of the separate frequency bands [44, 45]. After demodulation, the different data streams are recombined to form the original message. Cheng and Chang [41] demodulate the signal consisting of a pair of carriers by demodulating each carrier with the same method used as for the DSSS demodulation. The demodulation of both signals allows to compute the phase shift between the two carriers so that an estimate on the travelled distance can be obtained. Gonzalez and Bleakley demodulate the FHSS signals with bandpass filters followed by complex envelope and sequence search modules [42]. In the latter case, the obtained localization accuracy in a typical office room is below 1.5 cm in 95% of the measurements. FPGA implementation of the FHSS has also been carried out by Segers et al. [40], with accuracies generally remaining below 2 cm.

3.4. Chirp Spread Spectrum Ranging. The DSSS and FHSS methods allow to obtain a more fine-grained ranging compared to the pulse method. Whereas in the pulse method, a single short pulse of a given frequency is emitted; the signal is modulated with BPSK on a single carrier in the DSSS method and on a set of different frequency carriers in the FHSS method. The DSSS spreading scheme is more resilient to white noise. However, this method is not robust against in-band noise. This problem is attenuated with the FHSS due to the multiscarrier frequency spreading. A more recent technique used in ultrasound indoor localization consists in emitting an ultrasound signal with Chirp Spread Spectrum (CSS). Primarily used in the LoRa(WAN) due to its energy efficiency during radio transmission [46, 47], the advantages of a discrete number of frequency carriers are even more exploited by utilizing the complete frequency range between a given minimum \( f_{\text{min}} \) and maximum \( f_{\text{max}} \) frequency. CSS offers more robust communication regarding disturbances, multipath fading, and Doppler effects [48]. The signal is modulated by using a carrier with a linear increasing or decreasing frequency. The demodulation of chirp-based signals is based on immediate correlation with a matched filter. Since the signal covers a range of frequencies, the correlation peak is more narrow compared to the correlation peak obtained with a single frequency carrier. The principle of CSS in ultrasound ranging is demonstrated by Lazik and Rowe [49] and Khyam et al. [50, 51] with reported
localization accuracies of, respectively, 10 cm and 3.8 to 27 mm. In the latter experiments, an orthogonal chirp sequence is used to retrieve the distance to each of the transmitters. The CSS technique is also used in the RA dio Detection And Ranging (RADAR) system [52, 53] with high-resolution range estimation of radio-reflecting objects. The demodulation of the received signal results in a compressed correlation peak, hence the given name of Pulse Compression Modulation (PCM). This method is also used in the Sound Navigation and Ranging (SONAR) technique, where in recent research, the upcoming of Linear Frequency-Modulated Continuous Wave (LFM-CW) has been demonstrated for enhanced ranging accuracy [54]. Palmese et al. [55] demonstrate the possibility to use CSS in acoustic communication in shallow water with pseudonoise orthogonal sequences. Several CSS modulation schemes have been reported, such as upchirp, downchirp, On-Off Keying (OOK) chirp, and a combination such as Phase Shift Keying (PSK) over CSS.

4. Orthogonal Sequences

The aforementioned ranging methods based on modulation techniques generally make use of orthogonal code sequences [34, 36, 37, 43]. These allow to distinguish a given signal within a set of signals more easily compared to nonorthogonal sequences, and the retrieval of these signals is performed by means of a sequence correlator. Although orthogonal sequences can be obtained with a brute-force algorithm, several orthogonal sequence types, such as the Gold sequence, Walsh-Hadamard sequence, Kasami codes, Golay codes, and Barker codes, have been reported in literature [35, 56–60].

The correlation of a given sequence can be subdivided into autocorrelation and cross-correlation properties. The first correlation property defines the similarity of a given sequence with the shifted version of this same sequence, while the second property defines the similarity between a given sequence with another sequence. Most orthogonal sequences enhance one correlation property while providing a suboptimal value for the other property. In case of ultrasound ranging, several emitted signals can overlap in space and time leading to a degradation of the cross-correlation property. Noise between the emitter and the receiver might affect the quality of a received signal resulting in a degraded autocorrelation property. Therefore, the orthogonal sequences must offer a certain level of immunity for both correlation properties.

4.1. Orthogonal Gold Codes. Orthogonal Gold sequences offer good properties for both the autocorrelation and cross-correlation. Generating these sequences is performed by XORing the results of two maximum length sequences (i.e., “m-sequences” or linear feedback shift register polynomials) and by shifting the resulting sequence. The architecture for generating such sequences with length $N$ is given in Figure 2. Each sequence is formed from a collection of consecutive symbols or elements which in literature are also referred as “chips.” It is proven that the cross-correlation $r_{xy}(\tau)$ of two Gold codes of the same length $N = 2^u - 1$, with $u > 1$, can take one of the following possible correlation output values [35, 56]:

$$r_{xy}(\tau) = \begin{cases} 
-\frac{2^{(u+2)/2} + 1}{2^u - 1}, & \text{for } u \text{ even}, \\
-\frac{2^{(u+1)/2} + 1}{2^u - 1}, & \text{for } u \text{ odd}, 
\end{cases}$$

while the autocorrelation $r_{xx}(\tau)$ of each code is bounded by

$$r_{xx}(\tau) = \begin{cases} 
1, & \text{for } \tau = 0, \\
-\frac{2^{(u+2)/2} + 1}{2^u - 1}, & \text{for } \tau \neq 0, u \text{ even}, \\
-\frac{2^{(u+2)/2} + 1}{2^u - 1}, & \text{for } \tau \neq 0, u \text{ odd}. 
\end{cases}$$

For a given code length, only a few orthogonal sequences can be generated with one set of polynomial seeds [35, 56]. Sequences with a length $N$ equaling 31 and 63 permit to generate up to 6 orthogonal sequences in one set with a normalized cross-correlation value of, respectively, 0.29 and 0.27 [61]. Longer sequences with a length of 1023 chips can be obtained and are used in GPS. Longer sequences offer better auto- and cross-correlation properties. Longer sequences, however, require a longer emitting time which results in less

**Figure 1**: DSSS demodulation as proposed in [36]. The Gold code is expanded to fit the number of samples required to correlate the demodulated $I$ and $Q$ signals. The correlation is performed by applying the Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT).
frequent updates. This is especially the case since a minimum emitting time per chip is required to be efficiently demodulated by the receiver. Therefore, sequences with a length of 31 and 63 chips are used throughout the experiments.

4.2. Binary Chip Orthogonal Sequences. In DSSS and CSS, the signal can be modulated by following a binary pattern of sequences. The Gold codes can be computed by permuting seeds in the two m-sequences. In order to compute the location of a given target in a 3D room, at least four emitting beacons with their respective orthogonal sequences are required. More orthogonal sequences in one set allow to utilize more beacons, which in turn allows to cover a broader area.

A limited number of seeds results in a set of orthogonal codes. A family of criteria to generate appropriate seeds can be found in [62]. In order to validate the sequences, a correlation matrix \( R_{xy}(\tau) \) is defined so that the auto- and cross-correlation of all sequences in a given set can be assessed. In case of \( k_{\text{code}} \) sequences, the correlation matrix \( R_{xy}(\tau) \) can be defined as

\[
R_{xy}(\tau) = \begin{bmatrix}
    r_{1,1}(\tau) & r_{1,2}(\tau) & \cdots & r_{1,k_{\text{code}}}(\tau) \\
    r_{2,1}(\tau) & r_{2,2}(\tau) & \cdots & r_{2,k_{\text{code}}}(\tau) \\
    \vdots & \vdots & \ddots & \vdots \\
    r_{k_{\text{code}},1}(\tau) & r_{k_{\text{code}},2}(\tau) & \cdots & r_{k_{\text{code}},k_{\text{code}}}(\tau)
\end{bmatrix}.
\]

The autocorrelation of the sequences is described by the diagonal of the matrix, while all other elements define the cross-correlation between two sequences. The matrix is reversed symmetric reducing the number of unique cross-correlation to the upper or lower matrix triangle. To balance the correlations around zero, all chips are mapped as \( \{0 \rightarrow -1, 1 \rightarrow 1\} \).

Selecting a set of sequences can be done by comparing the maximum cross-correlation value in the matrix and the autocorrelations with the predefined boundaries. Figure 3(a) depicts the correlation matrix \( R_{xy}(\tau) \) for four orthogonal sequences. Each of the elements of the correlation matrix represents the correlation sequence of \( r_{xy} \) and is obtained by shifting and correlating one of the sequences against the other in a cyclic way for \( \tau \) times. If the length of each sequence corresponds to \( N \), then the total correlation length equals to \( 2 \cdot N - 1 \).

4.3. Multibit Chip Orthogonal Sequences. The FHSS modulation schemes contrast with DSSS and CSS since more than two frequency channels can be used to transmit an orthogonal sequence. An orthogonal pattern based on the Gold sequence with a 2-bit chip is generated here so that up to four different frequencies can be selected. A similar approach as in Section 4.2 can be used where all intermediate steps in the sequence generator (Figure 2) use a 2-bit register, including the feedback and XORing mechanisms.

Computing the correlation on a multibit chip sequence requires special attention. In case of binary chip sequences, it is sufficient to add or subtract 1 to the total correlation in case two chips are, respectively, equal or different. To keep the balance during the correlation process, computing the correlation \( r_{xy,\text{chip}} \) of two chips \( c_1 \) and \( c_2 \) with \( l \) bits per chip is extended to

\[
r_{xy,\text{chip}} = \begin{cases} 
    2^l - 1, & c_1 = c_2, \\
    -1, & c_1 \neq c_2.
\end{cases}
\]

A similar matrix as in Equation (3) can be defined for multibit chip sequences. Figure 3(b) shows the correlation of four
Figure 3: Continued.

$R_{xy}(\tau)$ with 1-bit chips

$(a)$
63 chip long 2-bit sequences which are used during the FHSS experiments, where four simultaneously emitted frequency channels are targeted. Since more multibit chips allow for more combinations, the auto- and cross-correlations with 2-bit chips are lower compared to the correlation properties of 1-bit chips.

5. FPGA Ready Spread Spectrum Modulation

The previously mentioned modulation schemes in Section 3 are all implemented in a similar fashion on the FPGA. Except for the pulse-based modulation scheme, the modulation blocks follow the similar pattern as shown in Figure 4. Emitting ultrasound signals can be achieved by means of transducers, and a small review of available transducers can be found in the previous chapter. The piezoelectric and electrostatic transducers are two viable options for emitting ultrasound signals. The former category of transducers is a low-cost possibility which is also widely available and can easily be employed in low-power embedded systems. This is especially the case since this type of transducers can be driven from a simple circuit or even directly be attached to the driving pins of an FPGA or a microcontroller. The latter alternative offers a much better response with lower transient effects, resulting in more accurate results at the receiver’s side. A drawback of the electrostatic transducers is the higher cost and the high bias voltages which are in the range of hundreds.
of volts. Moreover, these transducers are generally very directional making them less suitable for indoor ranging and localization. Therefore, the modulators are built so that the shortcomings of the piezoelectric transducers are taken into consideration.

5.1. Pulse Modulation. Piezoelectric transducers tend to transient from a first stable state to a second stable state in approximately 10 to 15 periods of an applied sine wave. In order to have sufficient ultrasound energy flowing from the emitter to the receiver, each pulse is modulated during a time frame of 25 sine wave periods. For a sine wave with a frequency $f_{\text{sine}}$ of 25 kHz, this corresponds to 25$f_{\text{sine}}$ or 1 ms.

5.2. DSSS Modulation. The DSSS modulation scheme is based on BPSK so that the phase of the signal changes with 180° when a new chip is applied. This phase change can occur at any time during the modulation. To minimize signal discontinuities as a consequence of the CSS modulation scheme between two consecutive chips, the up- and downchirps start and stop frequencies so that the respective start and stop phase of the signal are equal. This can be achieved by the following equality:

$$s(t_0) = s(t_1).$$

(6)

If we assume $t_0 = 0$, $t_1 = T_{\text{chip}}$, and that the phase $\phi = 0$, then Equality (6) can be rewritten so that

$$A \cdot \sin (0) = A \cdot \sin \left( \frac{\omega_0 \cdot T_{\text{chip}} + \omega_1 - \omega_0}{T_{\text{chip}}} \cdot \frac{T_{\text{chip}}^2}{2} \right)$$

(7)

which leads to

$$\omega_0 \cdot T_{\text{chip}} + \frac{\omega_1 - \omega_0}{T_{\text{chip}}} \cdot \frac{T_{\text{chip}}^2}{2} = 2\pi k_{\text{css}}$$

(8)

and results into

$$f_1 = f_0 + \frac{2k_{\text{css}}}{T_{\text{chip}}}.$$  

(9)

For a given frequency $f_0$ equaling 30 kHz and a $T_{\text{chip}}$ corresponding to 2 ms, all stop frequencies $f_1$ can be found on any offset representing a multiple of $2/T_{\text{chip}}$. The valid $f_1$ frequencies are thus located at 31, 32, 33, …, kHz.

Figure 5 depicts example spectrograms of the different modulation techniques. A code length of 31 chips is used in case of the spreading modulation techniques. The duration of each chip equals 2 ms resulting in a total signal duration of 62 ms. Signal discontinuities causing a wideband noise only occur when two successive chips differ. This noise is minimized when two consecutive chips have the same value.

5.4. CSS Modulation. The last modulation scheme is based on CSS. In this work, the linear chirp modulation scheme is utilized, where the carrier starts from frequency $f_0$ to linearly evolve to frequency $f_1$. The associated sine wave $s(t)$ with amplitude $A$ and constant phase offset $\phi$ for this modulation scheme can be described as [49]

$$s(t) = A \cdot \sin \left( \omega_0 \cdot t + \frac{\omega_1 - \omega_0}{T_{\text{chip}}} \cdot \frac{t^2}{2} + \phi \right).$$

(5)

with $\omega_0 = 2\pi f_0$ and $\omega_1 = 2\pi f_1$.

To help distinguish each emitter from the other emitters, a combination of alternating up- and downchirps is used on top of which a BPSK signal is applied. This BPSK signal follows the same principle as described for the modulating DSSS signals. To minimize signal discontinuities as a consequence of the CSS modulation scheme between two consecutive chips, the up- and downchirps start and stop frequencies so that the respective start and stop phase of the signal are equal. This can be achieved by the following equality:

$$s(t_0) = s(t_1).$$

(6)

...
6. FPGA Signal Demodulation, Correlation, and Ranging

6.1. Pulse Ranging. The ultrasound pulse ranging is applied by allocating a time slot to each ultrasound emitter of 50 ms. The four consecutive time slots allow to have an update rate of 5 Hz of the localization. The pulse emitted consists of a short sine wave of 15 periods at 25 kHz, corresponding to an emitting time of 600 μs. The received signal is demodulated with an envelope detector. From the resulting signal, a threshold method is applied. The distance between the emitter and the receiver is computed by estimating the time between synchronization and the first point at which the signal exceeds the threshold.

6.2. DSSS-Based Ranging. The DSSS method is the first wideband method described in ultrasound ranging and has been derived from the GPS. The ultrasound signal modulation of the DSSS method is done by applying BPSK, where the sequence of ones and zeros is determined by the orthogonal sequence as described in Section 4. Demodulating the received signal is achieved by multiplying the signal with an I and Q reference carrier signal. From here, the signal can be analyzed on similarity with an expected signal so that the distance between the emitter and the receiver can be estimated by searching the resulting peak value. This is performed by means of a signal correlation between 2 signals \( f \) and \( g \). In the time domain, the correlation \( (f*g) \) between the received signal \( g(t) \) and the complex conjugate \( f(t) \) of the expected signal \( f(t) \) can be obtained by

\[
(f*g)(\tau) = \int_{-\infty}^{\infty} f(\tau) \cdot g(t + \tau) dt. \tag{10}
\]

Signals captured by an FPGA by means of an Analog to Digital Converter (ADC) only contain real valued data. Moreover, an FPGA offers a limited amount of memory which limits the number of samples being processed during the correlation. Therefore, the discrete time domain correlation at sample \( k \) for real valued signals with \( m \) past samples can be expressed as

\[
(f*g)[k] = \sum_{i=0}^{m-1} f[i] \cdot g[i + k]. \tag{11}
\]

This first correlation step is graphically represented in Figure 6.

One can obtain the complete correlation of a signal by sliding Equation (11) over all samples when captured by the FPGA. The correlation of the captured signal with a reference signal of length \( m \) will take \( m \) multiplications and additions per demodulated channel, resulting in \( 2 \cdot m \) multiplications and additions per demodulated emitter. For an orthogonal sequence of length \( N = 63 \) and with \( n \) samples per code symbol, the number \( m = N \cdot n \) rapidly exceeds the capabilities of a small- to medium-sized FPGA. This high number of multiplications and additions can be reduced by utilizing a recursive subtraction together with a switch operator. In general, the correlation is computed between a measured signal and a reference signal. In case of correlation with a reference signal based on an orthogonal sequence, this reference signal is composed of long series of ones and zeros. Therefore, the correlation result \( f[i] \cdot g[i] \) of a given demodulated sampled \( g[i] \) can be simplified as

\[
f[i] \cdot g[i] = \begin{cases} g[i], & \text{chip } f[i] = 1, \\ -g[i], & \text{chip } f[i] = 0. \end{cases} \tag{12}
\]

It can be observed that the total correlation can be rewritten as the sum of smaller correlations. If each of the smaller correlations corresponds to the duration of one orthogonal chip with \( n \)-demodulated samples, then all the values of \( f[i] \) inside this correlation are equal. The complete sample chain can also be split into sections of duration \( n \) to form a cascade of sample buffers. We therefore denote \( f_{\text{chip}}[j] \) as the single element representing the \( j \)th chip of the orthogonal sequence so that the correlation \( (f*g)_{\text{chip}}[j][k] \) for each chip

![Figure 5: Spectrograms of the different modulation techniques. In case of the pulse spectrogram, two pulses are emitted at an interval of 50 ms at a frequency of 25 kHz. This same frequency is used for the DSSS modulation (second from left). The spectrum of the signal is enlarged when FHSS is used, while the CSS modulation scheme allows to fully utilize a 24-40 kHz band. In the last 3 cases, phase shifts causing wideband noise due to the orthogonal sequence at an interval of 2 ms can be observed.](image-url)
of all chip correlations results in the original correlation:

\[
(f*g)[k] = \sum_{j=0}^{N-1} (f*\text{chip}[j])[k] \cdot S_g[j][k].
\]  

(16)

For each orthogonal chip symbol, a delaying buffer keeps track of the last \(n\) past samples. The samples are shifted to the next buffer when \(n\) new data samples have been processed. The operation of demodulation, recursive subtraction, and buffering is shown in Figure 8. The switching operation of Equation (12) is shown in Figure 9. This matrix of cells computes the correlation sum for each emitter. Each column corresponds to a chip of an orthogonal sequence while each row corresponds to an emitter. The ranging is computed by finding the correlation peak after a synchronization signal has been sent. A total of \(N\) additions and switching operations are required per demodulation channel and are shown
in Figure 10. The savings in the required computation steps are more pronounced when the number of samples $n$ is increased. In case of a sampling frequency $f_s$ of 500 kSps and with a chip rate of 500 chips per second (cps), the number of samples $n$ per chip equals to 1000. If the orthogonal sequence equals $N = 63$ in length, the number of operations for the optimized form would require 63 switching operations and addition, and 2 additions for the recursive sum. This corresponds to approximately 0.103% of the required number of operations compared with a regular correlation method.
The derived optimized correlation method can be split up so that the recursive sum is computed immediately after the demodulation of the $I$ and $Q$ channels, while the correlation of the $S_n[k]$ values is applied in parallel for each of the ultrasound emitters. The resulting architecture for demodulation and correlation is shown in Figures 8 and 9. The recursive sum and buffering are shown in the first figure, while the final correlation step of Equation (17) is shown in the latter. Each row of the matrix corresponds with the correlation of each emitter, while the recursive sum data are shared so that multiple emitters can be correlated simultaneously.

### 6.3. Ultrasound Ranging Based on FHSS

The second scheme to be evaluated is based on the FHSS principle and relies on the ability to demodulate a signal, which switches several times from frequency carrier. The switching pattern is determined by the applied orthogonal sequence and needs to be demodulated at the receiver’s side to evaluate the distance between an emitter and a transmitter. The general correlation matrix architecture of FHSS is very similar to the DSSS correlation architecture. The major difference resides in the demodulation of the received signals, which is based on frequency demodulation instead of phase demodulation. One type of frequency demodulation is based on the Finite Impulse Response (FIR) filtering and has been proposed by [42], where each frequency is filtered via a bandpass filter. If one assumes that each of the resulting sine wave signals has an average value of zero, then the resulting correlation energy would be zero. Therefore, the absolute value of the obtained filtered signal is computed, after which a running average is taken. The principle of the FHSS demodulation based on the FIR approach is shown in Figure 11.

The FIR-based FHSS demodulation scheme allows to detect several frequency carriers in a given signal. One shortcoming of this principle is the relatively high number of taps required to obtain filters with a steep roll-off effect. This number increases when more frequencies are added in a same frequency range due to steeper roll-off to avoid frequency overlap. Beyond a certain number of taps, a pipelined version of the FIR filter needs to be provided to demodulate the signal in the FPGA, adding additional complexity.

Another demodulation method which allows to discriminate several frequencies in a signal is by implementing a Goertzel filter [63]. The adapted demodulation chain based on the Goertzel filter is shown in Figure 12. This filter can be attributed to the family of first-order Infinite Impulse Response (IIR) filters and allows to efficiently compute a single term of the Discrete Fourier Transform (DFT). The Goertzel filter is commonly known as the Dual-Tone Modulation Frequency (DTMF) filter, which has been used in the first commercial internet routers during the 80’s. The filter is said to be marginally stable due to the poles of the filtering being located on the unitary circle. Care must be taken when low precision arithmetic is used on long sample sequences.

An adapted version with a sliding mechanism with a comb stage is proposed in [63] and is further adapted towards the applications of indoor ranging. The resulting filter architecture and equations are, respectively, shown in Figure 13 and Equations (18) and (19).

\[
\begin{aligned}
Q[k] &= c_g \cdot Q[k-1] - Q[k-2] + x[k] - x[k-N_{bin}], \\
M[k] &= Q[k-1]^2 + Q[k-2]^2 - c_g \cdot Q[k-1] \cdot Q[k-2],
\end{aligned}
\]

with the Goertzel constant
\[
c_g = 2 \cdot \cos \left( \frac{2\pi \cdot f_{\text{target}}}{f_s} \right).
\]

It can be noted that the computed output is the direct computation of the magnitude $M[k]$ instead of the complex-valued DFT. Although the proposed method has been demonstrated with double precision arithmetic, a signed precision arithmetic variant for FPGAs is developed so that FHSS-modulated signals can be demodulated without the area intensive double precision arithmetic. A few boundary conditions, however, have to be met in order to make the sliding Goertzel effective.

1. The comb stage at the beginning of the filter must delay each sample for at least $N_{bin}$ samples. This delay corresponds to the number of bins $N_{bin}$ dividing the frequency spectrum with sampling frequency $f_s$ of a DFT. The value $N_{bin}$ can be found by computing the ratio between the sample frequency and the frequency resolution $f_{\text{res}}$ so that $N_{bin} = f_s / f_{\text{res}}$.
2. The coefficient $c_g$ is obtained via Equation (19). This coefficient is bounded between $c \in [0, 2]$ for $f_{\text{target}} \in [0, f_s/2]$.
3. The implementation of the algorithm on FPGA is preferably achieved with fixed-point or signed arithmetic. With the former type, bit truncation can arise leading to a runaway effect of the algorithm. With signed precision arithmetic, this can be avoided at the condition that a sufficient number of bits are provided at each step of the computation.

The bitwidth at each step of the computations can be computed in the assumption that a periodic and finite signal is processed. If each sample $x[k]$ has $b_s$ bits, then the differential delayed output at the comb stage requires $b_s + 1$ bits. The values in the $Q$ factors grow due to feedback mechanisms until the comb stage takes effect, which is when at least $N_{bin}$ samples have been processed. The number of additions before the comb effect takes place results in a bitwidth of $b_s + \log_2(N_{bin}) + 1$. The coefficient $c_g$ is bounded between 0 and 2, which in turn requires an additional bit. At last, the coefficient $c_g$ also needs to be scaled in a signed precision format with $b_s$ bits. This bitwidth is also temporarily reflected after the multiplication step of $Q[k-1]$ where the bitwidth equals to $b = b_s + b_s + \log_2(N_{bin}) + 2$ bits. At all other stages, the bitwidth equals to $b = b_s + \log_2(N_{bin}) + 2$ bits.

A major limitation of this filter, however, resides in the comb structure at the beginning of the filtering. In the...
assumption that a steady signal is applied at sample \( k \), then the filter stabilizes at sample \( k + N_{\text{bin}} \). This implies that for a higher demand in resolution, the filter requires more time to converge. Another limitation is due to the recursive structure of the filter with “infinite” response. Although the filter may be designed to avoid infinite valued outputs, the exact time at which an output occurs may slightly differ depending on the previous samples, resulting in a less accurate timing result compared to results obtained with a FIR counterpart.

6.4. Ultrasound Chirp Ranging. The last ultrasound-modulated signal technique is based on chirp signals. The modulation technique requires a matched filter in order to demodulate the signals properly. The orthogonal sequence, as described in Section 4, allows to subdivide the complete signal into two shorter sequences of up- and downchirps. Therefore, only these shorter sequences are generated as matched filters and stored into a lookup table of the FPGA. The principle of the chirp demodulation is shown in Figure 14. Due to the variability in the frequency and phase.

**Figure 11:** FPGA ready architecture for FHSS FIR-based demodulation. Two frequency channels are depicted here, but more channels can be implemented depending on the modulated signal. The signal is first demodulated by applying a bandpass FIR filter for each channel, which is then smoothed out with a running average (RA), before the recursive sum is computed.

**Figure 12:** FHSS Goertzel-based FPGA ready demodulator.

**Figure 13:** Adapted version of the Goertzel DTF algorithm computing the magnitude of a given signal at frequency \( f_i \) on the FPGA.
of the signal, the matched filter needs to correlate the sampled signal for each sample with the computed chirp signal according to Equation (11). When applying a full correlation on an FPGA, two parameters have to be taken into consideration. The first parameter is the clock frequency \( f_{\text{clk}} \) at which the programmable logic of an FPGA is running. Typical FPGAs are clocked at frequencies between 50 MHz and 200 MHz [64–66]. The second parameter is the sampling frequency \( f_s \), at which the signal is captured. In the applications with ultrasound signals from MEMS microphones, the sampling frequency is chosen between 100 kSps and 500 kSps to comfortably fulfill the Nyquist theorem. The total maximum allowed correlation time in terms of clock cycles \( C \) between two consecutive correlation cycles can be obtained by taking the floor of the ratio between \( f_{\text{clk}} \) and \( f_s \):

\[
C = \left\lfloor \frac{f_{\text{clk}}}{f_s} \right\rfloor. \tag{20}
\]

In case \( f_s \) equals to 500 kSps on a Zynq7020 FPGA running at 100 MHz, then \( C \) would be equal to 200 cycles. Considering that each orthogonal chip of the sequence can be emitted for a duration of 2 ms, the number of samples \( n \) to be processed for each chip corresponds to 1000. Therefore, a pipelined version of the chirp correlator is implemented, where the complete correlation is subdivided in subcorrelation banks. The correlation sum is also propagated along the pipelined structure. If \( N_{\text{corr}} \) subcorrelators are chained consecutively, each of the subcorrelators computes the correlation of \( n/N_{\text{corr}} \) samples so that the correlation of the complete chirp requires \( N_{\text{corr}} \) multiplications and additions.

The CSS modulation is expected to modulate on a large bandwidth with a flat frequency response. The transducers used in the experiments, especially the piezoelectric transducers, however, do not provide a flat response on a wide frequency range. Therefore, a limited bandwidth with a frequency response with a maximal variation of 3 dB in the magnitude is chosen during the experiments for a satisfying result.

The demodulation of the chirp-based signals is generally applied with a matched filter, the result of the filtering corresponds to a compressed pulse or sync pulse (Figure 15). The characteristics of this pulse are related to the total bandwidth \( B = f_2 - f_1 \) and the total duration \( T_{\text{chirp}} \) so that the compressed pulse output \( O_{\text{cp}}(t) \) corresponds to

\[
O_{\text{cp}}(t) \approx \sqrt{T_{\text{chirp}}} \cdot B \cdot \sin \left( \pi Bt \right) / \pi Bt. \tag{21}
\]

From Equation (21), the amplitude of the pulse \( \sqrt{T_{\text{chirp}}} \cdot B \) can be deduced. More importantly, the spatial resolution \( R_{\text{chirp}} \) of the main lobe is inversely proportional to \( B \) and linearly proportional to the speed of sound \( c \) so that [67]

\[
R_{\text{chirp}} = \frac{c}{2 \cdot B}. \tag{22}
\]

This last equation allows to understand that the total correlation will result in a thin peak. This contrasts with the correlation of the other modulation techniques where the correlations describe a linear slope towards the main peak. With CSS, overlapping signals from different emitters can be picked up separately. Correlation peaks from one emitter, however, may overwhelm correlation peaks from more distant emitters resulting in false positives. To avoid these false positives, a few additional changes to Equation (12) are proposed. If the CSS signal contains a BPSK-modulated signal, then the demodulated signals will contain an upchirp \( S_{g,\text{up}}[j][k] \) or a downchirp \( S_{g,\text{down}}[j][k] \) sync pulse with a zero or \( \pi \) phase. In case of a zero phase, the resulting sync pulse will point upwards, while a phase \( \pi \)
results in a downwards-oriented sync pulse. Therefore, the correlation of each chirp chip $f_{ch}^{j}$ results in

$$(f * g)_{ch}^{j}[k] = \begin{cases} S_{g_{\text{up}}}^{j}[k], & j \text{ even}, f_{ch}^{j}[j] = 1, S_{g_{\text{up}}}^{j}[k] \geq 0, \\
-S_{g_{\text{up}}}^{j}[k], & j \text{ even}, f_{ch}^{j}[j] = 0, S_{g_{\text{up}}}^{j}[k] < 0, \\
S_{g_{\text{down}}}^{j}[k], & j \text{ odd}, f_{ch}^{j}[j] = 1, S_{g_{\text{down}}}^{j}[k] \geq 0, \\
-S_{g_{\text{down}}}^{j}[k], & j \text{ odd}, f_{ch}^{j}[j] = 0, S_{g_{\text{down}}}^{j}[k] < 0, \\
0, & \text{ other cases.} \end{cases}$$

When the last condition is met, the correlation over all chips is set to zero to avoid false positives. An additional signal between the correlation modules in the matrix is added to propagate this error state. The latter is reflected in Figure 16.

### 7. FPGA Resource Consumption

The proposed algorithms in previous sections allow to achieve a significant reduction in terms of computation resources from embedded systems like FPGAs. In this section, the required FPGA logic for each technique is investigated so that the scalability towards small- and medium-sized FPGAs can be estimated. This scalability is estimated for the number of emitters which can be processed simultaneously as well as for the length of the orthogonal codes. The orthogonality of the sequences, i.e., the number of maximally allowed sequences in one group of sequences, is not taken into account. Tables 1–5 summarize the used resources of the FPGA per configuration. The target FPGA is a Zynq7020 mounted on the Z-turn board [65]. This FPGA offers a total of 53200 Look-Up Table (LUT), 106400 flip-flops (registers), 220 multiplier modules (Digital Signal Processing (DSP)), 39900 F7 and F8 multiplexers (muxes), and 140 BlockRAM (BRAM) tiles of 36 kB. For each of the implementations, the sampling frequency is set to 500 kSps with a chip duration $T_{ch}$ of 2 ms. The length of the orthogonal sequences is set to 31 or 63 chips, while the number of emitters ranges from 1 to 8. The length of the orthogonal sequence is omitted for the pulse ranging method since this method does not utilize these sequences. Special care is taken so that the available number of BRAMs and DSPs is optimally utilized.

#### 7.1. Pulse Ranging

In case of the pulse ranging method, the resource consumption remains very low with less than 1% of the total available logic. Table 1 depicts the FPGA occupancy. The only DSP and BRAM modules in use are utilized by the “signal conditioning” module (19). The number of LUTs and registers tends to grow linearly with the number of emitters to be processed.

#### 7.2. DSSS Ranging

The processing chain for the DSSS method requires much more logic from the FPGA. The resource consumption of each setup is shown in Table 2. The signal conditioning also requires 1 DSP and 1 BRAM, while the demodulation requires 1 BRAM for the running average of a length of 256 and the correlation structure requires the remaining logic. The bitwidth of each of the demodulated channels “delay buffer” is set to 18 bits, so 2 parallel channels (i.e., 36 kB of data) fit into one BRAM. With a total number of $n = 1000$ samples per chip to correlate, the number of BRAM equals to the number of chips to be correlated. Therefore, only $2 + 31$ or $2 + 63$ BRAM tiles are required. The number of DSPs increases with 2 for each new emitter to be processed. These additional DSPs are used in the “ranging” modules after the correlation matrix. The other 2 DSPs are used during the demodulation of the $I$ and $Q$ signals. The number of flip-flops and LUTs scales linearly with the number of chips and emitters added to the processing.

#### 7.3. FHSS Ranging

The second spread spectrum method to be evaluated on the FPGA is the FHSS and is detailed in Table 4. The FHSS method is evaluated with the same sampling and demodulation configuration as for the DSSS method. The number of demodulation channels, however, is set to 4. The results of the FIR- and Goertzel-based approaches are, respectively, shown in Tables 3 and 4. In both cases, a noticeable increase in resource consumption due to increased complexity of the correlation matrix can be observed. Instead of only selecting the appropriate demodulation channel, the channels are added or subtracted according to Equation (4). One can also note that the number of DSPs and BRAM only change as a function of the number of chips. In case of the FIR-based approach, only 5 DSP modules are required. These modules represent the multiplications of the signal conditioning (1 DSP) and the 4 modules in use by each of the demodulators.

Depending on the number of chips, a total of 69 or 133 BRAM are required to store the intermediate values. The signal conditioning, the FIR demodulators, and the running average modules require, respectively, 1, 4, and 4 BRAMs. The remaining BRAMs are used by the delay buffers. This contrasts with the Goertzel-based FHSS method. For this method, one DSP is used in the signal conditioning while 28 other DSP modules are used in the 4 Goertzel demodulators. The number of BRAM corresponds to one module used for signal conditioning and 2 in the Goertzel demodulators. The remaining 62 or 126 BRAMs delay the samples in the
correlation matrix. A bitwidth of 18 bits per demodulation is chosen for a sufficient demodulation resolution, resulting in 2 required BRAM modules for each delay buffer. The Goertzel-based method also requires more logic compared to the FIR-based method due to the increased complexity of the demodulators. This method also demands less BRAM since less intermediate sample-storing and filtering steps are required. Note that the last solution for both methods, i.e., the solution with an orthogonal sequence length of 63 and the number of emitters to be processed set to 8, cannot fit the Zynq FPGA targeted since the number of LUTs exceeds the available LUTs in the device. A different architecture, however, is proposed in [39]. A simplified version of the current correlation approach, with a sampling frequency set to 100 kSps and 2 demodulation channels, allows to detect up to 16 concurrent emitters. The proposed correlation matrix in [39] also does not offer a pipelined architecture which limits the maximum attainable sampling frequency.

7.4. CSS Ranging. The last method, the CSS method, requires more logic compared to the DSSS method but less than the FHSS method. Two demodulation channels for the up- and downchirp are used. The number of bits in the delay buffers, however, is higher than 18 bits resulting in a double demand of BRAM (i.e., 60 or 124). The remaining BRAMs are subdivided in 1 BRAM for the signal conditioning and 10 BRAMs for the pipelined chirp correlation. An additional optimization is performed where the first delay buffer is omitted due to redundancy so that 2 BRAM tiles can be spared. The number of LUTs and registers increases linearly with the length of the orthogonal sequence and the number of emitters to be processed. In this architecture, a switch operation is performed at each element of the correlation matrix resulting in a linear increase in muxes utilized. The number of DSP modules remains stable at 21 for all combinations of the CSS modulation scheme.

Table 1: FPGA resource consumption for the pulse ranging method.

<table>
<thead>
<tr>
<th># emitters</th>
<th>Pulse ranging</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>LUTs</td>
<td>696</td>
</tr>
<tr>
<td>Registers</td>
<td>423</td>
</tr>
<tr>
<td>Muxes</td>
<td>25</td>
</tr>
<tr>
<td>DSPs</td>
<td>1</td>
</tr>
<tr>
<td>BRAMs</td>
<td>1</td>
</tr>
</tbody>
</table>

7.5. Conclusions. It is possible to derive the maximum number of emitters which can be processed simultaneously on a Zynq 7020 FPGA. Table 6 lists the maximum number of emitters for each of the configurations. For all configurations, the longer length sequence approximately results in a double resource consumption. Therefore, compared with a sequence length $N = 31$, the number of the emitters which can be processed is at least reduced to 50% when a sequence length $N = 63$ is used. The number of correlation channels also affects this number. In case of the two-channel DSSS and CSS correlation methods, the number of emitters is almost the double compared with the four-channel FHSS correlation method. The pulse method offers the highest possible number of emitters which can be detected. It can be noted that this number is based on the duplication of the logic for each emitter. Since this method does not make use of multiple-access principles but relies on TDMA, only one ranging module could suffice, resulting in an infinite number of emitters for the pulse ranging method.

8. Implementation

Generating and processing ultrasound waves in the FPGA are possible with the methods described in the previous sections. In order to emit and to receive the ultrasound signals, and to evaluate the performance of the methods, a hardware
Table 2: FPGA resource consumption of the DSSS method.

<table>
<thead>
<tr>
<th># emitters</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUTs</td>
<td>3837</td>
<td>5387</td>
<td>8615</td>
<td>14847</td>
<td>6884</td>
<td>10145</td>
<td>16639</td>
<td>29344</td>
</tr>
<tr>
<td>Registers</td>
<td>4204</td>
<td>5796</td>
<td>9147</td>
<td>15584</td>
<td>7803</td>
<td>11057</td>
<td>17554</td>
<td>30255</td>
</tr>
<tr>
<td>Muxes</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>61</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>61</td>
</tr>
<tr>
<td>DSPs</td>
<td>5</td>
<td>7</td>
<td>11</td>
<td>19</td>
<td>5</td>
<td>7</td>
<td>11</td>
<td>19</td>
</tr>
<tr>
<td>BRAMs</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
</tr>
</tbody>
</table>

Table 3: FPGA resource consumption of the FHSS FIR-based method. In case of \( N = 63 \) and 8 emitters (bold numbers), the required amount of LUTs exceeds the available logic on the FPGA and can thus not be implemented.

<table>
<thead>
<tr>
<th># emitters</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUTs</td>
<td>6537</td>
<td>9438</td>
<td>15644</td>
<td>27680</td>
<td>11067</td>
<td>17428</td>
<td>30057</td>
<td>55150</td>
</tr>
<tr>
<td>Registers</td>
<td>9205</td>
<td>13843</td>
<td>23418</td>
<td>41959</td>
<td>17301</td>
<td>27016</td>
<td>46293</td>
<td>84534</td>
</tr>
<tr>
<td>Muxes</td>
<td>104</td>
<td>104</td>
<td>104</td>
<td>133</td>
<td>104</td>
<td>104</td>
<td>104</td>
<td>133</td>
</tr>
<tr>
<td>DSPs</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>BRAMs</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>133</td>
<td>133</td>
<td>133</td>
<td>133</td>
</tr>
</tbody>
</table>

Table 4: FPGA resource consumption of the FHSS Goertzel-based method. In case of \( N = 63 \) and 8 emitters (bold numbers), the required amount of LUTs exceeds the available logic on the FPGA and can thus not be implemented.

<table>
<thead>
<tr>
<th># emitters</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUTs</td>
<td>8798</td>
<td>11728</td>
<td>17996</td>
<td>30095</td>
<td>13380</td>
<td>19804</td>
<td>32476</td>
<td>57568</td>
</tr>
<tr>
<td>Registers</td>
<td>9783</td>
<td>14421</td>
<td>23996</td>
<td>42537</td>
<td>17879</td>
<td>27594</td>
<td>46871</td>
<td>85114</td>
</tr>
<tr>
<td>Muxes</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>61</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>61</td>
</tr>
<tr>
<td>DSPs</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
<td>29</td>
</tr>
<tr>
<td>BRAMs</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
</tr>
</tbody>
</table>

Table 5: FPGA resource consumption of the CSS method.

<table>
<thead>
<tr>
<th># emitters</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUTs</td>
<td>7922</td>
<td>8480</td>
<td>12746</td>
<td>19029</td>
<td>10906</td>
<td>13136</td>
<td>20572</td>
<td>33465</td>
</tr>
<tr>
<td>Registers</td>
<td>8273</td>
<td>9716</td>
<td>12659</td>
<td>18419</td>
<td>12851</td>
<td>15733</td>
<td>21469</td>
<td>33065</td>
</tr>
<tr>
<td>Muxes</td>
<td>83</td>
<td>134</td>
<td>236</td>
<td>469</td>
<td>83</td>
<td>193</td>
<td>236</td>
<td>469</td>
</tr>
<tr>
<td>DSPs</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>BRAMs</td>
<td>71</td>
<td>71</td>
<td>71</td>
<td>71</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
</tr>
</tbody>
</table>

Table 6: Maximum possible number of emitters for each method which can be processed simultaneously on the Zynq 7020 FPGA. The “X” in the column of the pulse ranging methods denotes that no sequences are used.

<table>
<thead>
<tr>
<th>Method</th>
<th>Pulse</th>
<th>DSSS</th>
<th>FHSS FIR</th>
<th>FHSS Goertzel</th>
<th>CSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N )</td>
<td>X</td>
<td>31</td>
<td>63</td>
<td>31</td>
<td>63</td>
</tr>
<tr>
<td># emitters</td>
<td>475</td>
<td>31</td>
<td>15</td>
<td>16</td>
<td>16</td>
</tr>
</tbody>
</table>
implementation with transducers is constructed. The complete implementation scheme is shown in Figure 17.

The implementation is based on the Z-turn board from the MYiR company [65]. This board houses a Zynq 7020 FPGA and offers 3 Peripheral Module Interface (PMOD) connectors so that up to 24 digital Input-Output (IO) can be used.

These connectors allow to emit ultrasound signals, which are emitted with the 250ST180 piezoelectric crystal [68]. An amplification circuit based on Metal-Oxide-Semiconductor Field-Effect Transistor (MOSFET) is built for each individual transmitter and allows to convert the 3.3 V from the digital pins from the FPGA to the maximum allowed voltage for the transducers, i.e., 20 V. Up to four amplifier channels are provided.

A connection with the analog receiving hardware enables to capture the ultrasound waves. The SPU0410LR5H-QB [69] MEMS microphone converts the acoustic information and transforms it into electrical signals. The receiving hardware is optimized to capture signals between 24 kHz and 45 kHz, with a higher amplification effect between 30 kHz and 38 kHz, while diminishing the effects of the resonance frequencies of the piezoelectric transducer at 25.5 kHz and 40 kHz. The gain of the circuit is variable with an amplification factor between 20 dB and 40 dB. The amplification is based on an active filtering circuit which minimizes the nonflat frequency responses resulting from the combination of both the 250ST180 piezoelectric crystal and the SPU0410LR5H-QB microphone. This board also provides the single-ended AD7091 ADC [70], which allows to sample signals up to 1 MSPs with a resolution of 12 bits. The FPGA collects the samples via this ADC by means of the Serial Peripheral Interface (SPI) protocol. Aside of the ADC, the board also provides a probe point which allows to debug the analog signals and to store the samples on a computer in case of later debugging of the proposed algorithms. The resulting MEMS receiving hardware is shown in Figure 18.

The FPGA implements the functional blocks of the ultrasound correlation, ranging, and emitting methods as described in previous sections. The developed algorithms are connected to the hardware via additional modules.

At first, the ultrasound correlation and ranging process the samples from a microphone. In this setup, the ultrasound analog signals are converted with the ADC before being processed by the FPGA logic. An ADC module in the FPGA allows to read the digitized samples via SPI, which are then
conditioned in the “signal conditioning”. The “ranging” module implements the different correlators as well as the ranging modules allowing to estimate the range between the emitters and the MEMS microphone.

At second, the ultrasound signals for the different emitters have to be generated. A block containing four emitters “US emitters” allows to instantiate the ultrasound modulator blocks.

At last, a UART communication link allows to transmit the computed ranges to a computer. This same link allows the computer to issue a new ranging request. A master block in the FPGA logic permits to start emitting ultrasound signals, to reset the ranging modules, and to send the ranges after a predefined amount of time to the computer. The “Serializer” block converts the individual ranges into a series of bytes which can be transmitted by the UART module.

The complete overview of the FPGA setup is shown in Figure 19. The “Sync” signal allows to synchronize external devices with the FPGA. The rising edge of this signal indicates the start of the emitted signals. In the experiments, it can be used to synchronize the samples of the Picoscope so that the ranging can also be performed on the computer afterwards.

9. Results

In this section, the proposed ranging methods are evaluated. The evaluation is performed in two steps. In the first step, all implementations are evaluated in a single-access environment. This allows to estimate the capability of a given algorithm to obtain the range between the receiving MEMS microphone and one emitter. In the second step, two emitters are used so that the ability of retrieving each emitter is calculated. In both situations, one emitter is placed at distances ranging from 20 cm up to 8 m, in steps of 20 cm. During the dual-access experiments, a second emitter is placed at a fixed position of 2.40 m from the MEMS microphone receiver. The latter allows to estimate the ratio up to which both emitters can be processed by the algorithms on FPGA. All methods are evaluated via this two-step evaluation, except for the pulse ranging where at all times, only one emitter is used simultaneously. During all experiments, the voltage for the emitters is set to 20 V and corresponds to the maximum allowed amplitude for the piezoelectric transmitters. The signal amplitude on receiving hardware is calibrated so that peak-to-peak voltage of the signal corresponds to approximately 75% of the maximum input range of the ADC when a single emitter is placed at a distance of 1 m. This amplitude allows to measure signals which are closer than 1 m or allows to have signal superposition of two signals without signal distortion. The complete measurement chain comprises all the modules as depicted in Figure 17. The Picoscope is the 2203 model and samples the signals at the same sampling frequency as the FPGA, i.e., 500 kSps. The samples are stored for later processing in case a problem occurs during the measurements. A small application on the computer ensures that the obtained distances from the FPGA are stored in a Comma-Separated Value (CSV) file. At each position, 300 distance measurements are taken. The post processing of the distances, such as outlier rejection, regression analysis, and the calculation of the estimation of the accuracy and the precision, is computed via a Matlab script. The accuracy and the precision, respectively, denote the ability to retrieve the range and to which extent the measurements are close to each other. For the former, the real distance is subtracted from the measured distance to obtain the error, while for the latter, a 90% confidence interval is used. The measurements are performed in the recently built I.2.03 auditorium at the Vrije Universiteit Brussel which is equipped with acoustic dampening materials. An additional acoustic dampening carpet is used to minimize reverberations from the floor. The measurement environment is shown in Figure 20.

9.1. Pulse Ranging. The pulse ranging method is evaluated by estimating the range between a single emitter and the receiver, where a short pulse with a frequency of 25 kHz and a duration of 1 ms is used. All emitters are programmed to emit sequentially after each other with an interval of 50 ms. During the measurements, no signal overlap is possible. Therefore, only single-access measurements are taken into account. Distance outlier rejection is applied when a measured distance corresponds to the maximally allowed range in a number of samples. In the case of pulse ranging, this value corresponds to 50 ms·500 kSps or 25000 samples. This ranging method also relies on a threshold which stops the counter of the ranging module when triggered by a pulse.
signal. During the experiments, this threshold is set to 6.25% of the total attainable signal strength. Lower values would result in false positives, while higher values would result in an insensitive ranging module. Figure 21 depicts the ranging between a single emitter and the receiver. One can note that with this method, ranging up to 5 m can be performed with an accuracy generally below 2 cm. The precision is quite high and in 90% of the cases also remains below 2 cm. Beyond 5 m, all samples indicate the maximum attainable value. The measurements were stopped after 6 m since no ranging could be established. In this case, 100% of the measured data is also rejected.

9.2. DSSS Ranging. The second method to be evaluated is the DSSS method. During the experiments, a chip rate of 500 cps is chosen and the orthogonal code is implemented with a length of 63 chips. With a sampling frequency \( f_s \) set at 500 kSps, the number of samples \( N \) per chip equals to 1000 samples, resulting in a total number of samples for each orthogonal sequence of 63000. Therefore, measured distances with a value lower than this threshold can be rejected since these lead to negative distance values. The measurements were performed in single and dual access, and the results are shown in Figure 22.

In a single-access environment, the distance between the emitter and the receiver can be estimated up to 7.8 m. The distances are estimated with an accuracy and precision which are generally lower than 3 cm. The 90% confidence interval, however, shows a tendency to increase while the distance between the emitter and the receiver is increased. The accuracy remains stable for all measured ranges.

In case of dual access, one of the emitters (Emitter 1) is placed at the different position while the other emitter (Emitter 2) is placed at 2.4 m from the receiver during all the measurements. In case of the first emitter, the distance can be estimated from 20 cm up to approximately 6 m, with an accuracy and precision which are comparable to the single-access environment. From the distance of 6 m, the number of rejected measurements increases drastically to reach a rejection rate of almost 100%. The emitter placed at the fixed position, by contrast, shows a very stable distance measurement from 2 m. The confidence interval and the accuracy generally remain below 1 cm. Measurements when the first emitter is placed at lower than 2 m result in a high rejection rate and in less accurate and precise range estimation.

9.3. FHSS Ranging. The two demodulation approaches of FHSS are evaluated, and in both cases, the carrier frequencies are set to 26, 30, 34, and 38 kHz. An orthogonal sequence with a length of 63 chips with a chip rate of 500 cps is chosen for the experiments. In a similar way as during the DSSS experiments, distances with a correlation peak located at less than 63000 samples are automatically rejected. The results of the FHSS method based on the FIR method are shown in Figure 23. In single-access mode, all distances from 20 cm up to 8 m can be estimated with an accuracy and precision generally remaining below 2 cm. Although the accuracy remains stable throughout the experiments, the precision shows a small tendency to slightly increase from 0.5 m to 1 cm, while the distance between the emitter and the receiver decreases. In dual-access mode, the distance between Emitter 1, which is placed at regular intervals of 20 cm, and the receiver can be estimated with an accuracy of approximately 3 cm. The precision interval increases from less than 1 cm at a distance of 20 cm and increases drastically beyond 6 cm. This precision interval shows less optimal results at 5.2 and 5.8 m which is probably caused by multipath effects of the signal. The precision and accuracy of the emitter positioned at
2.4 m, however, remain stable and below 1 cm except when the other emitter is at less than 1.8 m from the receiver.

The results of the FHSS method based on the Goertzel demodulator are shown in Figure 24. In single access, the Goertzel-based demodulator allows to obtain the range between the emitter and the receiver with an accuracy and precision generally remaining below 2.5 cm. The precision shows a tendency to slightly deteriorate with the increasing
range, while the accuracy remains constant. Both the accuracy and precision are slightly less optimal compared with the FIR based. This is mainly caused by the family of demodulator, which in case of the FIR filters offers a finite response, while a response with infinite effects can be expected from the IIR-based Goertzel filter. In case of dual-access mode, both the accuracy and the precision of the first emitter deteriorate but generally remain below 10 cm. These drastically deteriorate when the first emitter is beyond 6.4 m. The measurements beyond 7 m all result in a rejection rate of 100%, resulting in off-scale values, and are therefore not shown. The accuracy and precision of the fixed emitter (Emitter 2) improve when the first emitter is placed at more distant locations from the receiver.

9.4. CSS Ranging. The CSS ranging method is evaluated with the same chip rate and length of orthogonal sequences as the FHSS and DSSS methods. The same approach of distance rejection can be used so that correlated distance values lower than 63000 samples are automatically rejected. A frequency range of 4 kHz between 34 kHz and 38 kHz is chosen to emit the CSS signals. This frequency range offers the flattest response on the hardware and results in more accurate range estimations. The results of the single-access and dual-access ranging are shown in Figure 25. In both the single- and dual-access measurements, the emitter which is set at different distances is detected with an accuracy and precision of approximately 1 cm. In dual-access mode, this emitter is less reliably detected beyond a distance of 6 m. The measurements, however, depict a phenomenon at 3.6 and 3.8 m, where the distance is falsely estimated to be 1 m more distant compared to the real position. A plausible cause is that reflections at these distances result in a higher correlation peak than the original correlation peak. The distance estimation of the second emitter in dual-access mode beyond 1.6 m shows an accuracy and a precision which are both much lower than 1 cm. One can note that, except for the false distance reading at 3.6 and 3.8 m, the distances between the emitters and the receivers are well estimated and result in a constant accuracy and precision for all the measured distances. When a range cannot be detected, the algorithm results in a clear zero distance which can be discarded. The latter is depicted by the zero distances of the blue graph of the first emitter beyond 6 m.

9.5. Discussions. All methods allow to obtain the distances between the emitters and the receiver. In case of the pulse ranging method, distances up to 5 m can be measured, while distance of 8 m can be obtained with the multiple-access methods. All methods offer a centimeter level accuracy and precision. In general, the precision tends to increase while the distance between the emitter and the receiver increases. An exception to this rule is when the CSS method is used, where a constant precision can be observed. The accuracy tends to offer a less reliable result with increasing distances for all methods. In single access, all methods reliably obtain the range between the emitter and the receiver. In dual-access mode, the ranges between the two emitters and the receiver can be obtained up to approximately 6 m. This phenomenon can be interpreted as the near-far problem, which is known in the field of radio communications. The DSSS and CSS methods, however, can also compute the range with the most distant emitter for a few points beyond 6 m. At points where the distance could not be retrieved reliably, the CSS outperforms the DSSS by returning a zero distance, resulting in an automatic rejection of the distance.

The results also have to be put in perspective with the hardware and the FPGA used. The pulse-based ranging method requires the least resources. This method is therefore very popular in combination with microcontrollers due to the simplicity. Based on the resource consumption and hardware requirements, this method outperforms all other methods. Only single-access ranging is possible, and this method can be prone to in-band noise due to the use of a single frequency carrier. In case of the multiple-access schemes, the DSSS method requires the least FPGA resources. The DSSS method also relies on a single frequency carrier, allowing to use the same hardware as the pulse ranging method. Although multiple access is possible, this method can also suffer from in-band noise. This in-band noise problem can

![Figure 25: CSS range estimation. In the left graph, one can find the results of the ranging between a single emitter and the receiver while in the two other graphs, the ranging in dual-access mode can be observed. Note that, when the range against an emitter is well estimated, the accuracy and precision generally remain constant at approximately 1 cm.](image-url)
be alleviated by relying on multiple frequency carriers such as in the FHSS method. This method offers similar results compared with the DSSS method. A major shortcoming, however, is the increased resource consumption. The hardware must also be capable of supporting the multiple frequencies. The CSS method requires hardware with a flat frequency response between the lower and upper frequency. In case this condition is met, the CSS ranging method offers the most reliable distance estimations, while requiring a similar amount of FPGA logic as the DSSS method.

10. Conclusions

In this chapter, ultrasound-based indoor ranging techniques are studied and implemented. Several multiple-access modulation techniques are studied, and the complete implementation on embedded systems such as FPGAs is performed. These comprise the pulse-based ranging, ranging based on DSSS, FHSS, and CSS. Special care is taken so that the modulated waves are emitted in an optimal way and signal discontinuities are avoided whenever possible. The regular demodulation schemes proposed in literature are optimized so that the correlations can be performed with a much lower required number of processing logic. This demodulation is based on recursive logic and allows to save several factors of magnitude in DSP logic. The ultrasound-emitted waves are also modulated by using orthogonal sequences which allow to distinguish multiple signals from each other. The regular single-bit Gold coding scheme is adapted towards a multibit coding scheme for the FHSS method, where several carrier frequencies are used simultaneously. Different sets of orthogonal sequences can be derived and offer an optimum balance between code length and cross- and autocorrelation properties.

A dedicated analog MEMS-based microphone signal amplifier and signal conditioning is designed and built. This hardware compensates the nonlinear frequency response resulting from both the piezoelectric ultrasound emitter and the MEMS-based microphone so that signals between 24 kHz and 45 kHz can be picked up with a frequency response of approximately 3 dB. Experiments are performed for each method, in both single and dual accesses. The results show that the pulse ranging method is the less reliable ranging method, while the CSS ranging offers the most constant accuracy and precision. All multiple-access methods show a tendency of suffering from the near-far problem. In case of the DSSS and FHSS, this near-far problem can cause false measurements, while this results in a zero distance in case of the CSS method. In all cases, the accuracy and precision generally remain at a centimeter level. A slight degradation, however, can be observed for the DSSS and FHSS ranging methods, while a constant accuracy and precision is observed while using the CSS method. The CSS also offers the ability to automatically reject measured ranges when the ultrasound signal does not offer a sufficient quality, such as in dual-access ranging.

All methods are evaluated in a stationary environment. All emitters and receivers remain at a fixed position during each of the measurements. The proposed algorithms, in combination with the modulation and demodulation schemes, however, do not inherently support the ability to detect moving targets. Currently, the FHSS demodulation method offers the best resilience to Doppler shifts since it is based on frequency information, while the other methods rely on the signal phase. The current implementations require Doppler compensation in case of moving target detection.

Data Availability

All datasets and measurements are available on request via the corresponding author. Please send an email to laurent.segers@vub.be to obtain this datasets.

Conflicts of Interest

The authors declare no conflict of interest.

Authors’ Contributions

During this research, the work has been divided among the authors of this paper. Laurent Segers developed and implemented the ranging theorems on FPGAs, collected the results, and wrote the main manuscript. An Braeken and Abdellah Touhafi reviewed the manuscript and provided constructive feedback.

Acknowledgments

We would like to thank the Vrije Universiteit Brussel for providing the material and rooms to perform the measurements, especially the newly built acoustic dampened auditorium.

References


