Universidad de Alcalá

Escuela Politécnica Superior

Departamento de Electrónica

Efficient generation and correlation of binary codes obtained from complementary sets of sequences for ultrasonics systems

Author: Mª Carmen Pérez Rubio

Advisors: Dr. Jesús Ureña Ureña, Dr. Álvaro Hernández Alonso

Ph. D. Thesis
Extended Abstract

2008

Contents

Mathematical symbols

$\delta[\tau]$	Kronecker delta.
τ	Time delay.
θ	Correlation bound.
θ_{AC}	Auto-correlation bound.
θ_{CC}	Cross-correlation bound.
μ	Number of simultaneous users.
$C_{a,b}$	Aperiodic cross-correlation function between a and b.
DW	Data-width.
f_θ	Nontrivial correlation factor $f_\theta = 0.5 \cdot \theta_{AC} + 0.5 \cdot \theta_{CC}$.
K	Family size in LS codes.
L	Code-length.
L_0	Length of CSS when they are used to construct LS and T-ZCZ codes.
L_{Ms}	Length of a macro-sequence generated from CSS.
M	Family size of PR, CSS and T-ZCZ codes. In CSS it also represents the number of sequences in a code set (i. e., the family size is equal to the set size).
m	In codes derived from CSS is the base-two logarithm of the number of sequences in the set, i. e., $m = log_2(M)$.
N	In CSS, is a natural number, with value $N = \frac{log_2(L)}{2}$, representing the number of stages in the generator and correlator filter. In pseudo-random sequences, is a natural number representing the degree of the primitive polynomial used in their construction.
N_{SM}	Number of carrier periods.
O_f	Over-sampling factor.
S_i	i-th set of CSS.
$s_{i,j}$	Sequence j-th of the i-th CSS.
$\overline{s_{i,j}}$	Reverse of sequence $s_{i,j}$.

Z_C	Size of the zero correlation zone around the origin of LS and T-ZCZ codes.
Z_L	Size of the lateral zero correlation zones in T-ZCZ codes.

Abbreviations

AC	Auto-Correlation.
BPSK	Binary Phase Shift Keying.
CC	Cross-correlation.
CSS	Complementary Set of Sequences.
CDMA	Code Division Multiple Access.
DTOF	Difference in Times-Of-Flight.
EGC	Efficient Golay Correlator.
ETZG	Efficient Three Zero correlation zone codes Generator.
ETZC	Efficient Three Zero correlation zone codes Correlator.
FFT	Fast Fourier Transform.
FPGA	Field Programmable Gate Array.
FWT	Fast Walsh Transform.
IZ	Interference Zone.
ISI	Inter-Symbol Interference.
LFSR	Linear Feedback Shift Register.
LS	Loosely Synchronous codes.
LPS	Local Positioning System.
MAI	Multiple Access Interference.
M-CSS	Complementary Sets of M Sequences.
M-ESSG	Efficient Set of M Sequences Generator.
M-ESSC	Efficient Set of M Sequences Correlator.
PR	Pseudo-Random sequences.

PVDF Polyvinylidene Fluoride.

QPSK Quadrature Phase Shift Keying.

RF Radio Frecuency.

SIC Successive Interference Cancellation.

TOF Time Of Flight.

T-ZCZ Three Zero Correlation Zone codes.

UCSS Uncorrelated Complementary Sets of Sequences.

VHDL Hardware Description Language for VHSIC.

VHSIC Very High Speed Integrated Circuit.

ZCZ Zero Correlation Zone.

1 Introduction

1.1 Motivation

Many sensory systems based on time-of-flight (TOF) measurements have incorporated signal processing techniques previously used in radar theory. This is the case of ultrasonic systems, where binary codes with similar properties to Gaussian noise are emitted and, afterwards, the corresponding echoes are detected by means of correlation with the original pattern. Thus, a notable improvement in features such as their temporal precision, spatial resolution and robustness to noise is achieved when the selected codes have good auto-correlation (AC) properties. Furthermore, by choosing a set of codes with low cross-correlation (CC) values between them it is possible to have several transducers emitting at the same time over the same spatial zone, allowing measurements from different locations simultaneously. This multi-mode operation capability increases the velocity at which the system extracts information from the environment, and it is especially important when a set of measurements under similar conditions is required.

The overall performance of these systems, and in general of those that use Code Division Multiple Access (CDMA) techniques, strongly depends on the properties of the codes used [Fan04, Che07]. Nevertheless, the design approach of these codes do not usually take into account many real application scenarios, where asynchronous detection, multipath effect, aperiodic emission, etc. are quite common. This is the case of privacy-oriented ultrasonic Local Positioning Systems (LPS), where a set of beacons are located at the environment. They simultaneously emit their corresponding codes with a certain period. Then, a set of portable receivers compute their positions asynchronously, using multilateration algorithms, as the echoes could arrive at any moment depending on the transducer and receiver positions [HW03, UHJ$^+$07]. Many of these systems use traditional CDMA codes which are interference-limited [Syl67, Gol67a, Kas68]. In consequence, they have to heavily rely on complex auxiliary systems, such as precision power control or interference cancellation techniques to supress both the Inter-Symbol Interference (ISI) and Multiple Access Interference (MAI) [DMUH$^+$06, HH06]. Obviously, the key to improve the effectiveness of these systems implies finding out the adequate encoding scheme, what explains the considerable growth of the research devoted to improve the code design process. A clear example could be the active proposal of novel codes with zero correlation zones (ZCZ) for quasi-synchronous environments [FFT07, LGZ07, FFTL08, RC08], most of them constructed from complementary sets of sequences (CSS) [TL72].

On the other hand, another limitation is imposed by the matched-filter technique used in the reception stage to detect the emitted codes, as it means an increase of the signal processing tasks that can put into risk the real-time operation capability of the system. Hence, a lot of efforts are aimed at the development of efficient hardware implementations of the corresponding correlators so as to reduce their computational load and achieve real-

time requirements [Bud89, Pop97, Pop99, ÁUM+04, DMUH+07].

In this thesis the current CDMA codes are considered for their application on ultrasonic sensory systems. Furthermore, a new encoding scheme with three zero correlation zones is proposed. Finally, efficient structures for the generation and correlation of codes based on CSS are developed.

Notice that, even though the proposed structures and codes are thought to improve the performance of ultrasonic systems based on TOF measurements, they can be applied to many other fields such as non-destructive evaluation (NDE) and medical imaging [NSLT03], cryptography [KMM07] or mobile and wireless communications [CLY+06].

1.2 Background

The importance of the codes used in pulse compression-based systems is difficult to overemphasize. In fact, the overall performance of the system can be changed only by changing the type of code used or its length [Fan04].

Walsh-Hadamard sequences [Syl67] and orthogonal Gold codes [DO99] have perfect orthogonality at zero time shift, but at nonzero time delays the AC and CC functions are normally very poor (in fact, some codes are just the shifted versions of others in the same set so their CC at nonzero time delays can reach values equal to the main AC peak). Hence, an accurate synchronism of a small fraction of one code-bit is required which is in general not easy to achieve and, even so, interferences appear because of the delays introduced by the multipath propagation.

Other traditional CDMA codes are pseudo-random (PR) sequences, such as m-sequences [Gol67b], Gold codes [Gol67a] or Kasami sequences [Kas68]. All of them exhibit nonzero off-peak AC and CC values, being the small set of Kasami sequences the more optimal regarding the Welch bound [Wel74]. However, the number of codes in a Kasami family could be not enough and, what is more restrictive, they were designed based only on their periodic correlation properties. Nevertheless, many ultrasonic systems require aperiodic emissions: less energy is required, and echoes from the multipath are not mistaken with direct echoes coming from a consecutive emission. In these cases, codes with good aperiodic correlation properties are necessary.

Binary Barker sequences [Bar53] have been widely used because of their aperiodic AC properties (the absolute value of their aperiodic AC sidelobes is bounded by 1). Unfortunately, to date, only Barker binary sequences with length up to 13 are found and there is only one Barker code with this maximum length. Thus, they are neither adequate for noisy environments nor for multi-mode operation. By employing polyphase Barker codes [CG96] it is possible to obtain lengths up to 31, which are still not enough in most applications. Furthermore, to detect all the phases (up to 7200) complex correlators must be used, which means a high computational load.

Since with one single sequence it is impossible to remove completely and simultaneously sidelobes on both the AC and CC functions [Wel74], several authors rely on complementary codes [Che07, ÁUM$^+$06, WA07]. Complementary codes were firstly considered by Golay [Gol61] and each one consists of a pair of binary sequences with the same length whose addition of aperiodic AC and CC functions have perfect properties. Nevertheless, they provide only two mutually uncorrelated pairs and are not suitable for multi-user environments. Complementary sets of sequences (CSS) [TL72] are a generalization of Golay pairs containing more than two sequences. The elimination of the constrains in the number of sequences yields on a high process gain and also more than two mutually uncorrelated sets. However, the number of uncorrelated sets is limited by the number of sequences for every set. Besides, in many systems it is not possible to simultaneously transmit the sequences of a set, thus emission schemes that arrange the sequences of the set by concatenation or by interleaving are used. These transmission schemes degrade the ideal properties of the sum of the correlation functions, so ISI and MAI appear [DMUH$^+$06].

Recently, interest in codes with Zero Correlation Zones (ZCZ), as Loosely Synchronous (LS) codes, has considerably increased [Fan04, SBH01, Li03, ZYH05, FFTL08]. Such kind of codes are thought for applications in quasi-synchronous CDMA (QS-CDMA) systems where the relative time delay among different users does not exceed a certain limit. The correlation functions of these codes have zero values around the in-phase shift, providing null ISI and MAI when the length of the ZCZ is larger than the multipath spread and the relative multiple access delays. However, in periodic emissions it is necessary to insert guard intervals (or zero gaps) to maintain its properties [SBH01, WH06]. As a solution, in [ZLH05] new sequence pairs with ZCZ in the middle and terminal parts of the sum of their aperiodic AC and CC functions have been proposed, and called Three-ZCZ (T-ZCZ) pairs. These sequence pairs avoid to insert extra guard intervals in case of periodic emissions and allow more simultaneous users than Golay pairs at the expense of assuming a set of interferences in a restrained area. The interferences outside the ZCZ in both LS codes and T-ZCZ pairs are higher than those of traditional CDMA codes, so when these codes operate in a completely asynchronous manner they will encounter more serious ISI and MAI. Thus, it is important to assure that the relative time delay arrive within the ZCZ around the origin. In an asynchronous LPS system it can be achieved by placing the emitters close enough to each other while delimiting the positioning area of the mobile receivers, so that the arrival of the different TOF do not exceed the size of this ZCZ.

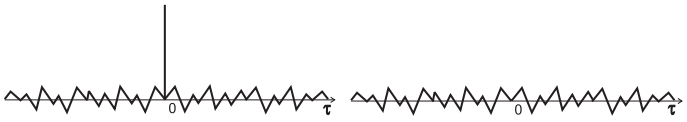
For the sake of giving a succinct overview, Table 1 gives a brief summary of the sequences discussed in this thesis, whereas Figure 1 represents their correlation properties.

Code	Synchronism required	For periodic emissions	$N, m, n \in \mathbb{N}$	Maximal length	Number of sequences in the code ($\star$)	Number of uncorrelated codes	ACF en $\tau = 0$	max. ACF in $\tau \neq 0$	max. CCF
Walsh-Hadamard [Syl67]	$\tau = 0$	No	$\forall N$	2^N	1	2^N	2^N	—	—
Barker [Bar53]	$\forall \tau$	No	—	13	1	1	13	-1	—
Preferred m-sequences [Gol67b]	$\forall \tau$	Yes	$\forall N$	$2^N - 1$	1	($\star\star$)	$2^N - 1$	-1	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$
Gold [Gol67a]	$\forall \tau$	Yes	$\forall N$	$2^N - 1$	1	$2^N + 1$	$2^N - 1$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$
Kasami small set [Kas68]	$\forall \tau$	Yes	$\mathrm{mod}(N, 2) = 0$	$2^N - 1$	1	$2^{\frac{N}{2}}$	$2^N - 1$	$\lvert 1 + 2^{\lfloor \frac{N}{2} \rfloor} \rvert$	$\lvert 1 + 2^{\lfloor \frac{N}{2} \rfloor} \rvert$
Kasami large set [Kas68]	$\forall \tau$	Yes	$\mathrm{mod}(N, 2) = 0$	$2^N - 1$	1	$2^{\frac{N}{2}}(2^N + 1)$	$2^N - 1$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$	$\lvert 1 + 2^{\lfloor \frac{N+2}{2} \rfloor} \rvert$
CSS [DMUH+07]	$\forall \tau$	No	$\forall\ N, m$	2^{mN}	2^m	2^m	2^{mN}	0	0
LS [ZYH05]	$\lvert \tau \rvert \leq Z_C$	No	$\forall\ N, m$	$2^{m(2+N)} + (2^m - 1)Z_C$	1	2^{2m}	$2^{m(2+N)}$	0, in $1 \leq \lvert \tau \rvert \leq Z_C$	0, en $0 \leq \lvert \tau \rvert \leq Z_C$
T-ZCZ: method 1 [ZLH05]	$\lvert \tau \rvert \leq Z_C$	No	$\forall\ m, n$	2^{2n+m+1}	2	2^{n+1}	2^{2n+m+2}	0, in $1 \leq \lvert \tau \rvert \leq Z_C$	0, en $0 \leq \lvert \tau \rvert \leq Z_C$
T-ZCZ: method 2 [ZLH05]	$\lvert \tau \rvert \leq Z_C$	No	$\forall\ m, n$	2^{2n+m}	2	2^{n+1}	2^{2n+m+1}	0, in $1 \leq \lvert \tau \rvert \leq Z_C$	0, in $0 \leq \lvert \tau \rvert \leq Z_C$
T-ZCZ: method 3 [ZLH05]	$\lvert \tau \rvert \leq Z_C$	No	$\forall\ m, n$	2^{2n+m-1}	2	2^{n+1}	2^{2n+m}	0, in $1 \leq \lvert \tau \rvert \leq Z_C$	0, in $0 \leq \lvert \tau \rvert \leq Z_C$

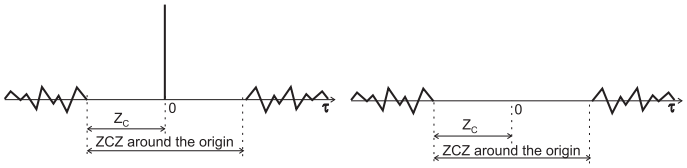
$\star$ When the number of sequences in a code is greater than one, the correlation results of the last three columns correspond to the sum of correlation functions of the sequences that form the code.

$\star\star$ The number of uncorrelated m-sequences vary depending on their length, and is not enough in most of the cases.

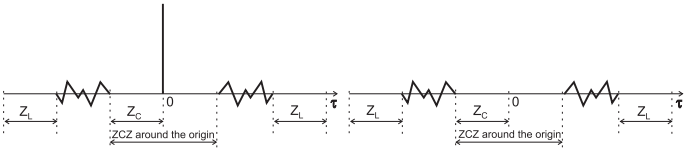
Table 1: Summary of various classes of codes.



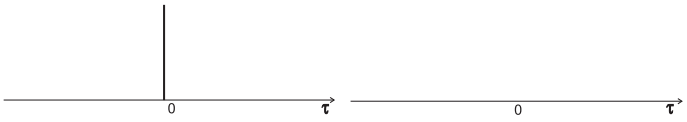
(a) Aperiodic AC and CC functions of pseudo-random sequences.



(b) Aperiodic AC and CC functions of LS codes.



(c) Sum of aperiodic AC and CC functions of T-ZCZ pairs.



(d) Sum of aperiodic AC and CC functions of CSS.

Figure 1: Correlation performance of different codes.

In practice, if the detection of the aforementioned codes are computed by means of straightforward correlators [PG93], which requiere high computational cost, the system throughput is constrained. A solution to reduce the number of operations to perform can be the use of a correlator based on the Fast Fourier Transform (FFT) algorithm [Bud89, TZW08]. However, the number of operations is reduced at the expense of performing a large number of multiplications (in contrast to the straightforward implementation that requires only additions and subtractions when binary sequences are used). Some authors propose efficient algorithms for specific codes. As an example, in [Bud89] an algorithm based on the Fast Walsh Transform (FWT) is developed, consisting only in additions and subtractions, for the periodic correlation of m-sequences. A similar proposal is described in [Pop97] for orthogonal Gold codes. In [Bud91] and [Pop99] an Efficient Golay Correlator (EGC) is proposed for the detection of Golay complementary pairs. The EGC reduces the number of additions and subtractions to perform from L (the sequence length) to $2{\cdot}log_2(L)$. In a similar way, in [ÁUM+04] an efficient generator and pulse compressor for complementary sets of four sequences is presented. Recently, this algorithm has been extended for the efficient generation and correlation of complementary sets of any number $M = 2^m$ of sequences with length $L = M^N$, where $m, N \in \mathbb{Z}^+$ [DMUH+07]. Regarding the ZCZ codes, to date, the correlation is performed by means of straightforward or FFT based schemes [XY05, TZW08].

As has been stated before, a higher evolution in TOF-based ultrasonic systems came

with the inclusion of signal coding and pulse compression techniques. First works were focused on NDE for medical and testing applications, starting with pseudo-random codes [WMRR72, EM78, Eli80], and later on, moving into Golay pairs [Fur82, HG88, NSLT03, LWF+07]. Other works have used the encoding of the ultrasonic signal to implement advanced sensors (with several emitters and/or receivers) mainly for the detection of obstacles in robotics, using Barker codes [PAC93, UMG+99], pseudo-random sequences [JB98], Golay pairs [DUM+99, HUG+04] or complementary sets of more than two sequences [ÁUM+06, DMUH+05].

In LPS, the use of ultrasound signals offers a low cost solution with a centimeter-level accuracy. However, several limitations can be identified in current LPS systems, such as the proposed in [HHWW99, PMBT01, RM01], due to the use of narrowband transducers emitting tone signals: single-user access, lack of identification encoding and noise sensitivity [HH06]. With the aim of solving the previous problems, the *Dolphin* units make use of hand made PVDF transducers with wide lobe and frequency range, encoding the emissions with 511-bit Gold codes and using a Succesive Interference Cancellation (SIC) [Vit90] postprocessing algorithm for minimizing the "near-far" effect caused by off-peak correlation values. Accuracy is 4.9 *cm* with a 95 % confidence level in the privacy-oriented configuration. Other LPS using Gold codes are [VUM+05, PSK07]. Similarly, the *3D-Locus* system [PJG+07] uses broadband acoustic transducers to emit unpaired Golay codes. Since only one sequence of each Golay pair is emitted, the sum of correlation functions is not accomplished and interferences appear. To minimize this effect, the authors have made a selection of a proper subset of unpaired Golay pairs with similar CC values as Gold codes. The advantage of using these Golay sequences is that the EGC can be used for compute the correlations. Other possibilities such as the concatenation, interleaving or even the QPSK modulation of both sequences of the Golay pair, also involves the loss of their ideal correlation properties. The *3D-Locus* system is capable of achieving an accuracy below 1 *cm* with a 90 % confidence level in an area of 4 m^2. In the search for an optimal family of codes to be emitted in a LPS, Genetic Algorithms (GA) have also been used. Thus, in [ÁGG+07] a new family containing 8 codes with length 127 bits, and a superior performance than those Gold codes with identical length, is presented for their use in an ultrasonic LPS. From this study, it can be stated that the pursuit of family codes with adequate correlation properties is still an active line of research in the field of ultrasonic signal processing.

All the improvements, derived from the use of encoded signals and their corresponding correlators, demand the use of suitable device or architectures for their implementation [HG88, HUM+05]. Programmable architectures have been used to process the ultrasonic signals of sonar systems in several works [KK95, APKC92, JB98, HOB+00]. However, they do not provide the desired performances for real-time processing. A good alternative is the use of configurable solutions as FPGAs since they allow a high parallelism degree, with several execution lines and pipeline-type estructures [GN98, HUM+05]. Examples using FPGAs to compute the correlation of Golay pairs and complementary sets of four

sequences can be found in [HUH+03] and [ÁHU+06] respectively.

1.3 The aims of this thesis

In this thesis an appropriate selection of the codes to be emitted in a simultaneous multi-emission system, with non-periodic emissions and asynchronous detection, is carried out. The goal is to find the optimal codes to improve the performance of ultrasonic sensory systems in terms of noise immunity, capability of simultaneous measurements and precision. With that purpose, current CDMA codes are evaluated, new encoding schemes are proposed and efficient generators and correlators are developed and implemented in configurable hardware. Specifically, the tasks involved are:

- First, an exhaustive search has been done to find the set of codes with lower aperiodic AC and CC sidelobes in each one of the most promising current CDMA families (Kasami, Golay, CSS, LS and T-ZCZ codes). Next, the performance of the previous codes is analyzed and compared taking into account the following criteria of goodness: low aperiodic AC and CC sidelobes, process gain, number of available pseudo-uncorrelated or uncorrelated codes in the family, and the ability of minimizing the blind zone that appear in ultrasonic sensor systems when the same transducer works as emitter and as receiver [HUM+06].

- The second objective of this thesis implies, on one hand, a new hardware design for the efficient generation and correlation of CSS (ESSG and ESSC respectively). On the other hand, it involves the proposal and hardware implementation of new generation and correlation algorithms for LS codes. In all cases, the developed implementations can be easily adapted to the requirements of the applications and have been designed to minimize their computational cost. Furthermore, the performance of every proposal has been tested in comparison with a straighforward implementation.

- Next, novel methods to generate sequence pairs with Three Zero Correlation Zones (T-ZCZ) in the sum of their correlation functions are proposed. With these new methods it is possible to obtain lower interferences in the area where interferences are constrained, called Interference Zone (IZ), and narrower IZ than those obtained with previous methods [ZLH05]. Also, efficient generation and correlation algorithms for each method are derived. These algorithms offer an efficient hardware implementation that significantly reduces the total number of operations to be performed in comparison with straightforward ones. Thus, real-time operation can be achieved.

- Finally, the performance of the studied codes is experimentally evaluated in an ultrasonic LPS system developed for that purpose.

2 On the search of optimal codes for asynchronous detection

In this section the proper selection of code sets that are most suitable for its use in a CDMA system with simultaneous, non-periodic emissions and asynchronous detection is carried out. The chosen codes are required to have a very sharp aperiodic AC function to ensure proper acquisition and high process gain, as well as very low aperiodic CC values to reduce MAI among different users. Furthermore, good partial AC properties would be desired to improve the detection efficiency when only a section of the emitted code is received.

2.1 Procedure of code selection

The selection and optimization of code families with good aperiodic correlation properties have received a considerable interest [Lah95, Kri03]; however, because the aperiodic correlation properties of codes are significantly difficult to analyze, only a few analytical results have been published [SP80, FD96]. Thus, the search of codes with good aperiodic correlation properties has been investigated mainly on empirical bases via exhaustive search. Nevertheless, since the number of possibilities increases considerable with the length of the codes, the search is limited by the current computation power (for instance, for a set of M codes with length L, L^M new sets can be constructed by combining the different phases of the codes; and for each set of phases, there are $\frac{1}{2}M(M-1)$ CC functions to be evaluated). A solution could be the use of sub-optimal procedures that limit the search to fewer possibilities, or to stop the search when the factor used to evaluate the codes reach a certain threshold. Due to the characteristics of the evaluated codes the first option is the one used in this thesis, as will be explained later.

To quantify the performance of the codes, the bound θ is employed. This is a commonly used figure of merit that gives the maximum non-trivial correlation value $\theta = \max(\theta_{AC}, \theta_{CC})$, where θ_{AC} stands for the maximum sidelobe obtained from all AC functions and θ_{CC} is the maximum value obtained in the CC between all the pairs contained in a family. Both the θ_{AC} and θ_{CC} usually appears divided by the AC main peak. Knowing that, the search procedure consists of the following steps:

- First, for a specific family set, the AC and CC between the codes are computed.
- Then, for the selected number of simultaneous users, the combinations that leads to lower $f_\theta = 0.5 \cdot \theta_{AC} + 0.5 \cdot \theta_{CC}$ values are identified, hence, the reduction of AC and CC sidelobes are equally weighted up. From these user combinations the ones that have closer θ_{AC} and θ_{CC} values are selected. Thus, when a group of codes have equal f_θ values, only the ones with lower θ are considered. From the remained group of codes, which can be considered optimum regarding to the bound, the ones with the lower θ_{CC} are selected.
- Finally, the two previous steps are repeated for every family set, so at the end of the

search the codes with better performance are selected.

After the better codes has been selected, they have also been evaluated in terms of partial AC. In many ultrasonic sensory systems, transducers work as emitters and receivers in the same scanning process. Consequently, during the emission process the reception stage is disabled to avoid the emission to be coupled. As a result, there is an area close to the transducers (the blind zone) that cannot be scanned. The use of larger codes to increase the process gain also involves to raise the duration of the emission interval, and thus, the size of the blind zone. In such cases, only the last part of the emitted code will be received, which implies a decrease of the AC mainlobe and the increase of the AC sidelobes. Obviously, depending on the partial AC properties of the emitted codes, the arrival of the AC mainlobe could still be detected, allowing the detection of reflectors in the blind zone.

The analyzed codes are: the small set of Kasami sequences, CSS, LS and T-ZCZ pairs. The small set of Kasami sequences is composed by $M = 2^{\frac{N}{2}}$ binary sequences with length $L = 2^N - 1$, where N is even, and it is constructed from the addition, modulo-2, of a m-sequence with its properly decimated version [Kas68]. In the performed search, Kasami sets obtained from all possible m-sequences with length $L = 2^N - 1$ (where N is even) and the sets obtained from the cyclic shift of these m-sequences are considered.

In this work, CSS generated as indicated in [DMUH$^+$07] are considered. Every set has $M = 2^m$ codes with length $L = M^N$, where $m, N \in \mathbb{Z}^+$. The sum of the aperiodic correlation functions of the M sequences of the set is zero for all non-zero shifts; furthermore, the sum of the corresponding sequences of two uncorrelated sets is zero for any shift. Nevertheless, since more than one sequence is assigned to a transducer, it is necessary to employ emission schemes that allow their emission in the smallest time and under similar conditions. A simple method consists in linking or interleaving the sequences of a set, constructing a new larger code with length $L_{Ms} = M \cdot L$, which has been called *macro-sequence*. After that, by using a digital phase modulation, the macro-sequence can be transmitted. However, this arranging method destroys the ideal properties of CSS, as was demonstrated in [DMUH$^+$06]. In this work, a computer search algorithm has been carried out to find preferred sets of macro-sequences that minimize the effects of the emission method. This study extends the results showed in [DMUH$^+$06] for a large number of simultaneous users and code-lengths. For both methods, concatenation and interleaving, all combinations of M and L that leads to a same macro-sequence length have been taken into account; furthermore, for each $M \cdot L$ combination, L different macro-sequences obtained by changing the initial CSS are considered. Notice that the phase of these macro-sequences have not been changed since the ESSC is coupled to CSS with zero shift.

LS and T-ZCZ codes exhibit an interference free window (Z_C) around the origin of the AC and CC functions. In this scenario, only the codes arriving outside this Z_C will cause ISI and MAI. The performance of LS and T-ZCZ codes inside the IZ has been studied and, to reduce the interferences when a echo falls within this zone, a search of the groups of

codes with lower bounds θ and lower number of non-zero values in the IZ has been carried out. Two different methods of generating LS codes are evaluated: one uses Golay pairs [SBH01] and the other CSS[1] [ZYH05]. In both cases, all the possibilities derived from using different Golay pairs or CSS have been considered. Any shift have been applied to these codes, because the algorithms proposed in this thesis for the efficient correlation of LS codes take advantage of the EGC and ESSC associated to non-shifted versions of the Golay pairs and CSS that compose them. Similar considerations can be applied to T-ZCZ codes.

2.2 Performance of the selected codes

The code selection results for each one of the family codes evaluated can be found in Appendix A.

In a completely asynchronous system the codes with better performance in terms of bound θ are the small set of Kasami sequences. It has been tested that an increase in the number of simultaneous users hardly affects the bound, as can be seen in Figure 2. In this figure the θ obtained in case of periodic (PER) emissions has also been represented. It can be observed that, apart from the singular case of $L = 15$, higher bounds are obtained in the aperiodic case. Increasing the length of the codes implies lower bounds; in fact, it asymptotically achieves twice the Levenshtein bound [Lev99]. Furthermore, the blind zone can be reduced even if large codes are used. For instance, assuming the reception of the last 51 bits: for a code with length $L = 64$ bits (the 80 % of the code is considered), it means a $\theta_{AC} = 0.3$; whereas for a code with length $L = 1023$ (only the 5 % of the code is considered) is $\theta_{AC} = 0.49$, so the mainlobe could still be detected.

The performance of preferred macro-sequences generated from the contatenation or interleaving of CSS is shown in Figure 3, for macro-sequence lengths of $L_{Ms} = 64$ and $L_{Ms} = 256$ bits. The highest bounds are obtained when $L = M$. These bounds also increase with the number of simultaneous users. When the sequences of the CSS are arranged by concatenation lower bounds are achieved than if interleaving is used. Furthermore, the concatenation method have a better performance when the beginning of the code is loss. Nevertheless, when long codes are used, with $L >> M$, the differences between concatenation and interleaving decrease. In these situations, it would be better the use of macro-sequences generated from interleaving, since changes in the environment will affect in the same manner to all sequences. On the other hand, although macro-sequences generated from CSS have higher interferences than Kasami codes, its use can be justified because of

[1] In [ZYH05] an iterative algorithm is proposed that allows, thorough P iterations, to obtain $K = M^{(1+P)}$ codes with length $L = M^{(1+P)} \cdot L_0 + (M-1)Z_C$ using CSS. The method proposed in [SBH01] can be considered as a special form deduced from this scheme, since Golay pairs are a special case of CSS with $M = 2$ sequences. However, increasing P means increasing to a great extent the complexity of the efficient generators and correlators that could be derived. As a solution, only one P iteration has been allowed, increasing the length and number of codes by using different families of CSS. Consequently, the aforementioned methods could be considered as different (they only coincide when $M = 2$ and $K = 4$).

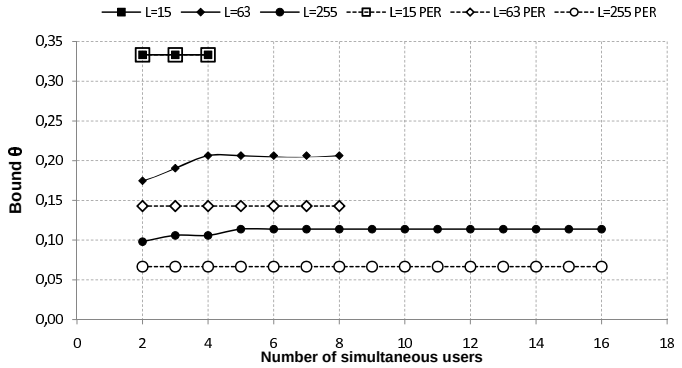
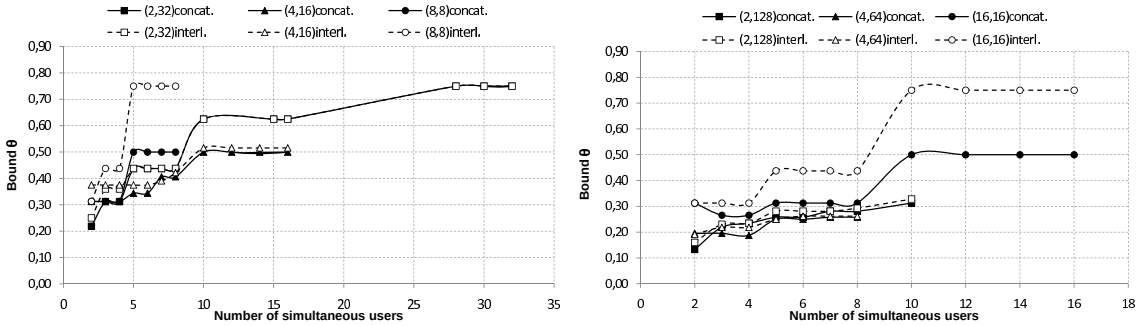


Figure 2: Bound θ in Kasami sequences depending on the number of simultaneous users and code length.



(a) $L_{Ms} = 64$ (b) $L_{Ms} = 256$

Figure 3: Bound θ in macro-sequences generated from CSS with size (M, L).

the ESSC, that reduces the total number of operations to be performed in the correlation process. Thus, when the length of the codes is high in comparison with the number of simultaneous users, the use of macro-sequences could be a good option. Furthermore, they allow more code lengths and families with a large number of pseudo-uncorrelated codes.

In QS-CDMA systems, codes with ZCZ are more robust if multipath propagation exists compared with the aforementioned codes, as long as the relative delays does not exceed the size of the ZCZ. Two types of encoding with those properties have been evaluated: LS codes and T-ZCZ pairs. LS codes generated from CSS [ZYH05] have lower bounds in the IZ than LS codes generated from Golay pairs [SBH01], as can be observed in Figure 4. Nevertheless, to achieve the same process gain and the same ZCZ sizes of those for Golay generation method, larger codes have to be used with the CSS one. In both cases, it has been checked that in the IZ null values also appear, being more numerous when the number of available codes in the family decreases. It is interesting to note that these zero values always appear in the same positions of the correlation functions. On the other hand, the size of the off-peak values in the IZ increases with the number of simultaneous users and available codes in the family, being, in general, higher than those for Kasami and macro-

sequences. More results regarding the behavioral of LS codes in the IZ can be found in [**Pérez et al. 2007c**].

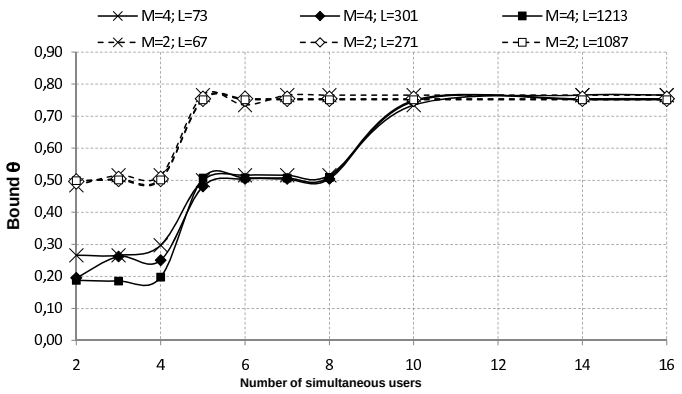


Figure 4: Bound θ in the IZ of LS families with 16 codes generated as shown in [SBH01] (M=2) and in [ZYH05] (M=4).

In [ZLH05] three different constructions of T-ZCZ pairs are proposed. The construction 1 has the wider ZCZ and lower number of non-zero values in the IZ. Nevertheless, the construction 2 has the lowest bounds in the IZ; in fact, the interferences within this area do not exceed half the magnitude of the mainlobe, as can be observed in Figure 5. When comparing LS codes and T-ZCZ pairs that leads to AC mainlobe of the same magnitude, it can be found that LS codes have larger ZCZ than T-ZCZ, and imply a bigger reduction of the blind zone. Another advantage of LS codes is that each code only consists of one sequence. On the contrary, since ZCZs in T-ZCZ codes only appear when the correlation functions of both sequences of the pair are added in phase, they need the use of QPSK modulations or other emission schemes.

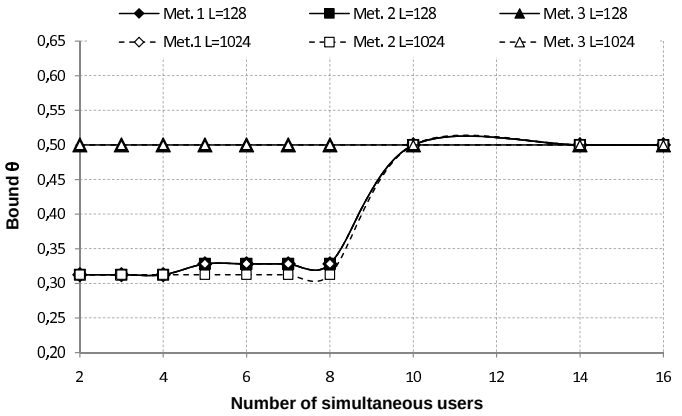


Figure 5: Bound θ in the IZ of T-ZCZ families with 16 codes.

3 Efficient hardware implementation of generators and correlators for CSS and LS codes

An optimal method for detecting the presence of a known waveform in Gaussian-noise environments consists of correlating the received signal with this known emitted waveform. However, as mentioned in the Introduction, standard straighforward correlators are often unfeasible for long sequences. Thus, the feasibility of the employed encoding schemes strongly depends on the availability of efficient architectures that reduce the computational load and hardware complexity of the despreading process.

Firstly, this section presents the FPGA hardware implementation of the generic algorithm proposed in [DMUH+07] for the correlation of CSS. Two design approaches utilizing reconfigurable hardware to gain the required flexibility in terms of correlator parameters have been proposed. One requires the configuration of the design at the pre-synthesis stage, whereas the other allows the run-time configuration of the system parameters. Secondly, the pre-synthesis implementation has been adapted to the transmission scheme with macro-sequences, generated either by concatenating or by interleaving the complementary sequences. Hence, it has been possible to determine what macro-sequences offer a better performance in terms of resource requirements and operating frequencies of their corresponding correlator implementation. Finally, novel methods to efficiently generate and correlate LS codes have been proposed. With these methods a great reduction of the total number of operations to be performed is achieved. Also, the hardware implementation of these new proposals has been accomplished in a FPGA architecture, allowing the real-time processing of the acquired signals in ultrasonic systems.

3.1 Straightforward implementation

A serial straighforward correlator has been implemented in reconfigurable hardware with the aim of comparing it with the proposed efficient structures detailed below. A parallel implementation using only combinational resources has been discarded since, when the length of the code increases, it is not possible to compute in a single clock cycle all the operations involved in the correlation.

Figure 6 depicts the scheme of the solution adopted. It requires a sequential buffer where the last L received samples are stored with a frequency f_L. This buffer is read at a frequency $f_H = L \cdot f_L$. In every reading, a sample is added or subtracted depending on the corresponding correlation symbol. Whenever a new sample is received, the accumulate value is reset in order to compute a new correlation. Also, due to the fact that M codes can be simultaneously emitted, M computational blocks are implemented, all of them sharing the buffer. Thus, the frequency operation is less constrained than when these M correlations are performed in a sequential mode. These two approaches, sequential and parallel, are considered for the detection of five simultaneous Kasami codes in the LPS developed

in [**Ureña et al. 2007b**]. Furthermore, in [**Pérez et al. 2006a**] different alternatives to manage the memory required by the straightforward correlator are evaluated: one stores the data necessary in slices of the FPGA, other in external memory, and finally, the one with better performance uses the internal BRAM blocks of the FPGA. In this work the limitations of a combinational parallel design are also shown.

This correlator can be easily adapted for its use in an ultrasonic-based system, where the codes are appropriately modulated to place the spectrum of the signal to be emitted at the maximum frequency response of the transducer. In these cases the amount of samples to be stored in the bufer is $L \cdot N_{SM} \cdot O_f$, where L is the length of the code; N_{SM} represents the number of carrier periods emitted per bit of the code; and O_f is the oversampling factor used in the acquisition of the signals. Then, the access to the buffer has to be carried out in gaps of $N_{SM} \cdot O_f$ positions, by taking into account the demodulation effect.

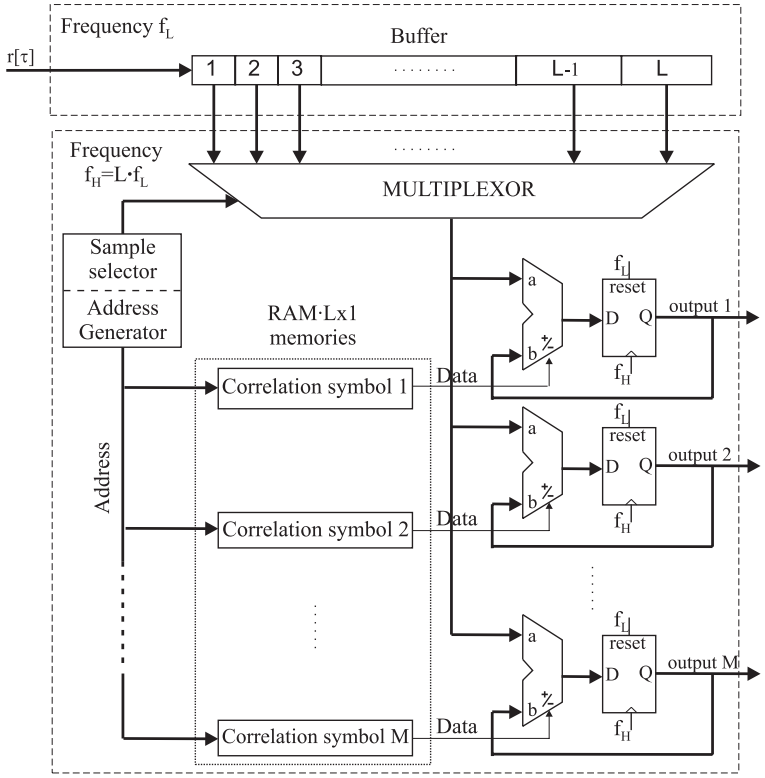


Figure 6: Parallel implementation of various serial straightforward correlators.

3.2 Efficient hardware implementation of a correlator for CSS codes

In [DMUH+07] recursive algorithms, which allow the construction of efficient architectures, are available for the generation and correlation of complementary sets of M sequences with length L, where the number M of sequences is a power of two ($M = 2^m$) and the length L

is a power of M ($L = M^N$), with $m, N \in \mathbb{Z}^+$. The algorithm proposed for the correlation, called M-ESSC, simultaneously performs the correlation of the input signal with the M sequences in the set, reducing the total number of operations required by the correlation. Thus, the M-ESSC carries out $3 \cdot 2^{m-1} \cdot m \cdot N$ operations, whereas the straightforward implementation computes $2^m \cdot (2^{mN+1} - 1)$ operations. Thanks to this reduction in the number of operations the hardware implementation of the M-ESSC can achieve real-time operation at ultrasonic frequencies.

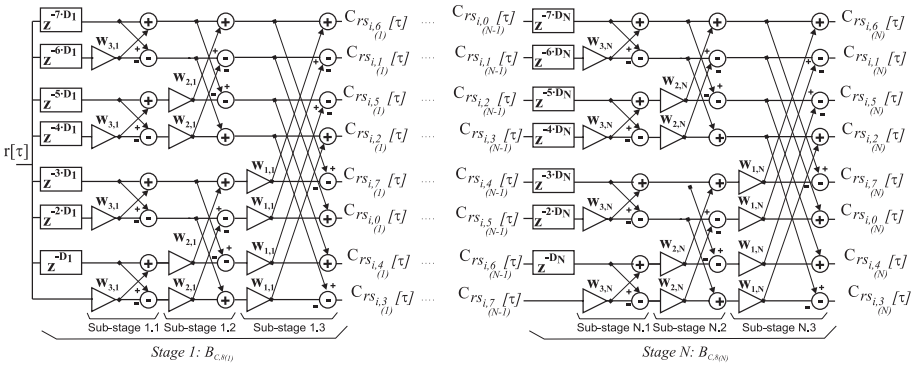
Nevertheless, the M-ESSC architecture proposed in [DMUH+07] is not suitable for a generic hardware implementation: the internal links at every stage, and the order of the partial correlation with the sequences of the set at the output of every stage, differ depending on the number of sequences in the set. Specifically, the relation between the outputs of every stage and the partial correlations obtained is computed using a complex method, difficult to implement in hardware. In this work, this architecture is modified into a regular and modular one that can be easily adapted to different values of M and L. The proposed transformation consists of rearranging the inputs at every stage. Thus, the coefficients multiplying delays D_n appear organized in a bit-reversed order. In this way, the delay $3 \cdot D_n$ ("011") will appear at the location previously occupied by the delay $6 \cdot D_n$ ("110"). As an illustration of this transformation, Figure 7.a shows the initial configuration of an 8-ESSC, and Figure 7.b the obtained after the transformation. It can be observed that, after the rearrangement of the delays, the internal connections between sub-stages are always the same, independently on the number M of sequences in the set. Furthermore, the stage outputs are always ordered in the same way, being very easy to implement in configurable hardware the connection between stages. Therefore, the architecture of the generic correlator has a regular data flow, allowing a more efficient hardware implementation. Details of the proposed transformation can be found in [**Pérez et al. 2006b**] and [**Pérez et al. 2007a**].

For brevity, only the ESSC hardware implementation is explained. Nevertheless, the same considerations can be applied to the efficient CSS generator (ESSG), since the difference among both architectures lies only in the arrangement of the delays in every stage.

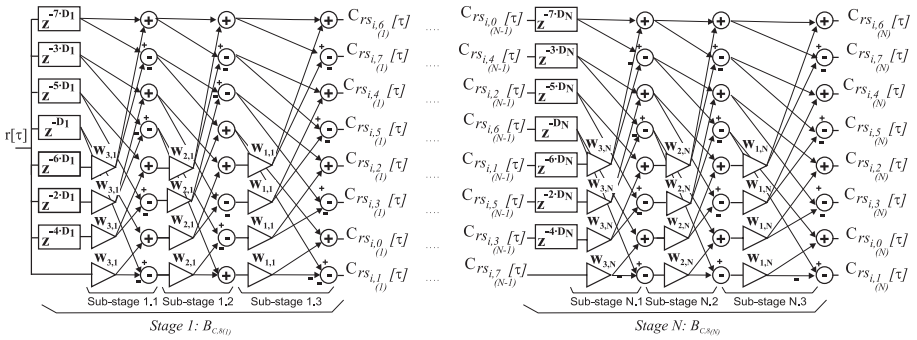
3.2.1 Configuration options for the design of a generic module

When considering the implementation of a correlator in hardware there are three design options that can be taken into account: custom design for a specific code; specific implementation with the code parameters specified at the time of hardware synthesis; and flexible implementation with the code parameters as an input to the design.

The first option has been discarded since no flexibility is allowed, i. e., the design is not able to be adapted to new requirements of the application. The second option is based on five generic parameters in the VHDL specification. It is possible to configure the number M



(a) Block diagram of an 8-ESSC.

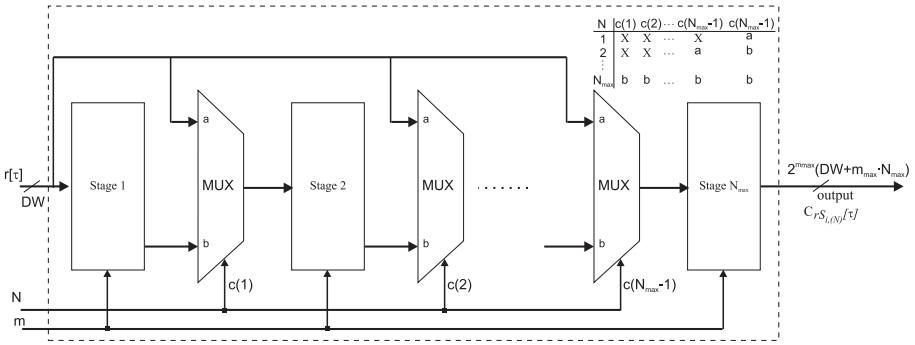


(b) Rearrangement of Figure 7.a, with the same geometry for every stage.

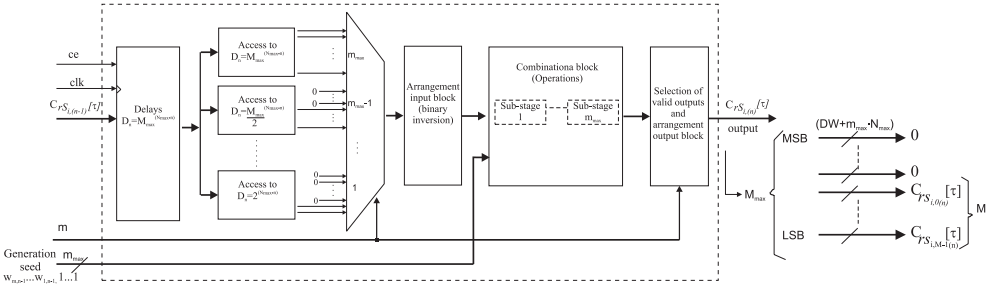
Figure 7: Transformation of the M-ESSC structure proposed in [DMUH$^+$07].

of sequences in the set. Also, it is possible to select the number N of stages in the correlator (and thus the code-length $L = M^N$). The data-width DW at the input can be configured as well, in order to have more accuracy in the obtained results. Considering the demodulation effect, the number N_{SM} of carrier periods used in the modulation, and the over-sampling rate O_f can be also selected by the user. Internally, the M-ESSC is based on N similar stages connected in cascade at synthesis time. The design of these stages has been divided into four blocks. The first one implements the required delays for each stage, by using SRL16 modules. The second block rearranges the delayed outputs according to the bit-reversed algorithm explained before. Then, a combinational block carries out the operations of the stage, where multiplications have been reduced to additions and subtractions because of the binary character of the CSS. Note that, due to the new geometry of the M-ESSC, it is enough to implement one basic operation block sub-stage and to repeat it at synthesis time, depending on the parameter M. Finally, an arrangement block orders the outputs from the operation block for their correct connection to next stage. This implementation is described in [**Pérez et al. 2006b**].

Finally, the third option is to change the number M of sequences and their length L in run-time. This configuration can be interesting for non-stable environments, as outdoors, so the sensory system can be adapted to different conditions without stopping the detection



(a) Connection of stages.



(b) Stage structure.

Figure 8: Implementation scheme of the M-ESSC configured in post-synthesis.

process. The implementation is based on the fact that the M-ESSC contains the $\frac{M}{2}$-ESSC, the $\frac{M}{2}$-ESSC contains the $\frac{M}{4}$-ESSC, and so on. Thus, the design has been configured for an upper limit of $M_{\max}$ complementary sequences in a set, with a maximum length of $L_{\max} = M_{\max}^{N_{\max}}$. If it is desired to work with less sequences $M < M_{\max}$, null values are inserted at the non-used inputs, and the proper outputs are taken, discarding the others. Also, a group of multiplexers allows the data input to be processed in the enabled stages $N \leq N_{\max}$, as can be seen in Figure 8. As a drawback, this implementation always uses the resources and memory needed for $M_{\max}$ and $L_{\max}$, with independence on the current number M of sequences in a set and their length L. On the contrary, in the pre-synthesis configuration only the necessary resources are generated, and it is also possible to increment the operating frequency if smaller sets are used.

The previous ESSC implementations optimize the execution times achieved by the straightforward implementation, at the expense of a large number of resources employed. For instance, when $M_{\max} = M = 4$, $L_{\max} = L = 64$, and the initial data-width is $DW = 8$ bits, the pre-synthesis implementation gives a new correlation result every 8.53 ns and the post-synthesis implementation every 9.86 ns, assuring real-time operation even at high frequencies. Nevertheless, with the straightforward implementation, correlation results are obtained every 487.87 ns.

3.2.2 Algorithm adaptation to the transmission scheme by macro-sequences

One advantage associated to the transmission scheme with macro-sequences is that only one correlator is necessary to detect the CSS from every emitter. The M-ESSC adapted to macro-sequences simultaneously performs the correlation of the input signal $r[\tau]$ with the M sequences of the set. It provides the AC with the first sequence of the set at the first branch of the M-ESSC, the AC with the second sequence at the second branch and so on. Nevertheless, these AC are not in phase, so it is necessary to insert delays D_c at every branch of the M-ESSC to compute the in-phase additions of the AC functions, as can be seen in Figure 9. When the macro-sequence is generated by concatenation of the sequences in the complementary set, the AC of two consecutive sequences are L samples apart, and thus, $D_c = L$. On the other hand, if the interleaving method is used, the distance among two bits from the same sequence in the transmitted macro-sequence is M bits. Hence, the correlations can not be carried out directly with the sequences of the CSS, but with interpolated versions of these sequences, where the interpolation factor is equal to M and $D_c = 1$. Finally, if the asynchronous demodulation is taken into account, the bits of the macro-sequence coming out of the demodulator have a separation of $N_{SM} \cdot O_f$ samples between them. Hence, it will be necessary to decimate the demodulated signal by the same factor prior to compute the correlations. This decimation can be easily achieved by multiplying the delays D_n from every stage and the delays D_c by $N_{SM} \cdot O_f$.

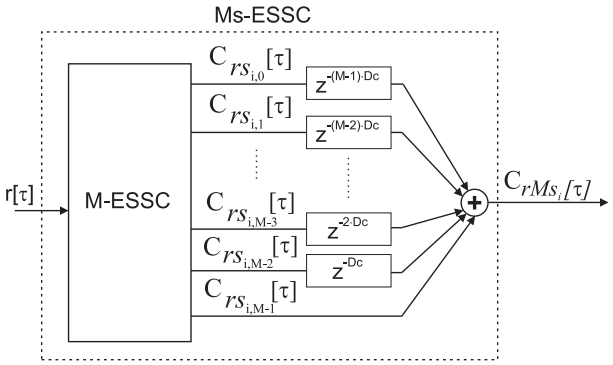


Figure 9: Correlation architecture of macro-sequences using the M-ESSC.

The resource requirements and maximum operating frequencies in the implementation of the M-ESSC adapted to macro-sequences depend on the number M of sequences in the complementary set and their length L. Obviously, large macro-sequences have high requirements. For equally-long macro-sequences, less memory resources are demanded in the cases of generating the macro-sequences with CSS with the lowest number of sequences. In regard to the operating frequency, the bigger number of sequences M in the set, the harder is to compute the combinational addition of the M output branches of the M-ESSC, so the operating frequency decreases. Note that macro-sequences generated from CSS with $M = L$ also achieve higher bounds. In Table 2 the resource requirements

of the M-ESSC for macro-sequences with length $L_{Ms} = 256$ are shown, and compared with a straightforward implementation. A XC2VP100 FPGA [Xil07] has been used. It can be observed that the M-ESSC implementation adapted to macrosequences is more suitable for real-time correlation. The description and results obtained with the M-ESSC adapted to the transmission method by interleaving can be found in [**Pérez et al. 2007a**].

xc2vp100	Ms(2,128)		Ms(4,64)		Ms(16,16)		straight-forward
	Concat.	Interl.	Concat.	Interl.	Concat.	Interl.	
Slices	410	365	732	540	2356	1096	71
LUTs	582	537	1024	832	3110	1850	117
SRL16	230	185	426	234	1560	300	0
BRAM	0	0	0	0	0	0	3
IOBs	33	33	32	32	30	30	27
Max. Freq. f_{FPGA}	125.502 MHz	115.447 MHz	88.067 MHz	86.648 MHz	71.490 MHz	66.903 MHz	139.509 MHz
Max. Through-put	125.502	115.447	88.067	86.648	71.490	66.903	0.55

Table 2: Resources required by the correlation of macro-sequences with length $L_{Ms} = 256$.

3.3 New algorithm for the efficient generation and correlation of LS codes

Since LS codes can be obtained from CSS, it is possible to use the efficient generation and correlation algorithms proposed for CSS. Thus, an effective reduction in the number of operations to be performed in the generation and correlation of LS codes can be obtained. As a result, the implementation in hardware devices as FPGAs and the use of long sequences are possible.

For brevity, only the efficient correlators of LS codes (ELSC) are shown here. The efficient generator filters (ELSG) can be easily derived from the corresponding correlators by interchanging the order of the delays as indicated in [Pop99].

3.3.1 ELSC for LS codes generated from Stańzak et al. method [SBH01]

The general scheme for the efficient correlation of LS codes, generated as shown in [ZYH05], can be observed in Figure 10. This ELSC exploits the properties of Golay pairs that form the LS sequences to simplify the detection process. It uses two EGC to perform the correlation of the input signal with each uncorrelated Golay pair (see step 1). Later, the obtained correlations are delayed (step 2) and added or subtracted (step 3) depending on the values of a binary vector Π and of the coefficients $h_{i,j}$ of a Hadamard matrix.

Table 3 shows a comparison among the number of operations and memory

bits that are necessary to store the data both in the ELSC and in the straightforward implementation. Actually, both implementations can be achieved without any multiplication since LS coefficients are $\{-1, 0, 1\}$, and thus multiplications are reduced to additions and subtractions. The number of operations is notably reduced with the proposed implementation.

A detailed description of the ELSC FPGA implementation can be found in [**Pérez et al. 2007b**]. The design has been developed as a parameterized module, able to be adapted to the requirements of different applications in the synthesis phase. Also, it has been checked that, for equal number of simultaneous users and code lengths, the implementation that demands higher requirements in terms of resources and operating frequencies, corresponds to the family with larger number K of codes.

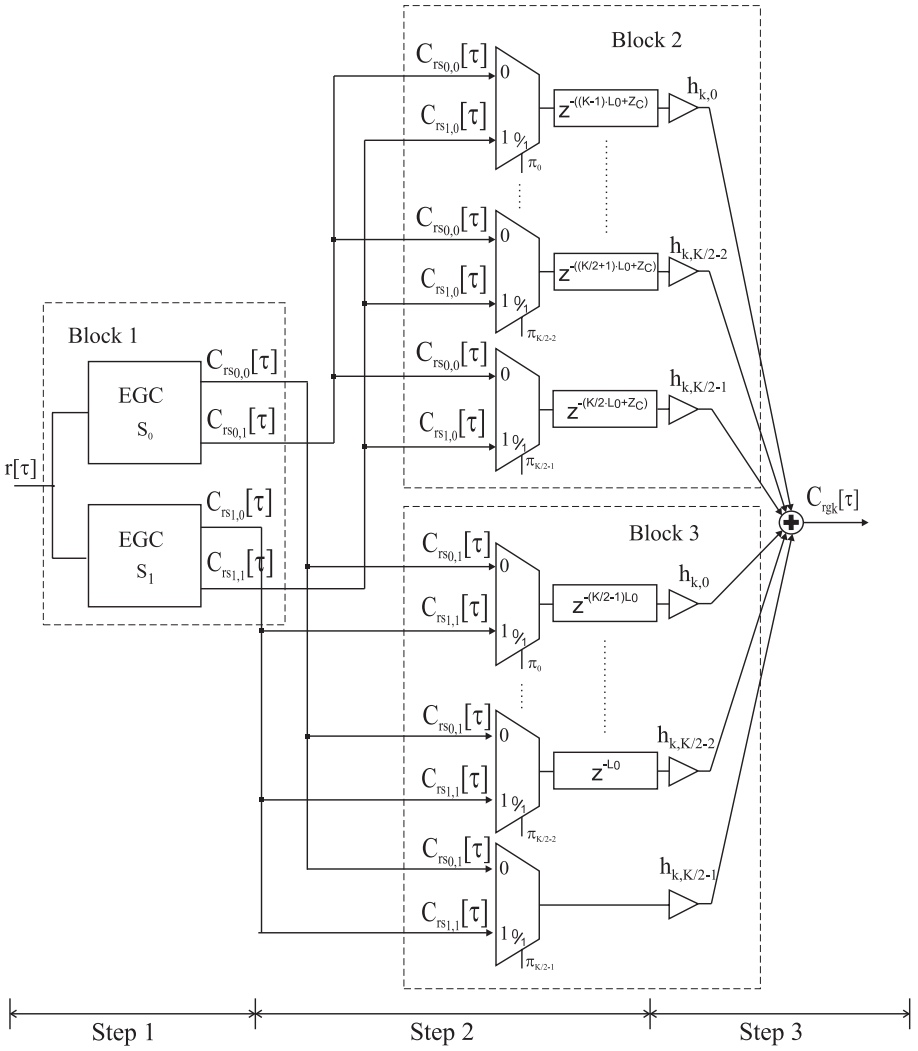


Figure 10: Block diagram of the Efficient LS code Correlator (ELSC) associated to LS codes generated from CSS as shown in [SBH01].

Implementation	Straightforward	ELSC	
Products	$K \cdot 2^N$	Step 1	$2N$
		Step 3	K
		TOTAL	$2N + K$
Additions	$K \cdot 2^N - 1$	Step 1	$4N$
		Step 3	$K - 1$
		TOTAL	$4N + K - 1$
Memory bits	$(N_{SM} \cdot O_f \cdot DW + 2)(K \cdot 2^N + Z_C)$ $+(DW + Nlog_2K)$	Step 1	$2N_{SM} \cdot O_f \left[DW(2^N - 1) + 2^N \sum_{i=1}^{N-1} \frac{i}{2^{i+1}}\right]$
		Step 2	$N_{SM} \cdot O_f \cdot (DW + N) \cdot \left[\frac{K(2^N(K-1)+Z_C)}{2}\right]$
		TOTAL	$N_{SM} \cdot O_f \left[2DW(2^N - 1) + 2^{N+1} \sum_{i=1}^{N-1} \frac{i}{2^{i+1}} + \right.$ $\left. +(DW + N) \cdot \left(\frac{K(2^N(K-1)+Z_C)}{2}\right)\right]$

Table 3: Number of operations in the straightforward and the proposed ELSC for LS codes generated as shown in [SBH01]. The LS code length is $L = K \cdot 2^N + Z_C$, where 2^N is the length of the uncorrelated Golay pairs; Z_C is the number of zeros that appear in the middle of the LS code and K the number of codes in the family.

3.3.2 ELSC for LS codes generated from Zhang et. al method [ZYH05]

Figure 11 is a block diagram of the architecture proposed for the correlation of LS codes, when they are generated with the method developed in [ZYH05]. The total number of operations and memory requirements of the proposed ELSC and the straightforward implementation can be observed in Table 4. As an example, in case of a LS family with $K = 16$ codes of length $L = 1213$ bits (and then generated from $M = 4$ uncorrelated CSS with length $L_0 = 64$ bits) the operations and memory required, when the initial data width is $DW = 8$, are:

- For ELSC: 163 operations and 116112 memory bits.
- For straightforward implementation: 2047 operations and 9663 memory bits.

Since there are enough resources available with current FPGAs, the decrease in the number of operations involved by the ELSC is more important than the increase in memory requirements in terms of hardware implementation.

The implementation of the ELSC has also been performed in a FPGA. The algorithm is structured in a spatial implementation, so all the required resources are always available to exploit parallelism. Therefore, the processing is divided into tasks that can be overlapped temporarily to rise the performance, at the expense of increasing the latency. Also, the correlator parameters (code length L, number K of available codes, length Z_C of the set of zeros that appears in the code, data-width DW, number N_{SM} of carrier periods

and oversampling factor O_f) can be specified at synthesis time. By using a Xilinx Virtex XC2VP100 [Xil07] in the ELSC implementation, the processing time of a sample can be up to 11.88 ns for $K = 16$, $L = 301$ and $DW = 8$. On the other hand, when the straightforward implementation is used, the time required is 2.23 μs.

The description of this ELSC correlator associated to CSS can be seen in [**Pérez et al. 2008a**].

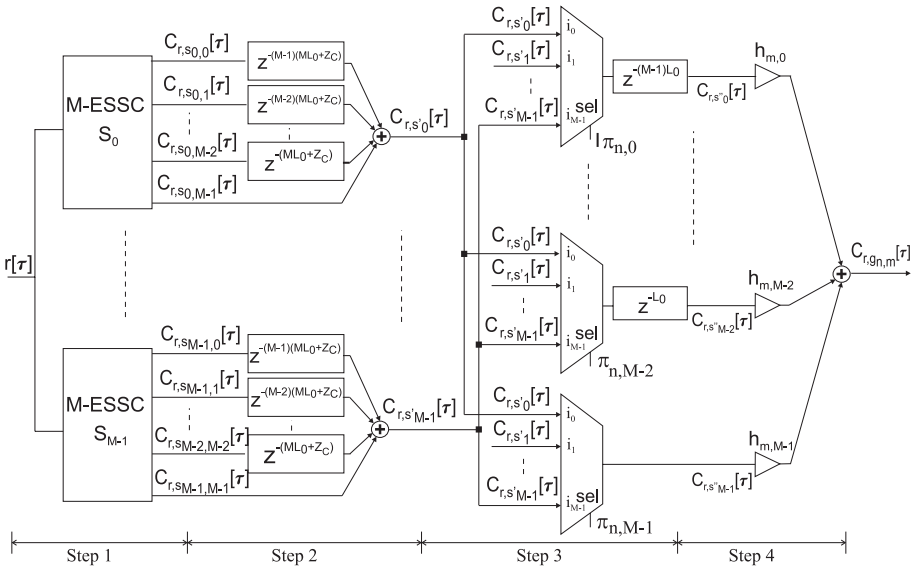


Figure 11: Block diagram of the Efficient LS code Correlator (ELSC) associated to LS codes generated from CSS as shown in [ZYH05].

Implementation	Straightforward	ELSC	
Products	$M^{(N+2)}$	Step 1	$\frac{N \cdot M^2}{2} log_2(M)$
		Step 4	M
		TOTAL	$M(1 + \frac{N \cdot M}{2} log_2(M))$
Additions	$M^{(N+2)} - 1$	Step 1	$N \cdot M^2 \cdot log_2(M)$
		Step 2	$M(M-1)$
		Paso 4	$M - 1$
		TOTAL	$N \cdot M^2 \cdot log_2(M) + M^2 - 1$
Memory bits	$(N_{SM} \cdot O_f \cdot DW + 2)(M^{N+2} + (M-1)Z_C) +$ $+(DW + N + 2)$	Step 1	$N_{SM} \cdot O_f \left[(M^{N+1} - M^N)DW + \frac{M^3 - M^2}{2} \left(\frac{M^{N-1}-1}{M-1} DW + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}} \right) \right]$
		Step 2	$N_{SM} \cdot O_f \cdot (DW + mN) \cdot \left(\frac{(M^{N+1}+Z_C)(M^3-M^2)}{2} \right)$
		Step 3	$N_{SM} \cdot O_f \cdot (DW + mN + m) \cdot \left(\frac{M^{N+2}-M^{N+1}}{2} \right)$
		TOTAL	$N_{SM} \cdot O_f \left[(M^{N+1} - M^N)DW + \frac{M^3 - M^2}{2} \left(\frac{M^{N-1}-1}{M-1} DW + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}} \right) \right] +$ $+N_{SM} \cdot O_f \left[(DW + mN) \cdot \left(\frac{M^{N+4} - M^{N+3} + M^{N+2} - M^{N+1} + Z_C M^3 - W M^2}{2} \right) + m \left(\frac{M^{N+2} - M^{N+1}}{2} \right) \right]$

Table 4: Number of operations in the straightforward and the proposed ELSC for LS codes generated as shown in [ZYH05].

4 New generation and correlation algorithms for T-ZCZ codes

In [ZYH05], T-ZCZ pairs are obtained from a generator matrix Δ. In practice, the iterative algorithm proposed to construct this Δ matrix coincides with the iterative algorithm proposed by Tseng and Liu (see [TL72]) for generating a matrix Δ containing M mutually Uncorrelated CSS (UCSS). In this thesis, new generator matrixes are obtained by using CSS generated from the recursive method proposed in [DMUH$^+$07]. T-ZCZ codes generated from these other generator matrixes have the following advantages:

- These T-ZCZ codes exhibit narrower IZ and lower bounds in the aforementioned IZ, if they are compared with those of T-ZCZ codes generated as shown in [ZYH05].
- By using the ESSG and ESSC, efficient structures for the construction and detection of T-ZCZ codes are possible. The proposed generation algorithms allow the simultaneous generation of all the sequence pairs in a family, starting from the elementary delta sequence $\delta[\tau]$. Furthermore, the developed correlation filters simultaneously perform the correlation of the input signal with the M sequence-pairs in the family, employing less operations than the required in the straightforward correlation.

  Thanks to this important reduction in the number of operations, the implementation of the Efficient T-ZCZ code Generator (ETZG) and of the Efficient T-ZCZ code Correlator (ETZC) in a configurable hardware architecture can achieve real-time operation even at high frequencies.

A complete description of the developed algorithms for the efficient construction and correlation of T-ZCZ codes can be found in **[Pérez et al. 2008b]**. Furthermore, In Section A.5 of Appendix A, several tables are shown with information regarding to the size of the lateral Z_L and central Z_C zero correlation zones. In these tables a selection of T-ZCZ codes with lower bounds and number of interferences in the IZ also appear.

A concise overview of the correlators that have a better performance in terms of hardware implementation (ETZC_1, $\text{ETZC}_{2'}$ and $\text{ETZC}_{3'}$) is given below.

4.1 Efficient correlator (ETZC$_1$) for T-ZCZ$_1$ codes

A set of M binary T-ZCZ pairs, $\{E_k = (e_{k,0}, e_{k,1}); 0 \leq k \leq M-1\}$ with length $L = M^N$ and elements $\{-1, +1\}$, called T-ZCZ$_1$ set, can be constructed from a single CSS $\{S_i = s_{i,j}[l]; 0 \leq j \leq M-1; 0 \leq l \leq M^N-1\}$ as shown in (1); where $\overline{s_{i,j}}[\tau] = s_{i,j}[M^N-1-\tau]$ imply reversing the order of the elements in $s_{i,j}$; $\lfloor x \rfloor$ represents the largest integer less than or equal to x; $M = 2^m$, with $m, N \in \mathbb{Z}^+$; and the index i is fixed to any value $0 \leq i \leq M^N - 1$.

$$\begin{cases} e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2} = s_{i,j}, & 0 \leq j \leq M-1 \\ e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2} = (-1)^{j+1} \cdot \overline{s_{i,j}}, & 0 \leq j \leq M-1 \end{cases} \tag{1}$$

The proposed algorithm for the efficient correlation of the T-ZCZ$_1$ set is shown in Figure 12. Note that the simultaneous correlation of the input signal $r[\tau]$ with every one of the M sequences $s_{i,j}$ is carried out by the ESSC; obtaining $C_{r,s_{i,j}}[\tau]$, and thus, $C_{r,e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2}} = C_{r,s_{i,j}}$, where $0 \leq j \leq M-1$. By changing the order of the coefficients $\{M-1, M-2, \cdots, 0, 1\}$ that multiplies the delays D_n at every stage of the ESSC to $\{0, 1, \cdots, M-2, M-1\}$, a correlator $\overline{\text{ESSC}}$ matched to sequences $\overline{s_{i,j}}[l]$ is achieved. Even outputs of the $\overline{\text{ESSC}}$ are negated so as to obtain the correlation with the last $\frac{M}{2}$ pairs $C_{r,e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2}} = (-1)^{j+1} \cdot C_{r,\overline{s_{i,j}}}$, where $0 \leq j \leq M-1$.

A performance comparison between the ETZC$_1$ and the straightforward implementation can be found in Table 5. The product operations performed by these correlators consist of a change in the sign (multiplications by -1 or $+1$). On the other hand, note that the memory requirements of both implementations are very similar, however a great reduction in the number of operations carried out is achieved by the ETZC$_1$.

Implementation	Products	Additions	Memory bits
Straightforward	$2M^{N+1}$	$2M \cdot (M^N - 1)$	$N_{SM} \cdot O_f \cdot DW \cdot M^N + 2M^{N+1} + 2M \cdot (DW + mN)$
M-ETZC$_1$	$M \cdot m \cdot N + \frac{M}{2}$	$2M \cdot m \cdot N$	$2N_{SM} \cdot O_f \cdot DW \cdot (M-1)M^{N-1} + N_{SM} \cdot O_f \cdot (M^2 - M) \left[DW \frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}} \right]$

Table 5: Number of operations and memory bits in the straightforward correlator and the proposed ETZC$_1$, depending on the number M of pairs in a T-ZCZ$_1$ family and its length $L = M^N$.

4.2 Efficient correlator (ETZC$_{2'}$) for T-ZCZ$_{2'}$ codes

According to the generation method of T-ZCZ$_{2'}$ codes, with length $L = \frac{M^{N+1}}{2}$, one pair is composed by the sequences $s_{i,j}$ of M uncorrelated CSS $\{S_i; 0 \leq i \leq M-1\}$ as follows: $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{2}-1}], [s_{\frac{M}{2},j}|s_{\frac{M}{2}+1,j}|\cdots|s_{M-1,j}])$, where $0 \leq j \leq M-1$ and $s_{0,j}|s_{1,j}$ denotes the concatenation of sequences $s_{0,j}$ and $s_{1,j}$. Thus, the AC of sequences

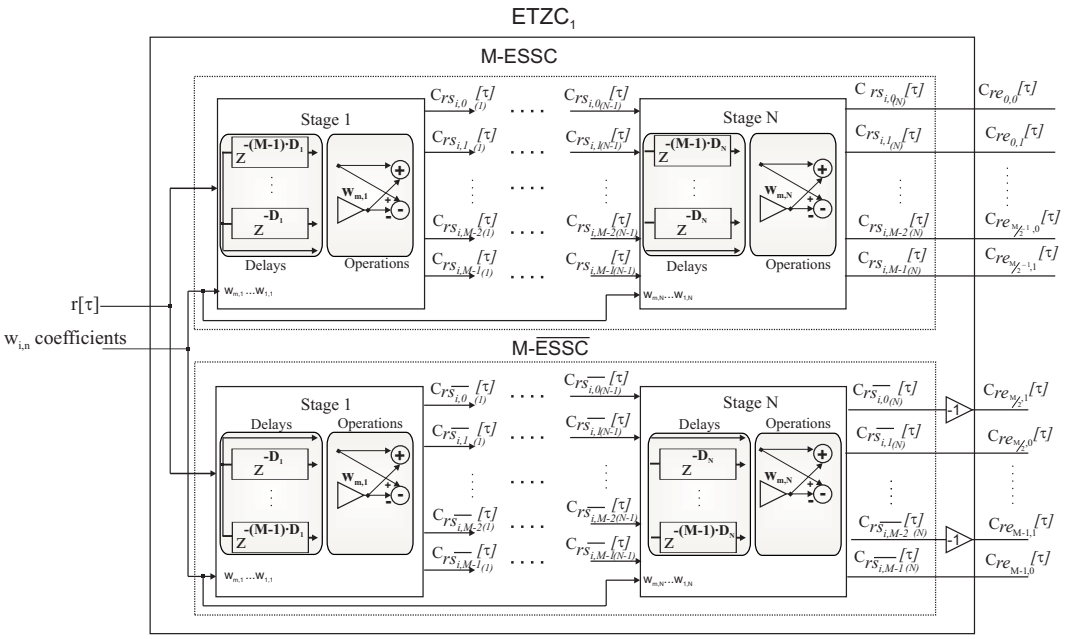


Figure 12: Block diagram for the Efficient T-ZCZ$_1$ Correlator (ETZC$_1$).

$e_{j,0}$ and $e_{j,1}$ can be expressed in terms of correlations with $s_{i,j}$ sequences -see (2)-, where the correlations $C_{r,s_{i,j}}$ can be achieved by means of the ESSC as shown in Figure 13.

$$\begin{aligned} C_{r,e_{j,0}} &= C_{r,s_{0,j}}[\tau] + C_{r,s_{1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{2}-1,j}}[\tau - (\tfrac{M}{2}-1)L_0] \\ C_{r,e_{j,1}} &= C_{r,s_{\frac{M}{2},j}}[\tau] + C_{r,s_{\frac{M}{2}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{M-1,j}}[\tau - (\tfrac{M}{2}-1)L_0] \end{aligned} \tag{2}$$

This filter, called ETZC$_{2'}$, performs the simultaneous correlation of the input signal with the sequences of the M pairs of a T-ZCZ$_{2'}$ family. By using it, a high reduction in the number of operations is achieved, when compared to the straightforward implementation, at the expense of increasing the memory requirements. Table 6 shows the computational requirements for both implementations.

Implementation	Products	Additions	Memory bits
Straightforward	M^{N+2}	$2M(\frac{M^{N+1}}{2}-1)$	$N_{SM}\cdot O_f \cdot DW \cdot \frac{M^{N+1}}{2} + M^{N+2} + 2M\cdot(DW+mN+m-1)$
M-ETZC$_{2'}$	$\frac{M^2}{2}mN$	$M^2mN + 2(\frac{M^2}{2}-M)$	$N_{SM}\cdot O_f \cdot DW \cdot (M-1)M^N + N_{SM}\cdot O_f \cdot \frac{(M^3-M^2)}{2}\left[DW\frac{M^{N-1}-1}{M-1} + mM^N\sum_{i=1}^{N-1}\frac{i}{M^{i+1}}\right] + N_{SM}\cdot O_f\cdot(DW+mN)\cdot\frac{M}{2}(\frac{M}{2}-1)\cdot M^N$

Table 6: Number of operations and memory bits in the straightforward correlator and the proposed ETZC$_{2'}$, depending on the number M of pairs in a T-ZCZ$_{2'}$ family and its length $L = \frac{M^{N+1}}{2}$.

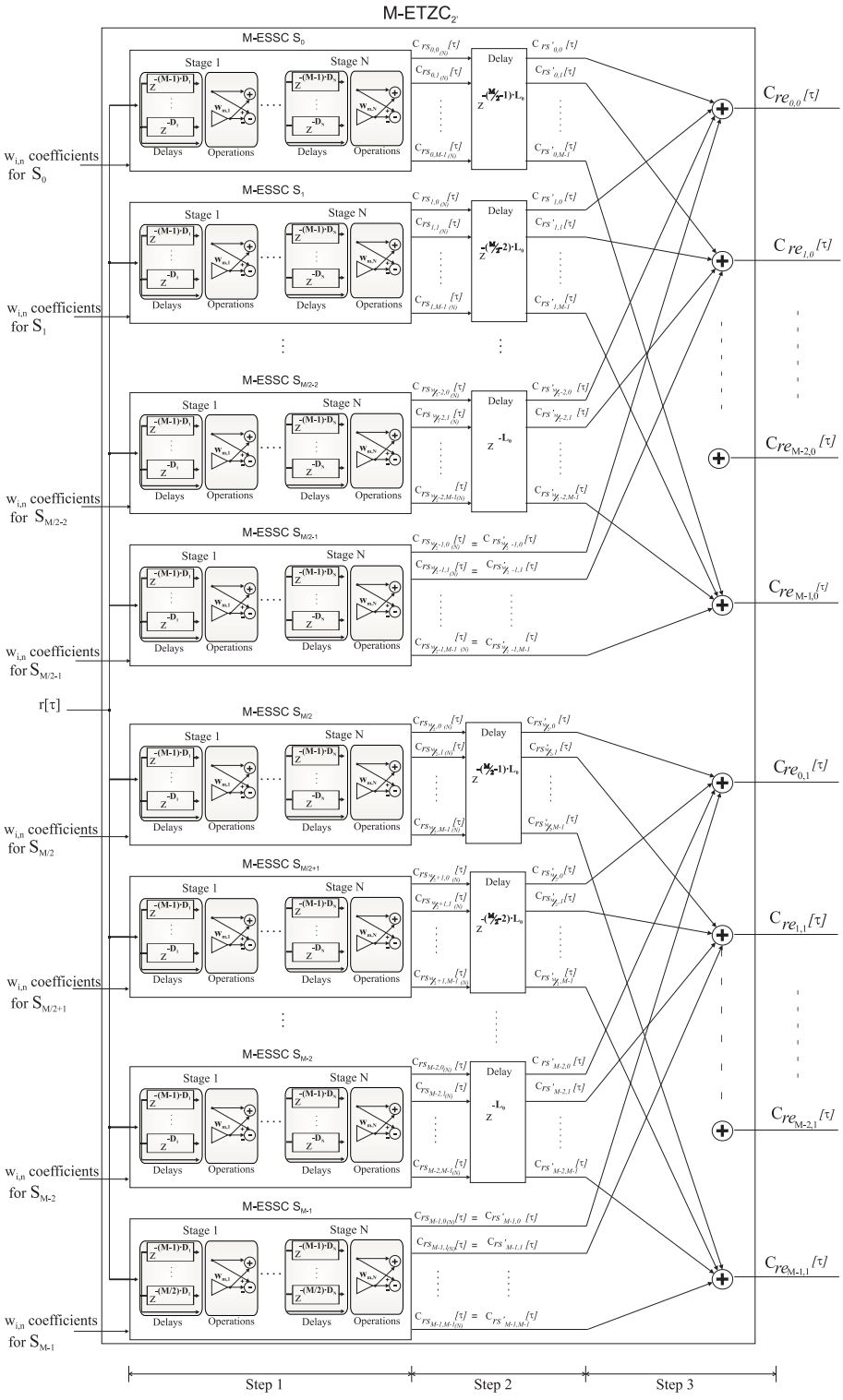


Figure 13: Block diagram for the Efficient T-ZCZ$_{2'}$ Correlator (ETZC$_{2'}$).

4.3 Efficient correlator ($\text{ETZC}_{3'}$) for T-$\text{ZCZ}_{3'}$ codes

T-$\text{ZCZ}_{3'}$ pairs with length $L = \frac{M^{N+1}}{4}$ can be obtained from M uncorrelated CSS as follows: $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{4}-1,j}], [s_{\frac{M}{4},j}|s_{\frac{M}{4}+1,j}|\cdots|s_{\frac{M}{2}-1,j}])$, with $M > 2$. Then, the correlation of the input signal $r[\tau]$ with every sequence of a pair T-$\text{ZCZ}_{3'}$ can be separated into the correlation of $r[\tau]$ with the corresponding sequences $s_{i,j}$, as shown in (3). The proposed correlator filter is represented in Fig. 14. The total number of operations and memory requirements of the $\text{ETZC}_{3'}$ and a straightforward implementation can be observed in Table 7. A considerable reduction in the number of operations is obtained for the $\text{ETZC}_{3'}$.

$$\begin{aligned} C_{r,e_{j,0}} &= C_{r,s_{0,j}}[\tau] + C_{r,s_{1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{4}-1,j}}[\tau - (\tfrac{M}{4} - 1)L_0] \\ C_{r,e_{j,1}} &= C_{r,s_{\frac{M}{4},j}}[\tau] + C_{r,s_{\frac{M}{4}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{2}-1,j}}[\tau - (\tfrac{M}{4} - 1)L_0] \end{aligned} \tag{3}$$

Implementation	Products	Additions	Memory bits
Straightforward	$\frac{M^{N+2}}{2}$	$2M(\frac{M^{N+1}}{4} - 1)$	$N_{SM} \cdot O_f \cdot DW \cdot \frac{M^{N+1}}{4} + \frac{M^{N+2}}{2} + 2M \cdot (DW + mN + m - 2)$
M-$\text{ETZC}_{3'}$	$\frac{M^2}{4}mN$	$\frac{M^2}{2}mN + 2(\frac{M^2}{4} - M)$	$N_{SM} \cdot O_f \cdot DW \cdot (M-1)\frac{M^N}{2} + N_{SM} \cdot O_f \cdot \frac{(M^3-M^2)}{4}\left[DW\frac{M^{N-1}-1}{M-1} + mM^N\sum_{i=1}^{N-1}\frac{i}{M^{i+1}}\right] + N_{SM} \cdot O_f \cdot (DW + mN) \cdot (\frac{M^2}{16} - \frac{M}{4})M^N$

Table 7: Number of operations and memory bits in the straightforward correlator and the proposed $\text{ETZC}_{3'}$, depending on the number M of pairs in a T-$\text{ZCZ}_{3'}$ family and its length $L = \frac{M^{N+1}}{4}$.

4.4 Discussion

The election of T-ZCZ_1, T-$\text{ZCZ}_{2'}$ or T-$\text{ZCZ}_{3'}$ codes mainly depends on the desired code length and the size of the ZCZ. T-ZCZ_1 codes have reduced ZCZ when N is odd, whereas T-$\text{ZCZ}_{3'}$ are the ones with larger ZCZ. On the other hand, considering the hardware implementation of the corresponding correlators, the one that requires less operations and less memory resources is the ETZC_1. $\text{ETZC}_{2'}$ and $\text{ETZC}_{3'}$ reduce the number of operations performed in comparison with a straightforward correlator, however they require more memory.

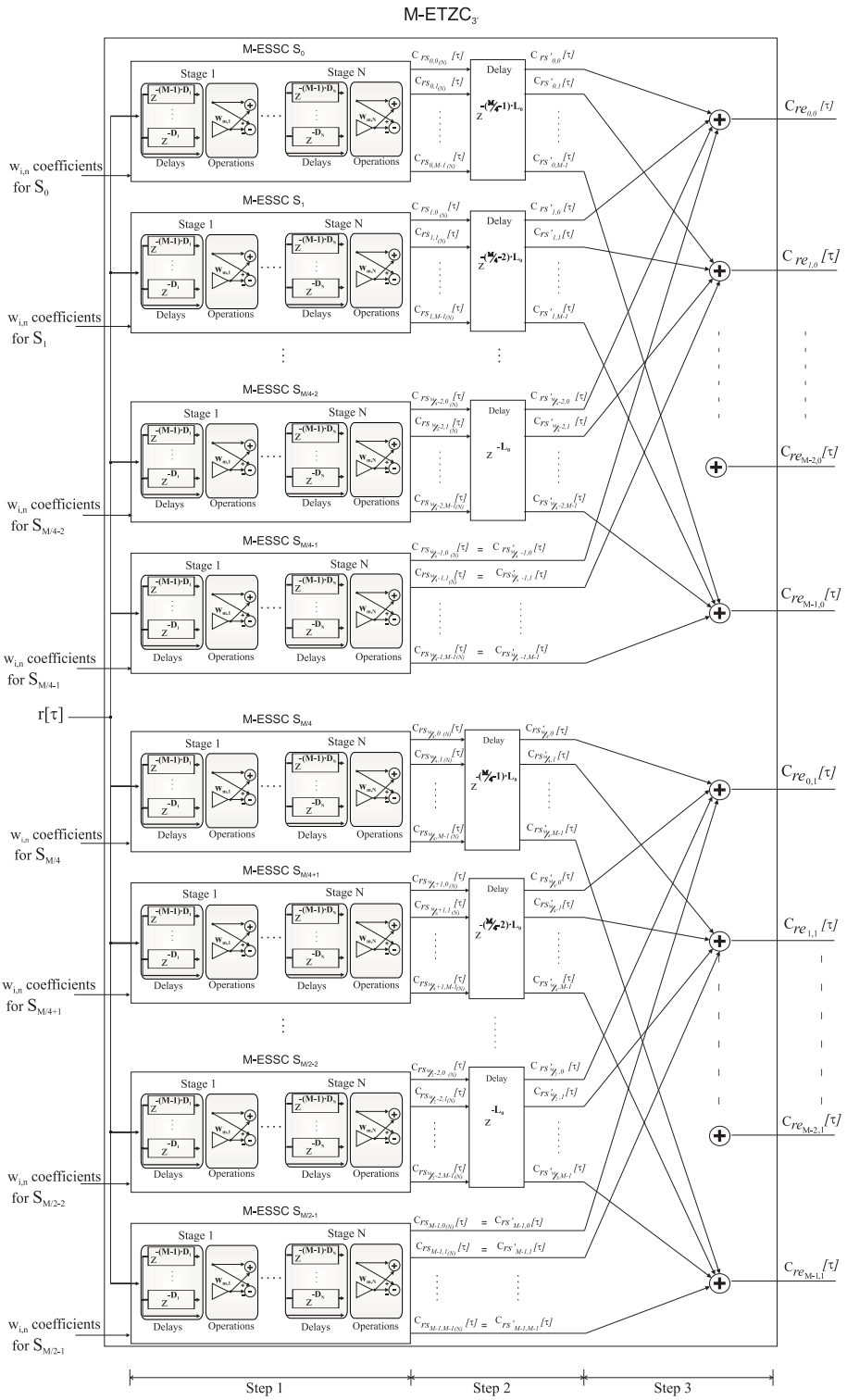


Figure 14: Block diagram for the Efficient T-ZCZ$_{3'}$ Correlator (ETZC$_{3'}$).

5 Experimental results

The purpose of this section is to evaluate the performance of Kasami, CSS, LS and T-ZCZ pairs as encoding codes for an ultrasonic LPS.

5.1 Architecture and functionality of the experimental set-up

Figure 15 shows a schematic representation of the global structure of the LPS. In this system, a set of five beacons is placed at known positions of the ceiling, with a height of 3.45 m and into a 1 $m \times 1$ m surface, what constitutes all the infrastructure to be installed. The beacons are hardware synchronized (there is a wire joining them), all of them cover an area on the floor of 4.5 $m \times 3.5$ m and are emitting periodically. To minimize the problem of interferences caused by the simultaneous emissions, a preferred set of codes with low CC among them encodes and identifies the ultrasonic signals emitted by the beacons (see Appendix A). Then, a non-limited number of portable receivers detects all these signals asynchronously. Every receiver independently carries out the correlation with the codes assigned to every beacon, obtaining the Differences of TOF (DTOF) between a reference beacon (the nearest one) and the others. Using these DTOFs, they can compute their absolute position by using a hyperbolic trilateration algorithm. With this method, the receivers do not need to know the instant in which the emission starts, and then the synchronism is only necessary among the beacons. This avoids the use of radio-frequency or infrared synchronization signals together with the ultrasonic ones.

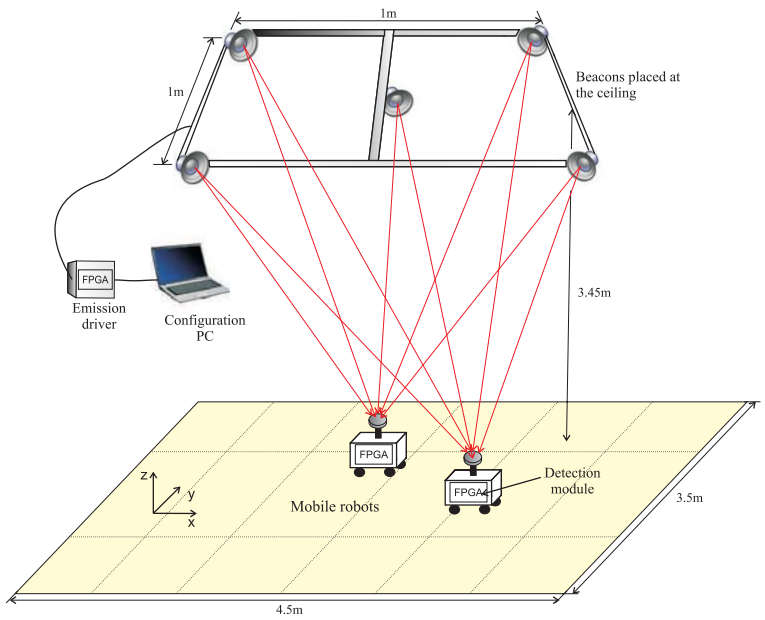


Figure 15: Schematic representation of the LPS.

The emission and reception modules have been implemented each one in a Digilent Nexys-2 [Inc08a] computing platform, which, among other resources, includes a SPARTAN3E-500 FPGA [Xil08], A/D and D/A converters and a USB port for communications with external devices.

At the emission stage, the design is flexible enough to allow the real-time configuration of the ultrasonic emission by using a PC connected to the FPGA via USB, as can be observed in Figure 16. Concerning the encoding, the type of binary sequence used (Kasami, CSS, LS or T-ZCZ) and their length can be configured. On the other hand, the type of carrier used in the modulation (sinusoidal or square), the number of carrier periods, the carrier frequency and the time interval between emissions can be also selected. The codes are generated off-line in the PC using the corresponding algorithms (efficient structures ESSG, ELSG and ETZG are used in case of CSS, LS and T-ZCZ codes respectively). Then, they are stored in the BRAM memory of the FPGA from which they are read at the proper frequency, BPSK modulated and emitted. The used emitters are omni-directional piezoelectric polymer transducers by MSI [Inc08b], which have a resonant frequency of 40 kHz with a 8 kHz bandwidth. Furthermore, they have been inserted into a conical reflector as shown in [VUM+07] to enlarge the common covered area.

Regarding the reception process, the receiver on board the mobile robot is a WM-61 Panasonic omnidirectional electret microphone [Mic08], which has a flat frequency response between 20 Hz and 45 kHz. On the other hand, the low level design operation parameters are also configurable, but during synthesis-time. The tasks involved are: digital demodulation, correlation and peak detection. A digital squared carrier symbol has been used in the modulation, thus the BPSK demodulation can be simply designed as a shift register whose samples are added or subtracted. After, the demodulated signal is correlated with the corresponding code that identify every beacon. These correlations can be performed in a straightforward manner, or by using the efficient schemes proposed in this thesis (ESSC, ELSC or ETZC). When the last sample of a received code is processed, a peak is obtained only in the correlator matched to this code. A peak detector system at the output of each correlator looks for the local maxima that exceed a certain threshold. If the peaks detected at each beacon are close enough to those detected in the other beacons, they are validated as correlation peaks. The size of the analysis window that determines the maximum separation among echoes is pre-synthesis configurable (in case of LS and T-ZCZ codes, it would be of $N_{SM} \cdot O_f \cdot Z_C$ samples). The high level algorithms are carried out in the PC embedded in the robot, which is a classical PIOONER 3-DX8 [Inc08c]. The absolute position of the mobile robot is obtained by using the DTOF between the first beacon and the others and the positions where the beacons are located, as shown in **[Ureña et al. 2007b]**.

Figure 17 is a picture of the LPS used in the experimental tests.

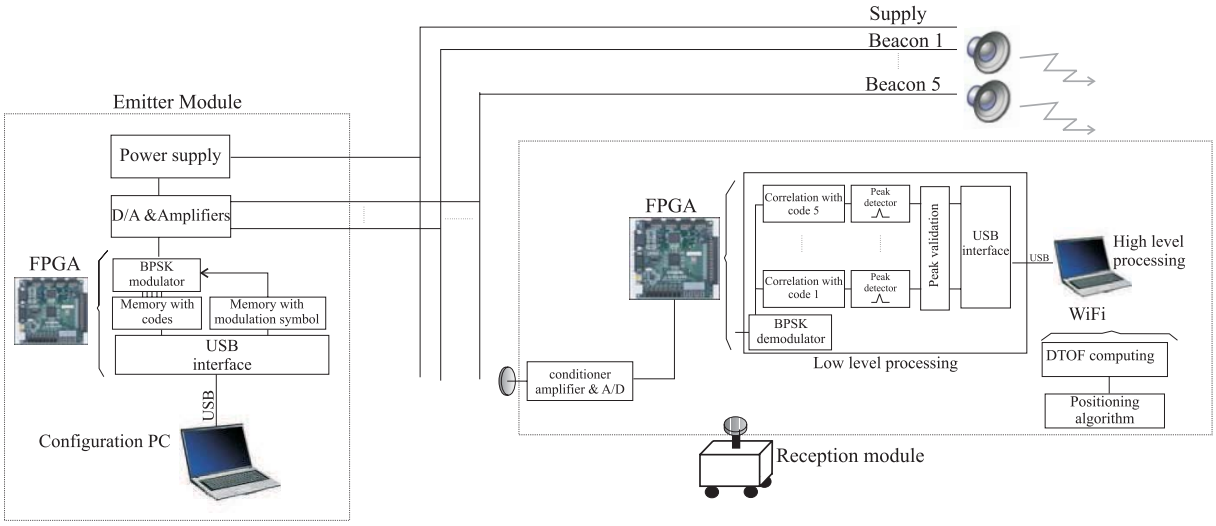


Figure 16: Block diagram of the proposed LPS.

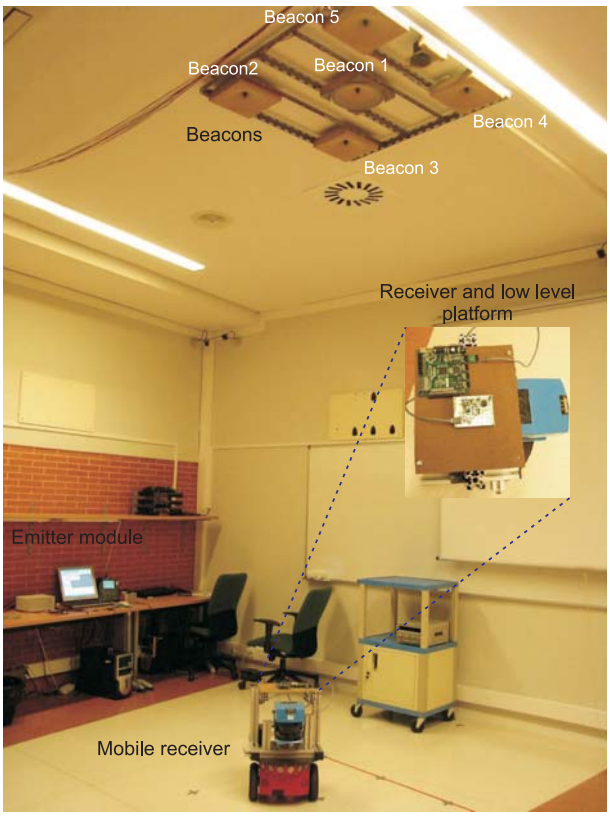


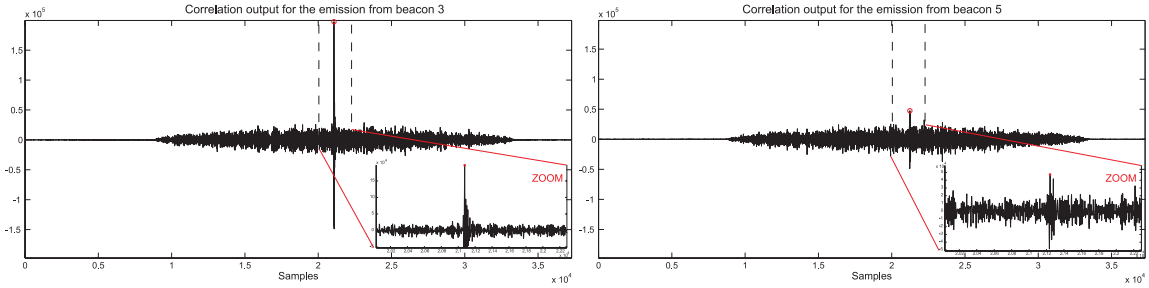
Figure 17: Experimental set-up.

5.2 Results

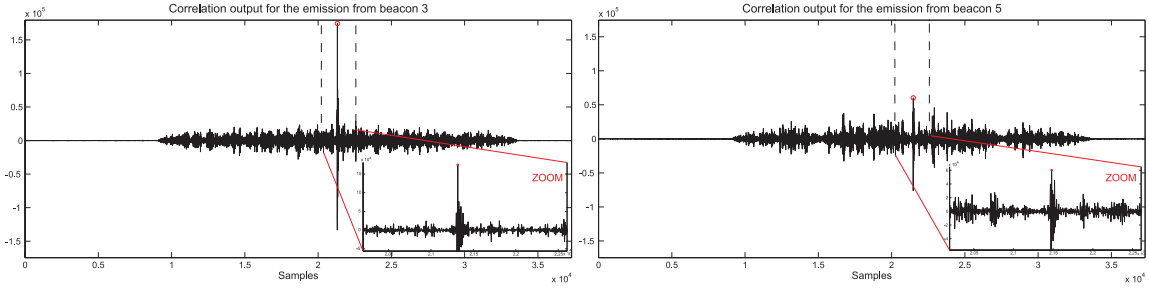
Considering the frequency response of the transducers, the emission frequency was selected to be 41.667 kHz. The modulation has been configured with a single period $N_{SM} = 1$ of a squared signal, not to increase the emission interval of every code. The acquisition module has a sampling frequency of 500 kHz, which implies twelve samples per carrier period ($O_f = 12$), and a resolution of $DW = 8$ bits.

With these parameters, and taking into account the beacon's location, the maximum distance among them and the size of the area to be covered, the maximum relative time-offset between the codes will be 2.051 ms (assuming that the propagation speed of ultrasounds is $c = 343.5$ m/s at 20°). Thus, the ZCZ around the origin must be higher than 173 bits ($Z_C \geq 86$). A set with at least $\mu = 5$ LS codes that fulfil this condition is generated as shown in [SBH01]; it has a length of $L = 1151$ bits, with $Z_C = 127$ bits and a mainlobe magnitude equal to 1024. With these conditions, T-ZCZ$_{3'}$ pairs of length $L = 1024$ and $Z_C = Z_L = 146$ bits can be chosen, achieving a process gain equal to 2048. Nevertheless, to transmit the two sequences in each pair, maintaining the BPSK modulation, they have to be rearranged in a new sequence of 2048 bits. The increase of the code length has two main disadvantages: there are more data to be processed and the robot could have changed its position without having received the whole code. If the speed of the robot is slow enough, this second problem can be neglected. However, to perform the comparison under similar conditions (with similar code-lengths and process gain), a set of $M = 4$ codes T-ZCZ$_{2'}$ with length $L = 512$ bits, $Z_C = Z_L = 153$ has been selected. The two sequences of each pair have been concatenated with a separation of 153 bits among them, thus obtaining a code length equal to 1171 bits. In this case the positioning is achieved with four beacons. A preferred set of five Kasami codes with length $L = 1023$ bits, and two preferred sets of macro-sequences generated from the concatenation and interleaving of Golay pairs (2-CSS) of length $L = 512$ bits ($L_{Ms} = 1024$), are also chosen.

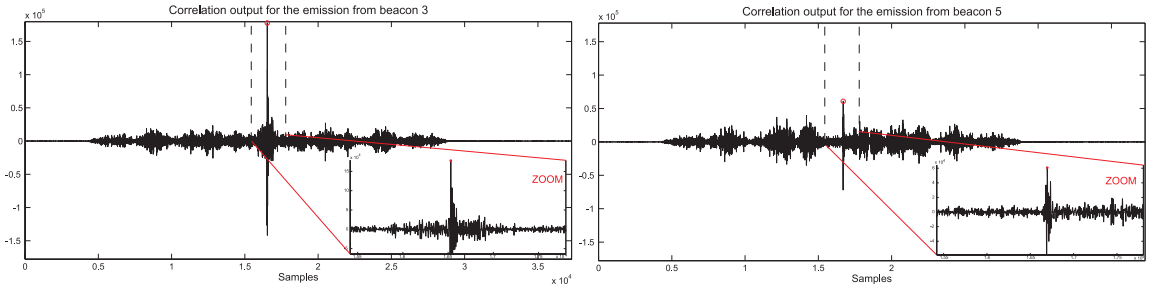
To check the feasibility of the aforementioned codes, test point positions every 50 cm have been defined in a grid over the floor, and 105 measurements were made for every position and family code. One of these test positions is in the floor just under one of the beacons (beacon 3 in Figure 17, with approximated coordinates of $x = 249$ cm and $y = 103$ cm). In this situation, the code emitted by the furthest beacon (which is beacon 5) can be affected by the near-far effect, not being detected after the correlation process. For each encoding scheme, Figure 18 depicts the results obtained at the outputs of the corresponding correlators for beacon 3 and beacon 5. A dot-line has been included to represent the analysis window where AC peaks are expected to be found, and a red circle indicates the peak validated. As can be observed, Kasami codes are more likely to fail in the detection of weak echoes, because of the CC sidelobes introduced by beacon 3. On the contrary, LS codes and T-ZCZ pairs significantly alleviates the MAI and multipath interferences within the analysis window (that coincides with the ZCZ around the origin).



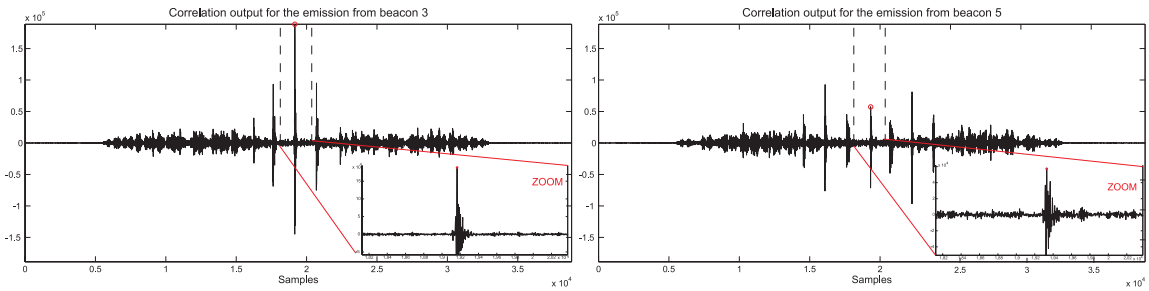
(a) Kasami codes.



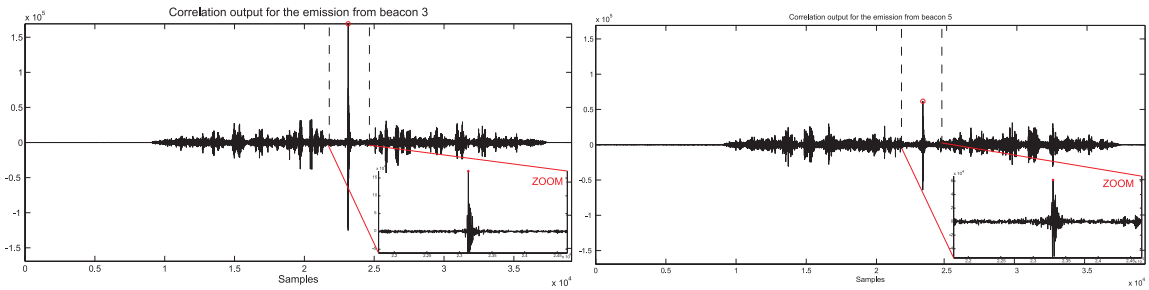
(b) Macro-sequences obtained by concatenation of CSS.



(c) Macro-sequences obtained by interleaving of CSS.



(d) LS codes.



(d) T-ZCZ$_{2'}$ codes.

Figure 18: Correlation outputs from beacon 1 and 5 using different encoding schemes.

To determine the absolute position, whenever a potential near-far interference is found, the weaker signal is simply not considered in the positioning algorithm (3 DTOF are enough in case of 2D positioning). In Figures 19 to 23 the coordinate estimation obtained with every code is shown for different test points (P1 to P13). For every test point, the ellipse represents the 95 % confidence level absolute position obtained; the blue circles depict the position estimated with each one of the 105 measurements carried out (note that most of them coincide); and, finally, the red cross inside the ellipses is the mean of these positioning values. Note that the results obtained with T-ZCZ$_{2'}$ codes have an offset with respect to the others, it is due to the use of only four beacons, instead of five. On the other hand, it can be observed that, in general, codes derived from CSS (macro-sequences, LS and T-ZCZ) give better precision in the results than what would be obtained by using Kasami codes, even in cases heavily affected by the multipath (such as the test point P1).

A LPS using Kasami codes has been presented in [**Ureña et al. 2007b**]. Furthermore, results of a LPS using LS codes are shown in [**Pérez et al. 2007c**]. Finally, the performance of Kasami, macro-sequences of interleaved CSS, and LS codes in a LPS is compared in [**Ureña et al. 2007a**].

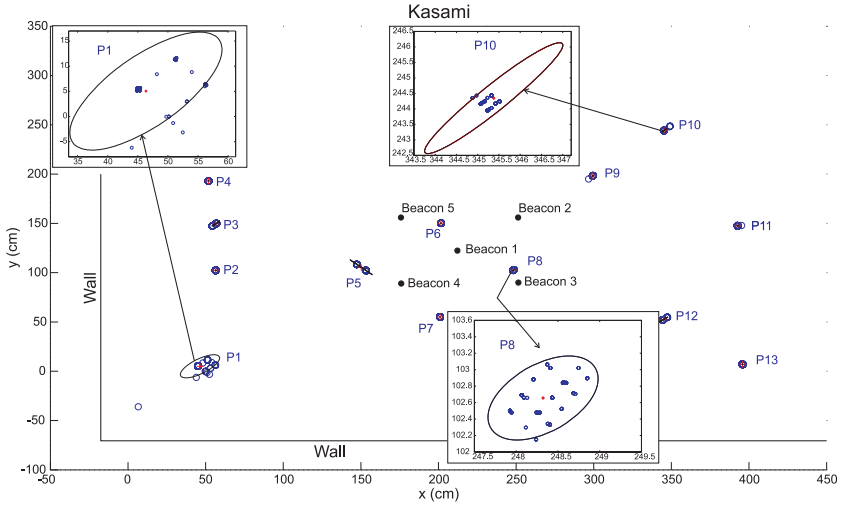


Figure 19: Position obtained with a 95 % confidence level when the emissions are encoded with Kasami codes.

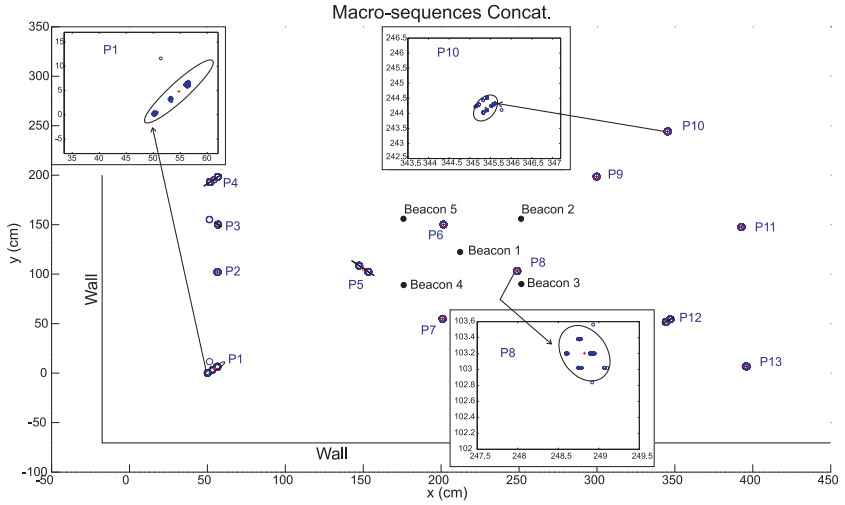


Figure 20: Position obtained with a 95 % confidence level when the emissions are encoded with macro-sequences generated from the concatenation of CSS.

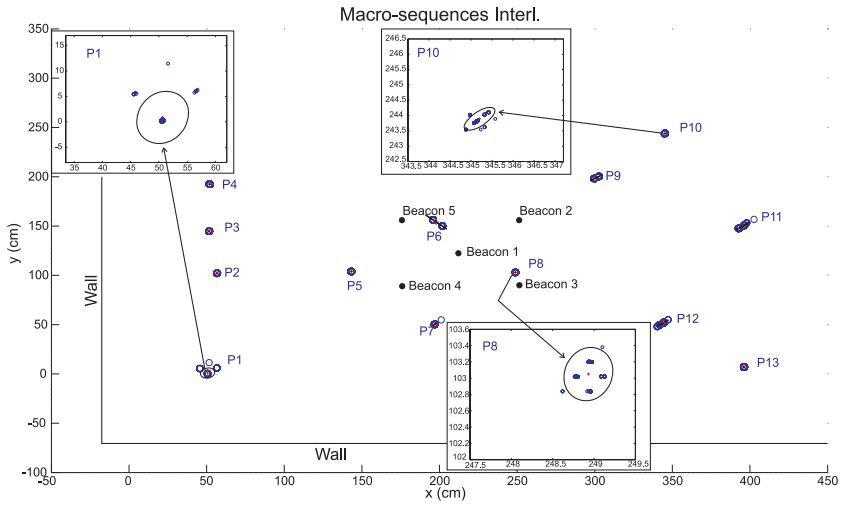


Figure 21: Position obtained with a 95 % confidence level when the emissions are encoded with macro-sequences generated from the interleaving of CSS.

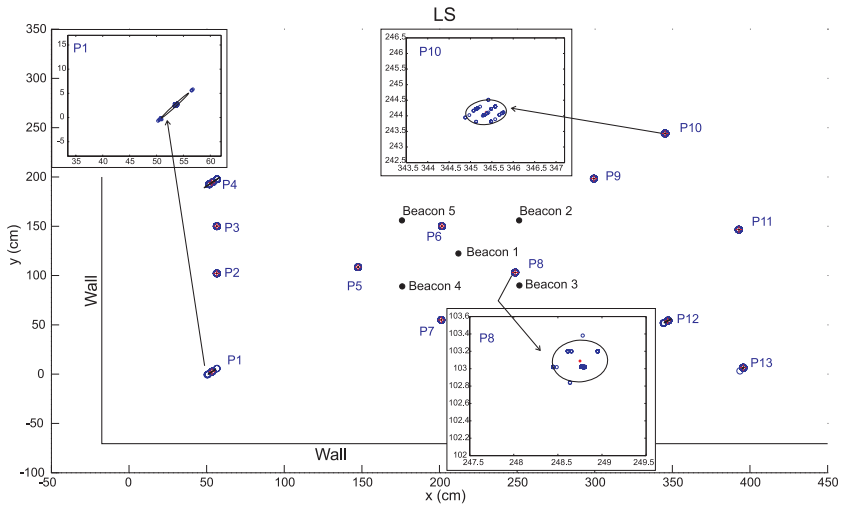


Figure 22: Position obtained with a 95 % confidence level when the emissions are encoded with LS codes.

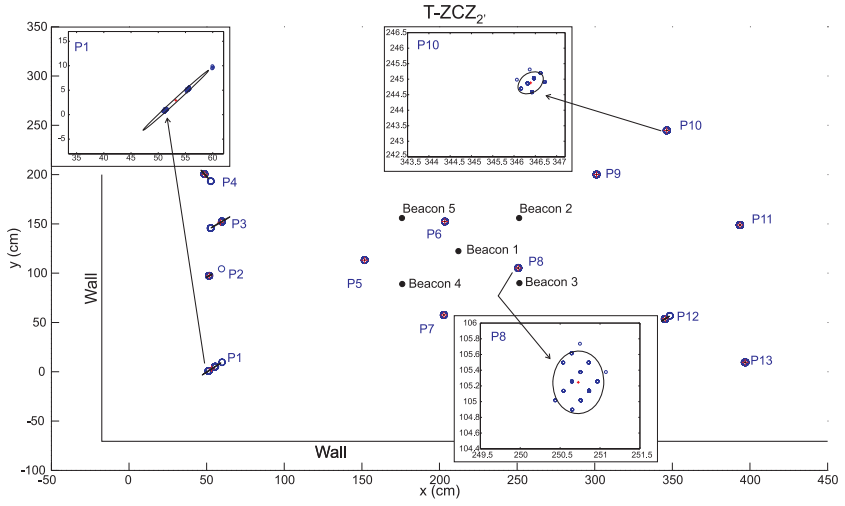


Figure 23: Position obtained with a 95 % confidence level when the emissions are encoded with T-ZCZ$_{2'}$ pairs.

6 Conclusions and future work

6.1 Conclusions

The aim of this thesis is the proposal of a code design approach for asynchronous ultrasonic sensory systems, capable to offer some kind of inherent immunity against multiple access interference (MAI) and inter-symbol interference (ISI). Bearing this in mind, existing and new sort of binary codes are evaluated using the following criteria: 1) high robustness to noise and the possibility of performing several simultaneous emissions without interference among them; 2) real-time processing with the emitted codes. The first criterion requires the use of codes with good aperiodic correlation properties. The second implies the availability of efficient correlation algorithms that reduce the number of operations to be carried out, allowing the implementation in reconfigurable hardware which is very adequate for the real-time computing of this kind of processing.

The main contributions of this thesis can be summarized as follows:

- Selection of binary codes with good aperiodic correlation properties and performance comparison.

  Subsets of Kasami, macro-sequences from CSS, LS and T-ZCZ codes with reduced aperiodic AC and CC interference values have been presented. The selected codes can be directly used in any CDMA system that requires them, assuring proper identification and high processing gains.

  The analytical and simulation results show that LS codes and T-ZCZ pairs are indeed more robust in rejecting the multipath and MAI interferes than the other codes, provided that the maximum relative time-offset among the codes is within the ZCZ. In a completely asynchronous application, macro-sequences from CSS offer similar bounds than Kasami codes when the length of the code is much greater than the number of simultaneous users.

- Hardware implementation of efficient algorithms for CSS.

  There exist efficient generation (ESSG) and correlation (ESSC) algorithms of CSS that give a reduction in the computational load in comparison with straighforward implementations. In this thesis, two new hardware design approaches of these algorithms are presented. One is a specific implementation that can be adapted to the requirements of the application at the time of hardware synthesis. The other can be configured in run-time, but it requires a large number of resources employed although not all of them are used all the time.

  Additionally, the pre-synthesis implementation has been adapted to the transmission scheme by macro-sequences, finding out that the ones generated from Golay pairs require less resources and increase the operating frequency. Furthermore, they achieve lower bounds.

- Development of efficient processing algorithms for LS codes.

  New algorithms have been developed for the generation and correlation of LS codes generated from Golay pairs and CSS with more than two sequences. By using them, the total number of operations to be performed is significantly reduced, and thus high speeds and long sequences are possible in applications that might require it. Furthermore, the hardware implementation of these algorithms has been accomplished in a FPGA architecture.

- Proposal of a novel encoding scheme with T-ZCZ pairs.

  T-ZCZ codes are of great interest for quasi-synchronous applications since they produce zero correlation zones in the middle and two terminal parts of the sum of their aperiodic correlation functions, and achieve a process gain of $2 \cdot L$ in the reception stage. In fact, they can replace CSS when the number of simultaneous emissions is greater than two: by using two sequences for each emitter is possible to have M simultaneous emitters and to get rid of ISI and MAI, if the maximum delay is shorter than the length of the ZCZ around the origin. With CSS, however, each emitter has to be encoded with M sequences.

  In this thesis, new methods to construct T-ZCZ pairs have been proposed. These new codes offer some advantages in comparison with the original T-ZCZ codes proposed in [ZYH05]: larger ZCZ for specific code lengths; lower values of interferences in the IZ; and the possibility of using efficient algorithms for its generation and correlation that decreases the computational load.

- Development of efficient processing algorithms for the proposed T-ZCZ pairs.

  A modular structure is provided for the efficient generation and correlation of T-ZCZ pairs, taking advantage of the properties of CSS from which these T-ZCZ codes are derived. These structures have two useful properties. On one hand, they allow the simultaneous generation of the M codes of a family. On the other hand, the $2 \cdot M$ outputs of each efficient correlator provide the correlation of the input signal with the M sequences pairs of the family code, decreasing the computational load if they are compared with straighforward implementations. This solution is also less complex than a FFT-based one. Furthermore, the proposed algorithms can be easily implemented in reconfigurable hardware to achieve real-time operation.

- Development of a LPS based on LS and T-ZCZ codes

  Encoding the emissions of the ultrasonic signals with LS codes and T-ZCZ pairs provides the system with a high immunity against multipath and near-far effect. The LPS has been designed to assure that the difference in times-of-flight among the codes does not exceed the size of the ZCZ. Thus, it is not necessary a synchronism trigger signal (radio-frequency, infrared, etc.) between the receiver and the beacons.

The system allows a non-limited number of mobile receivers working in the same environment.

6.2 Future work

This thesis leads to the crucial task of the sequence design for CDMA ultrasonic based systems. The results have shown that some improvements have been achieved thanks to the research in codes derived from CSS and the proposal of algorithms that decrease the detection computational load. However, a lot of work is yet to be done in this field; specifically, some of the tasks to be performed are:

- To devise novel codes with near perfect aperiodic correlation values.

  It is still a long way to construct the desirable optimal codes satisfying the theoretical bounds for different lengths. One of the contributions presented in this thesis is a new family code with three zero correlation zones, which is intended to handle the restrictions of CSS regarding to the number of sequences assigned to every emitter. Nevertheless, these T-ZCZ codes are not adequate when the number of simultaneous emissions increases, because very large codes have to be used to obtain a ZCZ wide enough.

  The efforts should be aimed towards the discovering of unitary codes with small or zero aperiodic correlations in a given window. LS could be an example, however the maximum number of codes of a specific length having this property is limited. Furthermore, it is not clear if the desired gain in terms of signal-to-noise ratio (SNR) is achieved through the insertion of zeros. It would be interesting to find unitary codes with only -1 and $+1$ elements and zero interferences in a window around the origin. Furthermore, it should be desirable to extend the size of this zero correlation zone to values near $\frac{L}{2}$, for an arbitrary code length L. Till now, only partial solutions have been found for short zero correlation zones and specific code lengths.

  Also, further research could be done in the development of less complex algorithms to search for subsets of codes with good asynchronous aperiodic correlation properties, so finding larger codes than those obtained from an exhaustive search would be possible.

- The implementation of fast correlators.

  Despite the current computation power of nowadays platforms, the up-going requirement for large simultaneous emissions and the use of long codes demand the use of efficient algorithms that give a reduction in the computational load and complexity of a hardware implementation. Hence, further research should be devoted to reduce the complexity and number of operations to be coped in the detection process.

- To study the application of the proposed hardware design approaches, for the efficient generation and correlation of CSS-based codes, to other technologies.

The results obtained from this thesis can be applied to improve the performance of other CDMA-based ultrasonic systems, such as NDE medical applications, ultrasounds under water or beamforming. As well as this, the wireless communications community is dealing now with the design of new codes for the incoming 4th generation network, being the most promising ones based on the use of CSS. Thus, the efficient implementations proposed in this thesis can be adapted to enhance the performance of such kind of communication systems.

With respect to the LPS developed in this thesis, it has been designed with the purpose of testing different encoding schemes in a experimental location system, and can be optimized in several aspects. For instance, the receiver units can be small and the emitters battery powered. Furthermore, a more sophisticated peak detector can be used, able to be adapted to the received signal level allowing the detection of weak echoes. Also, strategies to overcome the ISI and MAI problem can be used (as SIC post-processing). Again, these two previous suggestion try to minimize the influence of a poor code design. Other improvement could be the inclusion of several beacon structures to extend the area to be covered. In this proposal, the robots could move from the influence of one beacon structure to another. Soft-hand off can be possible by using the adequate codes. Also, it would be interesting the sensorial fusion between different localization strategies that takes advantage of the positive aspects of each system. For instance, if the fixed and mobile units can act as emitters or receivers, a more general system able to be adapted to different configurations can be achieved.

Appendix A Tables of preferred codes

In this appendix, tables containing a selection of preferred code sets in terms of aperiodic correlation are shown. The procedure to obtain these tables is detailed in Section 2.1.

On the other hand, the nomenclature used for the generation of the small set of Kasami sequences can be found in [FD96], where the *primitive polynomials* $h(x)$ that specify the *Linear Shift Register* (LFSR) from which Kasami codes are obtained (see Table A. 10). Notice that the initial value of the LFSR has been expressed in decimal base. The algorithm used to generate the CSS from which macro-sequences are derived can be found in [DMUH$^+$07], where the function of the *generation seed* is explained. The nomenclature used with LS codes is defined in [SBH01] when Golay pairs are the basis of the code; and in [ZYH05] when CSS are used. In both cases, the seeds needed to construct the initial Golay codes and CSS are given. However, it has been checked that the use of different seeds does not mean a significative change in the bound. Results in Section A.4 correspond to T-ZCZ pairs obtained when the generation method developed in [ZLH05] is used. Later, in Section A.5, results with the T-ZCZ codes proposed in this thesis are shown. These codes take advantage of CSS as shown in **[Pérez et al. 2008b]**. The generation seeds of these CSS are not included because they lead to T-ZCZ codes with similar properties, hence any seed can be chosen. In LS and T-ZCZ codes the percentage of non-zero values in the IZ is also given.

A.1 Small set of Kasami sequences

θ	θ_{AC}	θ_{CC}	$h(x)$	LFSR initialization	codes
L=15 and M=4					
$\mu = 2$					
0.3333	0.2667	0.3333	11001	11	0, 3
$\mu = 3$					
0.3333	0.3333	0.3333	11001	11	0, 1, 2
$\mu = 4$					
0.3333	0.3333	0.3333	11001	11	0, 1, 2, 3
L=63 y M=8					
$\mu = 2$					
0.1746	0.1587	0.1746	1011011	16	0, 3
$\mu = 3$					
0.1905	0.1746	0.1905	1110011	10	0, 5, 6
$\mu = 4$					
0.2063	0.1905	0.2063	1100001	50	0, 1, 2, 4

Continuation of **Table** 8

θ	θ_{AC}	θ_{CC}	$h(x)$	LFSR initialization	codes
$\mu = 5$					
0.2063	0.1905	0.2063	1100001	55	0, 1, 3, 5, 7
$\mu = 6$					
0.2063	0.2063	0.2063	1100001	50	0, 1, 2, 3, 4, 5
$\mu = 7$					
0.2063	0.2063	0.2063	1100001	50	0, 1, 2, 3, 4, 5, 6
$\mu = 8$					
0.2063	0.2063	0.2063	1100001	50	0, 1, 2, 3, 4, 5, 6, 7
L=255 y M=16					
$\mu = 2$					
0.0980	0.0902	0.0980	100101101	124	0, 13
$\mu = 3$					
0.1059	0.0941	0.1059	100101101	111	5, 7, 9
$\mu = 4$					
0.1059	0.1020	0.1059	100101101	55	0, 5, 7, 15
$\mu = 5$					
0.1137	0.0980	0.1137	100101101	169	0, 5, 7, 8, 10
$\mu = 6$					
0.1137	0.0980	0.1137	100101101	115	0, 5, 7, 8, 10, 14
$\mu = 7$					
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6, 7, 8
$\mu = 8$					
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6 7, 8, 10
$\mu = 9$					
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6 7, 8, 10, 13
$\mu = 10$					
0.1137	0.1020	0.1137	100101101	169	0, 2, 4, 5, 6, 7, 8, 10, 13, 14
$\mu = 11$					
0.1137	0.1059	0.1137	100101101	169	0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 13
$\mu = 12$					
0.1137	0.1059	0.1137	100101101	169	0, 2, 3, 4, 5, 6, 7, 8, 9, 11, 13, 14
$\mu = 13$					
0.1137	0.1098	0.1137	100101101	169	0, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14

Continuation of **Table** 8

θ	θ_{AC}	θ_{CC}	$h(x)$	LFSR initialization	codes
$\mu = 14$					
0.1137	0.1137	0.1137	100101101	41	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13
$\mu = 15$					
0.1137	0.1137	0.1137	100101101	41	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14
$\mu = 16$					
0.1137	0.1137	0.1137	100101101	41	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15

Table 8: Selection of Kasami codes depending on their length L, the number M of codes in a family and the number μ of simultaneous users.

A.2 Macro-sequences of CSS

$L_{Ms} = 16$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,8)	Concat.	0.3125	0.1875	0.3125	0, 3
(2,8)	Interl.	0.3125	0.3125	0.3125	1, 2
(4,4)	Concat.	0.3125	0.3125	0.25	0, 3
(4,4)	Interl.	0.4375	0.3125	0.4375	0, 3
$\mu = 3$					
(2,8)	Concat.	0.5	0.1875	0.5	0, 1, 2
(2,8)	Interl.	0.5	0.1875	0.5	0, 2, 4
(4,4)	Concat.	0.5	0.3125	0.5	0, 2, 3
(4,4)	Interl.	0.75	0.3125	0.75	0, 1, 2
$\mu = 4$					
(2,8)	Concat.	0.5	0.1875	0.5	0, 1, 2, 3
(2,8)	Interl.	0.5	0.3125	0.5	1, 3, 5, 7
(4,4)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3
(4,4)	Interl.	0.75	0.3125	0.75	0, 1, 2, 3
$\mu = 5$					
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 4
(2,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, 4, 5

Continuation of **Table** 9

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 6$					
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 4, 5
(2,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, 4, 5, 6
$\mu = 7$					
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 4, 5, 6
(2,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, 4, 5, 6, 7
$\mu = 8$					
(2,8)	Concat.	0.75	0.1875	0.75	0, 1, 2, $\cdots$, 7
(2,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, $\cdots$, 7

Table 9: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 16$ depending on the number μ of simultaneous users.

$\mathbf{L_{Ms} = 32}$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,16)	Concat.	0.2188	0.1875	0.2188	0, 8
(2,16)	Interl.	0.2813	0.2188	0.2813	1, 3
$\mu = 3$					
(2,16)	Concat.	0.3750	0.1875	0.3750	0, 2, 15
(2,16)	Interl.	0.3750	0.2813	0.3750	5, 6, 13
$\mu = 4$					
(2,16)	Concat.	0.3750	0.1875	0.3750	0, 2, 13, 15
(2,16)	Interl.	0.3750	0.2813	0.3750	5, 6, 13, 14
$\mu = 5$					
(2,16)	Concat.	0.6250	0.1875	0.6250	0, 1, 2, 3, 6
(2,16)	Interl.	0.6250	0.2188	0.6250	0, 1, 3, 4, 7
$\mu = 6$					
(2,16)	Concat.	0.6250	0.1875	0.6250	0, 1, 2, 3, 6, 7
(2,16)	Interl.	0.6250	0.2188	0.6250	0, 1, 3, 4, 7, 8
$\mu = 7$					
(2,16)	Concat.	0.6250	0.1875	0.6250	0, 2, 3, 6, 7, 11, 15
(2,16)	Interl.	0.6250	0.2188	0.6250	0, 1, 3, 4, 7, 8, 12
$\mu = 8$					
(2,16)	Concat.	0.6250	0.1875	0.6250	2, 3, 6, 7, 10, 11, 14, 15
(2,16)	Interl.	0.6250	0.2188	0.6250	0, 3, 4, 7, 8, 11, 12, 15

Continuation of **Table** 10

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 10$					
(2,16)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 6, 7, 8, 9, 10, 11
(2,16)	Interl.	0.75	0.2188	0.75	0, 1, 3, 4, 7 8, 11, 12, 13, 15
$\mu = 12$					
(2,16)	Concat.	0.75	0.1875	0.75	0, 1, 2, 3, 6, 7, 8, 9, 10, 11 14, 15
(2,16)	Interl.	0.75	0.2188	0.75	0, 1, 3, 4, 5 7, 8, 9, 11, 12, 13, 15
$\mu = 16$					
(2,16)	Concat.	0.75	0.1875	0.75	0, 1, 2, $\cdots$, 15
(2,16)	Interl.	0.75	0.2188	0.75	0, 1, 2, $\cdots$, 15

Table 10: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 32$ depending on the number μ of simultaneous users.

$\mathbf{L_{Ms} = 64}$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,32)	Concat.	0.2187	0.1094	0.2187	0, 16
(2,32)	Interl.	0.25	0.1719	0.25	2, 14
(4,16)	Concat.	0.3125	0.3125	0.1094	0, 12
(4,16)	Interl.	0.375	0.375	0.1719	0, 12
(8,8)	Concat.	0.3125	0.3125	0.125	0, 7
(8,8)	Interl.	0.3125	0.3125	0.2344	0, 7
$\mu = 3$					
(2,32)	Concat.	0.3125	0.1563	0.3125	0, 3, 12
(2,32)	Interl.	0.3594	0.1719	0.3594	1, 2, 13
(4,16)	Concat.	0.3125	0.3125	0.25	0, 3, 12
(4,16)	Interl.	0.375	0.375	0.3125	0, 5, 11
(8,8)	Concat.	0.3125	0.3125	0.25	0, 3, 5
(8,8)	Interl.	0.4375	0.3125	0.4375	0, 3, 5
$\mu = 4$					
(2,32)	Concat.	0.3125	0.1563	0.3125	0, 3, 12, 15
(2,32)	Interl.	0.3594	0.1719	0.3594	1, 2, 13, 14
(4,16)	Concat.	0.3125	0.3125	0.25	0, 3, 12, 15
(4,16)	Interl.	0.375	0.375	0.3125	0, 5, 10, 15
(8,8)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6
(8,8)	Interl.	0.4375	0.3125	0.4375	0, 3, 5, 6
$\mu = 5$					

Continuation of **Table** 11

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
(2,32)	Concat.	0.4375	0.1094	0.4375	0, 2, 9, 11, 16
(2,32)	Interl.	0.4375	0.1719	0.4375	1, 2, 13, 14, 19
(4,16)	Concat.	0.3438	0.3125	0.3438	0, 3, 8, 11, 12
(4,16)	Interl.	0.375	0.375	0.375	0, 3, 8, 11, 12
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4
(8,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, 3, 4
$\mu = 6$					
(2,32)	Concat.	0.4375	0.1094	0.4375	0, 2, 9, 11, 16, 18
(2,32)	Interl.	0.4375	0.1719	0.4375	1, 3, 5, 10, 12, 14
(4,16)	Concat.	0.3438	0.3125	0.3438	0, 3, 8, 11, 12, 15
(4,16)	Interl.	0.375	0.375	0.375	0, 7, 8, 11, 12, 15
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5
(8,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, 3, 4, 5
$\mu = 7$					
(2,32)	Concat.	0.4375	0.1563	0.4375	0, 2, 4, 6, 9, 11, 13
(2,32)	Interl.	0.4375	0.1719	0.4375	1, 3, 5, 10, 12, 26, 28
(4,16)	Concat.	0.4063	0.3125	0.4063	0, 3, 4, 7, 8, 11, 12
(4,16)	Interl.	0.3906	0.375	0.3906	1, 2, 5, 9, 10, 13, 14
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6
(8,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, 3, 4, 5, 6
$\mu = 8$					
(2,32)	Concat.	0.4375	0.1563	0.4375	0, 2, 4, 6, 9, 11, 13, 15
(2,32)	Interl.	0.4375	0.1719	0.4375	1, 3, 10, 12, 17, 20, 26, 28
(4,16)	Concat.	0.4063	0.3125	0.4063	0, 3, 4, 7, 8, 11, 12, 15
(4,16)	Interl.	0.4219	0.375	0.4219	1, 2, 5, 6, 9, 10, 13, 14
(8,8)	Concat.	0.5	0.3125	0.5	0, 1, 2, $\cdots$ 6
(8,8)	Interl.	0.75	0.3125	0.75	0, 1, 2, $\cdots$, 7
$\mu = 10$					
(2,32)	Concat.	0.625	0.1094	0.625	0, 1, 2, 3, 8, 9, 10, 11, 16, 17
(2,32)	Interl.	0.625	0.1719	0.625	1, 2, 3, 4, 5, 6, 9, 10, 11, 12
(4,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4 5, 6, 7, 8, 9
(4,16)	Entrel.	0.5156	0.375	0.5156	0, 1, 2, 3, 4 5, 6, 7, 8, 9
$\mu = 16$					
(2,32)	Concat.	0.625	0.1094	0.625	0, 1, 2, 3, 8, 9, 10, 11, 16, 17, 18, 19, 24, 25, 26, 27

Continuation of **Table** 11

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
(2,32)	Interl.	0.625	0.1719	0.625	1, 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 19, 20, 27, 28
(4,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
(4,16)	Interl.	0.5156	0.375	0.5156	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
$\mu = 32$					
(2,32)	Concat.	0.75	0.1562	0.75	0, 1, 2, $\cdots$, 31
(2,32)	Interl.	0.75	0.2031	0.75	0, 1, 2, $\cdots$, 31

Table 11: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 64$ depending on the number μ of simultaneous users.

$L_{Ms} = 128$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,64)	Concat.	0.1484	0.1406	0.1484	8, 40
(2,64)	Interl.	0.1953	0.1641	0.1953	6, 38
$\mu = 3$					
(2,64)	Concat.	0.2578	0.1094	0.2578	0, 2, 37
(2,64)	Interl.	0.2578	0.1484	0.2578	0, 37, 39
$\mu = 4$					
(2,64)	Concat.	0.2578	0.1094	0.2578	0, 2, 37, 39
(2,64)	Interl.	0.2734	0.1797	0.2734	0, 2, 37, 39
$\mu = 5$					
(2,64)	Concat.	0.3125	0.1094	0.3125	0, 2, 13, 15, 21
(2,64)	Interl.	0.3203	0.1797	0.3203	7, 18, 25, 39, 50
$\mu = 6$					
(2,64)	Concat.	0.3125	0.1094	0.3125	0, 2, 13, 15, 21, 23
(2,64)	Interl.	0.3203	0.1797	0.3203	7, 18, 25, 39, 50, 57
$\mu = 7$					
(2,64)	Concat.	0.3125	0.1406	0.3125	0, 2, 13, 15, 21, 23, 24
(2,64)	Interl.	0.3359	0.1797	0.3359	0, 2, 13, 15, 21, 23, 24
$\mu = 8$					
(2,64)	Concat.	0.3125	0.1406	0.3125	0, 2, 13, 15, 21, 23, 24, 26
(2,64)	Interl.	0.3516	0.1797	0.3516	0, 2, 13, 15, 21, 23, 24, 26
$\mu = 10$					

Continuation of **Table** 12

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
(2,64)	Concat.	0.4063	0.1406	0.4063	0, 13, 19, 21, 23, 24, 26, 30, 52, 58
(2,64)	Interl.	0.4063	0.1797	0.4063	1, 22, 25, 27, 31, 34, 53, 56, 58, 60
$\mu = 64$					
(2,64)	Concat.	0.75	0.1406	0.75	0, 1, 2, $\cdots$, 63
(2,64)	Interl.	0.75	0.1953	0.75	0, 1, 2, $\cdots$, 63

Table 12: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 128$ depending on the number μ of simultaneous users.

$\mathbf{L_{Ms} = 256}$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,128)	Concat.	0.1328	0.0859	0.1328	12, 76
(2,128)	Interl.	0.1601	0.1367	0.1602	0, 64
(4,64)	Concat.	0.1953	0.1953	0.0938	4, 18
(4,64)	Interl.	0.1914	0.1914	0.0977	4, 11
(16,16)	Concat.	0.3125	0.3125	0.0625	0, 15
(16,16)	Interl.	0.3125	0.3125	0.1211	0, 15
$\mu = 3$					
(2,128)	Concat.	0.2188	0.0859	0.2188	8, 54, 79
(2,128)	Interl.	0.2305	0.1289	0.2305	4, 74, 125
(4,64)	Concat.	0.1953	0.1953	0.1875	0, 11, 20
(4,64)	Interl.	0.2188	0.1914	0.2188	0, 11, 20
(16,16)	Concat.	0.2656	0.2656	0.25	8, 11, 13
(16,16)	Interl.	0.3125	0.3125	0.2383	0, 7, 13
$\mu = 4$					
(2,128)	Concat.	0.2344	0.0859	0.2344	8, 54, 79, 113
(2,128)	Interl.	0.2344	0.1211	0.2344	13, 62, 77, 126
(4,64)	Concat.	0.1953	0.1953	0.1875	0, 11, 20, 31
(4,64)	Interl.	0.2188	0.1992	0.2188	0, 11, 20, 31
(16,16)	Concat.	0.2656	0.2656	0.25	8, 11, 13, 14
(16,16)	Interl.	0.3125	0.3125	0.2461	3, 4, 8, 15
$\mu = 5$					
(2,128)	Concat.	0.2578	0.1016	0.2578	8, 57, 66, 70, 115
(2,128)	Interl.	0.2813	0.1211	0.2813	1, 17, 42, 52, 115
(4,64)	Concat.	0.25	0.1953	0.25	0, 3, 12, 15, 32
(4,64)	Interl.	0.25	0.1992	0.25	1, 13, 33, 36, 40

Continuation of **Table** 13

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9
(16,16)	Interl.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9
$\mu = 6$					
(2,128)	Concat.	0.2578	0.0977	0.2578	16, 22, 105, 107, 109, 111
(2,128)	Interl.	0.2813	0.1289	0.2813	8, 10, 24, 45, 61, 63
(4,64)	Concat.	0.25	0.1953	0.25	0, 3, 12, 15, 32, 35
(4,64)	Interl.	0.2617	0.1992	0.2617	1, 3, 5, 9, 10, 12
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10
(16,16)	Interl.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10
$\mu = 7$					
(2,128)	Concat.	0.2813	0.0977	0.2813	4, 9, 11, 20, 25, 27, 46
(2,128)	Interl.	0.2813	0.1367	0.2813	8, 10, 24, 45, 47, 61, 63
(4,64)	Concat.	0.2578	0.1953	0.2578	0, 3, 5, 6, 9, 10, 12
(4,64)	Interl.	0.2617	0.1992	0.2617	1, 3, 5, 9, 10, 12, 15
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10, 12
(16,16)	Interl.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10, 12
$\mu = 8$					
(2,128)	Concat.	0.2813	0.1016	0.2813	0, 13, 15, 16, 29, 31, 42, 58
(2,128)	Interl.	0.2930	0.1445	0.2930	9, 11, 25, 27, 44, 46, 60, 62
(4,64)	Concat.	0.2578	0.1953	0.2578	0, 3, 5, 6, 9, 10, 12, 15
(4,64)	Interl.	0.2617	0.207	0.2617	0, 3, 5, 6, 9, 10, 12, 15
(16,16)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10, 12, 15
(16,16)	Interl.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10, 12, 15
$\mu = 10$					
(2,128)	Concat.	0.3125	0.1016	0.3125	0, 2, 4, 6, 65, 67, 109, 111, 125, 127
(2,128)	Interl.	0.3281	0.1289	0.3281	4, 6, 11, 25, 39, 42, 55, 56, 58, 69
(16,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5 6, 7, 8, 9
(16,16)	Interl.	0.75	0.3125	0.75	0, 1, 2, 3, 4, 5 6, 7, 8, 9
$\mu = 16$					
(16,16)	Concat.	0.5	0.3125	0.5	0, 1, 2, $\cdots$, 15
(16,16)	Interl.	0.75	0.3125	0.75	0, 1, 2, $\cdots$, 15
$\mu = 64$					
(4,64)	Concat.	0.75	0.1953	0.75	0, 1, 2, $\cdots$, 63
(4,64)	Interl.	0.75	0.2070	0.75	0, 1, 2, $\cdots$, 63
$\mu = 128$					

Continuation of **Table** 13

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
(2,128)	Concat.	0.75	0.1016	0.75	0, 1, 2, $\cdots$, 127
(2,128)	Interl.	0.75	0.1445	0.75	0, 1, 2, $\cdots$, 127

Table 13: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 256$ depending on the number μ of simultaneous users.

$L_{Ms} = 512$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,256)	Concat.	0.0977	0.082	0.0977	24, 152
(2,256)	Interl.	0.1348	0.1152	0.1348	50, 178
(8,64)	Concat.	0.3281	0.3281	0.0527	16, 40
(8,64)	Interl.	0.375	0.375	0.0723	16, 40
$\mu = 3$					
(2,256)	Concat.	0.1797	0.0742	0.1797	64, 199, 233
(2,256)	Interl.	0.1934	0.0996	0.1934	8, 23, 153
(8,64)	Concat.	0.3125	0.3125	0.1172	8, 19, 61
(8,64)	Interl.	0.375	0.375	0.1328	0, 42, 63
$\mu = 4$					
(2,256)	Concat.	0.1914	0.082	0.1914	36, 60, 175, 183
(2,256)	Interl.	0.1953	0.1035	0.1953	14, 17, 36, 59
(8,64)	Concat.	0.3125	0.3125	0.1328	8, 16, 31, 52
(8,64)	Interl.	0.375	0.375	0.1465	19, 24, 39, 44
$\mu = 5$					
(2,256)	Concat.	0.2187	0.0742	0.2188	0, 23, 31, 148, 156
(2,256)	Interl.	0.2187	0.1074	0.2188	4, 12, 19, 148, 156
(8,64)	Concat.	0.2969	0.2969	0.25	1, 18, 21, 24, 27
(8,64)	Interl.	0.375	0.375	0.3535	1, 4, 6, 7, 9
$\mu = 6$					
(2,256)	Concat.	0.2187	0.0898	0.2188	0, 8, 23, 31, 148, 156
(2,256)	Interl.	0.2344	0.0996	0.2344	36, 73, 119, 164, 201, 23
$\mu = 7$					
(2,256)	Concat.	0.2344	0.082	0.2344	0, 12, 39, 43, 213, 217, 242
(2,256)	Interl.	0.2344	0.123	0.2344	4, 12, 27, 102, 143, 144, 242
$\mu = 8$					
(2,256)	Concat.	0.2422	0.082	0.2422	12, 39, 85, 126, 140, 167, 213, 254
(2,256)	Interl.	0.2441	0.1151	0.2441	14, 45, 81, 96, 146, 163, 223, 252

Continuation of **Table** 14

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 64$					
(8,64)	Concat.	0.5	0.3281	0.5	0, 1, 2, $\cdots$, 63
(8,64)	Interl.	0.5020	0.3750	0.5020	0, 1, 2, $\cdots$, 63
$\mu = 256$					
(2,256)	Concat.	0.75	0.0977	0.75	0, 1, 2, $\cdots$, 255
(2,256)	Interl.	0.75	0.1269	0.75	0, 1, 2, $\cdots$, 255

Table 14: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 512$ depending on the number μ of simultaneous users.

$\mathbf{L_{Ms} = 1024}$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,512)	Concat.	0.0811	0.0684	0.0811	240, 496
(2,512)	Interl.	0.1064	0.1064	0.1064	60, 316
(32,32)	Concat.	0.3125	0.3125	0.0313	0, 31
(32,32)	Interl.	0.3125	0.3125	0.0615	0, 31
$\mu = 3$					
(2,512)	Concat.	0.1484	0.0645	0.1484	32, 223, 229
(2,512)	Interl.	0.1484	0.0908	0.1484	65, 123, 132
(32,32)	Concat.	0.2539	0.2539	0.25	24, 29, 30
(32,32)	Interl.	0.3125	0.3125	0.1943	0, 14, 23
$\mu = 4$					
(2,512)	Concat.	0.1523	0.0645	0.1523	172, 223, 428, 479
(2,512)	Interl.	0.1523	0.0850	0.1523	173, 222, 429, 478
(32,32)	Concat.	0.2539	0.2539	0.25	24, 27, 29, 30
(32,32)	Interl.	0.3125	0.3125	0.2178	5, 10, 22, 25
$\mu = 5$					
(2,512)	Concat.	0.1758	0.0645	0.1758	92, 102, 201, 378, 469
(2,512)	Interl.	0.1758	0.0811	0.1758	28, 64, 151, 314, 407
(32,32)	Concat.	0.2656	0.2656	0.25	16, 25, 26, 28, 31
(32,32)	Interl.	0.3125	0.3125	0.249	1, 12, 20, 24, 31
$\mu = 6$					
(2,512)	Concat.	0.1885	0.0684	0.1885	0, 84, 169, 253, 319, 363
(2,512)	Interl.	0.1914	0.0908	0.1914	78, 81, 100, 123, 338, 359
(32,32)	Concat.	0.2656	0.2656	0.25	16, 19, 21, 22, 25, 26
(32,32)	Interl.	0.3125	0.3125	0.249	0, 7, 11, 12, 20, 25

Continuation of **Table** 15

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 7$					
(2,512)	Concat.	0.1914	0.0684	0.1914	4, 80, 173, 249, 315,
(2,512)	Interl.	0.2031	0.085	0.2031	33, 92, 136, 159, 245, 342, 386
(32,32)	Concat.	0.2656	0.2656	0.25	16, 19, 21, 22, 25, 26, 28
(32,32)	Interl.	0.3125	0.3125	0.2568	1, 7, 11, 12, 19, 20, 24
$\mu = 8$					
(2,512)	Concat.	0.1992	0.0684	0.1992	8, 24, 198, 217, 267, 283, 453, 474
(2,512)	Interl.	0.2070	0.083	0.207	8, 45, 50, 231, 267, 302, 305, 484
(32,32)	Concat.	0.2656	0.2656	0.25	16, 19, 21, 22, 25, 26, 28, 31
(32,32)	Interl.	0.3125	0.3125	0.2568	1, 7, 11, 12, 19, 20, 24, 31
$\mu = 10$					
(32,32)	Concat.	0.3125	0.3125	0.25	0, 3, 5, 6, 9, 10, 12, 15, 17, 18
(32,32)	Interl.	0.4375	0.3125	0.4375	0, 3, 5, 6, 9, 10, 12, 15, 17, 18
$\mu = 32$					
(32,32)	Concat.	0.5	0.3125	0.5	0, 1, 2, $\cdots$, 31
(32,32)	Interl.	0.75	0.3125	0.75	0, 1, 2, $\cdots$, 31
$\mu = 512$					
(2,512)	Concat.	0.75	0.0727	0.75	0, 1, 2, $\cdots$, 255
(2,512)	Interl.	0.75	0.1064	0.75	0, 1, 2, $\cdots$, 255

Table 15: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 1024$ depending on the number μ of simultaneous users.

$L_{Ms} = 2048$

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 2$					
(2,1024)	Concat.	0.0674	0.0576	0.0674	200, 712
(2,1024)	Interl.	0.0835	0.0649	0.0835	109, 621
$\mu = 3$					
(2,1024)	Concat.	0.1230	0.0518	0.123	40, 215, 637
(2,1024)	Interl.	0.123	0.0659	0.123	6, 249, 595
$\mu = 4$					
(2,1024)	Concat.	0.1230	0.0557	0.1230	8, 355, 520, 867
(2,1024)	Interl.	0.1284	0.0591	0.1284	163, 316, 675, 828
$\mu = 5$					
(2,1024)	Concat.	0.1484	0.0576	0.1484	36, 134, 629, 911, 1019
(2,1024)	Interl.	0.1523	0.0728	0.1523	109, 306, 464, 643, 799

Continuation of **Table** 16

(M,L)	Method	θ	θ_{AC}	θ_{CC}	Generation seed
$\mu = 6$					
(2,1024)	Concat.	0.1543	0.0557	0.1543	32, 317, 475, 544, 829, 987
(2,1024)	Interl.	0.1602	0.0747	0.1602	3, 41, 348, 543, 565, 832
$\mu = 1024$					
(2,1024)	Concat.	0.75	0.0635	0.75	0, 1, 2, $\cdots$, 1023
(2,1024)	Interl.	0.75	0.0874	0.75	0, 1, 2, $\cdots$, 1023

Table 16: Selection of macro-sequences $Ms(M, L)$ with length $L_{MS} = 2048$ depending on the number μ of simultaneous users.

A.3 LS codes

M = 2; K = 2

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
4	11	2	0.375	0.25	0.375	0	0, 1	85.71 %	85.71 %
8	23	2	0.375	0.1875	0.375	0	0, 1	80 %	80 %
16	47	2	0.2188	0.1875	0.2188	0	0, 1	77.42 %	77.42 %
32	95	2	0.2188	0.1094	0.2188	0	0, 1	76.19 %	76.19 %
64	191	2	0.1719	0.1094	0.1719	0	0, 1	75.59 %	75.59 %
128	383	2	0.1328	0.0859	0.1328	12	0, 1	75.29 %	75.29 %
256	767	2	0.0977	0.082	0.0977	24	0, 1	75.15 %	75.15 %
512	1535	2	0.0879	0.0566	0.0879	52	0, 1	75.07 %	75.07 %
1024	3071	2	0.0693	0.0459	0.0693	84	0, 1	75.04 %	75.04 %

Table 17: Selection of LS codes, with family size $K = 2$ and length of the ZCZ $Z_C = L_0 - 1$, depending on the number μ of simultaneous users. The codes have been generated from Golay pairs with length L_0 as indicated in [SBH01].

M = 2; K = 4

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
4	19	2	0.375	0.1875	0.375	0	0, 1	80 %	80 %
		3	0.5625	0.1875	0.5625	0	0, 1, 2		
		4	0.5625	0.1875	0.5625	0	0, 1, 2, 3		
8	39	2	0.2188	0.1875	0.2188	0	0, 1	77.42 %	77.42 %
		3	0.5313	0.1875	0.5313	0	0, 1, 2		
		4	0.5313	0.1875	0.5313	0	0, 1, 2, 3		
16	79	2	0.2188	0.1094	0.2188	0	0, 1	76.19 %	76.19 %
		3	0.5156	0.1094	0.5156	0	0, 1, 2		
		4	0.5156	0.1094	0.5156	0	0, 1, 2, 3		

Continuation of **Table** 18

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
32	159	2	0.1719	0.1094	0.1719	0	2, 3	75.59 %	75.59 %
		3	0.5078	0.1094	0.5078	0	0, 1, 2		
		4	0.5078	0.1094	0.5078	0	0, 1, 2, 3		
64	319	2	0.1328	0.0859	0.1328	12	0, 1	75.29 %	75.29 %
		3	0.5039	0.0742	0.5039	8	0, 1, 2		
		4	0.5039	0.0742	0.5039	8	0, 1, 2, 3		
128	639	2	0.0977	0.082	0.0977	24	0, 1	75.15 %	75.15 %
		3	0.502	0.0664	0.502	12	0, 1, 2		
		4	0.502	0.0664	0.502	12	0, 1, 2, 3		
256	1279	2	0.0879	0.0566	0.0879	52	0, 1	75.07 %	75.07 %
		3	0.501	0.0488	0.501	24	0, 1, 2		
		4	0.501	0.0488	0.501	24	0, 1, 2, 3		
512	2559	2	0.0693	0.0459	0.0693	84	0, 1	75.04 %	75.04 %
		3	0.5005	0.0405	0.5005	240	0, 1, 2		
		4	0.5005	0.0405	0.5005	240	0, 1, 2, 3		

Table 18: Selection of LS codes, with family size $K = 4$ and length of the ZCZ $Z_C = L_0 - 1$, depending on the number μ of simultaneous users. The codes have been generated from Golay pairs with length L_0 as indicated in [SBH01].

M = 2; K = 8

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
4	35	2	0.5312	0.5312	0.1875	1	4, 7	77.42 %	77.42 %
		3	0.5313	0.5313	0.3125	0	0, 5, 7		
		4	0.5313	0.5313	0.4375	0	0, 2, 5, 7		
		5	0.5313	0.5313	0.5313	0	0, 1, 2, 3, 4		
		6	0.5313	0.5313	0.5313	0	0, 1, 2, 3, 4, 6		
		7	0.5938	0.5313	0.5938	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5938	0.5313	0.5938	0	0, $\cdots$, 7		
8	71	2	0.5156	0.5156	0.1563	2	0, 3	76.19 %	76.19 %
		3	0.5156	0.5156	0.2656	0	0, 2, 7		
		4	0.5156	0.5156	0.2969	0	0, 2, 5, 7		
		5	0.5156	0.5156	0.5156	0	0, 1, 2, 3, 4		
		6	0.5156	0.5156	0.5156	0	0, 1, 2, 3, 4, 6		
		7	0.5469	0.5156	0.5469	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5469	0.5156	0.5479	0	0, $\cdots$, 7		
16	143	2	0.5078	0.5078	0.1406	0	0, 3	75.59 %	75.59 %
		3	0.5078	0.5078	0.2578	0	0, 2, 7		
		4	0.5078	0.5078	0.2734	0	0, 2, 5, 7		
		5	0.5078	0.5078	0.5078	0	0, 1, 2, 3, 4		
		6	0.5078	0.5078	0.5078	0	0, 1, 2, 3, 4, 6		

Continuation of **Table** 19

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
		7	0.5234	0.5078	0.5234	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5234	0.5078	0.5234	0	0, $\cdots$, 7		
		2	0.5039	0.5039	0.0859	0	0, 3		
		3	0.5039	0.5039	0.2539	0	0, 2, 7		
		4	0.5039	0.5039	0.2617	0	0, 2, 5, 7		
32	287	5	0.5039	0.5039	0.5039	0	0, 1, 2, 3, 4	75.29 %	75.29 %
		6	0.5039	0.5039	0.5039	0	0, 1, 2, 3, 4, 6		
		7	0.5117	0.5039	0.5117	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5117	0.5039	0.5117	0	0, $\cdots$, 7		
		2	0.502	0.502	0.082	4	0, 3		
		3	0.502	0.502	0.252	0	0, 2, 7		
64	575	4	0.502	0.502	0.2559	0	0, 2, 5, 7	75.15 %	75.15 %
		5	0.502	0.502	0.502	0	0, 1, 2, 3, 4		
		6	0.502	0.502	0.502	0	0, 1, 2, 3, 4, 6		
		7	0.5059	0.502	0.5059	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5059	0.502	0.5059	0	0, $\cdots$, 7		
		2	0.501	0.501	0.0645	8	0, 3		
		3	0.501	0.501	0.251	0	0, 2, 7		
		4	0.501	0.501	0.2529	0	0, 2, 5, 7		
128	1151	5	0.501	0.501	0.501	0	0, 1, 2, 3, 4	75.07 %	75.07 %
		6	0.501	0.501	0.501	0	0, 1, 2, 3, 4, 6		
		7	0.5015	0.501	0.5029	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5015	0.501	0.5029	0	0, $\cdots$, 7		
		2	0.5005	0.5005	0.0557	20	0, 3		
		3	0.5005	0.5005	0.2505	0	0, 2, 7		
		4	0.5005	0.5005	0.2515	0	0, 2, 5, 7		
256	2303	5	0.5005	0.5005	0.5005	0	0, 1, 2, 3, 4	75.04 %	75.04 %
		6	0.5005	0.5005	0.5005	0	0, 1, 2, 3, 4, 6		
		7	0.5015	0.5005	0.5015	0	0, 1, 2, 3, 4, 5, 6		
		8	0.5015	0.5005	0.5015	0	0, $\cdots$, 7		

Table 19: Selection of LS codes, with family size $K = 8$ and length of the ZCZ $Z_C = L_0 - 1$, depending on the number μ of simultaneous users. The codes have been generated from Golay pairs with length L_0 as indicated in [SBH01].

M = 2; K = 16

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
		2	0.4844	0.4844	0.2188	0	3, 6		
		3	0.5156	0.5156	0.4219	0	3, 7, 10		
		4	0.5156	0.5156	0.4531	0	3, 7, 10, 14		
4	67	5	0.7656	0.7656	0.2969	0	0, 1, 5, 10, 14	76.19 %	76.19 %

Continuation of **Table** 20

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
		6	0.7344	0.7344	0.4531	0	2, 5, 6, 8, 11, 15		
		7	0.7656	0.7656	0.4531	0	0, 1, 2, 5, 6, 11, 15		
		8	0.7656	0.7656	0.4688	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.7656	0.7656	0.5156	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.7656	0.7656	0.7344	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.7656	0.7656	0.7656	0	0, $\cdots$, 15		
8	135	2	0.4922	0.4922	0.1719	2	2, 7	75.59 %	75.59 %
		3	0.5078	0.5078	0.3984	0	3, 7, 10		
		4	0.5078	0.5078	0.4141	0	3, 7, 10, 14		
		5	0.7578	0.7578	0.2578	0	0, 1, 5, 10, 14		
		6	0.7578	0.7578	0.2891	0	2, 7, 8, 9, 12, 13		
		7	0.7578	0.7578	0.3984	0	0, 3, 4, 7, 9, 10, 13		
		8	0.7578	0.7578	0.4141	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.7578	0.7578	0.5078	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.7578	0.7578	0.7422	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.7578	0.7578	0.7578	0	0, $\cdots$, 15		
16	271	2	0.4961	0.4961	0.1641	0	3, 6	75.29 %	75.29 %
		3	0.5039	0.5039	0.3867	0	3, 7, 10		
		4	0.5039	0.5039	0.3945	0	3, 7, 10, 14		
		5	0.7539	0.7539	0.2539	0	0, 1, 5, 10, 14		
		6	0.7539	0.7539	0.2695	0	2, 7, 8, 9, 12, 13		
		7	0.7539	0.7539	0.3867	0	0, 3, 4, 7, 9, 10, 13		
		8	0.7539	0.7539	0.3945	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.7539	0.7539	0.5039	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.7539	0.7539	0.7461	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.7539	0.7539	0.7539	0	0, $\cdots$, 15		
32	543	2	0.498	0.0957	0.498	0	3, 6	75.15 %	75.15 %
		3	0.502	0.502	0.3809	0	3, 7, 10		
		4	0.502	0.502	0.3848	0	3, 7, 10, 14		
		5	0.752	0.752	0.252	0	0, 1, 5, 10, 14		
		6	0.752	0.752	0.2598	0	2, 7, 8, 9, 12, 13		
		7	0.752	0.752	0.3809	0	0, 3, 4, 7, 9, 10, 13		

Continuation of **Table** 20

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
		8	0.752	0.752	0.3848	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.752	0.752	0.502	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.752	0.752	0.748	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.752	0.752	0.752	0	0, $\cdots$, 15		
		2	0.499	0.0957	0.499	4	2, 7		
		3	0.501	0.501	0.3779	0	3, 7, 10		
		4	0.501	0.501	0.3799	0	3, 7, 10, 14		
		5	0.751	0.751	0.251	0	0, 1, 5, 10, 14		
64	1087	6	0.751	0.751	0.2549	0	2, 7, 8, 9, 12, 13	75.07 %	75.07 %
		7	0.751	0.751	0.3779	0	0, 3, 4, 7, 9, 10, 13		
		8	0.751	0.751	0.3799	0	0, 3, 4, 7, 9, 10, 13, 14		
		10	0.751	0.751	0.501	0	0, 1, 3, 4, 5, 7, 8, 10, 12, 14		
		14	0.751	0.751	0.749	0	0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14		
		16	0.751	0.751	0.751	0	0, $\cdots$, 15		

Table 20: Selection of LS codes, with family size $K = 16$ and length of the ZCZ $Z_C = L_0 - 1$, depending on the number μ of simultaneous users. The codes have been generated from Golay pairs with length L_0 as indicated in [SBH01].

M = 4; K = 16

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
		2	0.2656	0.2656	0.1094	0	1, 2		
		3	0.2656	0.2656	0.25	1	0, 3, 5		
		4	0.2969	0.2656	0.2969	0	8, 9, 11, 15		
		5	0.5	0.2656	0.5	0	0, 1, 3, 7, 15		
4	73	6	0.5156	0.2656	0.5156	0	0, 1, 2, 3, 8, 9	89.96 %	89.96 %
		7	0.5156	0.2656	0.5156	0	0, 1, 2, 3, 8, 9, 10		
		8	0.5156	0.2656	0.5156	0	0, 1, 2, 3, 8, 9, 10, 11		
		10	0.7344	0.2656	0.7344	0	0, 3, 5, 7, 8, 9, 10, 11, 13, 15		
		14	0.7656	0.3125	0.7656	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		
		16	0.7656	0.3125	0.7656	0	0, $\cdots$, 15		
		2	0.1953	0.1953	0.1094	0	1, 2		
		3	0.2617	0.1797	0.2617	1	3, 5, 7		

Continuation of **Table** 21

L_0	L	μ	θ	θ_{AC}	θ_{CC}	seed	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
16	301	4	0.25	0.1953	0.25	0	1, 3, 8, 10	80.70 %	78.60 %
		5	0.4805	0.1953	0.4805	1	7, 8, 9, 11, 15		
		6	0.5039	0.1953	0.5039	0	0, 1, 2, 3, 8, 9		
		7	0.5039	0.1953	0.5039	0	0, 1, 2, 3, 8, 9, 10		
		8	0.5039	0.1953	0.5039	1	0, 1, 2, 3, 8, 9, 10, 11		
		10	0.7641	0.1953	0.7461	0	0, 1, 2, 3, 4, 6, 9, 12, 14		
		14	0.7539	0.1953	0.7539	1	0, 1, 2, 3, 5, 7, 8, 9, 10, 11, 12, 13, 14, 15		
		16	0.7539	0.2266	0.7539	0	0, $\cdots$, 15		
64	1213	2	0.1875	0.1875	0.0675	12	9, 11	79.90 %	81.20 %
		3	0.1855	0.1367	0.1855	4	4, 5, 7		
		4	0.1973	0.1973	0.1367	4	4, 5, 6, 7		
		5	0.5068	0.1367	0.5068	0	4, 5, 6, 7, 13		
		6	0.5068	0.1367	0.5068	0	4, 5, 6, 7, 13, 15		
		7	0.5068	0.1367	0.5068	0	4, 5, 6, 7, 12, 13, 14		
		8	0.749	0.1875	0.749	0	4, 5, 6, 7, 12, 13, 14, 15		
		10	0.749	0.1875	0.749	0	0, 1, 6, 7, 8, 9 10, 11, 14, 15		
		14	0.751	0.1875	0.751	0	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		
		16	0.751	0.1875	0.751	0	0, $\cdots$, 15		
256	4861	2	0.0637	0.0637	0.0637	8	5, 7	78.61 %	77.83 %
		3	0.1875	0.0649	0.1875	0	4, 5, 6		
		4	0.1875	0.0649	0.1875	0	4, 5, 6, 7		
		5	0.5017	0.0649	0.5017	0	4, 5, 6, 7, 13		
		6	0.5017	0.0649	0.5017	0	4, 5, 6, 7, 13, 15		
		7	0.5022	0.0649	0.5022	0	4, 5, 6, 7, 12, 13, 14		
		8	0.5022	0.0649	0.5022	0	4, 5, 6, 7, 12, 13, 14, 15		
		10	0.7498	0.105	0.7498	0	4, 5, 6, 7, 10, 11, 12, 13, 14, 15		
		14	0.7502	0.1099	0.7502	0	0, 1, 2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15		
		16	0.7502	0.1108	0.7502	0	0, $\cdots$, 15		

Table 21: Selection of LS codes, with family size $K = 16$ and length of the ZCZ $Z_C = L_0 - 1$, depending on the number μ of simultaneous users. The codes have been generated from 4-CSS with length L_0 as indicated in [ZYH05].

A.4 T-ZCZ codes proposed in [ZLH05]

M = 4

L	$Z_C = Z_L$	μ	θ	θ_{AC}	θ_{CC}	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
16	4	2	0.375	0.25	0	0, 2	57.14 %	0 %
		3	0.375	0.375	0.375	0, 1, 2		85.71 %
		4	0.375	0.375	0.375	0, 1, 2, 3		85.71 %
32	8	2	0.1875	0.1875	0	0, 2	57.33 %	0 %
		3	0.375	0.1875	0.375	0, 1, 2		80 %
		4	0.375	0.1875	0.375	0, 1, 2, 3		80 %
64	16	2	0.2188	0.2188	0	0, 2	51.61 %	0 %
		3	0.2188	0.2188	0.2188	0, 1, 2		77.42 %
		4	0.2188	0.2188	0.2188	0, 1, 2, 3		77.42 %
128	32	2	0.2031	0.2031	0	0, 2	50.79 %	0 %
		3	0.2188	0.2031	0.2188	0, 1, 2		76.19 %
		4	0.2188	0.2031	0.2188	0, 1, 2, 3		76.19 %
256	64	2	0.1484	0.1484	0	0, 2	50.39 %	0 %
		3	0.2031	0.1484	0.2031	0, 1, 2		75.59 %
		4	0.2031	0.1484	0.2031	0, 1, 2, 3		75.59 %
512	128	2	0.1367	0.1367	0	0, 2	50.20 %	0 %
		3	0.1484	0.1367	0.1484	0, 1, 2		75.29 %
		4	0.1484	0.1367	0.1484	0, 1, 2, 3		75.29 %
1024	256	2	0.0801	0.0801	0	0, 2	50.10 %	0 %
		3	0.1367	0.0801	0.1367	0, 1, 2		75.15 %
		4	0.1367	0.0801	0.1367	0, 1, 2, 3		75.15 %

Table 22: Selection of T-ZCZ codes generated with construction 2, with family size $M = 4$, depending on the number μ of simultaneous users.

M = 8

L	$Z_C = Z_L$	μ	θ	θ_{AC}	θ_{CC}	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
16	2	2	0.5	0.5	0	0, 4	63.64 %	0 %
		3	0.5	0.5	0.25	0, 3, 4		54.55 %
		4	0.5	0.5	0.25	0, 3, 4, 7		54.55 %
		5	0.5	0.5	0.5	0, 1, 2, 3, 4		81.82 %
		6	0.5	0.5	0.5	0, 1, 2, 3, 4, 5		81.82 %
		7	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6		81.82 %
		8	0.5	0.5	0.5	0, 1, $\cdots$, 7		81.82 %
		2	0.3125	0.3125	0	0, 4		0 %
		3	0.3125	0.3125	0.25	0, 3, 4		60.87 %

Continuation of **Table** 23

L	$Z_C = Z_L$	μ	θ	θ_{AC}	θ_{CC}	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
32	4	4	0.3125	0.3125	0.25	0, 3, 4, 7	43.38 %	60.87 %
		5	0.5	0.3125	0.5	0, 1, 2, 3, 4		60.87 %
		6	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5		60.87 %
		7	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6		60.87 %
		8	0.5	0.3125	0.5	0, 1, $\cdots$, 7		60.87 %
64	8	2	0.3438	0.3438	0	0, 4	38.30 %	0 %
		3	0.4063	0.4063	0.2188	0, 3, 4		51.06 %
		4	0.4063	0.4063	0.2188	0, 3, 4, 7		51.06 %
		5	0.5	0.4063	0.5	0, 1, 2, 3, 4		51.06 %
		6	0.5	0.4063	0.5	0, 1, 2, 3, 4, 5		51.06 %
		7	0.5	0.4063	0.5	0, 1, 2, 3, 4, 5, 6		51.06 %
		8	0.5	0.4063	0.5	0, 1, $\cdots$, 7		51.06 %
128	16	2	0.25	0.25	0	0, 4	35.79 %	0 %
		3	0.25	0.25	0.2188	0, 3, 4		50.53 %
		4	0.25	0.25	0.2188	0, 3, 4, 7		50.53 %
		5	0.5	0.25	0.5	0, 1, 2, 3, 4		50.53 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.53 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.53 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.53 %
256	32	2	0.25	0.25	0	0, 4	34.55 %	0 %
		3	0.25	0.25	0.1406	0, 3, 4		50.26 %
		4	0.25	0.25	0.1406	0, 3, 4, 7		50.26 %
		5	0.5	0.25	0.5	0, 1, 2, 3, 4		50.26 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.26 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.26 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.26 %
512	64	2	0.25	0.25	0	0, 4	33.94 %	0 %
		3	0.25	0.25	0.1328	0, 3, 4		50.13 %
		4	0.25	0.25	0.1328	0, 3, 4, 7		50.13 %
		5	0.5	0.25	0.5	0, 1, 2, 3, 4		50.13 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.13 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.13 %
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.13 %
1024	128	2	0.25	0.25	0	0, 4	33.64 %	0 %
		3	0.25	0.25	0.1211	0, 3, 4		50.13 %
		4	0.25	0.25	0.1211	0, 3, 4, 7		50.13 %
		5	0.5	0.25	0.5	0, 1, 2, 3, 4		50.13 %
		6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5		50.13 %
		7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6		50.13 %

Continuation of **Table** 23

L	$Z_C = Z_L$	μ	θ	θ_{AC}	θ_{CC}	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
		8	0.5	0.25	0.5	0, 1, $\cdots$, 7		50.13 %

Table 23: Selection of T-ZCZ codes generated with construction 2, with family size $M = 8$, depending on the number μ of simultaneous users.

M = 16

L	$Z_C = Z_L$	μ	θ	θ_{AC}	θ_{CC}	codes	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
64	4	2	0.5	0.5	0	0, 8	56.36 %	0 %
		3	0.5	0.5	0.125	0, 7, 8		76.36 %
		4	0.5	0.5	0.125	0, 7, 8, 15		76.36 %
		5	0.5	0.5	0.25	0, 3, 5, 6, 8		76.36 %
		6	0.5	0.5	0.25	0, 3, 5, 6, 8, 11		76.36 %
		7	0.5	0.5	0.25	0, 3, 5, 6, 7, 11, 13		76.36 %
		8	0.5	0.5	0.25	0, 3, 5, 6, 8, 11, 13, 14		76.36 %
		10	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		76.36 %
		14	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		76.36 %
		16	0.5	0.5	0.5	0, 1, $\cdots$, 15		76.36 %
128	8	2	0.3125	0.3125	0	2, 10	37.84 %	0 %
		3	0.3125	0.3125	0.125	2, 5, 10		54.05 %
		4	0.3125	0.3125	0.125	2, 5, 10, 13		54.05 %
		5	0.3281	0.3281	0.25	0, 3, 5, 6, 8		52.25%
		6	0.3281	0.3281	0.25	0, 3, 5, 6, 8, 11		52.25 %
		7	0.3281	0.3281	0.25	0, 3, 5, 6, 7, 11, 13		52.25 %
		8	0.3281	0.3281	0.25	0, 3, 5, 6, 8, 11, 13, 14		52.25 %
		10	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		54.05 %
		14	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		54.05 %
		16	0.5	0.5	0.5	0, 1, $\cdots$, 15		54.05 %
256	16	2	0.3359	0.3359	0	0, 8	33.18 %	0 %
		3	0.3984	0.3984	0.1172	0, 7, 8		43.05 %
		4	0.3984	0.3984	0.1172	0, 7, 8, 15		43.05 %
		5	0.3984	0.3984	0.25	0, 5, 6, 8, 13		43.05%
		6	0.3984	0.3984	0.25	0, 5, 6, 8, 13, 14		43.05 %
		7	0.4141	0.4141	0.25	0, 3, 5, 6, 7, 11, 13		43.05 %
		8	0.4141	0.4141	0.25	0, 3, 5, 6, 8, 11, 13, 14		43.05 %
		10	0.5	0.4141	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		43.05 %
		14	0.5	0.4141	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		43.05 %
		16	0.5	0.4141	0.5	0, 1, $\cdots$, 15		43.05 %
		2	0.3125	0.3125	0	0, 8		0 %

Continuation of **Table** 24

L	$Z_C = Z_L$	μ	θ	θ_{AC}	θ_{CC}	codes	$\%\frac{Int.\ IZ_{ACF}}{IZ}$	$\%\frac{Int.\ IZ_{CCF}}{IZ}$
512	32	3	0.3125	0.3125	0.3125	0, 7, 8	30.87 %	42.95 %
		4	0.3125	0.3125	0.3125	0, 7, 8, 15		42.95 %
		5	0.3125	0.3125	0.3125	0, 3, 5, 6, 8		42.95%
		6	0.3125	0.3125	0.3125	0, 3, 5, 6, 8, 11		42.95 %
		7	0.3125	0.3125	0.3125	0, 3, 5, 6, 7, 11, 13		42.95 %
		8	0.3125	0.3125	0.3125	0, 3, 5, 6, 8, 11, 13, 14		42.95 %
		10	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		42.95 %
		14	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		42.95 %
		16	0.5	0.3125	0.5	0, 1, $\cdots$, 15		42.95 %
1024	64	2	0.3125	0.3125	0	0, 8	29.72 %	0 %
		3	0.3125	0.3125	0.3125	0, 7, 8		42.91 %
		4	0.3125	0.3125	0.3125	0, 7, 8, 15		42.91 %
		5	0.3125	0.3125	0.3125	0, 1, 6, 7, 8		42.91%
		6	0.3125	0.3125	0.3125	0, 1, 6, 7, 8, 9		42.91 %
		7	0.3125	0.3125	0.3125	0, 1, 6, 7, 8, 9, 14		42.91 %
		8	0.3125	0.3125	0.3125	0, 1, 6, 7, 8, 9, 14, 15		42.91 %
		10	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9		42.91 %
		14	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13		42.91 %
		16	0.5	0.3125	0.5	0, 1, $\cdots$, 15		42.91 %

Table 24: Selection of T-ZCZ codes generated with construction 2, with family size $M = 16$, depending on the number μ of simultaneous users.

A.5 T-ZCZ codes proposed in this thesis [Pérez et al. 2008b]

a) Method 1

M = 4

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
16	2	0.25	0.25	0	0, 3	6	6	66.67 %	0 %
	3	0.5	0.25	0.5	0, 1, 2				100 %
	4	0.5	0.25	0.5	0, 1, 2, 3				100 %
64	2	0.25	0.25	0	0, 3	38	6	31.58 %	0 %
	3	0.25	0.25	0.25	0, 1, 2				42.11 %
	4	0.25	0.25	0.25	0, 1, 2, 3				42.11 %
256	2	0.1875	0.1875	0	0, 2	102	102	23.53 %	0 %
	3	0.375	0.1875	0.375	0, 1, 2				31.37 %
	4	0.375	0.1875	0.375	0, 1, 2, 3				31.37 %

Continuation of **Table** 25

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CCF}}{IZ}$
1024	2	0.125	0.125	0	0, 2	614	102	11.73 %	0 %
	3	0.1875	0.125	0.1875	0, 1, 2				15.64 %
	4	0.1875	0.125	0.1875	0, 1, 2, 3				15.64 %
4096	2	0.1094	0.1094	0	0, 2	1638	1638	8.79 %	0 %
	3	0.2188	0.1094	0.2188	0, 1, 2				11.72 %
	4	0.2188	0.1094	0.2188	0, 1, 2, 3				11.72 %
16384	2	0.0625	0.0625	0	0, 2	9830	1638	4.39 %	0 %
	3	0.1094	0.0625	0.1094	0, 1, 2				5.86 %
	4	0.1094	0.0625	0.1094	0, 1, 2, 3				5.86 %

Table 25: Selection of T-ZCZ$_1$ codes, with family size $M = 4$ and length L, depending on the number μ of simultaneous users.

M = 8

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
64	2	0.25	0.25	0	0, 4	12	12	30.77 %	0 %
	3	0.25	0.25	0.25	0, 3, 4				35.90 %
	4	0.25	0.25	0.25	0, 3, 4, 7				38.46 %
	5	0.5	0.25	0.5	0, 1, 2, 3, 4				38.46 %
	6	0.5	0.25	0.5	0, 1, 2, 3, 4, 5				38.46 %
	7	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6				38.46 %
	8	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7				38.46 %
512	2	0.25	0.25	0	1, 5	268	12	26.84 %	0 %
	3	0.3125	0.3125	0.1406	1, 4, 7				38.10 %
	4	0.3125	0.3125	0.1406	0, 3, 4, 7				38.10 %
	5	0.3125	0.3125	0.25	0, 1, 2, 4, 5				38.10 %
	6	0.3125	0.3125	0.25	0, 1, 2, 4, 5, 6				38.10 %
	7	0.3125	0.3125	0.3125	0, 1, 2, 3, 5, 6, 7				38.10 %
	8	0.3125	0.3125	0.3125	0, 1, 2, 3, 4, 5, 6, 7				38.10 %
4096	2	0.1875	0.1875	0	0, 4	780	780	9.94 %	0 %
	3	0.1875	0.18735	0.1523	0, 3, 4				13.89 %
	4	0.1875	0.1875	0.1523	0, 3, 4, 7				13.89 %
	5	0.375	0.1875	0.375	0, 1, 2, 3, 4				13.89 %
	6	0.375	0.1875	0.375	0, 1, 2, 4, 5, 6				13.89 %
	7	0.375	0.1875	0.375	0, 1, 2, 3, 4, 5, 6				13.89 %
	8	0.375	0.1875	0.375	0, 1, 2, 3, 4, 5, 6, 7				13.89 %
32768	2	0.125	0.125	0	1, 5	17164	780	10.87 %	0 %
	3	0.1406	0.1406	0.0732	0, 3, 5				16.62 %
	4	0.1406	0.1406	0.0732	0, 1, 4, 7				16.62 %
	5	0.1875	0.1406	0.1875	0, 1, 4, 5, 7				16.62 %

Continuation of **Table** 26

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
	6	0.1875	0.1406	0.1875	0, 1, 4, 5, 6, 7				16.62 %
	7	0.1875	0.1406	0.1875	0, 1, 2, 3, 4, 5, 6				16.62 %
	8	0.1875	0.1406	0.1875	0, 1, 2, 3, 4, 5, 6, 7				16.62 %

Table 26: Selection of T-ZCZ$_1$ codes, with family size $M = 8$ and length L, depending on the number μ of simultaneous users.

M = 16

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
256	2	0.25	0.25	0	1, 8	24	24	26.09 %	0 %
	3	0.25	0.25	0.125	0, 7, 8				28.99 %
	4	0.25	0.25	0.125	0, 7, 8, 15				28.99 %
	5	0.25	0.25	0.25	0, 3, 5, 6, 8				33.82 %
	6	0.25	0.25	0.25	0, 3, 5, 6, 8, 11				33.82 %
	7	0.25	0.25	0.25	0, 3, 5, 6, 8, 11, 13				33.82 %
	8	0.25	0.25	0.25	0, 3, 5, 6, 8, 11, 13, 14				33.82 %
	10	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				33.82 %
	14	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				33.82 %
	16	0.5	0.25	0.5	0, 1, 2, $\cdots$, 15				33.82 %
4096	2	0.25	0.25	0	3, 11	2072	24	27.11 %	0 %
	3	0.2969	0.25	0.2969	3, 7, 11				36.52 %
	4	0.2969	0.25	0.2969	3, 7, 11, 15				36.52 %
	5	0.3125	0.3125	0.1523	1, 2, 7, 9, 10				40.42 %
	6	0.3125	0.3125	0.1523	1, 2, 7, 9, 10, 15				40.42 %
	7	0.3281	0.3281	0.1523	0, 3, 5, 6, 8, 11, 14				40.42 %
	8	0.3281	0.3281	0.1523	0, 3, 5, 6, 9, 10, 12, 15				40.42 %
	10	0.3281	0.3281	0.25	0, 1, 2, 4, 5, 8, 9, 10, 12, 13				41.62 %
	14	0.3281	0.3281	0.3125	0, 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 15				41.62 %
	16	0.3281	0.3281	0.3281	0, 1, 2, $\cdots$, 15				41.62 %

Table 27: Selection of T-ZCZ$_1$ codes, with family size $M = 16$ and length L, depending on the number μ of simultaneous users.

b) Method 2

M = 4

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
8	2	0.25	0.25	0	0, 1	2	2	66.67 %	0 %
	3	0.5	0.25	0.5	0, 1, 2				100 %
	4	0.5	0.25	0.5	0, 1, 2, 3				100 %
32	2	0.375	0.375	0	0, 2	9	9	30.77 %	0 %
	3	0.375	0.375	0.375	0, 1, 2				46.15 %
	4	0.375	0.375	0.375	0, 1, 2, 3				46.15 %
128	2	0.1875	0.1875	0	0, 1	38	38	23.53 %	0 %
	3	0.375	0.1875	0.375	0, 1, 2				31.37 %
	4	0.375	0.1875	0.375	0, 1, 2, 3				31.37 %
512	2	0.2188	0.2188	0	0, 2	153	153	11.71 %	0 %
	3	0.2188	0.2188	0.2188	0, 1, 2				17.56 %
	4	0.2188	0.2188	0.2188	0, 1, 2, 3				17.56 %
2048	2	0.1094	0.1094	0	0, 3	614	614	8.79 %	0 %
	3	0.2188	0.1094	0.2188	0, 1, 2				11.72 %
	4	0.2188	0.1094	0.2188	0, 1, 2, 3				11.72 %
8192	2	0.1484	0.1484	0	0, 3	2457	2457	4.39 %	0 %
	3	0.1484	0.1484	0.1484	0, 1, 2				6.59 %
	4	0.1484	0.1484	0.1484	0, 1, 2, 3				6.59 %

Table 28: Selection of T-ZCZ$_{2'}$ codes, with family size $M = 4$ and length L, depending on the number μ of simultaneous users.

M = 8

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
32	2	0.3125	0.3125	0	0, 1	4	4	43.38 %	0 %
	3	0.3125	0.3125	0.25	0, 1, 6				60.87 %
	4	0.3125	0.3125	0.25	0, 1, 6, 7				60.87 %
	5	0.5	0.3125	0.5	0, 1, 2, 3, 4				60.87 %
	6	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5				60.87 %
	7	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6				60.87 %
	8	0.5	0.3125	0.5	0, 1, 2, 3, 4, 5, 6, 7				60.87 %
256	2	0.125	0.125	0	0, 4	33	33	22.22 %	0 %
	3	0.25	0.25	0.25	0, 3, 4				29.63 %
	4	0.25	0.25	0.25	0, 3, 4, 7				29.63 %
	5	0.375	0.375	0.375	0, 1, 2, 3, 4				29.63 %
	6	0.375	0.375	0.375	0, 1, 2, 3, 4, 5				29.63 %
	7	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6				29.63 %
	8	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6, 7				29.63 %

Continuation of **Table** 29

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
2048	2	0.1797	0.1797	0	0, 5	268	268	15.88 %	0 %
	3	0.1953	0.1953	0.1523	0, 1, 7				23.30 %
	4	0.1953	0.1953	0.1523	0, 1, 6, 7				23.30 %
	5	0.375	0.1953	0.375	0, 1, 2, 3, 4				23.30 %
	6	0.375	0.1953	0.375	0, 1, 2, 3, 4, 5				23.30 %
	7	0.375	0.1953	0.375	0, 1, 2, 3, 4, 5, 6				23.30 %
	8	0.375	0.1953	0.375	0, 1, 2, 3, 4, 5, 6, 7				23.30 %
16384	2	0.2188	0.2188	0	0, 5	2145	2145	7.97 %	0 %
	3	0.2188	0.2188	0.1162	0, 3, 4				11.61 %
	4	0.2188	0.2188	0.1162	0, 3, 4, 7				11.61 %
	5	0.2188	0.2188	0.2188	0, 1, 2, 3, 4				12.97 %
	6	0.2188	0.2188	0.2188	0, 1, 2, 3, 4, 5				12.97 %
	7	0.2188	0.2188	0.2188	0, 1, 2, 3, 4, 5, 6				12.97 %
	8	0.2188	0.2188	0.2188	0, 1, 2, 3, 4, 5, 6, 7				12.97 %

Table 29: Selection of T-ZCZ$_{2'}$ codes, with family size $M = 8$ and length L, depending on the number μ of simultaneous users.

M = 16

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
128	2	0.3125	0.3125	0	2, 3	8	8	37.84 %	0 %
	3	0.3125	0.3125	0.125	2, 3, 12				54.05 %
	4	0.3125	0.3125	0.125	2, 3, 12, 13				54.05 %
	5	0.3281	0.3281	0.25	0, 1, 6, 7, 10				52.25 %
	6	0.3281	0.3281	0.25	0, 1, 6, 7, 10, 11				52.25 %
	7	0.3281	0.3281	0.25	0, 1, 6, 7, 10, 11, 12				52.25 %
	8	0.3281	0.3281	0.25	0, 1, 6, 7, 10, 11, 12, 13				52.25 %
	10	0.5	0.3281	0.5	2, 3, 4, 5, 6, 7, 8, 9, 10, 11				54.05 %
	14	0.5	0.3281	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				54.05 %
	16	0.5	0.3281	0.5	0, 1, 2, $\cdots$, 15				54.05 %
2048	2	0.375	0.125	0	0, 8	129	129	20.12 %	0 %
	3	0.375	0.375	0.0762	0, 7, 8				29.96 %
	4	0.375	0.375	0.0762	0, 7, 8, 15				29.96 %
	5	0.375	0.375	0.2031	0, 3, 5, 6, 8				29.96 %
	6	0.375	0.375	0.2031	0, 3, 5, 6, 8, 11				29.96 %
	7	0.375	0.375	0.2031	0, 3, 5, 6, 8, 11, 13				29.96 %
	8	0.375	0.375	0.2031	0, 3, 5, 6, 8, 11, 13, 14				29.96 %

Continuation of **Table** 30

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
	10	0.375	0.25	0.25	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				29.96 %
	14	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				29.96 %
	16	0.375	0.375	0.375	0, 1, 2, $\cdots$, 15				29.96 %

Table 30: Selection of T-ZCZ$_{2'}$ codes, with family size $M = 16$ and length L, depending on the number μ of simultaneous users.

c) Method 3

M = 4

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
	2	0.5	0.5	0	0, 2				0 %
4	3	0.5	0.5	0.5	0, 1, 2	1	1	100 %	100 %
	4	0.5	0.5	0.5	0, 1, 2, 3				100 %
	2	0.25	0.25	0	0, 3				0 %
16	3	0.5	0.25	0.5	0, 1, 2	6	6	66.67 %	100 %
	4	0.5	0.25	0.5	0, 1, 2, 3				100 %
	2	0.375	0.375	0	0, 2				0 %
64	3	0.375	0.375	0.375	0, 1, 2	25	25	30.77 %	46.15 %
	4	0.375	0.375	0.375	0, 1, 2, 3				46.15 %
	2	0.1875	0.1875	0	0, 1				0 %
256	3	0.375	0.1875	0.375	0, 1, 2	102	102	23.53 %	31.37 %
	4	0.375	0.1875	0.375	0, 1, 2, 3				31.37 %
	2	0.2188	0.2188	0	0, 2				0 %
1024	3	0.2188	0.2188	0.2188	0, 1, 2	409	409	11.71 %	17.56 %
	4	0.2188	0.2188	0.2188	0, 1, 2, 3				17.56 %
	2	0.1094	0.1094	0	0, 3				0 %
4096	3	0.2188	0.1094	0.2188	0, 1, 2	1638	1638	8.79 %	11.72 %
	4	0.2188	0.1094	0.2188	0, 1, 2, 3				11.72 %

Table 31: Selection of T-ZCZ$_{3'}$ codes, with family size $M = 4$ and length L, depending on the number μ of simultaneous users.

M = 8

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
	2	0.5	0.5	0	0, 2				0 %
	3	0.5	0.5	0.25	0, 2, 5				54.55 %
	4	0.5	0.5	0.25	0, 2, 5, 7				54.55 %
16	5	0.5	0.5	0.5	0, 1, 2, 3, 4	2	2	63.64 %	81.82 %

Continuation of **Table** 32

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
	6	0.5	0.5	0.5	0, 1, 2, 3, 4, 5				81.82 %
	7	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6				81.82 %
	8	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7				81.82 %
	2	0.375	0.375	0	0, 3				0 %
	3	0.375	0.375	0.1875	0, 2, 5				39.56 %
	4	0.375	0.375	0.1875	0, 2, 5, 7				39.56 %
128	5	0.5	0.375	0.5	0, 1, 2, 3, 4	18	18	24.18 %	39.56 %
	6	0.5	0.375	0.5	0, 1, 2, 3, 4, 5				39.56 %
	7	0.5	0.375	0.5	0, 1, 2, 3, 4, 5, 6				39.56 %
	8	0.5	0.375	0.5	0, 1, 2, 3, 4, 5, 6, 7				39.56 %
	2	0.375	0.375	0	0, 3				0 %
	3	0.375	0.375	0.1406	0, 2, 5				26.67 %
	4	0.375	0.375	0.1406	0, 2, 5, 7				26.67 %
1024	5	0.375	0.375	0.375	0, 1, 2, 3, 4	146	146	15.32 %	26.67 %
	6	0.375	0.375	0.375	0, 1, 2, 3, 4, 5				26.67 %
	7	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6				26.67 %
	8	0.375	0.375	0.375	0, 1, 2, 3, 4, 5, 6, 7				26.67 %
	2	0.2188	0.2188	0	3, 4				0 %
	3	0.2188	0.2188	0.082	1, 3, 4				19.69 %
	4	0.2188	0.2188	0.082	1, 2, 4, 7				19.69 %
8192	5	0.375	0.2188	0.375	0, 1, 2, 3, 4	1170	1170	10.46 %	19.69 %
	6	0.375	0.2188	0.375	0, 1, 2, 3, 4, 5				19.69 %
	7	0.375	0.2188	0.375	0, 1, 2, 3, 4, 5, 6				19.69 %
	8	0.375	0.2188	0.375	0, 1, 2, 3, 4, 5, 6, 7				19.69 %

Table 32: Selection of T-ZCZ$_{3'}$ codes, with family size $M = 8$ and length L, depending on the number μ of simultaneous users.

M = 16

L	μ	θ	θ_{AC}	θ_{CC}	codes	Z_L	Z_C	$\%\frac{Int.\ IZ_{AC}}{IZ}$	$\%\frac{Int.\ IZ_{CC}}{IZ}$
64	2	0.5	0.5	0	0, 2	4	4	56.36 %	0 %
	3	0.5	0.5	0.125	0, 2, 13				50.91 %
	4	0.5	0.5	0.125	0, 2, 13, 15				50.91 %
	5	0.5	0.5	0.125	0, 2, 5, 7, 9				76.36 %
	6	0.5	0.5	0.25	0, 2, 5, 7, 9, 11				76.36 %
	7	0.5	0.5	0.25	0, 2, 5, 7, 9, 11, 12				76.36 %
	8	0.5	0.5	0.25	0, 2, 5, 7, 9, 11, 12, 14				76.36 %
	10	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				76.36 %
	14	0.5	0.5	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				76.36 %
	16	0.5	0.5	0.5	0, 1, 2, $\cdots$, 15				76.36 %
1024	2	0.375	0.375	0	0, 4	66	66	21.10 %	0 %
	3	0.375	0.375	0.1016	0, 4, 11				37.71 %
	4	0.375	0.375	0.1016	0, 4, 11, 15				37.71 %
	5	0.375	0.375	0.2031	0, 3, 4, 7, 9				33.67 %
	6	0.375	0.375	0.2031	0, 3, 4, 7, 9, 10				33.67 %
	7	0.375	0.375	0.2031	0, 3, 4, 7, 9, 10, 13				33.67 %
	8	0.375	0.375	0.2031	0, 3, 4, 7, 9, 10, 13, 14				33.67 %
	10	0.5	0.25	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9				37.71 %
	14	0.5	0.375	0.5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13				37.71 %
	16	0.5	0.375	0.5	0, 1, 2, $\cdots$, 15				37.71 %

Table 33: Selection of T-ZCZ$_{3'}$ codes, with family size $M = 16$ and length L, depending on the number μ of simultaneous users.

References

[ÁGG+07] F. J. Álvarez, H. M. González, C. J. García, M. Macías and R. Gallardo. Using gas to obtain an optimal set of codes for an ultrasonic local positioning system. In *11th International Conf. on Computer Aided System Theory (EUROCAST'07)*, pages 845–852, Las Palmas de Gran Canaria, (Spain), February 2007.

[ÁHU+06] F. J. Álvarez, A. Hernández, J. Ureña, J. J. García, A. Jiménez, M. C. Pérez and P. S. Teresa. Detection module in a complementary set of sequences-based pulse compression system. In *Proc. of IEEE International Conference on Field Programmable Logic and Applications FPL'06*, pages 1–6, Madrid (Spain), August 2006.

[APKC92] K. Audenaert, H. Peremans, Y. Kawahara and Van J. Campenhout. Accurate ranging of multiple objects using ultrasonic sensors. In *Proc. of International Conference on Robotics and Automation*, pages 1733–1738, Niza (Francia), May 1992.

[ÁUM+04] F. J. Álvarez, J. Ureña, M. Mazo, A. Hernández, J. J. García and J. A. Jiménez. Efficient generator and pulse compressor for complementary sets of four sequences. *IEEE Electronics Letters*, 40(11):703–704, May 2004.

[ÁUM+06] F. J. Álvarez, J. Ureña, M. Mazo, A. Hernández, J. J. García and C. De Marziani. High reliability outdoor sonar prototype based on efficient signal coding. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 53:1862–1872, October 2006.

[Bar53] R. H. Barker. Group synchronizing of binary digital sequences. In *Communication Theory (Proc. of the Symposium on the application of Communication Theory, Butterworth, London, septiembre 1952)*, pages 273–287, W. Jackon, ed., Academic Press, New York, 1953.

[Bud89] S. Z. Budisin. Fast PN sequence correlation by FWT. In *Proc. of IEEE Integration Research, Industry and Education in Energy and Communication Engineering, MELECON'89*, pages 513–515, Lisbon (Portugal), April 1989.

[Bud91] S. Z. Budisin. Efficient pulse compressor for Golay complementary sequences. *IEEE Electronics Letters*, 27(3):219–220, January 1991.

[CG96] N. Chang and S. W. Golomb. 7200- phase generalized Barker sequences. *IEEE Transactions on Information Theory*, 42(4):1236–1238, July 1996.

[Che07] H. Chen. *Next generation CDMA technologies*. John Wiley & sons, Ltd, West Sussex PO19 8SQ, England, 2007.

[CLY+06] H. Chen, J. Li, Y. Yang, X. Du and H. Liu. Challenges and futuristic perspective of CDMA technologies: OCC-CDMA/OS for 4G wireless networks. In *IEEE International Conference on Communications ICC'06*, volume 9, pages 3984–3989, Istanbul, (Turkey), June 2006.

[DMUH+05] C. De Marziani, J. Ureña, A. Hernández, M. Mazo and F. J. Álvarez. Simultaneous measurement of times-of-flight and communications in acoustic sensor network. In *Proc. of IEEE International Symposium on Intelligent Signal Processing*, University of Algarve, Faro (Portugal), September 2005.

[DMUH+06] C. De Marziani, J. Ureña, A. Hernández, M. Mazo, J. J García, J. M. Villadangos, M. C. Pérez, A. Ochoa and F. J. Álvarez. Inter-symbol interference reduction on macro-sequences generated from complementary sets of sequences. In *Proc. of IEEE 32nd Annual Conference on Industrial Electronics IECON'06*, pages 3367–3372, Paris (France), November 2006.

[DMUH+07] C. De Marziani, J. Ureña, M. Hernández, Mazo, F. J. Álvarez, J. J. García and P. Donato. Modular architecture for efficient generation and correlation of complementary set of sequences. *IEEE Trans. on Signal Processing*, 55(5):2323–2337, May 2007.

[DO99] H. Donelan and T. O'Farrell. A new method for generating sets of orthogonal sequences for a synchronous CDMA system. *IEEE Electronics Letters*, 35(8):1537–1538, September 1999.

[DUM+99] V. Díaz, J. Ureña, M. Mazo, J. J. García, E. Bueno and A. Hernández. Using Golay complementary sequences for multi-mode ultrasonic operation. In *Proc. of 7th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'99)*, pages 599–604, Barcelona (Spain), October 1999.

[Eli80] C. M. Elias. An ultrasonic pseudorandom signal-correlation system. *IEEE Transactions on Sonics and Ultrasonics*, SU-27(1):1–7, January 1980.

[EM78] C. M. Elias and T. J. Moran. A pseudorandom binary noise NDE ultrasonic correlation system. In *1978 Ultrasonics Symposium Proceedings*, pages 311–315, New Jersey (USA), September 1978.

[Fan04] P. Fan. Spreading sequence design and theoretical limits for quasisynchronous CDMA systems. *EURASIP Journal on Wireless Communications and Networking*, 2004(1):19–31, March 2004.

[FD96] P. Fan and M. Darnell. *Sequence Design for communications applications.* Research studies press LTD., 24 Belvedere Road, Taunton, Somerset, (England), 1996.

[FFT07] L. Feng, P. Fan and X. Tang. A general construction of OVSF codes with Zero Correlation Zone. *IEEE Signal Processing Letters*, 14(12):908–911, December 2007.

[FFTL08] L. Feng, P. Fan, X. Tang and K. Loo. Generalized pairwise Z-complementary codes. *IEEE Signal Processing Letters*, 15:377–379, January 2008.

[Fur82] E. S. Furgason. Optimal operation of ultrasonic correlation systems. In *Proc. of Ultrasonic Symposium*, pages 821–825, San Diego (USA), October 1982.

[GN98] P. Graham and B. Nelson. FPGA-based sonar processing. In *Proc. of the 6th ACM/SIGDA International Symposium on Field Programmable Gate Array (FPGA'98)*, pages 201–208, San Jose (USA), February 1998.

[Gol61] M. J. Golay. Complementary series. *IRE Transactions on Information Theory*, IT-7:82–87, abril 1961.

[Gol67a] R. Gold. Optimal binary sequences for spread spectrum multiplexing. *IEEE Transactions on Information Theory*, IT-13:619–621, October 1967.

[Gol67b] S. W. Golomb. *Shift Register Sequences.* Holden-Day, Inc, San Francisco, 1967.

[HG88] G. Hayward and Y. Gorfu. A digital hardware correlation system for fast ultrasonic data acquisition in peak power limited applications. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 35(6):800–808, noviembre 1988.

[HH06] M. Hazas and A. Hooper. Broadband ultrasonic location systems for improved indoor positioning. *IEEE Transactions on Mobile Computing*, 5:536–547, May 2006.

[HHWW99] A. Harter, P. Hopper, A. Ward and P. Webster. The anatomy of context-aware application. In *Proc. of 5th Annual ACM/IEEE International Conference on Mobile Computing Networking (Mobicom 1999)*, pages 1–59, Seatle, Washington (USA), August 1999.

[HOB+00] G. Hueber, T. Ostermann, T. Bauernfeind, R. Raschhofer and R. Hagelauer. New approach of ultrasonic distance measurement technique in robots applications. In *Proc. of 5th International Conference on Signal Processing (ICSP'2000)*, pages 2066–2069, Beijing (China), August 2000.

[HUG+04] A. Hernández, J. Ureña, J. J. García, M. Mazo, D. Herranz, P. Dérutin and Serot J. Ultrasonic ranging sensor using simultaneous emissions from different transducers. *IEEE Trans. on Ultrasonics, Ferroelectrics and Frequency Control*, 51(12):1660–1670, December 2004.

[HUH+03] A. Hernández, J. Ureña, D. Hernanz, J. J. García, M. Mazo, J. P. Dérutin, J. Sérot and S. Palazuelos. Real-time implementation of an efficient golay correlator (EGC) applied to ultrasonic sensorial systems. *Microprocessor and Microsystems*, 27(8):397–406, September 2003.

[HUM+05] A. Hernández, J. Ureña, M. Mazo, A. Jiménez, J. J. García, A. Ochoa, J. M. Villadangos and J. A. Jiménez. A comparison of computing architectures for ultrasonic signal processing. In *Proc. of IEEE International Symposium on Intelligent Signal Processing*, University of Algarve, Faro (Portugal), September 2005.

[HUM+06] A. Hernández, J. Ureña, M. Mazo, J. J. García, A. Jiménez and F. J. Álvarez. Reduction of blind zone in ultrasonic transmitter/receiver transducers. *Sensors and Actuators*, 133(1):96–103, March 2006.

[HW03] M. Hazas and A. Ward. A high performance privacy-oriented location system. In *Proc. of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom 2003)*, pages 216–223, Dallas (USA), March 2003.

[Inc08a] Digilent Inc. Digilent nexys2 board reference manual, noviembre 2008.

[Inc08b] Measurement Specialist Inc. 40 khz omni-directional ultrasound transmitter, application specification documment, noviembre 2008.

[Inc08c] Mobile Robots Inc. Pioneer 3-dx: The general purpose robot base, noviembre 2008.

[JB98] K. W. Jörg and M. Berg. Sophisticated mobile robot sonar sensing with pseudo-random codes. *Robotics and Autonomous Systems*, 25(3–4):241–251, November 1998.

[Kas68] T. Kasami. Weight distribution formula for some class of cyclic codes. Technical Report R–285, Coordinated Science Lab. University of Illinois, April 1968.

[KK95] L. Kleeman and R. Kuc. Mobile robot sonar for target localisation and classification. *The international Journal of Robotics Research*, 14(4):295–318, August 1995.

[KMM07] V. A. Kumar, A. Mitra and P. S. R. Mahadeva. On the effectivity of different pseudo-noise and orthogonal sequences for speech encryption from correlation properties. *International Journal Of Information Technology*, 4(2):145–152, September 2007.

[Kri03] R. Kristiansen. On the aperiodic autocorrelation of binary sequences, 2003.

[Lah95] J. Lahtonen. On the odd and the aperiodic correlation properties of the Kasami sequences. *IEEE Transactions on Information Theory*, 41(5):1506–1508, September 1995.

[Lev99] V. I. Levenshtein. New lower bounds on aperiodic cross-correlation of binary codes. *IEEE Transactions on Information Theory*, 45(1):284–288, January 1999.

[LGZ07] Q. Li, J. Gao and X. Zhao. The application of the ZCZ sequence pairs set in QS-CDMA system. In 3^{rd} *International Workshop on Signal Design and Its Applications in Communications (IWSDA'07)*, pages 288–291, Chengdu, (China), September 2007.

[Li03] D. Li. The perspectives of Large Area Synchronous CDMA technology for the fourth-generation mobile radio. *IEEE Communications Magazine*, 41(3):114–119, March 2003.

[LWF+07] C. Leavens, R. Williams, F. Foster, P. Burns and M. Sherar. Golay pulse encoding for microbubble contrast imaging in ultrasound. *IEEE Trans. on Ultrasonics, Ferroelectrics and Frequency Control*, 54(10):2082–2090, October 2007.

[Mic08] Panasonic Industrial Components Audio Electret Condenser Microphone. Omnidirectional back electret condenser microphone cartridge, noviembre 2008.

[NSLT03] A. Nowicki, W. Secomski, J. Litniewski and I. Trots. On the application of signal compression using golay's codes sequences in ultrasound diagnostic. *Archives of acoustics*, 28:313–324, 2003.

[PAC93] H. Peremans, K. Audenaert and J. Van Campenhout. A high resolution sensor based on tri-aural perception. *IEEE Transactions on Robotics and Automation*, 9(1):36–48, febrero 1993.

[PG93] G. J. R. Povey and P. M. Grant. Simplified matched filter receiver designs for spread spectrum communications applications. *Electronics and Communications Engineering Journal*, 5(2):59–64, April 1993.

[PJG+07] J. C. Prieto, A. R. Jiménez, J. I. Guevara, J. L. Ealo, F. A. Seco, J. O. Roa and F. X. Ramos. Subcentimeter-accuracy localization through broadband acoustic transducers. In *IEEE International Symposium on Intelligent Signal Processing WISP'2007*, pages 929–934, Madrid, (Spain), October 2007.

[PMBT01] N. B. Priyantha, A. K. L. Miu, H. Balakrishnan and S. Teller. The Cricket compass for context-aware mobile applications. In *Proc. of 7th International Conference on Mobile Computing Networking (Mobicom 2000)*, pages 1–14, Roma (Italy), July 2001.

[Pop97] B. M. Popovic. Efficient despreaders for multi-code CDMA systems. In *Proc. of IEEE Universal Personal Communication Record*, volume 2, pages 516–520, San Diego (USA), October 1997.

[Pop99] B. M. Popovic. Efficient Golay correlator. *IEEE Electronics Letters*, 35(17):1427–1428, August 1999.

[PSK07] H. Piontek, M. Seyffer and J. Kaiser. Improving the accuracy of ultrasoung-based localisation systems. *Personal and Ubiquitous Computing*, 11(6):439–449, August 2007.

[RC08] A. Rathinakumar and A. K. Chaturvedi. Complete mutually orthogonal Golay complementary sets from Reed-Muller codes. *IEEE Transactions on Information Theory*, 54(3):1339–1346, March 2008.

[RM01] C. Randell and H. Muller. Low cost indoor positioning system. In *Proc. of the 3rd International Conference on Ubiquitous Computing*, pages 42–48, Atlanta (USA), September 2001.

[SBH01] S. Stańzak, H. Boche and M. Haardt. Are LAS-codes a miracle? In *Proc. of IEEE Global Telecommunications Conference (GLOBECOM'2001)*, volume 1, pages 589–593, San Antonio (USA), November 2001.

[SP80] D. V. Sarwate and M. B. Pursley. Crosscorrelation properties of pseudorandom and related sequences. *Proceedings of the IEEE*, 68:593–619, May 1980.

[Syl67] J. J. Sylvester. Thoughts on inverse orthogonal matrices, simultaneous sign successions, and tessellated pavements in two or more colours, with applications to Newton's rule, ornamental tile-work, and the theory of numbers. *Philosophical Magazine*, 34:461–475, 1867.

[TL72] C. C. Tseng and C. L. Liu. Complementary sets of sequences. *IEEE Transactions on Information Theory*, IT-18(5):644–652, September 1972.

[TZW08] Y. Tsai, G. Zhang and X. Wang. Polyphase codes for uplink OFDM-CDMA systems. *IEEE Transactions on Communications*, 56(3):435–444, March 2008.

[UHJ+07] J. Ureña, A. Hernández, A. Jiménez, J. M. Villadandos, M. Mazo, J. C. García, J. J. García, F. J. Álvarez, C. De Marziani, M. C. Pérez, J. A. Jiménez, A. R. Jiménez, and F. Seco. Advanced sensorial system for an acoustic LPS. *Microprocessor and Microsystems*, 31:393–401, September 2007.

[UMG+99] J. Ureña, M. Mazo, J. J. García, A. Hernández and E. Bueno. Correlation detector based on a FPGA for ultrasonic sensors. *Microprocessors and Microsystems*, 23(1):25–33, June 1999.

[Vit90] A. J. Viterbi. Very low rate convolutional codes for maximum theoretical performance of spread-spectrum multiple-access channels. *IEEE Journal on Selected Areas in Communications*, 8:641–649, May 1990.

[VUM+05] J. M. Villadangos, J. Ureña, M. Mazo, A. Hernández, F. Álvarez, J. J. García, C. Marziani and D. Alonso. Improvement of ultrasonic beacon-based local position system using multi-acces techniques. In *Proc. of IEEE International Symposium on Intelligent Signal Processing*, Universidad of Algarve, Faro (Portugal), September 2005.

[VUM+07] J. M. Villadangos, J. Ureña, M. Mazo, A. Hernández, C. Marziani, M. C. Pérez, F. J. Álvarez, J. J. García, A. Jiménez, and I. Gude. Ultrasonic Local Positioning System with large covered area. In *IEEE International Symposium on Intelligent Signal Processing WISP'2007*, pages 935–940, Madrid, (Spain), October 2007.

[WA07] S. Wang and A. Abdi. MIMO ISI channel estimation using uncorrelated Golay complementary sets of polyphase sequences. *IEEE Trans. on Vehicular Technology*, 56(5):3024–3039, September 2007.

[Wel74] L. R. Welch. Lower bounds on the maximum cross-correlation of signals. *IEEE Transactions on Information Theory*, 20(3):397–399, May 1974.

[WH06] H. Wei and L. Hanzo. On the uplink performance of LAS-CDMA. *IEEE Transactions on Wireless Communications*, 5(5):1187–1196, May 2006.

[WMRR72] R. C. Waag, J. B. Myklebust, W. L. Rhoads and Gramiak R. Intrumentation for noninvasive cardiac chamber flow rate measurement. In *Proc. of Ultrasonics Simposium*, pages 74–77, Boston, Massachussetts (USA), octubre 1972.

[Xil07] Xilinx. *Virtex-II Pro and Virtex-II Pro X FPGA User Guide*, noviembre 2007.

[Xil08] Xilinx. *Spartan-3E FPGA Family Complete Datasheet, product specification*, abril 2008.

[XY05] H. Xu and S. Yang. LAS-OFDMA for power line carrier communications in home networks. In *IEEE International Conference on Consumer Electronics ICCE'05*, pages 387–388, Las Vegas (USA), January 2005.

[ZLH05] C. Zhang, X. Lin and M. Hatori. Novel sequence pair and set with Three Zero Correlation windows. *IEICE Transactions on Communications*, E88-B(4):1517–1522, April 2005.

[ZYH05] C. Zhang, S. Yamada and M. Hatori. General method to construct LS codes by complete complementary sequences. *IEICE Trans. on Wireless Communications Technologies*, E88-B(8):3484–3487, August 2005.

International publications derived from this thesis

[Pérez et al. 2006a] M. C. Pérez, A. Hernández, J. Ureña, C. De Marziani and A. Jiménez. FPGA-based Implementation of a Correlator for Kasami Sequences. *Proc. 11th IEEE International Conference on Emerging Technologies and Factory Automation ETFA'06*, pages 1141–1144, Prague, Rep. Czech, September, 2006.

[Pérez et al. 2006b] M. C. Pérez, J. Ureña, A. Hernández, C. De Marziani, A. Ochoa and W. P. Marnane. FPGA Implementation of an Efficient Correlator for Complementary Sets of Sequences. *Proc. of International Conference on Field Programmable Logic and Applications FPL'06*, pages 697–700, Madrid, Spain, August, 2006.

[Pérez et al. 2007a] M. C. Pérez, J. Ureña, A. Hernández, W. P. Marnane, C. De Marziani and A. Jiménez. Hardware Implementation of an Efficient Correlator for Interleaved Complementary Sets of Sequences. *Journal of Universal Computer Science*, 13(3):388–406, March, 2007.

[Pérez et al. 2007b] M. C. Pérez, J. Ureña, A. Hernández, W. P. Marnane, A. Jiménez and F. J. Álvarez. Efficient Real-Time Correlator for LS Sequences. *Proc. of IEEE International Symposium on Industrial Electronics ISIE'07*, pages 1663–1668, Vigo, Spain, June, 2007.

[Pérez et al. 2007c] M. C. Pérez, J. Ureña, A. Hernández, C. De Marziani, A. Jiménez, J. M. Villadangos and F. J. Álvarez. Ultrasonic beacon-based Local Positioning System using Loosely Synchronous codes. *Proc. IEEE International Symposium on Intelligent Signal Processing WISP'2007*, pages 923–928, Madrid, Spain, October, 2007.

[Ureña et al. 2007a] J. Ureña, M. C. Pérez, A. Ochoa, A. Hernández, C. De Marziani, F. J. Álvarez, J. J. García, A. Jiménez, J. A. Jiménez. Separation of concurrent echoes depending on the emitting source using DS-CDMA. *Proc. of 19th International Congress on Acoustics ICA'07*, pages 1–6, Madrid, Spain, September, 2007.

[Ureña et al. 2007b] J. Ureña, A. Hernández, A. and Jiménez, J. M. Villadandos, M. and Mazo,J. C. García, J. J. and García, F. J. Álvarez, C. De Marziani, M. C. Pérez, J. A. Jiménez,A. R. Jiménez F. and Seco. Advanced sensorial system for an acoustic LPS. *Microprocessor and Microsystems*, 31(6):393–401, September, 2007.

[Pérez et al. 2008a] M. C. Pérez, J. Ureña, A. Hernández, F. J. Álvarez, A. Jiménez and C. De Marziani. Efficient correlator for LS codes generated from Orthogonal CSS. *IEEE Communications Letters*, 12(10):764–766, October, 2008.

[Pérez et al. 2008b] M. C. Pérez, J. Ureña, A. Hernández, A. Jiménez and C. De Marziani. Efficient Generation and Correlation of Sequence Pairs with Three Zero Correlation Zones. *Accepted for its publication on IEEE Transactions on Signal Processing*, October, 2008.

A copy of the text of each of the publications is attached.

FPGA-based Implementation of a Correlator for Kasami Sequences

M. C. Pérez, A. Hernández, J. Ureña, C. De Marziani, A. Jiménez
Dept. of Electronics. University of Alcalá
Ctra. Madrid-Barcelona Km 33.600, E-28871
Alcalá de Henares, Madrid, Spain
email: carmen@depeca.uah.es

Abstract

Kasami sequences have been successfully used in communications, navigation and related systems due to their low cross-correlation values, compared to those from other binary sequences. In this work, different alternatives for the hardware implementation of a correlator of Kasami sequences are presented: for short sequences a combinational design is proposed, whereas three sequential designs are suggested for longer Kasami sequences. These three sequential designs differ about the management of the memory: one stores the necessary data in slices of the FPGA; another uses external memory; and finally, the last one uses the internal RAM blocks in the FPGA.

1. Introduction

In Spread Spectrum (SS) communication systems, the energy of the transmitted signal is spread along a frequency range, which is much wider than the minimum required to transmit the information. This is possible thanks to a spreading sequence which is independent of the data information. At the receiver, the original signal is recovered by using a synchronized copy of the spreading sequence. In Spread-Spectrum Code Division Multiple Access (SS-CDMA) all users share the same bandwidth, but every transmitter has a different spreading code [1]. The effectiveness of SS-CDMA systems strongly depends on the used SS codes. Therefore, those codes should provide very low cross-correlations (CC) values to avoid mutual interferences among the different users. Besides, the codes should have a high Auto-Correlation (AC) function, in order to distinguish them from noise.

Examples of SS codes are pseudo-random (PR) sequences (m-sequences, Gold, Kasami) [2]. Maximal length sequences (m-sequences) are easily generated from linear feedback shift registers (LFSR) and, in case of periodic emissions, they have a bi-level AC, -1 and N, where N is the length of the sequence. On the other hand, the CC of m-sequences depends on the specific sequences used to operate in the multi-user environment. There exist sets of m-sequences, called preferred pairs, which yield low cross-correlation values. Nevertheless, the numbers of preferred pairs of the same length are not enough for many applications. Gold and Kasami sequences yield a larger set of sequences than m-sequences with the same length, although worsening their AC characteristics. Gold sequences provide more number of different codes, and therefore, support a higher amount of users; but their CC are similar to the m-sequences CC values. Kasami sequences provide less codes than Gold sequences, but they have lower CC peaks and a better performance in a non-synchronized environment [3].

This work is included in a more general one, consisting in a Local Positioning System (LPS) for indoor positioning of mobile robots, based on the transmission of ultrasonic signals. The LPS consists of several ultrasonic beacons located in the environment, all of them emitting a different code in a simultaneous and continuous way. The mobile robot has an ultrasonic receiver to determine its absolute position by hyperbolic triangulation of the distances obtained from the measured Times-of-Arrival (TOAs) [4]. Thereby, the receiver detects the TOAs coming from the beacons by carrying out the correlations with the emitted codes. Due to the system features, where several beacons are simultaneously emitting, it is desirable that the emitted codes have very low CC values to provide a suitable discrimination among beacons. Kasami sequences have been employed for that purpose.

This work describes several hardware implementations of a correlator for the detection of Kasami sequences used to encode an ultrasonic emission. This paper is organized as follows. Section 2 reviews the properties of Kasami sequences. In Section 3 four different hardware implementations have been proposed, for the correlation of Kasami sequences. Section 4 shows a comparison of the required resources and operating frequencies for every correlator implementation. Finally, conclusions are discussed in Section 5.

1-4244-0681-1/06/$20.00 ©2006 IEEE

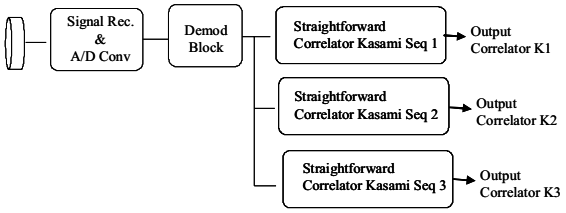


Figure 1. Block diagram of the reception stage.

2. Kasami Sequences

Kasami sequences were first discussed in [5], and were developed to improve the CC of other PR codes. The small set of Kasami sequences has been used in this work, due to have better CC features than the large set [3].

These sequences are generated from m-sequences. The procedure is explained in [2] and [3]: firstly, an m-sequence *m1* of length $N_c=2^n-1$ is generated, by using an LFSR with *n* cells, where *n* is an even integer. Next, a second m-sequence *m2* is generated by decimating *m1* by a factor $q=2^{n/2}+1$. It can be verified that the resulting sequence is periodic with period $T_{m2}=2^{n/2}-1$. A Kasami sequence is obtained by taking 2^n-1 bits of the sequences *m1* and *m2* and adding them in module-2. The set *K* of Kasami sequences consists of *m1*, and of the module-2 additions of *m1* and *m2*, and of all the $2^{n/2}-2$ cyclic shifts of *m2* (1).

$$K = \left\{ m1,\ m1 \oplus m2,\ m1 \oplus Dm2,\ldots,\ m1 \oplus D^{2^{n/2}-2} m2 \right\} \quad (1)$$

Where D^i is an operator which cyclically shifts *m2* to the left by *i* places, with $i=1, 2, 3, \ldots 2^{n/2}-2$; and $\oplus$ is the operator addition in module-2.

Globally, the set of Kasami sequences consists of $2^{n/2}$ sequences, with period $N_c=2^n-1$, and with low CC values between them. In the case of periodic emission of Kasami sequences, their AC and CC values are (2) and (3) respectively:

$$A_c(\tau) = \begin{cases} N_c & \tau = 0 \\ \left\{-1, -2^{n/2}-1, 2^{n/2}-1\right\} & \tau \neq 0 \end{cases} \quad (2)$$

$$C_c(\tau) = \left\{-1, -2^{n/2}-1, 2^{n/2}-1\right\} \quad (3)$$

3. Different Approaches for the Hardware Implementation of a detector of Kasami Sequences

The receiver system has to detect the different Kasami sequences emitted from the beacons, by carrying out all the correlations with the emitted codes. Figure 1 shows the block diagram of the reception system. It is divided into three main modules:

- An acquisition system that converts the signal received by the ultrasonic transducer into a digital signal.

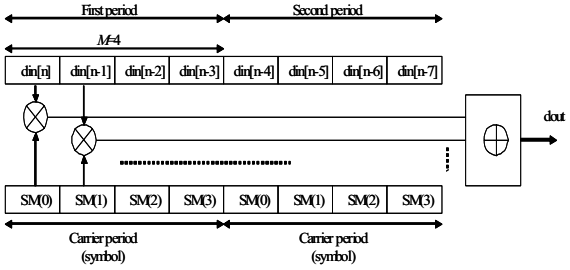


Figure 2. Block diagram of the BPSK demodulator.

- A BPSK demodulator that extracts the transmitted information from the received signal.
- A correlation stage that detects the emissions carried out by the beacons. It consists of as many correlators as beacons emitting Kasami codes.

Considering the demodulator, it is configured in pre-synthesis time according to the features of the application. Thus, it is possible to change the datawidth *DW* of the input, the number of periods of the symbol used in the modulation N_{SM}, and the oversampling factor *M*. Also, a sampling period of the carrier sequence should be provided, so the corresponding demodulation can be performed. In Figure 2 the internal scheme of the demodulator is shown, for the case of $M=4$ and $N_{SM}=2$.

The correlation of Kasami sequences is carried out by means of direct structures (straight-forward), in case of short sequences, or by FFTs in case of larger ones. In this work, the length of the used Kasami sequences does not exceed 255 bits, so a straight-forward implementation of the correlator has been considered. Four different approaches for the hardware implementation of the straight-forward correlator have been evaluated. In all of them, the multiplication operation has been reduced to additions and subtractions, since Kasami sequences are binary.

3.1. Combinational Operation Mode

As a first proposal, a combinational design has been done. At every rising edge of the clock signal a new input sample is received, which is demodulated and stored in a buffer of length $N_c \cdot M \cdot N_{SM}$, together with the last received samples. A combinational block accesses to the buffer data in gaps of $M \cdot N_{SM}$ positions, and it carries out the N_c additions and subtractions of the correlation, as can be seen in Figure 3. Nevertheless, this design is only feasible for short Kasami sequences ($N_c \leq 63$) and low oversamplig factors ($M \leq 4$). For instance, for Kasami

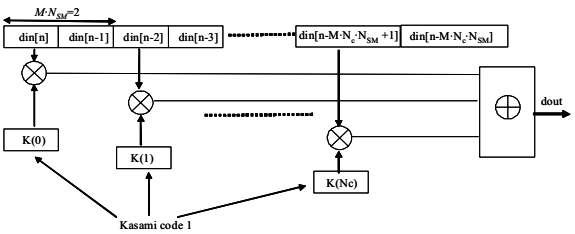


Figure 3. Combinational Implementation of one correlation stage.

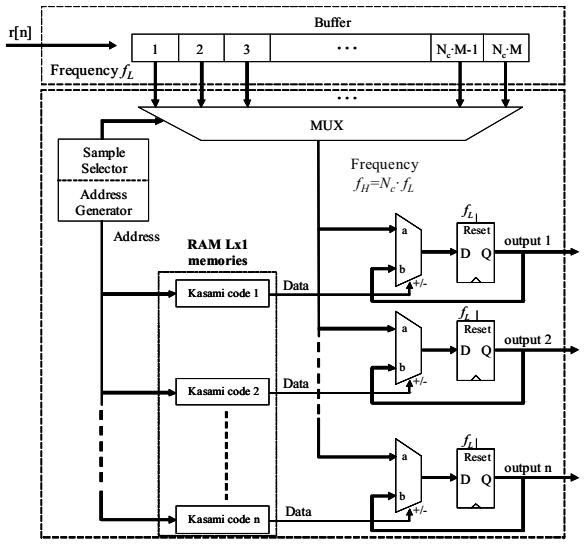


Figure 4. Sequential implementation of the straight-forward correlator.

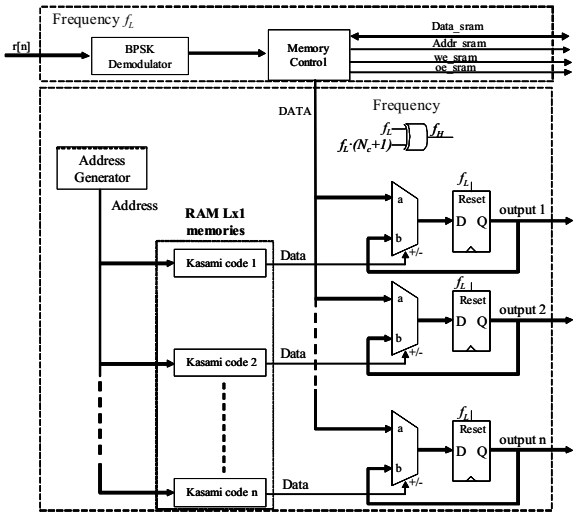


Figure 5. Hardware implementation of the sequential straight-forward correlator based on an external SRAM memory.

sequences up to 255 bits, with a Virtex–II Pro xc2vp100 FPGA [6], the design requires more resources than the available ones in this FPGA.

3.2. Sequential Operation Mode

Another possibility is a sequential design, in which the input data are received with a frequency f_L, demodulated and stored in a buffer of size $N_c \cdot M \cdot N_{SM}$. The sampling buffer is read with a frequency $f_H = f_L \cdot N_c$. In every reading, a sample is added o subtracted with the previous accumulated result, depending on the used correlation symbol. Whenever a new sample is received, the accumulated value is reset in order to compute the new correlation. It is important to remark that the access to the sampling buffer is carried out in gaps of $M \cdot N_{SM}$ positions, by taking into account the demodulation effect.

The computational blocks that carry out the additions and subtractions of the correlator, should be repeated N_k times (where N_k is the number of beacons emitting different Kasami codes). Furthermore, the symbols of the correlation are stored in N_k RAM memories of size $N_c x1$. The sampling buffer is shared by all the correlators, as can be observed in Figure 4.

The drawback of this design is the high requirements of resources, mainly because of the large amount of samples stored in the mentioned buffer. As an example, for Kasami sequences of N_c=255 bits, with an oversampling factor of M=10, one modulation symbol N_{SM}=1, and a datawidth of DW= 8 bits, it will be necessary the use of 20.400 flip-flops. In this design, the flip-flops will be mapped into the registers of the slices of the FPGA, which are not enough in most of the cases. As a solution, two different designs are proposed. In the first one, the sampling buffer has been implemented in an external SRAM memory; in the second one, the buffer has been mapped into the internal BRAM memory blocks of the FPGA [6].

3.3. External Memory Operation Mode

Here, the sampling buffer, storing the $N_c \cdot M \cdot N_{SM}$ samples coming out from the demodulator, is replaced by an external SRAM memory; in order to minimize the high utilization of slices in the previous design.

It is necessary to include a memory management block to organize the accesses to the external memory. Writing accesses are carried out to consecutive memory positions with a frequency f_L. On the other hand, reading accesses to memory are done every $M \cdot N_{SM}$ positions, for a frequency of $f_H = (N_c+1) \cdot f_L$ (it is necessary to add an additional read cycle, due to the fact that the reading process is disabled during writing). This hardware implementation is detailed in Figure 5.

3.4. Internal Memory based on BRAM Operation Mode

By using the internal RAMB16_Sn memory blocks [6] of the FPGA, it is possible to reduce the number of required slices, and also the complexity of the memory management. The RAMB16_Sn blocks have a capacity of 16.384 bits, and they can be configured to store data of 1, 2, 4, 9, 18 or 36 bits [6]. To make a suitable configuration of these memory blocks, it is necessary to consider the maximum requirements of the application.

In this case, the maximum length of Kasami sequences is N_c=255 bits, with an oversampling factor M=10, and a datawidth DW=8 bits. Therefore, it is necessary to have a memory with an address bus of $\log_2(N_c \cdot M) = 12\,bits$ and a data bus of $DW + \log_2(M) = 12\,bits$. In order to meet these specifications, three RAMB16_S4, with an address bus of 12 bits and a data width of 4 bits, have been linked with the aim of obtaining a data width of 12 bits (see Figure 6).

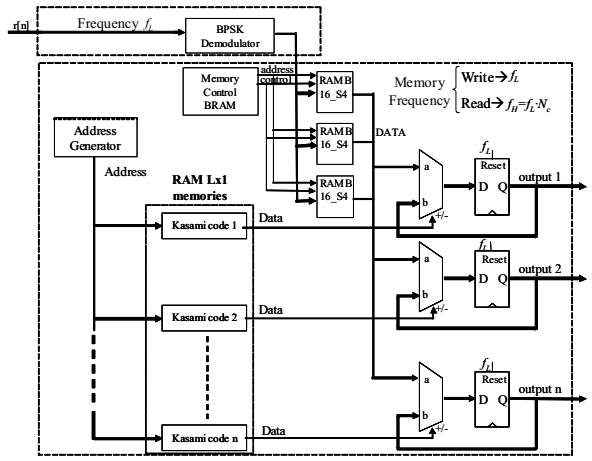


Figure 6. Hardware implementation scheme of the sequential straight-forward correlator based on internal BRAM memory.

The input signal is demodulated and stored in the RAMB16_S4 memories, in the position indicated by the memory management block. It is important to note that the RAMB16_Sn memories can be configured in WRITE_FIRST mode [6], so it is possible to write a data in memory and read it in the same clock cycle; in this way, the data processing rate is higher than in case C. The writing and reading accesses to memory are carried out with a frequency f_L and $f_H=f_L \cdot N_c$ respectively.

4. Results

The implementation of the system has been carried out in a Virtex-II Pro xc2vp100 FPGA [6]. Three different Kasami codes of N_c=255 bits has been assumed. The oversampling factor used is M=4 and the number of periods of the symbol used in the modulation is N_{SM}=1. Table 1 depicts the results obtained for the cases B, C and D, in terms of resource requirements and operating frequencies. In case A, the requirements of resources and operating frequencies cannot be supported by the FPGA.

As can be observed in Table 1, cases C and D are those with a better use of the resources available in the FPGA. This is because the data to be correlated are not stored in the slices of the FPGA. Furthermore, the case D, corresponding to the implementation of the correlator with internal BRAM of the FPGA, is more efficient than the case C, which uses external memory. The main reason is that, in case D, writing and reading accesses can be achieved at the same clock cycle, increasing the operating frequency.

To carry out a fair comparison among the proposed designs, a large FPGA (Virtex-II Pro) has been used, since it is required by the design B for the specified values of N_k, N_c, M and DW. Nevertheless, cases C and D could be implemented in smaller FPGAs.

TABLE I.
DEVICE UTILIZATION COMPARISON

xc2vp100 N_k=3, N_c=255, M=4, DW=4	Sequential (B)	Sequential External Memory (C)	Sequential BRAM (D)
Slices	14,22 %	0,25 %	0,24 %
LUTs	5,40 %	0,20 %	0,19 %
IOBs	5,10 %	6,83 %	5,10 %
BRAM	0,68 %	0,68 %	1,13 %
Max. Freq.	74,063 MHz	111,01 MHz	132,77 MHz

5. Conclusions

Four different hardware implementations of a straight-forward correlator for Kasami sequences have been carried out. A combinational one is proposed for short Kasami sequences; whereas for longer sequences three sequential designs are considered, which differ on the management of the necessary memory. Storing the samples in slices of the FPGA is not convenient, since the resources of the FPGA are rapidly ran out. Other option consists of using an external memory, although it implies the use of other devices apart from the FPGA. As has been tested, the best solution is to use the RAM blocks of the FPGA: no additional devices are needed, and the resources requirements are not so high as in the case of using the slices of the FPGA.

Acknowledgement

This work has been possible by funding from the Madrid Community (FPI fellowship, ANESUS project: CAM-UAH2005/016), and from the Spanish Ministry of Science and Technology (PARMEI project: DIP2003-08715-C02-xx).

References

[1] R. L. Pickholtz, D. L. Schilling, L. B. Milstein, "Theory of Spread Spectrum Communications-A Tutorial" *IEEE Trans. Commun.*, vol. COM-30, nº 5, pp. 855-884, 1982.

[2] D. Sarwate and M. Pursley, "Crosscorrelation Properties of Pseudorandom and Related Sequences," *Proc. IEEE*, vol. 68, no. 5, pp. 593-619, May 1980.

[3] E. H. Dinan, B. Jabbari, "Spreading Codes for Direct Sequence CDMA and Wideband CDMA Cellular Networks". *IEEE Comm. Magazine*, September 1998.

[4] A. Mahajan, M. Walworth, "3D Position Sensing Using the Differences in the Time-of Flights from a Wave Source to Various Receivers", *IEEE Trans. on Robotics and Automation*, vol. 17, no. 1, pp. 91-94, 2001.

[5] T. Kasami, "Weight distribution formula for some class of cyclic codes", *Combinational Mathematics and its Appl.*, Chapel Hill, N.C.: Univ. of North Carolina Press, 1969.

[6] Xilinx, Inc., Virtex II PRO Family: Complete Datasheet, *Product Documentation*, October 2005.

FPGA IMPLEMENTATION OF AN EFFICIENT CORRELATOR FOR COMPLEMENTARY SETS OF SEQUENCES

M.C.Pérez, J. Ureña, A. Hernández , C. De Marziani, A. Ochoa [*]

Dept. of Electronics. University of Alcalá
Campus universitario s/n, Alcalá de Henares, Spain
email: carmen@depeca.uah.es

W. P. Marnane [†]

Dept. of Electrical & Electronic Engineering
University College Cork, Cork, Ireland
email: marnane@ucc.ie

ABSTRACT

The use of complementary sets of sequences is especially relevant in multisensorial systems, with simultaneous emissions from different sources and low signal-to-noise ratio (SNR). There exist efficient generation and correlation algorithms that give a reduction in the computational load and complexity of a hardware implementation. In this work a new hardware design approach for an efficient correlator of complementary sets of sequences (CCS) is presented. It is based on utilising reconfigurable logic to gain the required flexibility in terms of correlator parameters. The configuration of the design is carried out at the pre-synthesis stage, so the user can change this configuration according to the requirements of the application.

1. INTRODUCTION

The effectiveness of CDMA systems (Code Division Multiple Access) strongly depends on the properties of the spread spectrum codes used. Examples of codes are PN sequences (M-sequences, Gold, Kasami) [1], Barker codes [2], Walsh-Hadamard [3], Golay sequences [4], etc. Each one of these sequences has auto-correlation (AC) and cross-correlation (CC) properties that determine the kind of application in which they are used. There are simultaneous multi-emission systems that are characterised by no periodic emissions, low signal-to-noise ratio (SNR) and asynchronous detection. In these cases, low aperiodic cross-correlation sequences should be used in order to avoid mutual interference. In [5] Complementary Sets of *M* Sequences (*M*-CSS) are analysed, where these sequences are a generalization of Golay complementary pairs. Furthermore, with these sequences it is possible to work with more than two mutually orthogonal sets, so they are very attractive for systems with several simultaneous emissions.

The generation and correlation of these sets can be accomplished by applying recursive methods that allow the design of efficient generators and correlators. Therefore, the computational load and the hardware complexity are reduced (in comparison with those required in the straightforward implementation) [6][7].

In [7] a regular structure is provided, which allows the implementation of an efficient correlator of *M*-CSS, based on the structure for *M/2*-CSS. In this work, a generic implementation in field programmable gate arrays (FPGA) of the algorithm proposed in [7] has been developed.

The paper is organized as follows: Section 2 reviews the properties of *M*-CCS. In Section 3 the architecture of the efficient correlator and its hardware implementation are explained. Section 4 shows a detection system prototype based on the efficient correlation algorithm. Finally conclusions are discussed in Section 5.

2. COMPLEMENTARY SETS OF SEQUENCES

Let be a set of *M* sequences, all of them with length $L=M^N$, where $M=2^m$, and *m* and *N* are any natural number different from zero. This set is a complementary set if the Addition of the Auto-Correlation (AAC) functions of the sequences in the set is zero except for a zero-shift term. Furthermore, if the elements of the sequence are {+1,-1}, the AAC function has a maximum value of $M \cdot L$ for the zero shift (1).

$$AAC = \sum_{i=1}^{M} \phi_{S_{i,M}\, S_{i,M}}[j] = \phi_{S_{1,M}\, S_{1,M}}[j] + \ldots + \phi_{S_{M,M}\, S_{M,M}}[j] \quad (1)$$
$$AAC = M \cdot L \cdot \delta[j] = M \cdot M^N \cdot \delta[j] = M^{N+1} \cdot \delta[j]$$

Where $S_{i,M}$, with i=1, 2, 3, …, M, is the i-th sequence of the set; $\delta[j]$ is a Krönecker delta function; and $\phi_{x,y}$ is the operation of correlation between x and y.

It is possible to obtain *M* mutually orthogonal CCS: *M* sets whose Addition of Cross-Correlations (ACC) functions of the corresponding sequences in these *M* sets is always zero. Let be two different orthogonal sets, with sequences $S'_{i,M}[k]$ and $S''_{i,M}[k]$ respectively, the ACC functions of the corresponding sequences in both sets are (2):

$$ACC = \phi_{S'_{1,M}\, S''_{1,M}}[j] + \cdots + \phi_{S'_{M,M}\, S''_{M,M}}[j] = 0 \qquad \forall j \quad (2)$$

1-4244-0 312-X/06/$20.00 ©2006 IEEE.

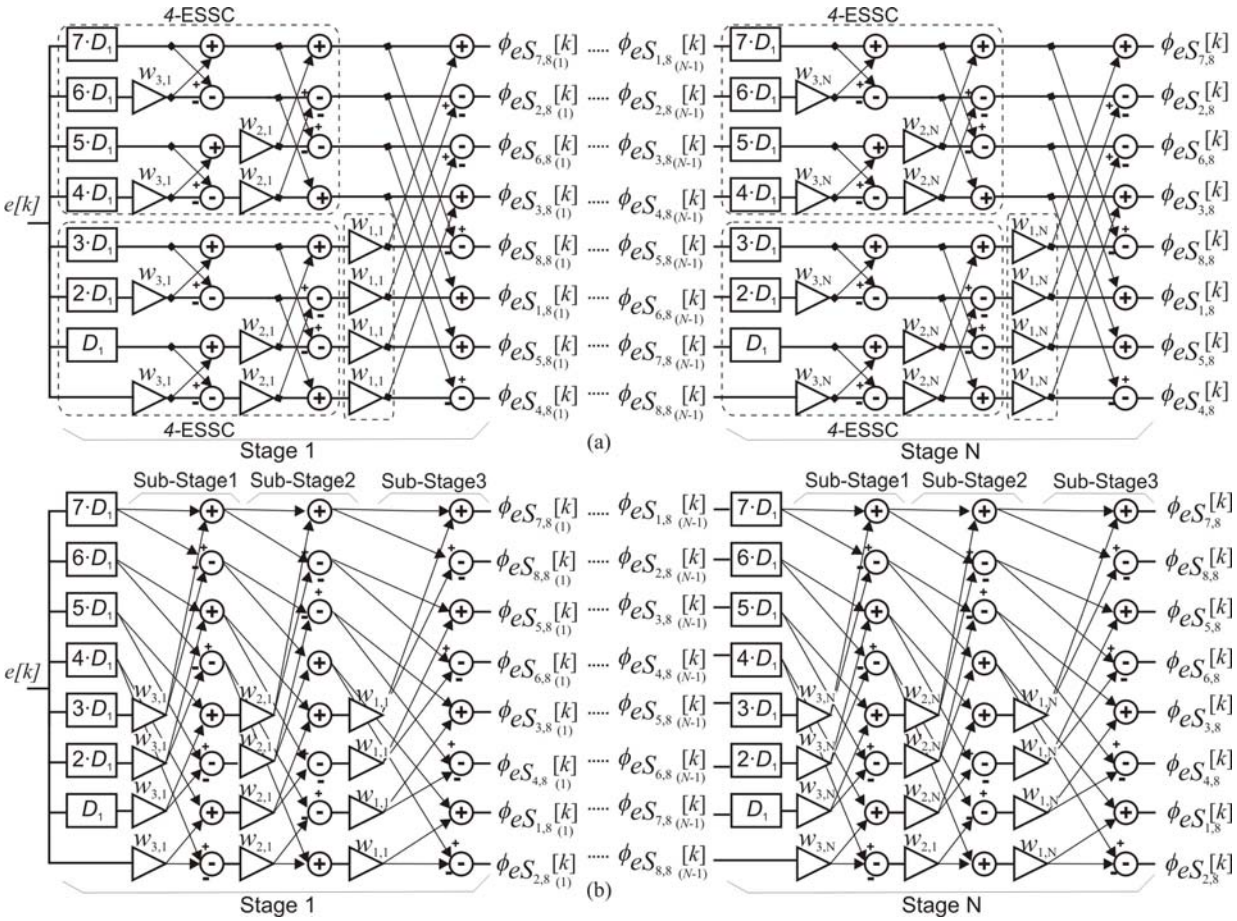


Fig.1. a) Block diagram of a *8-ESSC* [7], b) Rearrangement of the Fig. 1.a, with the same geometry for every stage.

3. HARDWARE CONSIDERATIONS

In [7] a method is developed, which allows the construction of an efficient correlator of *M*-CSS, called *M*-ESSC. It consists of repeating the architecture for the *M/2*-CSS correlator, changing the delay coefficients D_n of the repeated structure, and adding a set of multipliers $w_{i,k}$ at the output of the initial structure. D_n is a positive delay defined as $D_n=2^{Pn}$, where *Pn* is any permutation of the numbers {0,1,2…,*N*-1}, and $w_{i,k}$, with i=1,2,…,M and k=1,2,…,N, the generation seeds of the set represented in binary code (notice that the multiplication operation is reduced to additions and subtractions, due to the binary character of the seed $w_{i,k}$). Also, it is necessary to add a new set of adder-subtracters, with the sign inverted in the initial structure. It is important to note that the order of the sequences at the output of a stage does not fit with the input order of the next stage, so they have to be re-ordered. Fig. 1.a shows an example of a *8*-ESSC generated from a *4*-ESSC (an initial *2*-ESSC has been also used to generate the *4*-ESSC).

The described architecture presents drawbacks for its generic hardware implementation:

- The internal links in every stage are not regular, so depending on the number of sequences of the set, these links vary.
- The order of the sequences at the output of the stage differs depending on the number of sequences of the set.

In consequence, it is desirable to have a modular and regular architecture, which can be easily adapted to the different values of the number *M* of sequences in the set and their length *L*.

3.1. Transformation of the algorithm structure

The proposed transformation exploits the similarities between the structure of the *M*-ESSC and the structure of the decimation in frequency FFT (Fast Fourier Transform) algorithm; so the properties of the FFT can be used to make regular the structure of the *M*-ESSC.

It is possible to arrange the decimation in frequency FFT algorithm so that the same geometry for every stage is obtained, thereby permitting sequential data access and storage [8]. By rearranging in the same way the data of the *M*-ESSC, a regular distribution of the internal links in every stage is obtained. Thus, it is necessary to rearrange the inputs of every stage in inverted binary order; in other words, reading backward the binary representation of the position of every input: i. e., the input 3 (011) will appear now in position 6 (110). The stage outputs are ordered in the same way, independently of the number of sequences in the set, and therefore the architecture of the generic

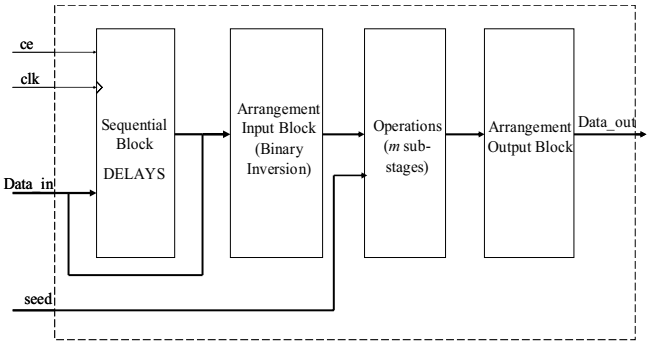


Fig. 2. Implementation scheme in VHDL of a stage of the *M*-ESSC.

Table 1. Device utilization comparison.

xc3s1500	Case A *M*=4, *L*=64	Case B *M*=2, *L*=2
Slices	2,62 %	0,20 %
LUTs	2,17 %	0,16 %
IOBs	14,78 %	5,95 %
Max. Freq.	33,8 MHz	100,817 MHz

correlator has a regular data flow, allowing a more efficient hardware implementation.

As an illustration of this transformation, an example with *8*-CSS is illustrated in Fig. 1. It is clear that the internal data flow of the stages in the *8*-ESSC are similar to those in the decimation in frequency algorithm for the calculation of an 8 point DFT [8].

3.2. Implementation notes

The pre-synthesis parameters for a parametric design of the *M*-ESSC include: the number *M* of sequences in the set, their length *L*, the oversampling factor, the number of symbols used in the modulation, and the data width at the input. The main unit of the design is shown in Fig. 2, and corresponds with a stage of the *M*-ESSC. The first sequential block implements the required delays for every stage, by using specific shift registers in Xilinx FPGA's architecture (modules SRL16 [9]). The outputs of these delays are rearranged according to the inversion binary algorithm explained above, and are connected to a combinational block, that carries out the operations of multiplication, addition and subtraction of the stage. It is important to emphasize that the multiplication operation is reduced to additions and subtractions, because of the fact that the elements of the generation seed are {+1,-1}. Furthermore, due to the new geometry of the algorithm, it is only necessary to implement one basic sub-stage of the operation block, and to repeat it at synthesis, depending on the parameter *m* (where $M=2^m$ is the number of sequences in the set). Also, the introduction of pipelining registers between sub-stages increase the operation frequency of the system. Finally, the output of the operation block is ordered for its correct connection to the next stage.

The resource requirements and maximum operation frequencies of the hardware implementation of the *M*-ESSC depends of the number *M* of sequences and its length *L*. Table 1 shows the results obtained for a case A with *M*=4 and *L*=64; and a case B with *M*=2 and *L*=2. An 8 bits data bus is assumed, and a Spartan3 xc3s1500 FPGA by Xilinx is used [9].

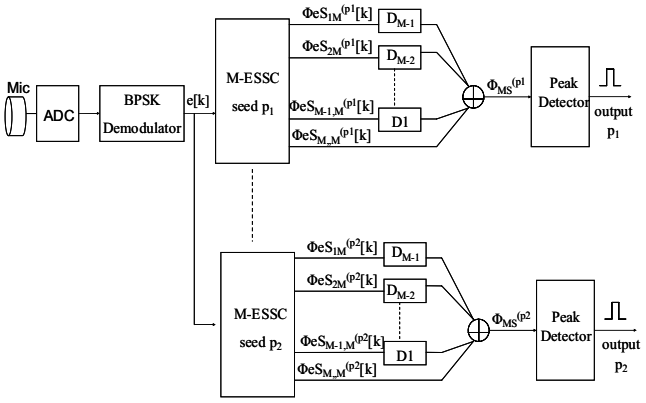


Fig. 3. Hardware implementation scheme of the correlation of macro-sequencies using *M*-ESSCs.

4. APPLICATION EXAMPLE

In order to demonstrate the suitability of the design methods employed, the efficient correlator has been included in the reception system showed in Fig. 3, allowing the simultaneous detection of different complementary sets.

The bits that form the sequences $S_{i,M}^p$ of the complementary sets, has been grouped by means of interleaving in macrosequences Ms^p of length $M{\cdot}L$ [10]. Let be $S_{i,M}^p[k]=\left[x_{i1}^p \quad x_{i2}^p \quad \ldots \quad x_{i(L-1)}^p \quad x_{i(L)}^p\right]$ the *i*-th (*i*=1, 2…*M*) of the *M*-CSS*;* the macrosequence Ms^p is generated by:

$$Ms^p[k]=\left[S_{i1}^p \otimes S_{i2}^p \quad \ldots \quad \otimes S_{i(M)}^p\right]=$$
$$=\left[x_{11}^p \quad x_{21}^p \quad \ldots \quad x_{(M)1}^p \quad x_{12}^p \quad \ldots \quad x_{(M)L}^p\right]$$

Where $\otimes$ is the interleaving operator; x_{ik}^p, with *k*=1,2,…,*L,* are the bits of the sequence $S_{i,M}^p$; and *p* is the decimal representation of the binary generation seed w_{ik} of the set, which appears in the structures of the efficient generator and correlator [7].

In Fig. 3 the hardware implementation of the reception system is shown. The received signal is adapted and demodulated for its later correlation. Since no temporal reference is available, the demodulation is carried out asynchronously by sampling the received signal at a high enough sample rate, and then correlating it with the symbol used in the modulation. The signal coming out from the demodulator is not directly correlated with the sequences of the CCS, but with zero interpolation versions of those sequences. The interpolation factor depends on the number *M* of sequences, the number of symbols used in the modulation and the oversampling factor used to capture the data input. This requires a modification to the efficient correlator structure by multiplying the delays in every stage by the interpolation factor. This modified correlator makes the simultaneous correlation between the

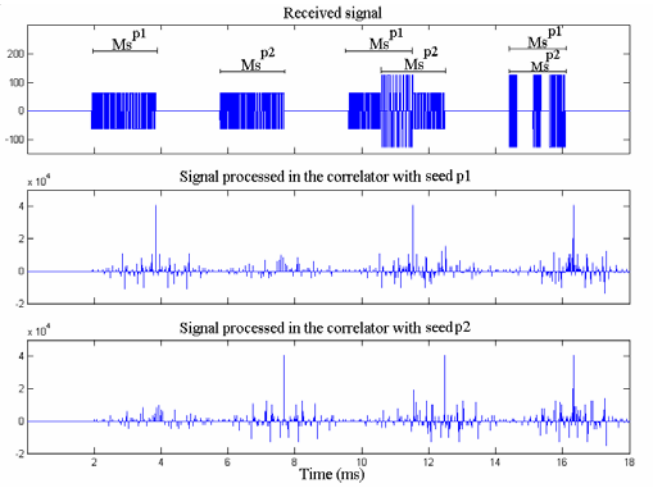


Fig. 4. Example of signal detection, with *M=8, L=8* and f_e*=33,33 kHz*.

input sequence and every sequence in the CCS. It is enough to connect directly the output of the demodulator to the modified correlator, obtaining the desired auto-correlation at every branch of the correlator. These auto-correlations are not in phase (the phase shift depends on the interleaving order, on the number of symbols in the modulation, and on the oversampling factor). Thus, it is necessary to insert delays in every branch of the correlator, in order to perform the in-phase addition of the auto-correlation functions. Finally, a peak detector based on a static threshold is included. The peak detector emits a digital pulse at a specific pin of the FPGA in cases that the result exceeds a certain threshold.

In the proposed example an xc3s1500 Xilinx FPGA [9] is used. Two different macrosequences, Ms^{p1} and Ms^{p2}, are obtained, by using two *8*-CCS with length 8 generated with different seeds. These macrosequences are modulated in BPSK, using a squared modulation carrier. The emission frequency f_e=33,33 kHz, and the oversampling factor of the received signal is 10.

Firstly, the macrosequences Ms^{p1} and Ms^{p2} are separately received; then they are received partially overlapped; and finally completely overlapped (as can be seen in the first graph in Fig. 4. The second and third graphs show the correlation of the received signal with the macrosequences generated by seeds *p1* and *p2*, respectively. It is possible to remark that both emitted macrosequences are suitably detected at their corresponding correlators.

5. CONCLUSIONS

A generic hardware implementation of an efficient correlator of *M*-CSS has been developed. The correlator parameters, that is, the number of sequences, its length, the oversampling factor, the number of symbols used in the modulation and the data width, can be changed through synthesizing the design.

Furthermore, a reception prototype in a FPGA has been developed. The sequences of the set are organized as a macrosequence, and the structure of the correlator has been adapted to detect the emitted macrosequence in only one correlator. Multiemission has been verified, by receiving two differents macrosequences partially and completely overlapped.

6. ACKNOWLEDGEMENTS

This work has been possible thanks to the Madrid Community (FPI fellowship, ANESUS: CAM-UAH2005/016), the European Social Fund, and from the Spanish Ministry of Science and Technology (PARMEI: DIP2003-08715-C02-01).

7. REFERENCES

[1] D. Sarwate and M. Pursley, "Crosscorrelation Properties of Pseudorandom and Related Sequences," *Proc. IEEE*, vol. 68, no. 5, pp. 593-619, May 1980.

[2] S. W. Golomb and R.A. Scholtz, "Generalized Barker sequences," *IEEE Trans. Inf. Theory*, vol. IT-11, no. 4, pp. 533-537, Oct. 1965.

[3] H. Harmuth, "Application of Walsh Functions in Communications," *IEEE Spectrum*, vol. 6, pp. 82-91, Nov. 1969.

[4] M. J. E. Golay, "Complementary series," *IRE Trans. On Inform. Theory,* vol. IT-7, no. 2, pp. 82-87, Apr. 1961.

[5] C. C. Tseng and C. L. Liu, "Complementary sets of sequences," *IEEE Trans. Inform. Theory*, vol. IT-18, no. 5, pp. 644–652, Sept. 1972.

[6] B. M. Popovic, "Efficient Golay correlator," *Electron. Lett*., vol. 35, no. 17, pp. 1427–1428, Aug. 1999.

[7] C. Marziani, J. Ureña, A. Hernández, M. Mazo, F. Álvarez, J. J. García, P. Donato, "Modular Architecture for Efficient Generation and Correlation of Complementary Set of Sequences," *IEEE Trans on Signal Processing*, Accepted for its publication, T-SP-03461-2005.R1, Dec. 2005.

[8] A. V. Oppenhein, R. W. Shafer, J. R. Buck, "Discrete-Time Signal Processing," *Prentice Hall*, 2nd edition, ISBN: 0137549202, 1999.

[9] Xilinx, Inc., Spartan-3E FPGA Family: Complete Datasheet, *Product Documentation*, Nov. 2005.

[10] C. Marziani, J. Ureña, A. Hernández, M. Mazo, F. Álvarez, J.J. García, J. M. Villadangos, A. Jiménez, "Simultaneous Measurement of Times-of-Flight and Communications in Acoustic Sensor Networks*", Conf. Proc.International Symposium on Intelligent Signal Processing*, *WISP 2005,* pp. 122-127, Sept. 2005.

Journal of Universal Computer Science, vol. 13, no. 3 (2007), 388-406
submitted: 23/11/06, accepted: 19/2/07, appeared: 28/3/07 © J.UCS

Hardware Implementation of an Efficient Correlator for Interleaved Complementary Sets of Sequences

María del Carmen Pérez, Jesús Ureña, Álvaro Hernández, Carlos De Marziani, Ana Jiménez
(Dept. of Electronics, University of Alcalá, Spain
carmen@depeca.uah.es; urena@depeca.uah.es; alvaro@depeca.uah.es;
marziani@depeca.uah.es; ajimenez@depeca.uah.es)

William P. Marnane
(Dept. of Electrical & Electronic Engineering, University College Cork, Ireland
marnane@ucc.ie)

Abstract: Some sensor systems are characterized by multiple simultaneous aperiodic emissions, with low signal-to-noise ratio and asynchronous detection. In these systems, complementary sets of sequences can be used to encode emissions, due to their suitable auto-correlation and cross-correlation properties. The transmission of a complementary set can be accomplished by interleaving the sequences of the set, generating a macro-sequence which is easily transmitted by a BPSK modulation. The detection of the macrosequence can be performed by means of efficient correlation algorithms with a notably decreased computational load and hardware complexity. This work presents a new hardware design in configurable logic of an efficient correlator for macrosequences generated from complementary sets of sequences. A generic implementation has been achieved, so the configuration can be changed according to the requirements of the application. The developed correlator has been tested in an ultrasonic pulse compression system in which real-time is needed. However, it is applicable in any multi-sensor or communication system where the goal is to make simultaneous emissions from different independent sources, minimizing mutual interference.

Key Words: Compression techniques, Complementary Sets of Sequences, efficient correlation, FPGA-based implementation, Ultrasonics.
Category: B.4.0, B.5.1, B.5.2

1 Introduction

Many sensor systems based on time-of-flight measurements, for example ultrasonic systems, have incorporated signal processing techniques previously used in radar theory [Audenaert et al. 1992] [Jörg and Berg 1998]. In these systems, signals are encoded with binary sequences with similar properties to Gaussian noise. Afterwards, the corresponding echoes are detected by means of correlation with the original binary sequence. Thus, an improvement of temporal precision, spatial resolution and robustness to noise is obtained, in comparison with methods based on threshold detection. In the case of simultaneous multi-emission systems, with non-periodic emissions and asynchronous detection, sequences with

a high Auto-Correlation (AC) function and low sidelobes should be used, as well as with low aperiodic Cross-Correlation (CC) values among them. As a result, it is possible to detect them even in the case of high noise or attenuation with minimum interference.

Different sequences are used to encode the transducer emissions of these systems: pseudo-random sequences [Sarwate and Pursley 1980], Walsh-Hadamard [Harmuth 1969], Barker codes [Golomb and Scholtz 1965], Golay codes [Golay 1961] or Complementary Sets of M sequences (M-CSS) [Tseng and Liu 1972]. M-CSS are characterized by the ideal properties of the sum of their AC and CC functions: they provide a high process gain and also M mutually orthogonal sets. Therefore, M signals can be simultaneously emitted and receivers are able to distinguish them without interference. The problem is how to emit the M-CSS assigned to every transducer without a big change in their ideal characteristics.

There exist several methods of transmitting M-CSS [Álvarez et al. 2004]. A simple one consists of interleaving the bits from the sequences of complementary sets, obtaining a longer sequence, called *macro-sequence*. Thus, all changes in the environment have the same effect on the M sequences of the set. This is important since the detection capability of the system is based on the complementary property of sequences. Furthermore, efficient algorithms for M-CSS processing have been developed [De Marziani et al. 2005], reducing their computational cost and hardware complexity, in comparison with those required in straightforward matched filter implementations [Hahm et al. 1997].

This work presents a generic implementation in Field Programable Gate Arrays (FPGAs) of an efficient correlator of interleaved M-CSS, and its application to an ultrasonic pulse compression system. The rest of the paper is organized as follows: Section 2 reviews the properties that characterize the M-CSS, and details the transmission method based on the interleaving of the sequences of the set. Section 3 presents the efficient correlator of macro-sequences. The implementation of this system is considered in Section 4. Section 5 shows an ultrasonic detection system prototype based on the developed correlator. Finally, conclusions are outlined in Section 6.

2 Encoding scheme based on interleaved M-CSS

2.1 Complementary sets of sequences

A set of M equally long binary sequences ($S_{i,M}[k] = [x_{i,1}\, x_{i,2} \cdots x_{i,L}]$; $1 \leq i \leq M$) has been defined, where $x_{i,j} = \pm 1$ are the elements of the sequences, $L = M^N$ is their length, and $M = 2^m$, with N and $m \in \mathbb{N} - \{0\}$. This set is said to be complementary if the Addition of the Auto-Correlation (AAC) functions

of the sequences from that set is zero for all non-zero time shifts (1).

$$\begin{aligned} AAC &= \sum_{i=1}^{M} \phi_{S_{i,M}S_{i,M}}[k] = \phi_{S_{1,M}S_{1,M}}[k] + \cdots + \phi_{S_{M,M}S_{M,M}}[k] \\ &= M \cdot L \cdot \delta[k] = M \cdot M^N \cdot \delta[k] = M^{N+1} \cdot \delta[k] \end{aligned} \tag{1}$$

Where $\delta[k]$ is a Krönecker delta function; and $\phi_{x,y}$ is the correlation between x and y.

Furthermore, it is possible to obtain M mutually orthogonal sets. Complementary sets of sequences are mutually orthogonal if the Addition of Cross-Correlation (ACC) functions from the corresponding sequences of any two sets is zero with any shift. Consider two different orthogonal sets, with sequences $(S'_{i,M}[k], S''_{i,M}[k];\ 1 \le i \le M)$ respectively, the ACC is (2):

$$ACC = \phi_{S'_{1,M}S''_{1,M}}[k] + \cdots + \phi_{S'_{M,M}S''_{M,M}}[k] = 0 \qquad \forall k \tag{2}$$

2.2 Transmission method by interleaving

In many systems, transducers used to transmit and receive the encoded signals have limited bandwidths. Therefore, it can be necessary to modulate the sequences in order to have their spectra at the region of maximum response of trasducers. In asynchronous detection, the process gain increase, provided by multi-level modulation techniques, is reduced because of the lower energy transmitted per bit [Álvarez et al. 2004]. Also, if the number M of sequences is high, the asynchronous detection scheme becomes more complex. Thus, a BPSK modulation has been adopted. It is desirable that all the sequences of the set be equally affected by changes in the environment, in order not to lose the complementary properties among them. A transmission method based on the construction of macro-sequences by interleaving sequences of complementary sets has been carried out.

In order to construct a macro-sequence $Ms[k]$ of length $M \cdot L$, the bits $x_{i,j}$ that form the sequences $S_{i,M}$ have been grouped by interleaving as in (3), where $\otimes$ is the interleaving operation.

$$\left.\begin{array}{rl} S_{1,M} &= [\ x_{1,1}\ \ x_{1,2}\ \cdots\ x_{1,L}\ ] \\ S_{2,M} &= [\ x_{2,1}\ \ x_{2,2}\ \cdots\ x_{2,L}\ ] \\ \vdots &= [\ \vdots\ \ \vdots\ \ \cdots\ \ \vdots\ ] \\ S_{M,M} &= [\ x_{M,1}\ x_{M,2}\ \cdots\ x_{M,L}\ ] \end{array}\right\} \Rightarrow \begin{aligned} Ms &= [S_{1,M} \otimes S_{2,M} \otimes \cdots \otimes S_{M,M}] \\ &= [x_{1,1} \cdots x_{M,1} \cdots x_{1,L} \cdots x_{M,L}] \end{aligned} \tag{3}$$

After interleaving, the ideal AC and CC properties of the M-CSS are degraded: neither the AC sidelobes, nor the CC values for two orthogonal sets are null anymore. A method to reduce this degradation is explained in [De Marziani

et al. 2006]. It consists in choosing macro-sequences with better AC and CC properties, and in applying a post-processing algorithm. As a result, the non-ideal effects in correlation are minimized, and the interleaving method still assures the reliability of the detection system, even when operating in noisy environments.

3 Efficient correlator for interleaved M-CSS

3.1 Efficient correlation algorithm for complementary sets of sequences

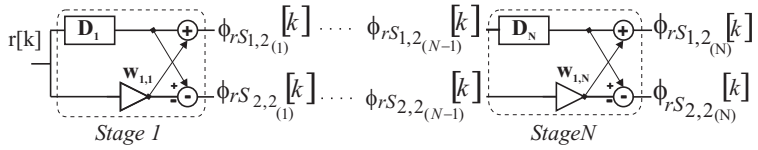
In [De Marziani et al. 2005] a recursive method is developed to allow the construction of an efficient correlator for M-CSS, called M-ESSC. It can be considered as a digital filter of N similar stages. This filter simultaneously performs the correlation of the input signal with the M sequences in the set, reducing the total number of operations required by the correlation. Thus, the M-ESSC carries out $3 \cdot 2^{m-1} \cdot m \cdot N$ operations, whereas a straightforward implementation of the correlator computes $2^m \cdot (2^{mN+1} - 1)$ operations. Thanks to this reduction in the number of operations, the M-ESSC can be implemented in configurable hardware, achieving real-time operation.

The efficient correlator is composed of a set of adders, subtractors and delay elements. Figure 1 shows the implementation scheme of an M-ESSC for the case of *2*-CSS (a), *4*-CSS (b), and finally, for the general case of M-CSS (c). As can be observed in Figure 1.c, the M-ESSC mainly consists of repeating the architecture of the $M/2$ -ESSC. However, some modifications have to be considered with every iteration:

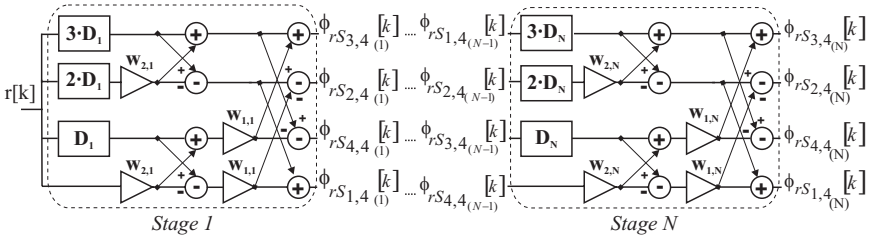
- Firstly, coefficients $\{(\frac{M}{2}-1), (\frac{M}{2}-2), \cdots, 1\}$ multiplying delay elements D_n of the repeated structure are changed by coefficients $\{(M-1), \cdots, (\frac{M}{2}+1), (\frac{M}{2})\}$. D_n is a positive delay defined as $D_n = 2^{Pn}$, where Pn is any permutation of numbers $\{0, 1, 2, \cdots, N-1\}$.

- Secondly, it is necessary to add a set of multipliers $\{w_{i,k}, 1 \leq i \leq m, 1 \leq k \leq N\}$ at the output of the initial structure. These multiplication coefficients $w_{i,k}$ have values +1 or -1, and represent the generation seed $W = [w_{m,N}\, w_{m-1,N} \cdots w_{m,N-1}\, w_{m-1,N-1} \cdots w_{1,1}]$ of the complementary set of sequences. Since these coefficients $w_{i,k}$ are binary, multiply any signal by them is reduced to negate or not the signal. In practice, the value of these coefficients determine the final configuration of the adders and subtractors in the stage.

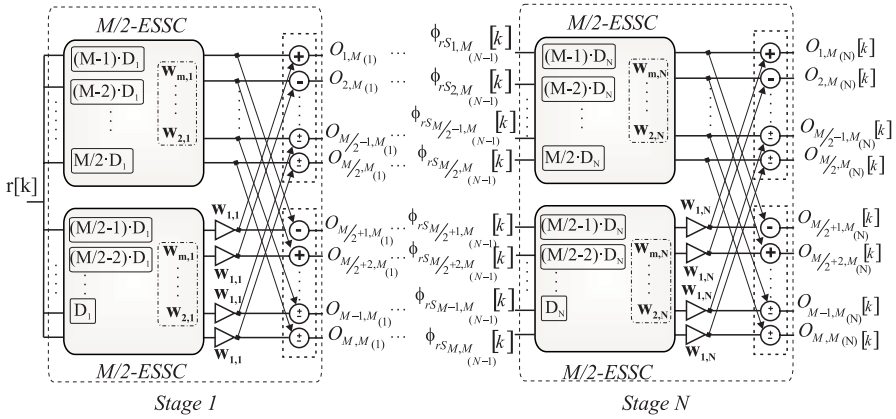
- A new set of adders and subtractors should be incorporated. Notice that adders and subtractors in the initial structure have the sign inverted.



(a) Block diagram of the *2*-ESSC.



(b) Block diagram of the *4*-ESSC.



(c) Block diagram of the M-ESSC.

Figure 1: Efficient correlator structures for M-CSS [De Marziani et al. 2005].

– Finally, it is necessary to reorder outputs $\{O_{i,M};\, 1 \leq i \leq M\}$ from every stage in the correlator, before connecting them to the next stage, since there does not exist a direct correspondence. In [De Marziani et al. 2005] a general recursive method is proposed to determine the output order.

This architecture is not suitable for a generic hardware implementation. The internal links in every stage and the order of the sequences at the output of every stage, differ depending on the number of sequences in the set. Furthermore, it does not take into account the interleaving transmission scheme. Therefore, it is

necessary to carry out the following modifications:

- Transformation of the M-ESSC architecture into a regular and modular one, so it can be easily adapted to the different values of M.
- Adaptation of the architecture for the interleaving-based transmission method.

3.2 Transformation of M-ESSC structure

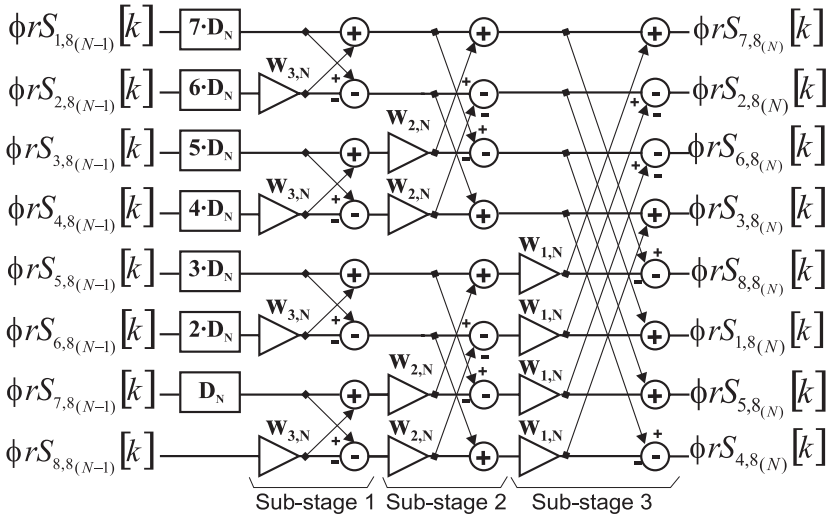
The first proposed transformation exploits the similarities between the structure of the M-ESSC (see Figure 2.a) and the structure of the decimation in frequency FFT (*Fast Fourier Transform*) algorithm (see Figure 2.b); so the properties of the FFT can be used to make the structure of the M-ESSC regular.

It is possible to arrange the decimation in frequency FFT algorithm, so the same geometry is obtained at every stage, allowing sequential data access and storage [Oppenhein et al. 1999]. With this purpose, the input should be organized in a bit-reversed order: if $(n_2\, n_1\, n_0)$ represents the index n of the input sequence $x[n]$ in binary form, the sample $x[n_2\, n_1\, n_0]$ should appear at location $(n_0\, n_1\, n_2)$ after reordering. In this way, the sample $x[3]$ should appear at position 6, as can be observed in Table 1.

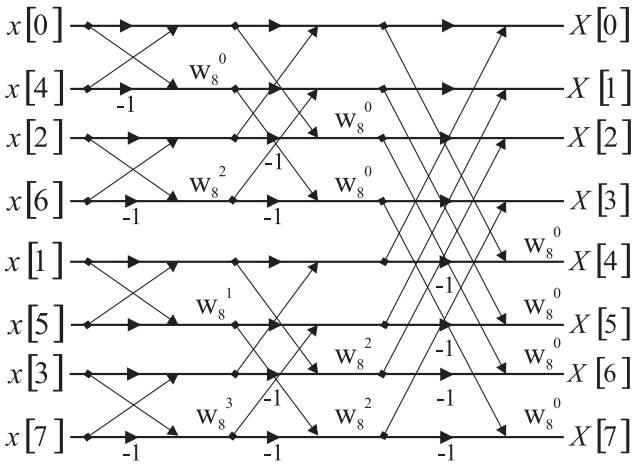
$n_2\, n_1\, n_0 \Rightarrow$ Bit reversed $\Rightarrow n_0\, n_1\, n_0$	
(0 0 0)	(0 0 0)
(0 0 1)	(1 0 0)
(0 1 0)	(0 1 0)
(0 1 1)	(1 1 0)
(1 0 0)	(0 0 1)
(1 0 1)	(1 0 1)
(1 1 0)	(0 1 1)
(1 1 1)	(1 1 1)

Table 1: Bit-reversed order algorithm.

If the same technique is used to rearrange data in the M-ESSC, a regular distribution of the internal links for every stage is obtained, and the order of the outputs can be easily achieved. Thus, coefficients multiplying delays D_n should be organized according to the bit-reversed order algorithm. In this way, the delay $3 \cdot D_n$ appears at the location previously occupied by the delay $6 \cdot D_n$. As an illustration of this transformation, an example with an *8*-ESSC is shown in Figure 3. It can be observed that, after the transformation, the internal connections between sub-stages are always the same, independently of the number M



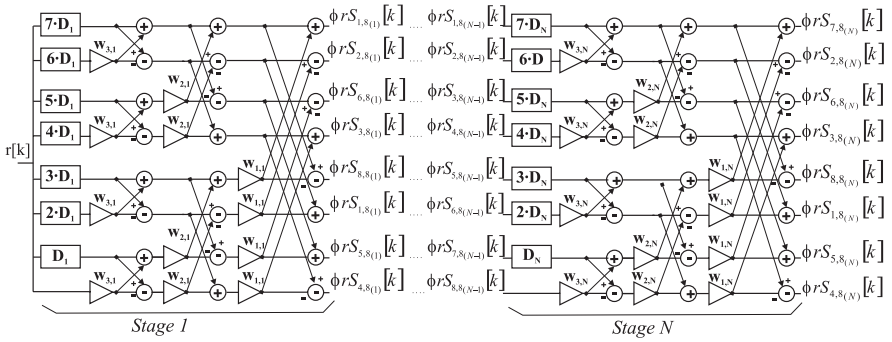
(a) Block diagram of an *8*-ESSC stage.



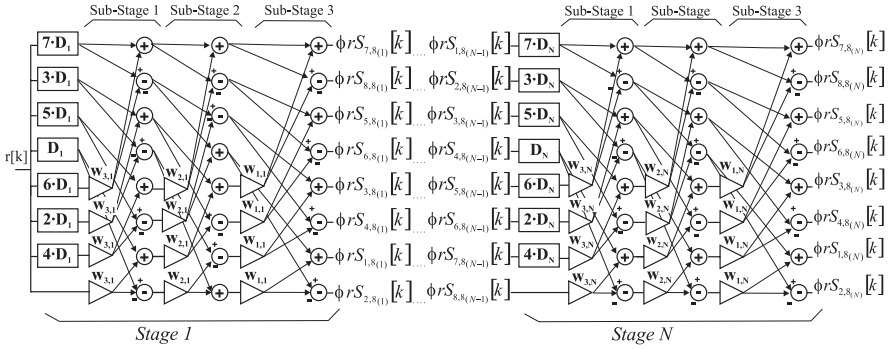
(b) FFT algorithm of an 8 point DFT.

Figure 2: Similarities between *M*-ESSC and FFT.

of sequences in the set. Furthermore, the stage outputs are always ordered in the same way. Therefore, the architecture of the generic correlator has a regular data flow, allowing a more efficient hardware implementation.



(a) Block diagram of an *8*-ESSC.



(b) Rearrangement of Figure 3.a, with the same geometry for every sub-stage.

Figure 3: Transformation of the M-ESSC structure proposed in [De Marziani et al. 2005].

3.3 Algorithm adaptation to the transmission scheme by interleaving

Since a macro-sequence is generated by interleaving the M sequences of the complementary set, the correlation can not be carried out directly with the sequences of the set, but with the interpolated versions of these sequences. The distance between two bits from the same sequence in the transmitted macro-sequence is M bits, so the interpolation factor is equal to M. The received signal is demodulated to extract the emitted macro-sequence. The demodulation is carried out asynchronously, by firstly sampling the received signal at a high acquisition rate, and then correlating it with the symbol used in the modulation. Let $O_f = \frac{f_a\ (acquisition\ frequency)}{f_e\ (emission\ frequency)}$ be the over-sampling factor, and N_{SM} the number of carrier periods in the modulating symbol. The bits corresponding to a sequence $S_{i,M}$ are obtained every $M \cdot\ O_f \cdot\ N_{SM}$ samples, so it is necessary

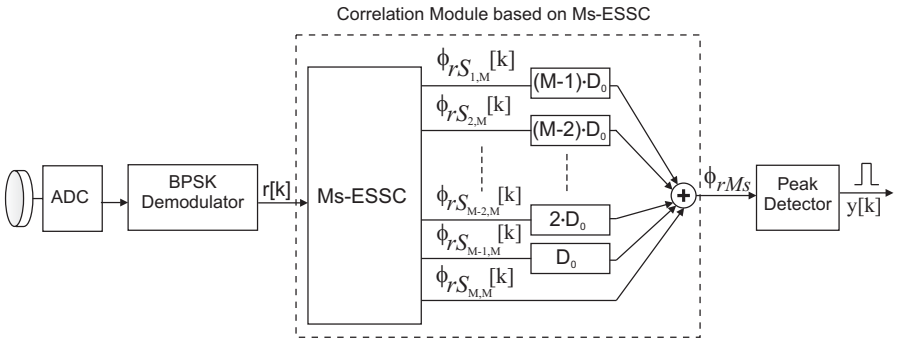


Figure 4: Block diagram of the detection of a macro-sequence using a *Ms*-ESSC.

to decimate the demodulated signal by the same factor prior to computing the correlations. This decimation can be easily achieved by multiplying the delays D_n from every stage by $M \cdot O_f \cdot N_{SM}$.

One advantage associated with this transmission scheme is that only one correlator is necessary to detect the signal from every emitter. The M-ESSC adapted to interleaving (hereafter *Ms*-ESSC) simultaneously performs the correlation of the input signal $r[k]$ with the M sequences of the set, providing the AC $\phi_{rS_{1,M}}$ with the first sequence of the set at the first branch of the *Ms*-ESSC, the AC $\phi_{rS_{2,M}}$ with the second sequence of the set at the second branch, and so on. Nevertheless, these ACs are not in phase, so, it is necessary to insert delays D_o at every branch of the *Ms*-ESSC, in order to compute the in-phase addition of the AC functions (4).

$$\begin{aligned}\phi_{rMs}[k] &= \phi_{rS_{1,M}}[k-(M-1)\cdot D_o] + \cdots + \phi_{rS_{M-1,M}}[k-D_o] + \phi_{rS_{M,M}}[k] \\ &= \sum_{i=1}^{M} \phi_{rS_{i,M}}[k-(M-i)\cdot D_o] \end{aligned} \quad (4)$$

Where D_o is a delay depending on the over-sampling factor O_f, and on the number N_{SM} of carrier periods in the modulating symbol: $D_o = O_f \cdot N_{SM}$. Figure 4 shows the detection scheme of a macro-sequence using *Ms*-ESSC.

4 Hardware implementation of the efficient correlator for interleaved M-CSS

4.1 Design strategy

The design of the efficient correlator for interleaved M-CSS has been based on five generic parameters in the VHDL specification. It is possible to configure the

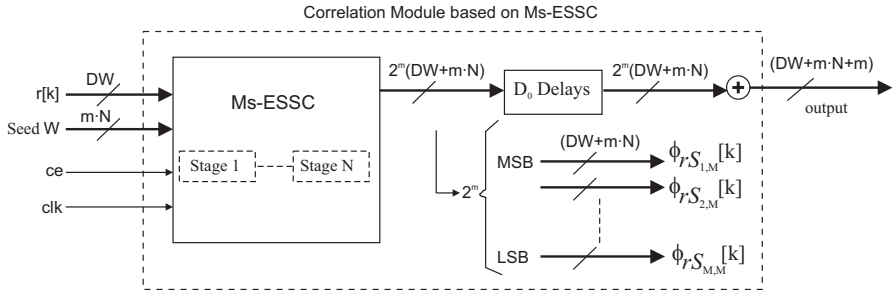


Figure 5: Input and output ports of the correlation module based on a *Ms*-ESSC structure.

number M of sequences in the set through the parameter m ($M = 2^m$). It is possible to select the number N of stages in the correlator, and therefore the length $L = M^N$ of the sequences to be processed. The data-width DW at the input can be configured as well, in order to have more accuracy in the obtained results. Considering the demodulation effect, the number N_{SM} of carrier periods used in the modulation and the over-sampling rate O_f can be also selected by the user.

Figure 5 shows the structure of the hierarchical root block. As the number of sequences in the set are defined during hardware synthesis, the number of output ports in the *Ms*-ESSC block is specified at the same moment. Since it is not possible in VHDL to have a generic number of ports, only one single port with variable size has been defined. This port consists of M *Ms*-ESSC outputs concatenated. The correlation with the first sequence of the set $\phi_{rS_{1,M}}$ comes from the $DW + m \cdot N$ most significant bits; and the correlation with the last sequence of the set $\phi_{rS_{M,M}}$ comes from the $DW+m\cdot N$ least significant bits. Note that every *Ms*-ESSC output $\{\phi_{rS_{i,M}},\ 1 \leq i \leq M\}$ requires $DW + m \cdot N$ bits, to avoid overflow in internal computations. These outputs are delayed according to (4). With this purpose, specific shift registers in Xilinx FPGA's architecture (SRL16 modules [Xilinx 2005]) has been used. Finally, the in-phase addition of the *Ms*-ESSC ouputs is performed.

The *Ms*-ESSC is based on N similar stages connected in cascade. Internally the design of these stages has been divided into four blocks, as shown in Figure 6.

- The first block implements the required delays for each stage, by using SRL16 modules.
- The second block rearranges the delayed outputs according to the bit-reversed order algorithm explained in Subsection 3.2.

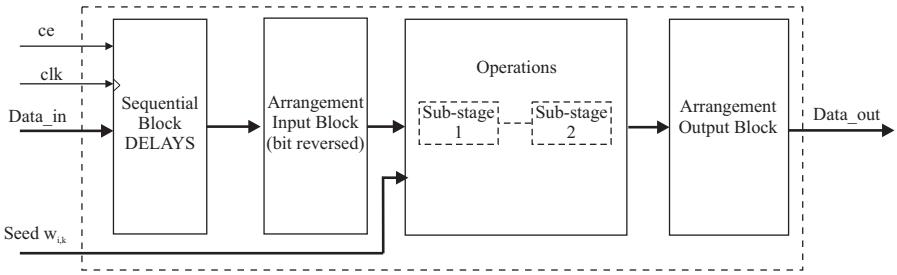


Figure 6: Internal configuration of a generic stage.

– A combinational block carries out the operations of the stage. Multiplication operations by coefficients $w_{i,k}$ have been reduced to additions and subtractions. Thereby, the values of these coefficients determine the configuration of the adders and subtractors at every stage (see Figure 7). Due to the new geometry of the efficient correlator, it is enough to implement one basic operation block sub-stage, and to repeat it in synthesis time, depending on the parameter m. Also, the introduction of pipeline registers between sub-stages increases the operation frequency of the system.

– Finally, an arrangement block orders the outputs from the operation block for its correct connection to the next stage.

It should be remarked that, at every stage, the number of bits required to store the partial results is incremented, so the larger delays of the algorithm should be placed at the first stages of the correlator, and the smaller ones at the last stages. Thus, a reduction of the total memory required by the correlation is obtained. Furthermore, the delays at the first stage of the correlator share the input of the system $r[k]$, so it is enough to implement the largest delay at this stage, $(M-1)\cdot D_1 \cdot DW$, and to allow the access to intermediate positions, as can be observed in Figure 8. It implies a reduction of $\left(\frac{M^2-3\cdot M+2}{2}\right)\cdot M \cdot O_f \cdot N_{SM} \cdot DW$ memory positions. It would be also possible to optimize the memory requirements by using range-propagation analysis.

4.2 Results

The resource requirements and maximum operating frequencies in the Ms-ESSC implementation depend on the length of macro-sequences. A macro-sequence of length $L_{Ms} = M \cdot L$ can be generated from complementary sets with different number M of sequences, keeping the final value L_{Ms} by changing the length L

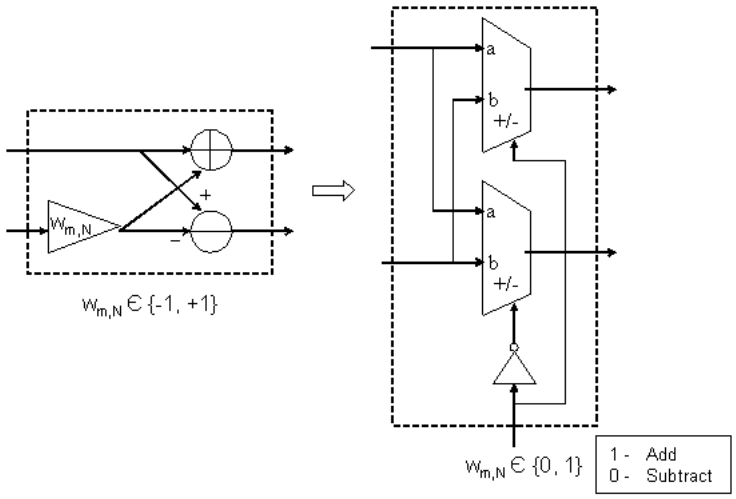


Figure 7: Implementation scheme of a multiplier, adder and subtracter for a stage of the *Ms*-ESSC.

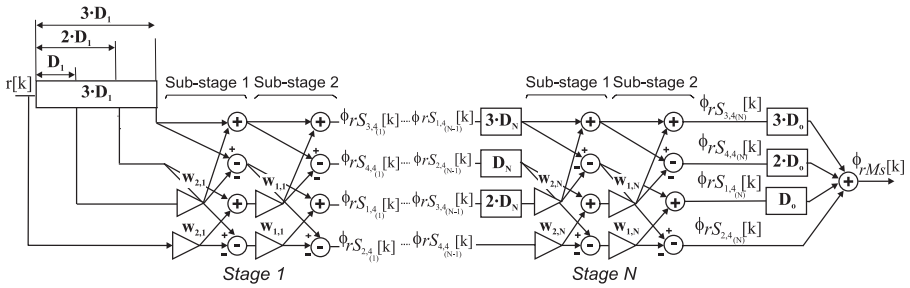


Figure 8: Structure of a correlation module based on a *4s*-ESSC.

of the sequences. In Table 2 the resource requirements of the correlation module for macro-sequences with length $L_{Ms} = 64$ are shown. It have been generated from *2*-CSS with length $L = 32$, *4*-CSS with length $L = 16$, and *8*-CSS with length $L = 8$. In Table 3 the resource requirements of the correlation module for macro-sequences with length $L_{Ms} = 256$ are shown, generated from *2*-CSS with length $L = 128$, *4*-CSS with length $L = 64$, and *16*-CSS with lenght $L = 16$. In both Tables, a single carrier period $N_{SM} = 1$ has been used, with an over-sampling factor $O_f = 10$, and the number of bits of the input signal is $DW = 8$. The correlation system has been implemented in a Spartan3 xc3s1500-5fg676 FPGA [Xilinx 2005], and results are given after place and route. It can be

xc3s1500-5fg676	$M = 2,\ L = 32$	$M = 4,\ L = 16$	$M = 8,\ L = 8$
Slices	323	375	804
Luts	432	522	1079
IOBs	29	28	27
Flip-Flops	123	160	254
Max. Freq. $f_{FPGA} = f_a$	106.135 MHz	81.473 MHz	73.438 MHz
Max. Throughput $(Msamples/s)$	106.135	81.473	73.438

Table 2: Resources required by the correlation of macro-sequences with length $L_{Ms} = 64$.

xc3s1500-5fg676	$M = 2,\ L = 128$	$M = 4,\ L = 64$	$M = 16,\ L = 16$
Slices	985	1131	2980
Luts	1152	1386	3734
IOBs	33	32	30
Flip-Flops	183	270	688
Max. Freq. $f_{FPGA} = f_a$	110.011 MHz	92.920 MHz	62.477 MHz
Max. Throughput $(Msamples/s)$	110.011	92.920	62.477

Table 3: Resources required by the correlation of macro-sequences with length $L_{Ms} = 256$.

observed that, for equally long macro-sequences, less resources are required for cases generated from M-CSS with low number M of sequences. In regard to the operating frequency, the more sequences M in the set, the harder is to compute the combinational addition of the M output branches of the Ms-ESSC, so the operating frequency decrease. Also, by comparing Tables 2 and 3, it is possible to verify that large macro-sequences have high requirements.

Finally, a performance comparison between Ms-ESSC and a straight-forward matched filter implementation has been achieved. Figure 9 shows the hardware implementation scheme of a straight-forward correlator based on internal BRAM memory [Pérez et al. 2006]. The demodulated signal is acquired with a frequency f_a, and stored in a buffer of size $L_{Ms} \cdot O_f \cdot N_{SM}$, that has been implemented in BRAM blocks [Xilinx 2005]. This sampling buffer is read every $f_{FPGA} = L_{Ms} \cdot f_a$. At each reading, a sample is added or subtracted with the previous accumulated result, depending on the used macro-sequence

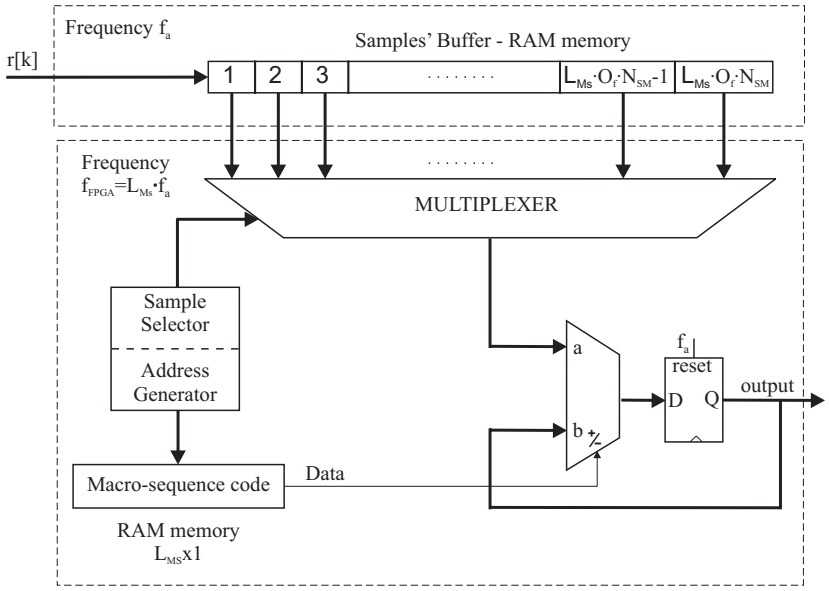


Figure 9: Sequential implementation of a straight-forward correlator based on internal BRAM memory.

symbol. Whenever a new sample is received, the accumulated value is reset in order to compute a new correlation. The access to the sampling buffer is carried out in gaps of $O_f \cdot N_{SM}$ positions, by taking into account the demodulation effect. To achieve a comparison between this structure and the correlation module based on a *Ms*-ESSC, a macro-sequence of length $L_{Ms} = 64$ has been generated from *2*-CSS of length $L = 32$. In Table 4 the requirements for the straight-forward implementation are shown. Correlation results are obtained every $t_a = t_{FPGA} \cdot L_{Ms} = 8.29\ ns \cdot 64 = 531.14\ ns$, being $t_a = \frac{1}{f_a}$ and $t_{FPGA} = \frac{1}{f_{FPGA}}$. Nevertheless, the *2s*-ESSC implementation provides a new correlation result every FPGA clock cycle (9.42 ns), assuring real-time operation, although it requires more resources. To increase the operation frequency of the straight-forward implementation, tasks have to be overlapped temporarily, which implies introducing registers to store intermediate data. Therefore, the amount of required resources increases, approaching that required by the *2s*-ESSC implementation. Thus, it can be stated that a *Ms*-ESSC implementation is more suitable for real-time correlation of macro-sequences generated from *M*-CSS.

xc3s1500-5fg676	M=2, L=32
Slices	49
Luts	85
IOBs	26
Flip-Flops	59
Max. Freq. f_{FPGA}	120.496 MHz
Max. Freq. f_a	1.88 MHz
Max. Throughput $(Msamples/s)$	1.88

Table 4: Resources required by a straight-forward implementation.

5 Application example

The suitability of the *Ms*-ESSC has been tested by including it in the ultrasonic pulse compression system shown in Figure 10. Four different macro-sequences Ms^0, Ms^{11}, Ms^{20} and Ms^{31} with length $L_{Ms} = 256$ are obtained, by using four different *4*-CSS, generated with seeds $0, 11, 20$ and 31 respectively. Each one of these macro-sequences is assigned to a different emitter from which is transmitted once modulated. Figure 11 depicts the position of the emitters with regard to the receiver in a xy-plane; although three coordinates are given (x: lenght, y: width, z: height). Four REGAL-RH16E [REGAL 2006] high power speakers have been used to continuously transmit the four macro-sequences in a simultaneous way. These speakers are often used in the audible frequency range from 280 Hz to 20 kHz. Nevertheless, after analysing the spectral characteristics of theses speakers, it was observed that the speakers can also operate at a 25 kHz ultrasonic frequency. Thus, the macro-sequences have been BPSK modulated by using a symbol with one period of a 25 kHz squared signal.

All the emitted macro-sequences are received by a condenser microphone capsule Avisoft Bioacoustics CM16 [Avisoft 2006], which has a flat frequency response between 15 kHz and 180 kHz. Afterwards, the received signal is amplified and digitalized at a sampling frequency of 500 kHz, so $O_f = 20$; then, it is demodulated by correlation with the modulation symbol. Four *4s*-ESSCs simultaneously correlate the demodulated signal with the four ideal emitted macro-sequences. As was explained in Subsection 3.3, it is necessary to add a set of additional delays at the output of every *4s*-ESSC in order to compute the in-phase sum of the AC functions. When the last sample from a macro-sequence is processed, a peak in the correlator for this macro-sequence is obtained. A peak detector confirms the peaks exceeding a static threshold U_e, assuming that there does not exist a higher peak in the neighborhood.

Figure 12 shows the results obtained at the output from each correlator. A

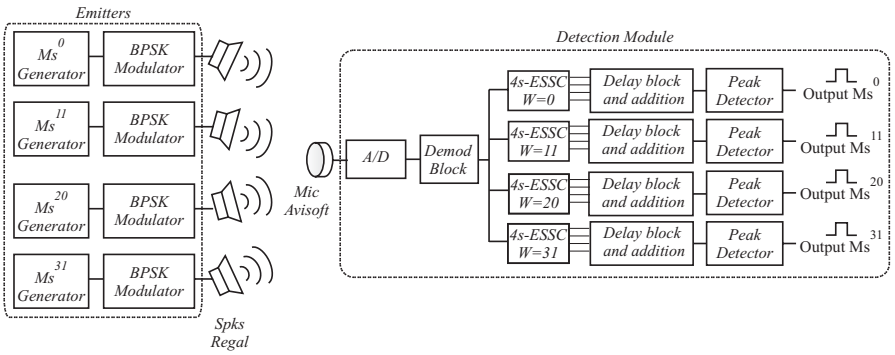


Figure 10: Block diagram of a macro-sequences detection system.

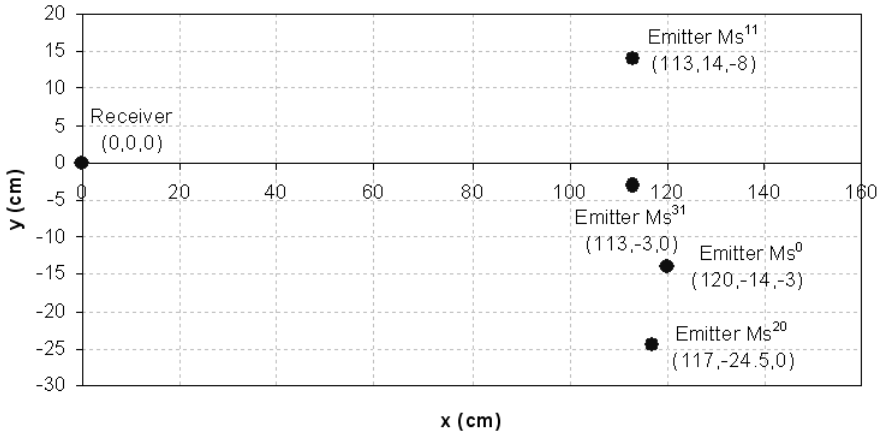


Figure 11: Diagram of the transducer position in the experimental tests.

circle has been introduced to mark the values confirmed by the peak detector. Every time a macro-sequence is completely received, a new peak is obtained in the corresponding output. Separation between peaks from one macro-sequence is equal to the emission interval $t_e = 10.24$ ms. The detection system is able to distinguish between the four macro-sequences, despite they are received practically overlapped in an environment with an approximate signal-to-noise ratio of $SNR = 18\ dB$. Figure 13 represents an enlargement of Figure 12, where the distance between peaks can be observed. The presence of sidelobes caused by the interleaving process can be reduced by applying the post-processing algorithm detailed in [De Marziani et al. 2006].

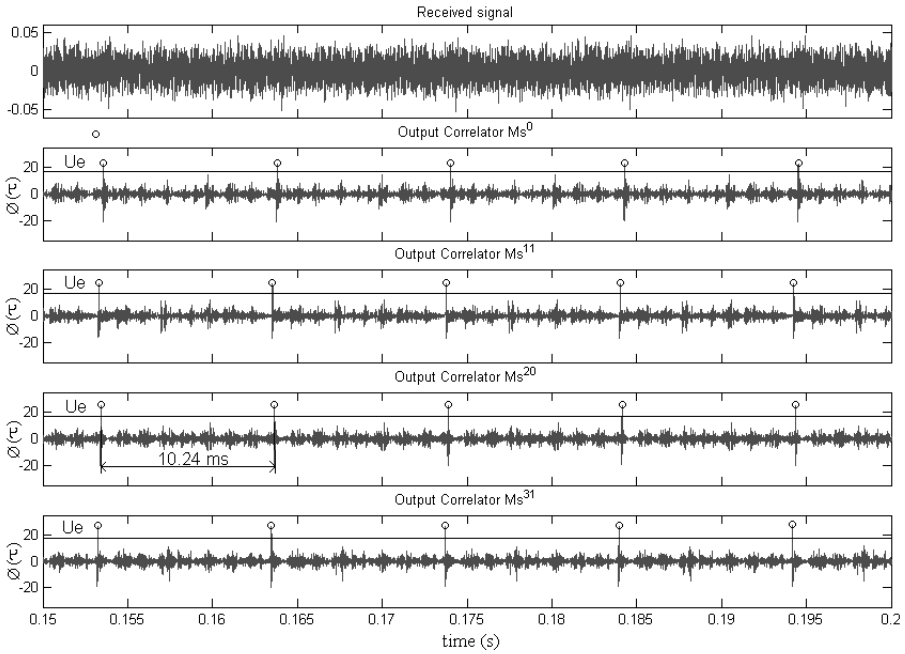


Figure 12: Results obtained from the macro-sequences detection system.

6 Conclusions

A generic hardware implementation of an efficient correlator for interleaved complementary sets of sequences has been presented. The design has been developed as a parameterized module, able to be adapted to the requirements from different sensor systems. By synthesizing the design, the number of sequences of the complementary set, their length, the sampling factor, the number of periods of the symbol used in the modulation, and the data width can be changed.

The performance of the correlator has been tested on a Xilinx Spartan3 FPGA. Furthermore, it has been included on an ultrasonic sensory system to verify the detection of simultaneous ultrasonic emissions in real-time.

Acknowledgements
This work has been possible thanks to the Madrid Community (FPI grant and ANESUS project: CAM-UAH2005/016), and from the Spanish Ministry of Science and Technology (PARMEI: DIP2003-08715-C02-01).

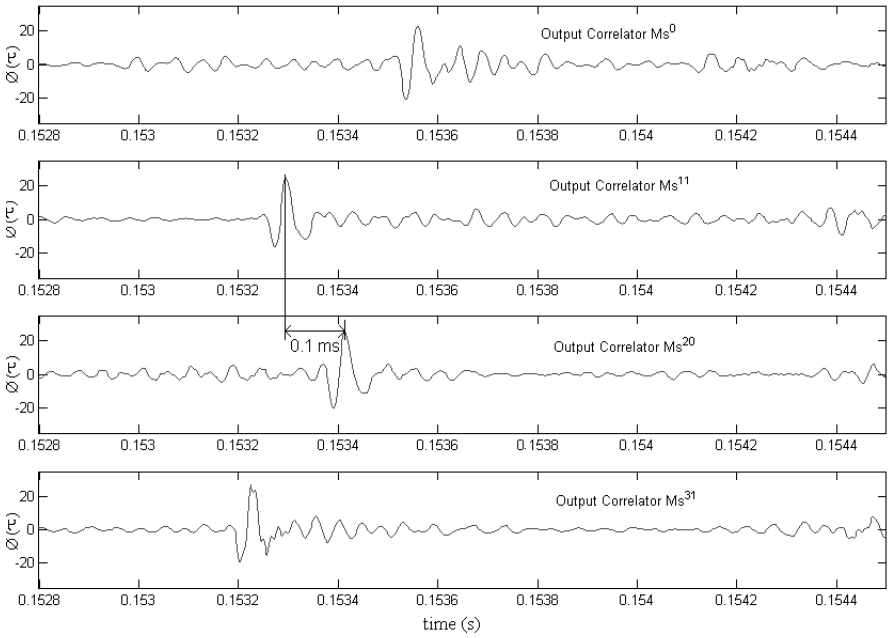


Figure 13: Enlargement of Figure 12.

References

[Audenaert et al. 1992] Audenaert, K., Peremans, H., Kawahara, Y., Campenhout, J.: "Accurate Ranging of Multiple Objects using Ultrasonic Sensors"; Proc. IEEE International Conference on Robotics and Automation, Vol. 2, (May 1992), 1733-1738.

[Avisoft 2006] AVISOFT BIOACOUSTICS: "Avisoft Bioacoustics UltraSoundGate microphone capsule CM16"; Product Spec. (2006).

[Álvarez et al. 2004] Álvarez, F., Ureña, J., García, J., Mazo, M., De Marziani, C., Hernández, A.: "A comparative analysis of two modulation schemes for the efficient transmission of complementary sequences in a pulse compression system"; Proc. IADT-tcn2004 Int. Conf. on Telecommunications and Computer Networks, (2004), 26-30.

[Budisin et al. 1989] Budisin, S., popovic, P., Indjin, I.: "Designing radar signals using complementary sequences"; Proc. on Radar, 87, (1989), 593-597.

[De Marziani et al. 2005] Marziani, C., Ureña, J., Hernández, A., Mazo, M., Álvarez, F., García, J., Donato, P.: "Modular Architecture for Efficient Generation and Correlation of Complementary Sets of Sequences"; IEEE Trans. on Signal Processing, Accepted for its publication, Ref. SP33377 (2005).

[De Marziani et al. 2006] Marziani, C., Ureña, J., Hernández, A., Mazo, M., García, Jiménez A., Villadangos, J., Pérez, M., Álvarez, F.: "Inter-Symbol Interference Reduction on Macro-Sequences Generated from Complementary Sets of Sequences"; IEEE Proc. 32nd Annual Conference of the IEEE Industrial Electronics Society (November 2006).

[Golay 1961] Golay, M.: "Complementary Series"; IRE Trans. On Inform. Theory, Vol. IT-7, N 2, (1961), 82-87.

[Golomb and Scholtz 1965] Golomb, S., Scholtz, R.: "Generalized Barker sequences"; IEEE Trans. Inf. Theory, Vol. IT-11, N 4, (1965), 533-537.

[Hahm et al. 1997] Hahm, M., Friedman, G., Titlebaum, E.:"A comparison of Analog and Digital Circuit Implementations of Low Power Matched Filters for Use in Portable Wireless Communication Terminals"; IEEE Trans. on circuits and systems II: analog and digital signal processing, Vol. 44, N 6, (June 1997), 498-506.

[Harmuth 1969] Harmuth, H.: "Application of Walsh Functions in Communications"; IEEE Spectrum, Vol. 6, (1969), 82-91.

[Jörg and Berg 1998] Jörg, K., Berg, M.: "Sophisticated Mobile Robot Sonar Sensing with Pseudo-Random Codes"; Robotics and Autonomous Systems, Vol. 25, (1998), 241-251.

[Oppenhein et al. 1999] Oppenhein, A., Shafer, R., Buck, J.: "Discrete-Time Signal Processing"; Prentice Hall, 2^{nd} edition, (1999), ISBN: 0137549202.

[Pérez et al. 2006] Pérez, M., Hernández, A., Ureña, J., De Marziani, C., Jiménez, A.: "FPGA-based Implementation of a Correlator for Kasami Sequences"; Proc. 11th IEEE International Conference on Emerging Technologies and Factory Automation ETFA'06, (2006), 1141-1144.

[REGAL 2006] REGAL ELECTRONIC: "Regal Electronic Speakers"; Product Spec. (2006).

[Sarwate and Pursley 1980] Sarwate, D., Pursley , M.: "Crosscorrelation Properties of Pseudorandom and Related Sequences"; Proc. IEEE, Vol. 68, N 5, (1980), 593-619.

[Tseng and Liu 1972] Tseng, C., Liu, C.: "Complementary Sets of Sequences"; IEEE Trans. Inf. Theory, Vol. IT-18, N 5, (1972), 644-652.

[Xilinx 2005] Xilinx, Inc.: "Spartan-3E FPGA Family: Complete Datasheet"; Product Documentation, (November 2005).

Efficient Real-Time Correlator for LS Sequences

M. C. Pérez[1], J. Ureña[1], A. Hernández[1], A. Jiménez[1]
[1]Department of Electronics
University of Alcalá
Alcalá de Henares, Spain
Email: carmen@depeca.uah.es

W. P. Marnane[2], F. Álvarez[3]
[2]Dept. of Electrical & Electronic Engineering
University College Cork, Cork, Ireland
Email: marnane@ucc.ie
[3] Dept. of Electronics & Electromechanical Engineering
University of Extremadura, Cáceres, Spain
Email: fafranco@unex.es

***Abstract*— The cross-correlation function and the side-lobes of the auto-correlation function of LS codes are zero in a certain vicinity of the zero shift. Therefore, the effects of the Inter-Symbol-Interference and Multiple-Access-Interference that appears in CDMA and multi-sensor systems are mitigated. Conventionally, the detection of these sequences has been achieved by means of straight-forward matched-filter correlators. In this paper, a new correlator which significantly reduces the total number of operations to be performed is proposed, allowing real-time operation.**

I. Introduction

Pulse compression sensor systems and CDMA (Code Division Multiple Access) systems are interference limited. There exists Inter-Symbol-Interference (ISI) due to the non zero auto-correlation (AC) sidelobes of the used sequences. They also have Multiple-Access-Interference (MAI) due to the non zero cross-correlations (CC) values. To avoid the effects of both ISI and MAI, the sidelobes of the AC and the CC values should be as small as possible.

Hence, a lot of research efforts have been devoted to finding sequences which fulfill both conditions as far as possible. Barker codes [2] have been widely used due to their good aperiodic AC. However, their maximal length is limited, so it is not possible to detect them in case of high noise. Furthermore, there are no Barker codes with low CC among them. Pseudo-random sequences such as m-sequences, Gold codes or Kasami codes [3], exhibit non zero off-peak AC and CC values in the case of asynchronous transmissions. Another possibility is Golay pairs [4], which have good aperiodic correlation properties, and their length is not limited. The detection of Golay sequences can be performed by means of efficient correlation algorithms which notably decrease the computational load and hardware complexity [5]. Nevertheless, they provide only two mutually orthogonal pairs, which is not useful for multi-user environments. Complementary sets of M sequences (M-CSS) [6] are a generalization of Golay codes containing more than two sequences. The elimination of the constraint in the number of sequences of a set yields on a high process gain and also M mutually orthogonal sets. To achieve these goals, it is necessary to add the AC functions of the sequences of the set, or the CC functions of the corresponding sequences in the M sets, respectively. The problem is that, in many systems, transducers have limited bandwidths, so it is not possible to transmit or receive the *M* sequences of the set simultaneously. Unfortunately, it implies an undesired increase of both ISI and MAI.

Loosely Synchronized (LS) codes [7] exhibit an Interference Free Window (IFW), where the aperiodic AC sidelobes and CC values become zero. Consequently, ISI and MAI are completely reduced if the maximum transmission delay is less than the length of the IFW. Usually, the correlation of these codes is carried out by means of straightforward matched filter implementations [8]. These implementations provide large processing cost; and, in case of a large amount of data to be processed, real-time operation can not be possible without the aid of high complexity hardware.

The contribution of this paper is to propose an *Efficient LS correlator (ELSC)*, which significantly reduces the number of operations performed, in comparison with an straight-forward matched filter implementation. The presented ELSC is designed for the correlation of LS codes, generated from Golay complementary pairs, as it is explained in [7]. Also, the practical implementation of the ELSC in a Field Programmable Gate Arrays (FPGA) has been carried out.

The paper is organized as follows. Section II introduces the family of LS codes. In Section III the Efficient Correlator of LS sequences is presented. Section IV shows the hardware implementation of the correlator and some results. In Section V tests with real signals are considered. Finally, conclusions are outlined in Section VI.

II. LS Codes

In [7], a systematic method to construct LS codes by using Golay complementary sequences is presented. It basically consists of using two orthogonal Golay pairs of length N, and of linking the sequences of these pairs directly, or negated, depending on the coefficients of a $(P \times P)$ Hadamard matrix. Also, a set of W_0 zeros has to be inserted in the centre of the LS code G_k. The total length of the LS codes is given by $L = KN + W_0$, where $K = 2P$ is the number of available codes with orthogonal properties in the IFW (with $K = 2^n$, n $\in \mathbb{N}$). The correlation properties of LS sequences $(G_k = [g_{k,1},\ g_{k,2},\ \cdots,\ g_{k,L}]; 1 \leq k \leq K)$ are given by (1). It can be noted that the AC function is obtained when $k_1 = k_2$, whereas the CC function when $k_1 \neq k_2$.

1-4244-0755-9/07/$20.00 ©2007 IEEE

$$R_{G_{k_1},G_{k_2}}[m] = \sum_{i=0}^{L-m-1} G_{k_1}[i]G_{k_2}[i+m] =$$

$$\begin{cases} K \cdot N, & \text{for} \quad m = 0, & k_1 = k_2 \\ 0, & \text{for} \quad 1 \le |m| \le W_0, & k_1 = k_2 \\ 0, & \text{for} \quad 0 \le |m| \le W_0, & k_1 \ne k_2 \end{cases} \quad (1)$$

Prior to describe the proposed ELSC, it is necessary to give more details about the generation method of LS codes presented in [7]. Also, the notation $LS(N, P, W_0)$ mentioned in [9] has been adopted.

Firstly, two orthogonal Golay pairs (c_0, s_0) and (c_1, s_1) of length *N* are defined. The AC and CC properties of these pairs should satisfy (2):

$$R_{c_i,c_j}([m] + R_{s_i,s_j}[m] = \begin{cases} 2 \cdot N, & \text{for} \quad m = 0, & i = j \\ 0, & \text{for} \quad m \ne 0, & i = j \\ 0, & \text{for} \quad \forall m, & i \ne j \end{cases} \quad (2)$$

Golay pairs (c_0, s_0) and (c_1, s_1) are linked depending on the coefficients of a Hadamard matrix H, and on a vector π. $H = [h_{i,j}]$ is a $P \times P$ Hadamard matrix, whose $h_{i,j} = \pm 1$ elements determine the final polarity of the Golay pairs. The vector $\pi = [\pi_1, \cdots, \pi_P]$, $\pi_k \in \{0, 1\}$ denotes a binary expansion of an arbitrary integer n, $0 \le n < 2^P$ so that $n = \sum_i \pi_i \cdot 2^i$. This vector π indicates which Golay pair, (c_0, s_0) or (c_1, s_1), is taken into account at every moment. Furthermore, a second vector $\pi^* = [\pi_1^*, \cdots, \pi_P^*]$, $\pi_k* = \pi_k + 1(mod\ 2)$ for each $1 \le k \le P$, has to be considered. The first P sequences of the set $G_1, \cdots, G_P$ are constructed as it is shown in (3). Whereas the next $G_{P+1}, \cdots, G_K$ are obtained when the vector π in (3) is substituted by its complement π^*. For brevity, polynomial notation has been used. In this notation, any bipolar sequence $a = [a_0,\ a_1,\ \cdots,\ a_{N-1}]$, is simply replaced by the polynomial $A(z) = \sum_{n=0}^{N-1} a_n z^n$, in the Laurent Polynomial ring $\mathbb{Z}[z, z^{-1}]$.

$$G_k(z) = \sum_{i=1}^{P} h_{k,i}[C_{\pi_i}(z) + z^{PN+W_0} S_{\pi_i}(z)]z^{(i-1)N} \quad (3)$$

Thus, a set of K LS codes with length $L = KN + W_0$ is obtained. For periodic emissions it is necessary to insert a guard interval of, at least, W_0 zeros between sequences, preventing them for overlapping. In these cases $L = KN + 2W_0$. The LS codes generated exhibit an IFW of length $W = min\ \{2N - 1, 2W_0 + 1\}$, considering $W_0 = N - 1$ in most of cases [7][9].

Fig. 1 shows the aperiodic AC of a $LS(16, 2, 15)$ sequence; whereas in Fig. 2 can be seen the aperiodic CC among diferent $LS(16, 2, 15)$ codes from the same set.

III. Efficient LS Correlator

When longer LS sequences have to be processed in real-time, the increasing in the computational load demands the use of efficient correlators, able to perform the detection

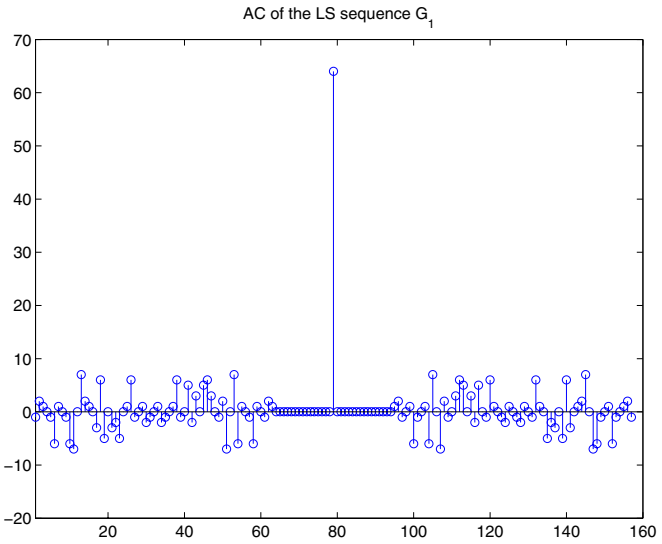


Fig. 1. AC function for a sequence LS(16,2,15).

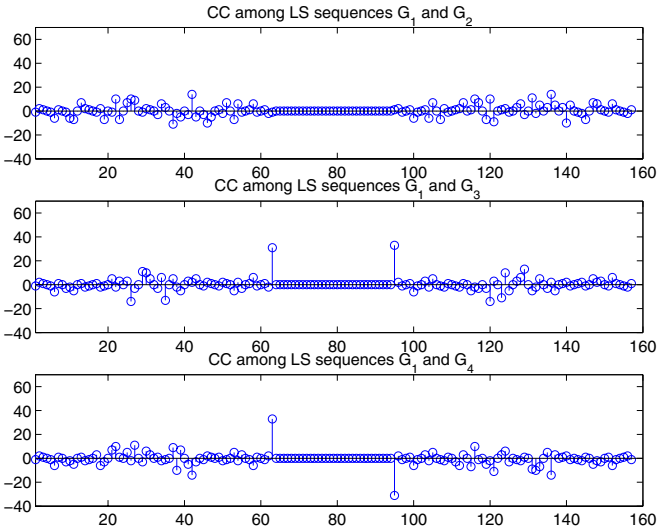


Fig. 2. CC function among the sequences of a set generated as LS(16,2,15).

of these sequences with affordable computational cost. In this paper, an efficient correlator for LS sequences (ELSC) has been developed; which decreases the total number of operations performed, in comparison with a straight-forward matched filter implementation. This correlator can be easily implemented on reconfigurable hardware to achieve real-time operation.

The proposed ELSC correlator exploits the properties of the Golay codes to simplify the correlation process. Golay pairs, (c_0, s_0) and (c_1, s_1), that form the LS sequences can be correlated by means of the Efficient Golay Correlator (EGC) developed in [5]. Later, the corresponding outputs of the EGCs are delayed and added or subtracted, depending on the values of the vector π and the Hadamard matrix H.

The block diagram of Fig. 3 shows the EGC algorithm structure. It is composed by a set of M similar stages, where $N = 2^M$, with $M \in \mathbb{N} - \{0\}$, is the length of the Golay

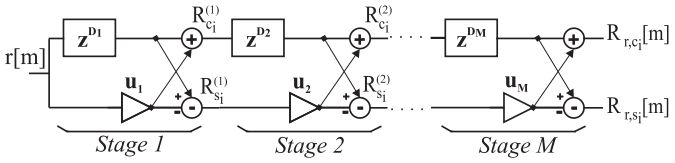


Fig. 3. Block Diagram of the Efficient Golay Correlator.

pairs. Every stage is composed of a delay module z^{D_i}, where $D_i = 2^{Ps}$, and Ps is any permutation of the numbers $0, 1, ..M-1$; a multiplier u_i, where $U = [u_0,\ u_1, \cdots,\ u_{M-1}]$ is the generation seed of the Golay pair; an adder; and a subtracter. It is important to note that, since the coefficients u_i are binary, multiplications are reduced to additions and subtractions. Starting from any input signal $r[m]$, the EGC provides two outputs, $R_{r,c_i}[m]$ and $R_{r,s_i}[m]$; which are the correlation between the input signal $r[m]$, and the sequences of the Golay pair (c_i, s_i). The number of multiplications in the EGC is equal to $log_2 N$, whereas in the straightforward matched filter implementation it would be N. Also, the number of additions is reduced from the $N-1$ operations in the straightforward to $2 \cdot log_2 N$ in the EGC.

Considering the EGC, the scheme has to be adapted to the features of LS codes. Since two orthogonal pairs, (c_0, s_0) and (c_1, s_1), form the LS codes, two EGC are needed. One EGC carries out the correlation among the input sequence $r[m]$ and the Golay pair (c_0, s_0), generated with seed U_0. It provides the outputs $R_{r,c_0}[m]$ and $R_{r,s_0}[m]$. The other EGC performs the correlation with the Golay pair (c_1, s_1), generated with seed U_1, and gives the outputs $R_{r,c_1}[m]$ and $R_{r,s_1}[m]$.

Vectors π or π^* used to generate the LS sequence determine which correlation outputs, $R_{r,c_0}[m]$, $R_{r,s_0}[m]$, $R_{r,c_1}[m]$ or $R_{r,s_1}[m]$, have to be delayed. Note that the order of the delays in (3) has to be interchanged for the correlation. Therefore, the output corresponding with the correlation between the input signal and the sequence of the Golay pairs that firstly appear in the LS code, has to be delayed by $z^{PN+W_0+(P-1)N}$. Later, the delayed outputs are added or subtracted, according to the column of the Hadamard matrix used in the LS code generation. Eq. (4) summarizes the operations performed by the ELSC. Furthermore, Fig. 4 depicts the block diagram of the ELSC.

$$R_{G_k} = \sum_{i=1}^{P} h_{k,i}[z^{PN+W_0} R_{C_{\pi_i}}(z) + R_{S_{\pi_i}}(z)] z^{(P-i)N} \quad (4)$$

This optimization presents some advantages compared to the implementation of a classical straight-forward correlation. Table I shows the operations carried out by the two correlation schemes. As has been stated before, multiplications are reduced to additions and subtractions, due to the binary nature of their coefficients. Regarding the notation, DW is the number of bits of the input signal; and $O_s = \frac{acquisition\ frequency}{emission\ frequency}$ is the over-sampling factor used when the sequences are acquired. It can be observed that the number of operations is

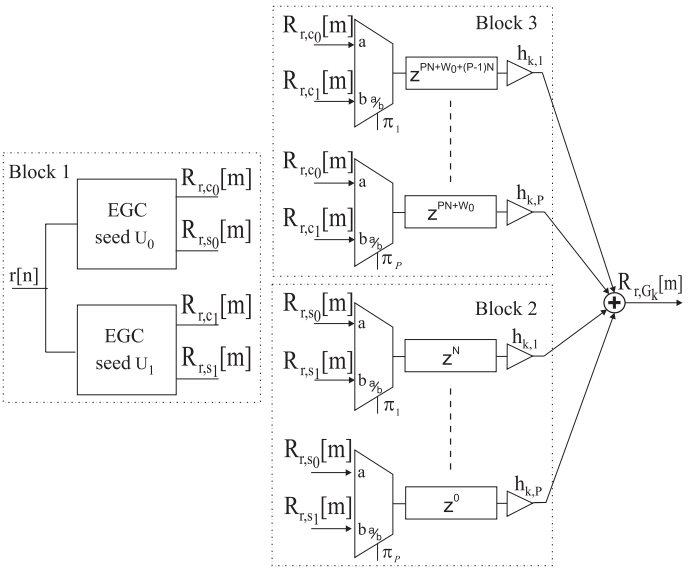


Fig. 4. Block Diagram of the Efficient Loosely Synchronized Correlator.

significantly reduced in the case of the ELSC implementation. Nevertheless, the total number of memory bits to store the data in the ELSC, is larger than the corresponding in the straightforward implementation, as can be seen in Table II. To expound further, Fig. 5 and Fig. 6 depict the information given by Tables I and II respectively. To generate both figures, a set of LS codes with $K = 16$ sequences has been used, with $DW = 8$, and $O_s = 10$. The length N of the initial Golay sequences can take values $N = [8,\ 16,\ 32,\ 64,\ 128,\ 256,\ 512,\ 1024]$; and the number of zeros in the centre of the LS code is $W_o = N - 1$.

IV. ELSC Implementation

A. Design Strategy

A generic hardware implementation in a FPGA of the ELSC has been developed. The correlator parameters can be changed through synthesizing the design. Therefore, it is possible to configure the number of LS codes availables in the set through

TABLE I

Operations to perform for the correlation of LS codes, using an straightforward implementation and an ELSC.

Implementation	Multiplications	Additions
Straight-forward	$L = KN + W_0$	$L - 1 = KN + W_0 - 1$
ELSC	$2log_2 N + K$	$4log_2 N + 1$

TABLE II

Memory required to store the data in the correlation of LS codes, using an straightforward implementation and an ELSC.

Implementation	Memory requirements
Straight-forward	$L \cdot O_s \cdot DW + L + (DW + log_2 L)$
ELSC	$2 \cdot O_s \cdot [DW \cdot \frac{N}{2} + (DW+1) \cdot \frac{N}{4} + (DW+2) \cdot \frac{N}{8} + \cdots (DW + log_2 N - 1)] +\ O_s \cdot (DW + log_2 N - 1) \cdot (\frac{K}{2} \cdot N + W_0) \cdot \frac{K}{2} + \sum_{i=1}^{\frac{K}{2}-1} [2 \cdot N \cdot i]$

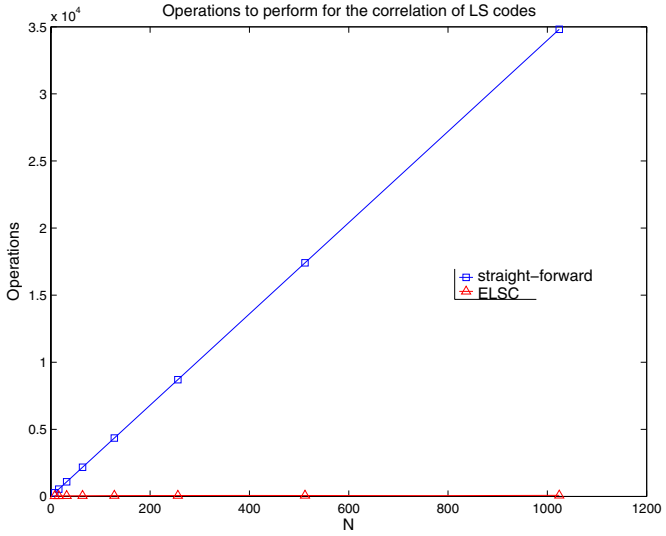


Fig. 5. Operations versus N performance comparison of an ELSC and a straight-forward matched filter implementation of a correlator for LS codes. $K = 16$ users are supported, with $DW = 8$ and $O_s = 10$.

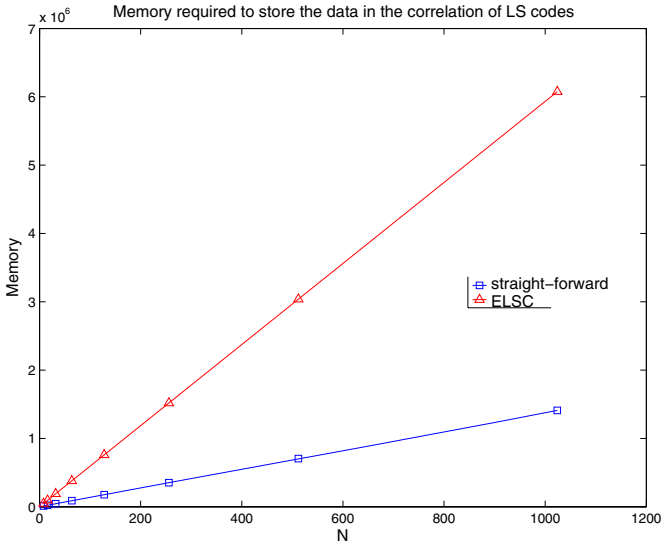


Fig. 6. Memory versus N performance comparison of an ELSC and a straight-forward matched filter implementation of a correlator for LS codes. $K = 16$ users are supported, with $DW = 8$ and $O_s = 10$.

the parameter n ($K = 2^n$). The number of zeros W_0 inserted in the centre of the LS code can be selected as well. It is possible to configure the number M of stages in the EGC, and therefore the length $N = 2^M$ of the initial Golay codes. The data-width DW can be selected, in order to have more accuracy in the obtained results. Finally, the the over-sampling rate O_s can be also selected by the user.

Fig. 7 shows the structure of the hierarchical root block for the ELSC, where $r[m]$ is the input signal; U_0 and U_1 are the generation seeds of the Golay pairs; Pi and Had are the vector π and the column of the Hadamard matrix H used in the generation of the LS codes; and finally R_{r,G_k} is the correlation

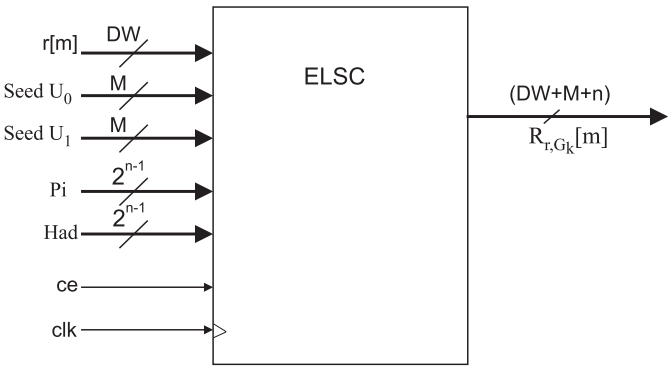


Fig. 7. Input and output ports of the proposed ELSC structure.

function between the input signal $r[m]$ and the considered LS sequence G_k. Note that the output R_{r,G_k} requires $DW+M+n$ bits if overflow is avoided in internal operations.

The internal architecture of the ELSC is illustrated in Fig. 8. It is based on six different modules. The first one implements the EGC (see Fig. 3), and it is based on M similar stages. As has been mentioned before, the number of stages is configurable in order to increase the adaptability of the system. Every stage is divided into two modules: a sequential and a combinational one. The sequential block contains the specific delay of the stage. Note that, at every stage, the number of bits required to store the partial results is increased. Hence, to reduce the total memory required by the correlator, the larger delays should be placed at the first stages, and the smaller ones at the last stages. On the other hand, all these delay elements have been increased by O_s, to consider the sampling factor used in the acquisition of the input signal. As a result, $D = [D_1,\ D_2,\ \cdots,\ D_M] = [2^{M-1}\cdot O_s,\ 2^{M-2}\cdot O_s,\ \cdots,\ 2^0\cdot O_s]$. These delays have been implemented by making use of specific shift registers in Xilinx FPGA's architecture (SRL16 modules [10]). The combinational module carries out the operations of the stage. The binary values of the seeds U_0 and U_1 determine the final configuration of the adders and subtractors in the $i-th$ stage.

The second block, called MUX, generates a set of P multiplexers governed by the input Pi. Every multiplexer has the same two inputs, a and b; which can be connected either with (R_{r,c_0}, R_{r,c_1}), or with (R_{r,s_0}, R_{r,s_1}). If $Pi(i) =$'0' then the multiplexer output is a, whereas if $Pi(i) =$'1' the multiplexer output is b.

When the outputs of the EGCs are selected, they have to be delayed according to (4). There are two different delays blocks: one implements P delays, all of them with value $z^{(PN+W_0)}$; the other contains the $[z^0,\ z^1,\ \cdots,\ z^{(P-1)N}]$ delays. Both blocks use the basic module SRL16 of Xilinx FPGA's architectures.

The Hadamard block determines the polarity of the delayed EGC's outputs. $Had(i) =$'0' denotes negating the input sequence i; while $Had(i) =$'1' means that the output sequence i and the input sequence i are both equal.

Finally, the P outputs of both Hadamard blocks are added

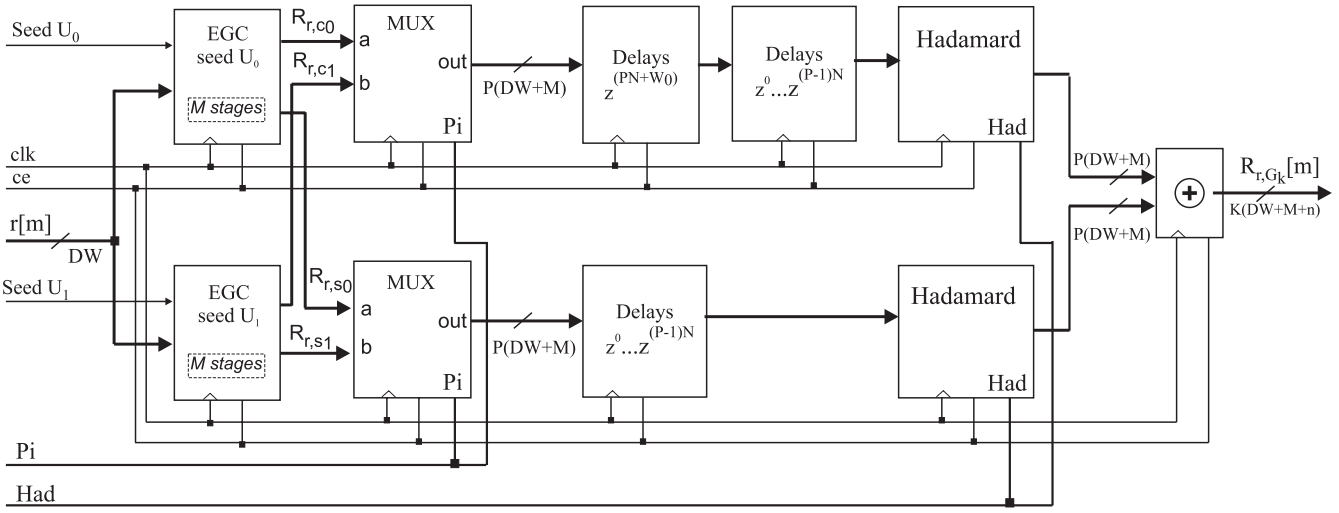


Fig. 8. Internal architecture of the ELSC.

to obtain the correlation output R_{r,G_k}.

B. Performance of the Hardware Implementation

The result requirements and maximum frequencies in the ELSC implementation depend on the (N, P, W_0) parameters of the LS code. In Table III the resource requirements of $LS(8,2,7)$, $LS(16,2,15)$ and $LS(8,4,7)$ are shown. An oversampling factor $O_s = 10$ has been used, with a data-width $DW = 8$. The correlation system has been implemented in a Spartan3 xc3s1500 FPGA by Xilinx [10].

A performance comparison between the ELSC and a straight-forward matched filter implementation has been achieved. Fig. 9 depicts the hardware implementation scheme of a straightforward correlator based on internal BRAM memory [11]. The input signal is digitalized with a sampling frequency f_s, and stored in a buffer of size $L \cdot O_s$, that has been implemented in BRAM blocks [10]. This sampling buffer is read every $f_{FPGA} = L \cdot f_s$. At each reading, a sample is added or subtracted with the previous accumulated result, depending on the LS code evaluated. Whenever a new sample is received, the accumulated value is reset in order to compute a new correlation. The access to the sampling buffer is carried out in gaps of O_s positions, by taking into account the sampling frequency. In Table IV the requirements for this implementation are shown, in case of using a $LS(16,2,15)$. Correlation results are obtained every $t_s = t_{FPGA} \cdot L = 8.01\ ns \cdot 79 = 632.95\ ns$, being $t_s = \frac{1}{f_s}$ and $t_{FPGA} = \frac{1}{f_{FPGA}}$. Nevertheless, the ELSC implementation provides a new correlation result every FPGA clock cycle $t_{FPGA} = t_s = 10.23\ ns$, assuring real-time operation, although it requires more resources.

To increase the operation frequency of the straight-forward correlator, tasks have to be overlapped temporarily, which implies introducing registers to store intermediate data. Therefore, the amount of required resources increases, approaching to the required by the ELSC implementation. Thus, it can be stated that the ELSC implementation is more suitable for real-time correlation of LS sequences.

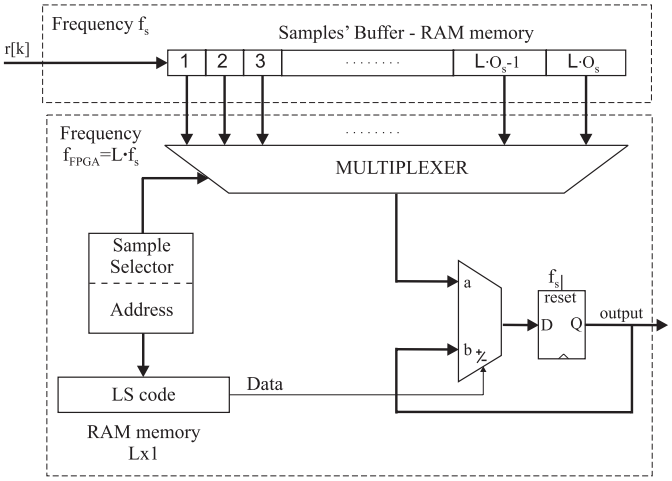


Fig. 9. Sequential implementation of a straight-forward correlator based on internal BRAM memory.

TABLE III
RESOURCES REQUIRED BY AN ELSC IMPLEMENTATION.

xc3s1500	LS(8,2,7)	LS(16,2,15)	LS(8,4,7)
Slices	720	1369	2112
Luts	929	1636	2432
IOBs	33	36	38
Max. Freq. $f_s = f_{FPGA}$	116.550 MHz	97.771 MHz	84.374 MHz
Correlation time $t_s = t_{FPGA}$	8.58 ns	10.23 ns	11.85 ns

TABLE IV
RESOURCES REQUIRED BY A STRAIGHT-FORWARD IMPLEMENTATION.

xc3s1500	LS(16,2,15)
Slices	51
Luts	83
IOBs	27
Max. Freq. f_{FPGA}	124.813 MHz
Max. Freq. f_s	1.58 MHz
Correlation time t_s	632.95 ns

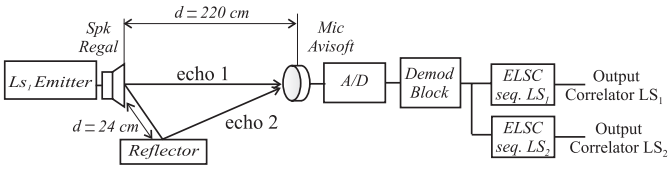


Fig. 10. Experimental setup for the tests carried out.

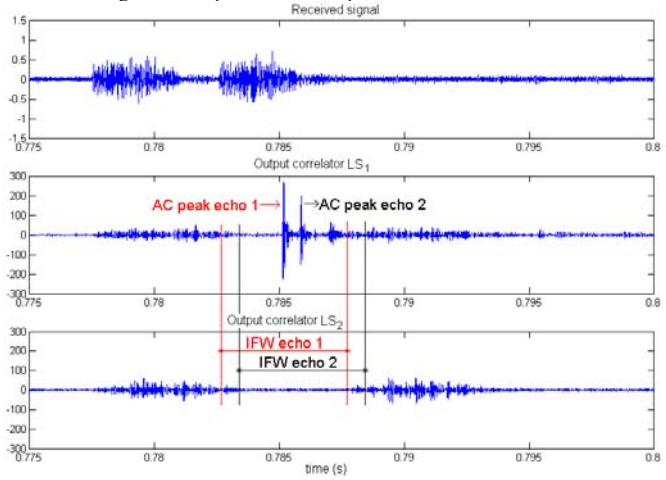


Fig. 11. Reception and correlation results of a real ultrasonic signal.

V. Application Example

The suitability of the ELSC has been tested in the ultrasonic pulse compression system shown in Figure 10. A set of two LS codes $LS(64, 1, 63)$ is generated, although only one sequence (LS_1) is transmitted. A REGAL-RH16E [12] high power speaker has been used to transmit LS_1 with a BPSK modulation scheme, which employs a symbol with one period of a 25 kHz squared signal. Two echoes (echo 1 and echo 2) of LS_1 are received, due to the multipath caused by a reflector. These echoes are captured by a condenser microphone capsule Avisoft Bioacoustic CM16 [13], amplified and digitalized at a sampling frequency of 500 kHz, so $O_s = 20$. Afterwards, the resulting signal is demodulated and, by using two ELSCs, it is correlated with the two original sequences (LS_1 and LS_2) of the set. Figure 11 shows the results obtained in every ELSC. At the output of the correlator LS_1 only echo 1 will be validated. Echo 2 arrives within the IFW of echo 1, and, due to the AC property, there is no interference impact in this zone. Also, because of the CC property, there is no interference at the output of the correlator LS_2.

VI. Conclusions

Traditional sequences used in pulse compression and CDMA systems, present ISI and MAI. LS sequences have ideal aperiodic correlation properties in a small window around the zero shift, removing the ISI and MAI completely in this window. If real-time operation is a requirement of the application, traditional straight-forward matched filter implementations are unsuitable. In this paper, an efficient correlator for LS sequences, generated with the method detailed in [7], is presented. It notably decreases the total number of operations to carry out, achieving real-time operation.

A generic hardware implementation in FPGAs of the proposed LS correlator has been developed. The design is characterized by five generic parameters: the number of codes in the set; the number of ceros in the centre of the LS code; the length of the Golay pairs used to generate the LS codes; the width of the data-path; and the sampling factor. Hence, the result is a configurable module, able to be adapted to the requirements of the application. The performance of the correlator has been tested on a Xilinx Spartan3 FPGA, and compared with the results obtained with an straight-forward implementation. Tests have shown that, for real-time operation, the correlator presented in this paper is more adequate. Furthermore, it has been included on an ultrasonic sensory system, to verify the detection capability in case of working in a multipath environment.

Acknowledgment

This work has been possible thanks to the Madrid Community (FPI grant), from the Spanish Ministry of Science and Technology (RESELAI: TIN2006-14986-CO2-01) and Spanish Ministry of Promotion (VIATOR: ref 70025-T05).

References

[1] Pingzhi Fan, Michaell Darnell, "Sequence Design for communications applications," *Research Studies Press LTD*, ISBN: 0 471 96557 X, 1996.
[2] S. W. Golomb, R. A. Scholtz, "Generalized Barker sequences" *IEEE Trans. Inf. Theory*, vol. IT-11, no. 4, Oct. 1965, pp. 533-537.
[3] D. Sarwate, M. Pursley, "Crosscorelation Properties of Pseudorandom and Related Sequences," *Proc. IEEE*, vol.68, no. 5, May 1980, pp. 593-619.
[4] M. J. E. Golay, "Complementary series," *IRE Trans. Inform. Theory*, vol. 7, no.2, Apr. 1961, pp. 82-87.
[5] B. Popovic, "Efficient Golay Correlator," *Electron. Lett.*, vol. 35, no. 17, Aug. 1999, pp. 1427-1428.
[6] C. C. Tseng, C. L. Liu, "Complementary sets of sequences," *IEEE Trans. Inf.*, vol. IT-18, no. 5, Sept. 1972, pp. 644-652.
[7] S. Stańczak, H. Boche," Are LAS-codes a miracle?,"*Proc. IEEE Global Telecommunications Conf. GLOBECOM'01*, San Antonio, Texas, vol. 1, Nov. 2001, pp. 589-593.
[8] M. Hahm, G. Friedman, E. Titlebaum, "A comparison of Analog and Digital Circuit Implementations of Low Power Matched Filters for Use in Portable Wireless Communication Terminals," *IEEE Trans. on circuits and systems II: analog and digital signal processing*, Vol. 44, no. 6, Jun. 1997, pp. 498-506.
[9] S. Ni, H. Wei, J. S. Blogh, L. Hanzo, "Network Performance of Asynchronous UTRA-like FDD/CDMA Systems using Loosely Synchronised Spreading Codes," *Proc. IEEE Vehicular Technology Conf. 2003*, Orlando, FL, USA, vol. 2, Oct. 2003, pp. 1359-1363.
[10] Xilinx, Inc., "Spartan-3E FPGA Family: Complete Datasheet"; *Product Documentation* , Nov. 2005.
[11] M. C. Pérez, A. Hernández, J. Ureña, C. De Marziani, A. Jiménez, "FPGA-based Implementation of a Correlator for Kasami sequences," *Proc. 11th IEEE International Conference on Emerging Technologies and Factory Automation ETFA'06*, Sept. 2006, 1141-1144.
[12] REGAL ELECTRONIC: "Regal Electronic Speakers"; Product Spec. (2006),
[13] AVISOFT BIOACOUSTICS: "Avisoft Bioacoustics UltraSoundGate microphone capsule CM16"; Product Spec. (2006).

Ultrasonic beacon-based Local Positioning System using Loosely Synchronous codes

M. C. Pérez[1], J. Ureña[1], A. Hernández[1], C. De Marziani[1], A. Jiménez[1], J. M. Villadangos[1], F. Álvarez[2]
[1]Department of Electronics, University of Alcalá, Alcalá de Henares, Spain
[2]Department of Electronics & Electromechanical Engineering, University of Extremadura, Cáceres, Spain
Email: carmen@depeca.uah.es

Abstract **– This work presents the development of a Local Positioning System (LPS), based on the transmission of ultrasonic signals, which have been previously encoded by Loosely Synchronous (LS) codes. The LPS consists of several ultrasonic emitters located at known positions in the environment, and of a portable receiver that computes its position by measuring the Differences in Times of Arrival (DTOA) between a reference emitter and the others. LS codes exhibit an Interference Free Window (IFW) in the auto-correlation and cross-correlation functions. Therefore, if the relative time-offset of the codes are within the IFW, it is possible to have simultaneous emissions without interference, as well as to reduce the multipath effect.**

Keywords **– Acoustic LPS, asynchronous detection, DS-CDMA, LS-codes, IFW, ISI, MAI.**

I. Introduction

Local Positioning Systems (LPS) allow to determine the location of a person or object in a certain area and, by using ultrasonic transducers, a suitable resolution degree is obtained [1]. Most of them consist of a set of beacons situated at known positions in the environment, and synchronized by radio-frequency (RF). The receptor is placed in a mobile robot and measures the Times of Arrival (TOAs) of the ultrasonic signals, starting from the emission interval [2] [3] [4]. Nevertheless, the use of RF signals implies an increase of the power consumption, interferences, etc. In [5] it is proposed an LPS where only acoustic emissions are used. It determines the absolute position of a mobile robot by hyperbolic triangulation of the distances obtained from the measurement of the Difference in TOAs (DTOAs) among a reference beacon (the nearest one) and the others, considering that all of them emit simultaneously and in a continuous way. Cancelation of interferences among simultaneous emissions is possible by using the SS-CDMA (Spread Spectrum Code Division Multiple Access) technique. Thus, a different orthogonal code is assigned to every beacon. The receiver on board the robot carries out the simultaneous correlations with the codes associated to every beacon, in order to detect the DTOAs between the reference beacon and the others. As the mobile robot does not need to know the emission instant, the RF signal can be supressed (it is enough a common synchronism among all the beacons).

The effectiveness of these systems strongly depends on the SS codes that codify the ultrasonic emissions. Those codes should provide very low cross-correlation (CC) values to avoid the Multiple Access Interference (MAI); and very low auto-correlation (AC) sidelobes to reduce the Inter-Symbol-Interference (ISI). In absolute positioning with ultrasounds, several previous works use pseudo-random (PR) sequences: Gold sequences [1] [6], and Kasami sequences [5]. Other works that codify the ultrasonic signal for the detection of obstacles in robotics use Barker codes [7], Golay codes [8] or Complementary Sets of M sequences (M-CSS) [9]. None of these sequences eliminate both ISI and MAI completely and simultaneously [10]. Golay pairs and M-CSS eliminate them by using more than one sequence per user, what implies some restrictions in the AC and CC functions [11].

Loosely Synchronous (LS) codes [12] exhibit an Interference Free Window (IFW), where the aperiodic AC sidelobes and CC values are zero. Consequently, if the time-offsets between the codes, expressed in terms of number of chips intervals, are within the IFW, both ISI and MAI can be eliminated thoroughly. Furthermore, the correlation of these codes can be carried out by means of an Efficient LS Correlator (ELSC) [13], which significantly reduces the total number of operations performed, in comparison with an straight-forward matched filter implementation. Thus, real-time operation can be possible without the aid of high complexity hardware.

This paper proposes the design of a LPS, which uses LS sequences to encode the ultrasonic signals. The paper is organized as follows: Section II presents the global arquitecture of the LPS. In Section III LS codes are described. Some results are shown in Section IV. Finally, conclusions are outlined in

1-4244-0830-X/07/$20.00 ©2007 IEEE.

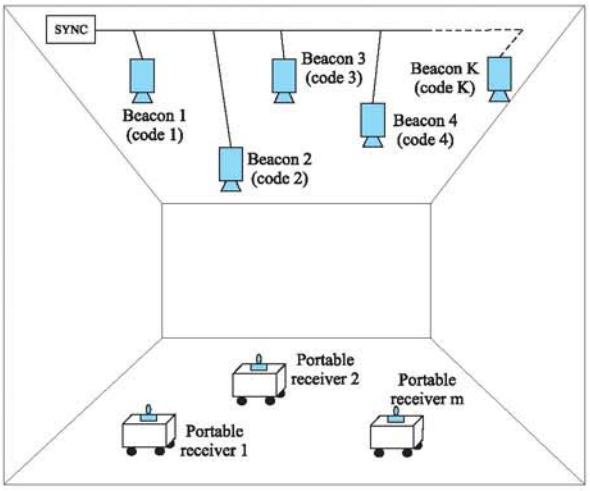


Fig. 1. SCHEMATIC REPRESENTATION OF THE LPS.

Section V.

II. GLOBAL STRUCTURE

Fig. 1 shows a schematic representation of the global structure of the proposed LPS. A set of hardware synchronized beacons (there exist a wire joining them) are placed at known positions of the environment, and all of them cover a determined area by emitting periodically. A non-limited number of portable receivers compute their own positions from the measurement of DTOAs, according to an hyperbolic triangulation algorithm (all the signals are asynchronously detected). At every position, the specific portable receiver has to detect at least 4 beacons for 2D positioning, or 5 beacons for 3D positioning.

In order to avoid multipath and multi-user interferences, the SS-CDMA technique is used, by encoding the ultrasonic signal with a different LS code for every beacon. On the other hand, the ultrasonic transducer used to emit the codes imposes its frequency response. Thus, it is necessary to modulate the signal to be emitted to place its spectrum at the maximum frequency response of the emitter. Concerning the reception process, all the signals can be simultaneously received by a group of portable detection modules (the total number depends on the particular application). Every module is able to distinguish between the different transmissions thanks to LS properties. It is important to notice that the received transmissions should arrive within the IFW to mitigate ISI and MAI. The detection process carried out in these modules involves four different tasks, namely, demodulation, efficient correlation, peak detection and positioning algorithm, as can be seen in Fig. 2.

The acquisition system converts the signal received by the ultrasonic transducer into a digital signal, which is demodulated to extract the transmitted information from the received signal. As there is no temporal reference in the receiver, the non-coherent demodulation is carried out by digital correlation with the modulation symbol. Later, K correlators simultaneously

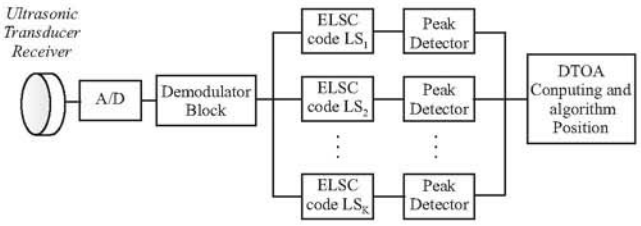


Fig. 2. BLOCK DIAGRAM OF THE RECEPTION STAGE.

correlate the demodulated signal with the K ideal emitted LS sequences. These correlations are performed by an efficient correlation system (ELSC) that notably decreases the total number of operations to carry out. In the precise moment at which the last sample of a received sequence is processed, a peak in the correlator for this LS sequence is obtained. A peak detector confirms the maximum values exceeding a static threshold, assuming that there is not a higher peak in the neighborhood. The maximum value nearest to time origin determines the beacon reference. The DTOAs between this reference beacon and the others are obtained by computing the difference in samples among the other maximum values in the different correlator outputs, and by multiplying them by the sampling frequency. The resulting values are used in the positioning algorithm to determine the robot's absolute position.

III. LS CODES

In [12] is proposed a method to construct LS codes from the well-known Golay pairs. It basically consists of linking with a specific order and directly or negated, depending on the coefficients of a Hadamard matrix, the sequences of two orthogonal Golay pairs of length N. Also, a set of W_0 zeros has to be inserted in the middle of the LS code to obtain the IFW in the correlation functions. Thus, it is obtained a set of K LS codes $\{G = g_k[l]; 1 \leq k \leq K; 0 \leq l \leq L-1\}$ with length $L = KN + W_0$, composed of $g_k[l] = \{0, \pm 1\}$ elements, and with aperiodic correlation functions equal to zero in a certain vicinity of the zero shift, that is:

$$R_{g_{k_1},g_{k_2}}[m] = \sum_{i=0}^{L-1} g_{k_1}[i] g_{k_2}[i+m] =$$

$$\begin{cases} K \cdot N, & \text{for} \quad m = 0, & k_1 = k_2 \\ 0, & \text{for} \quad 1 \leq |m| \leq W_0, & k_1 = k_2 \\ 0, & \text{for} \quad 0 \leq |m| \leq W_0, & k_1 \neq k_2 \end{cases} \tag{1}$$

Where $R_{g_{k_1},g_{k_2}}$ is the aperiodic correlation among g_{k_1} and g_{k_2} (note that the AC function is obtained when $k_1 = k_2$, whereas the CC function when $k_1 \neq k_2$); $K = 2^n$, being n any natural number, is the number of available LS codes with orthogonal properties in the IFW, that is, the maximum number of beacons that can simultaneously access to the medium; as has been stated before, N is the length of the Golay pairs used in the construction of LS codes; and $2W_0 + 1$ denotes the length of the IFW, being $W_0 \leq N - 1$.

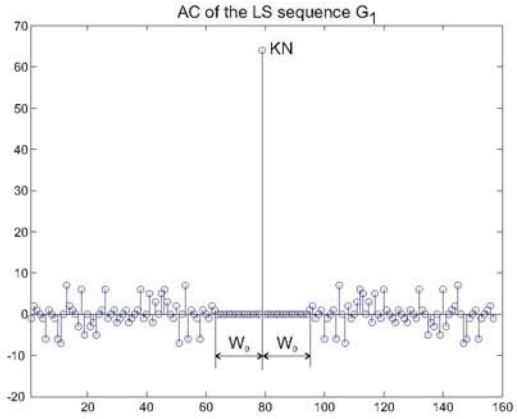


Fig. 3. AC FUNCTION FOR A SEQUENCE LS(16,4,15).

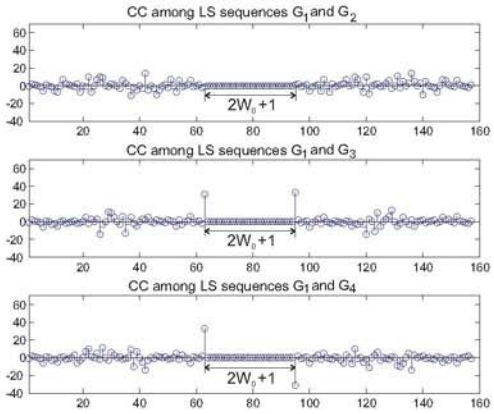


Fig. 4. CC FUNCTIONS AMONG THE SEQUENCES OF A SET GENERATED AS LS(16,4,15).

Since every LS family set can be defined by its parameters N, K and W_0, notation $LS(N, K, W_0)$ has been adopted. In Fig. 3 is shown the aperiodic AC of a $LS(16, 4, 15)$; whereas in Fig. 4 is depicted the aperiodic CC among different $LS(16, 4, 15)$ from the same set. It can be observed that interferences outside the IFW can be higher, even more than those of classic PR codes, so it is important to assure that the maximum dispersion delay of the channel (denoted by T_d) satisfy $T_d \leq W_0$ to minimize ISI and MAI interferences.

A. LS characterization outside the IFW

Several simulations have been carried out in order to study the performance of LS codes outside the IFW. Both the number of interferences outside the IFW and their magnitude have been evaluated.

It is well known that LS codes exhibit an IFW where neither AC sidelobes nor CC interferences appear. Outside the IFW there also exist zones with no interferences. Through experimental tests, it has been checked that the zone out of the IFW consists of approximately 75.72% of interferences (with an standard deviation of $\sigma = 0.0173$), and 24.28% of zeros. In Fig. 5 the percentage of CC interferences in LS sequences is represented for K simultaneously users. It can be observed that, the more users there are, the higher percentage of interferences in the specific sequence is.

To evaluate the magnitude of the AC sidelobes and CC values outside the IFW, the bound θ of LS sequences has been represented in Fig. 6. These bounds θ values are calculated according to (2) and expressed in percentages. Note that θ_{AC} corresponds to the AC bound, whereas θ_{CC} to the CC bound. The more simultaneous users there are, the higher bounds are obtained: 75.11% and 87.57% for $K = 16$ and $K = 32$ users respectively. It should be noted that, for a specific number of users, the percentage on interferences and correlation bounds are not affected by the sequence length (and specifically, they are not affected by the length N of the initial Golay codes). Thus, the length N has to be selected so as to have an enough IFW for the specific application, with the shortest LS codes (the larger a code is, the more resources and processing time are required for its correlation).

$$\theta = \max(\theta_{AC}, \theta_{CC}) \begin{cases} \theta_{AC} = \dfrac{\max\left\{\left\| R_{g_{k_1}, g_{k_1}}[m] \right\|\right\} \forall m \neq 0}{R_{g_{k_1}, g_{k_1}}[0]} \\ \theta_{CC} = \dfrac{\max\left\{\left\| R_{g_{k_1}, g_{k_2}}[m] \right\|\right\} \forall m}{R_{g_{k_1}, g_{k_1}}[0]} \end{cases} \quad (2)$$

A search for LS codes with the lowest bounds outside the IFW has been carried out, so as to reduce the interference in case that any echo falls outside the IFW. It has been checked that bounds obtained with LS codes generated with the same parameters (N, K, W_0) but from different Golay pairs, are very similar. When the number k of users is not a power of two, a set G with K LS codes, where K is the nearest power of two higher than k, has to be generated. In this case, it is possible to select the k LS codes in the set G with lower bounds. However it is not worthy to generate LS codes with $K > 2k$: in case that N is maintained constant, it means an unnecessary increment of the code length L without a better bound; and in case that N is reduced to maintain the length L constant, higher bounds are obtained.

To improve the performance of LS codes, it is also possible an enlargement of the IFW by inserting more zeros than $N - 1$ ($W_0 > N - 1$). It means an increase of the code length while maintaining the AC peak value, and the apparition of a set of controlled interferences within the IFW. This controlled interferences can be later eliminated by using ISI and MAI cancellation techniques [15].

B. Efficient LS Correlator

When longer LS sequences have to be processed in real-time, the increasing in the computational load demands the use of efficient correlators, able to perform the detection of these sequences with affordable computational cost. In [13] an efficient correlator for LS sequences, named ELSC, is proposed. It notably decreases the total number of performed operations, in comparison with a straight-forward implementation, as can

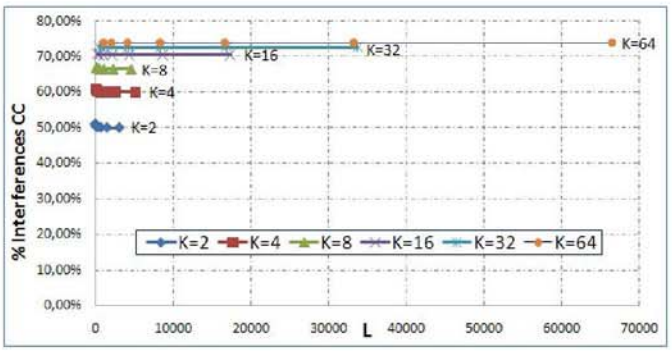


Fig. 5. NONZERO VALUES OF CC OF LS SEQUENCES AS A FUNCTION OF NUMBER OF USERS K AND CODE LENGTH L.

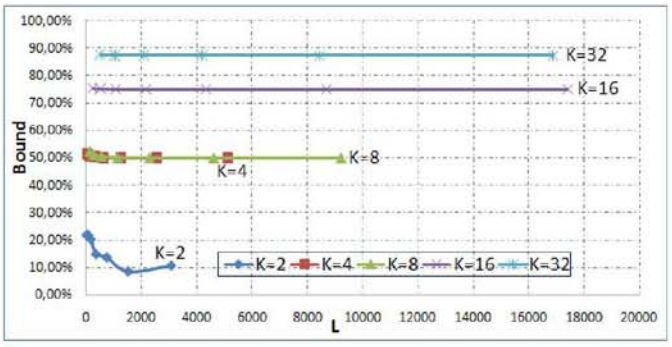


Fig. 6. BOUND IN LS SEQUENCES AS A FUNCTION OF NUMBER OF USERS K AND CODE LENGTH L.

TABLE I.

OPERATIONS TO PERFORM FOR THE CORRELATION OF LS CODES, USING AN STRAIGHTFORWARD IMPLEMENTATION AND AN ELSC.

Implementation	Multiplications	Additions
Straight-forward	$L = KN + W_0$	$L - 1 = KN + W_0 - 1$
ELSC	$2log_2 N + K$	$4log_2 N + 1$

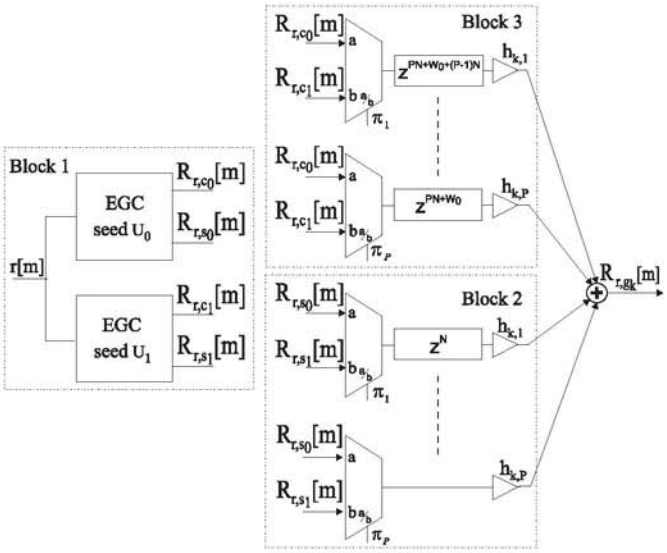


Fig. 7. BLOCK DIAGRAM OF THE EFFICIENT LOOSELY SYNCHRONIZED CORRELATOR.

be observed in Table I. Fig. 7 depicts the block diagram of the ELSC. It exploits the Efficient Golay Correlator (EGC) developed in [16] to simplify the correlation process. Two EGCs are used to simultaneously perform the correlation among the input LS sequence and the two orthogonal pairs composing it. Afterwards, the corresponding outputs of the EGCs are delayed, and added or subtracted according to the values of a binary vector π and the coefficients $h_{i,j}$ of a Hadamard matrix H.

Another advantage is that this correlator can be easily implemented in configurable hardware to achieve real-time operation (see [13] for further details).

IV. RESULTS

The proposed LPS has been simulated to verify its feasibility. The system is composed by a group of five beacons. Every beacon has an assigned $LS(128, 8, 127)$: a 1151-bit LS code with an IFW of 255 bits. These codes have been BPSK modulated by using a symbol with one period of a 50 kHz square signal. Thus, the duration of the signals is $1151 \cdot 20\ \mu$s$=$ 23.02 ms, and the maximum delay among the arrival of the different emissions is $T_{d_{max}} = 127 \cdot 20\ \mu$s$=$ 2.54 ms. The beacons' location has been chosen so as to assure that this time $T_{d_{max}}$ is not exceeded. Taking this into account, the five beacons are placed over the floor with a height of 3 m, and into a 1 m $\times$ 1 m surface, whereas the robot has been located at the coordinates ($x = 0.2$ m, $y = 0.1$ m, $z = 0.3$ m), as can be seen in Fig. 8.

All the beacons are synchronized by a common clock and they periodically emit a LS code every 50 ms. The emissions are acquired and digitalized with a sampling frequency of 500 kHz. The resulting signal is demodulated and the corresponding correlations are carried out. Fig. 9 shows the signal received assuming no noise, and Fig. 10 depicts the outputs of every correlator. Beacon 1 has the maximum value near the time origin, so it is the reference beacon. It can be also observed the IFW in every output correlator. Since all the echoes arrive in a time lower than $T_{d_{max}}$, both ISI and MAI interferences are mitigated in the area of the IFW. Outside the IFW the AC sidelobes and CC values are elevated, and can cause serious problems due to MAI and multipath interference in case of asynchronous detection. To avoid this, a special frame or window structure should be used: a maximum will be considered as a valid peak if its value is higher then a determined threshold and if it is surrounded by a window with only noise, but not interferences. Fig. 11 represents an enlargement of Fig. 10, where the distance among AC peaks can be observed.

In order to verify how the proposed system can operate in presence of noise, a similar detection has been carried out with a signal-to-noise ratio of SNR$=$ 0 dB. Fig. 12 shows the reception after the simultaneous emission of the five beacons, and Fig. 13 the correlation outputs. It can be observed that the detection of the different echoes is still achieved by the system.

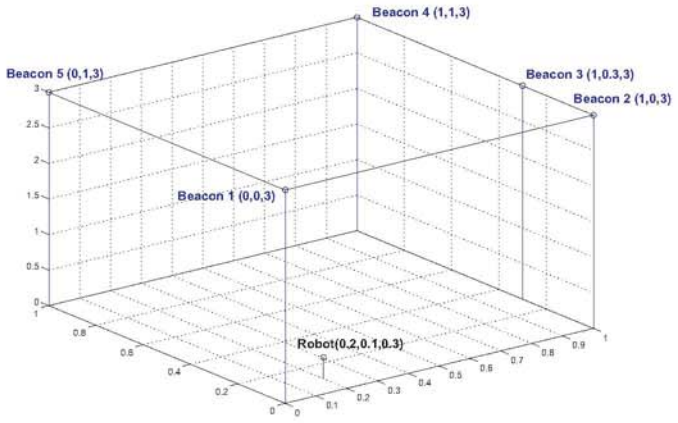


Fig. 8. SCHEME OF THE BEACON DISTRIBUTION AND OF THE POSITION OF THE MOBILE RECEIVER.

Finally, the five beacons are located into a 5 m × 5 m surface, with a height of 3 m. The robot maintain its former coordinates, and the SNR= 0 dB. This new beacon distribution is designed in such way that the delay between the arrivals of the different emissions can be higher than $T_{d_{max}}$. In this example, the emissions from beacons 2, 4 and 5 arrive outside the IFW of beacon 1 and 2 (see Fig. 14). It is not possible to determine the arrival of echoes corresponding to beacons 2, 4 and 5, so the DTOAs can not be computed, and therefore, the position of the portable receiver can not be obtained.

V. CONCLUSIONS

A local positioning system based on the use of LS sequences has been presented. It consists of using simultaneous emission from ultrasonic beacons. Every beacon is encoded with a LS sequence, which has the particularity of having a perfect auto-correlation and cross-correlation functions in the IFW. Every beacon periodically and continuously transmits its corresponding sequence. The mobile receiver does not require to know the emission instant, and therefore, it is not necessary a synchronism trigger signal between the receiver and the beacons. The system allows a non-limited number of mobile receivers working in the same environment, and thanks to the use of efficient correlators, real-time operation is achieved.

The suitability of LS codes for LPS has been analyzed. If the relative time-offsets between the codes are within the IFW, it is possible to easily identify the AC peaks, as both ISI and MAI are considerable reduced, even in case of low signal-to-noise ratios. Nevertheless, AC sidelobes and CC values are considerable outside the IFW, which is critical for asynchronous systems. A special analysis window is used in the reception to restrain the aperiodic interference components outside the IFW. As well as this, an analysis of the number of interferences and its magnitude outside the IFW has been carried out. Finally, the performance of the system has been tested by simulation.

ACKNOWLEDGMENT

This work has been possible thanks to the Madrid Community (FPI grant, INCUBUS: CCG06-UAH/DPI-0725), from the Spanish Ministry of Science and Technology (RESELAI: TIN2006-14986-CO2-01) and Spanish Ministry of Promotion (VIATOR: ref 70025-T05).

REFERENCES

[1] M. Hazas and A. Ward, "A novel broadband ultrasonic location system", *Proc. of UbiComp 2002: Ubiquitous Computing*, Göteborg, Sweden, vol. 2498, pp. 264-280, Sept. 2002.

[2] A. Ward, A. Jones and A. Hopper, "A New Location Technique for the Active Office", *IEEE Trans. on Personal Communications*, vol. 4, no. 5, pp. 42-47, Oct. 1997.

[3] S. Aoyagi, M. Tacaño and H. Noto, "Development of Indoor Mobile Robot Navigation System Using Ultrasonic Sensors", *Proc. of the IFAC Workshop on Intelligent Components for Vehicles (ICV'98)*, pp. 345-349, Sept. 1998.

[4] B. Priyantha, A. Chakraborty and H. Balakrishnan, "The Cricket location-support system", *Proc. of the Sixth International Conference on Mobile Computing and networking (ACM MobiCom)*, Boston, Massachussets, USA, pp. 345-349, Aug. 2000.

[5] J. Ureña et al, "Advanced Sensorial System for an Acoustic LPS", *Proc. of Journal of Microprocessors and Microsystems*, vol. 31, no. 6, pp. 393-401, Sept. 2007.

[6] J. M. Villadangos et al, "Improvement of Ultrasonic beacon-based Local Position System using Multi-Access Techinques", *IEEE International Symposium on Intelligent Signal Processing (WISP'05)*, Faro, Portugal, pp. 252-357, Sept. 2005.

[7] H. Peremans, K. Audeaert and J. Van Campenhout, "A high-resolution sensor based on tri-aural perception", *IEEE Trans. on Robotics and Automation*, vol. 9, no. 1, pp. 36-48, Feb. 1993.

[8] A. Hernndez et al, "Ultrasonic signal processing using configurable computing", *Proc. of the 15th Triennial Wold Congress of the International Federation of Automatic control (IFAC'02)*, Barcelona, Spain, July 2002.

[9] F. lvarez et al, "Detection module in a complementary set of sequences-based pulse compression system",*IEEE Proc. of the International Conference on Field Programmable Logic and Applications (FPL'06)*, Madrid, Spain, pp. 73-78, Aug. 2006.

[10] L.R. Welch, "Lower bounds on the maximum cross correlation of signals", *IEEE Trans. Inform. Theory*, vol. 20, no. 3, pp. 397-399, May 1974.

[11] De Marziani et al, "Inter-Symbol Interference Reduction on Macro-Sequences Generated from Complementary Sets of Sequences", *IEEE Proc. 32nd Annual Conference of the IEEE Industrial Electronics Society* Paris, France, pp. 3367-3372, Nov. 2006.

[12] S. Stańczak, H. Boche, "Are LAS-codes a miracle?", *IEEE Proc. IEEE Global Telecommunications Conf. GLOBECOM'01*, San Antonio, Texas, vol. 1, pp. 589-593, Nov. 2001.

[13] M. C. Pérez et al, "Efficient Real-Time Correlator for LS Sequences", *IEEE International Symposium on Industrial Electronics*, Vigo, Spain, Accepted for its publication, June 2007.

[14] S. Ni, H. Wei, J. S. Blogh, L. Hanzo, "Network Performance of Asynchronous UTRA-like FDD/CDMA Systems using Loosely Synchronised Spreading Codes," *IEEE Proc. of Vehicular Technology Conf. 2003*, Orlando, FL, USA, vol. 2, pp. 1359-1363, Oct. 2003.

[15] S. Moshavi, "Multi-User Detection for DS-CDMA Communications", *IEEE Communications Magazine*, vol. 34, no. 10, pp. 124-136, Oct. 1996.

[16] B. Popovic, "Efficient Golay Correlator", *Electron. Lett.*, vol. 35, no. 17, pp. 1427-1428, Aug. 1999.

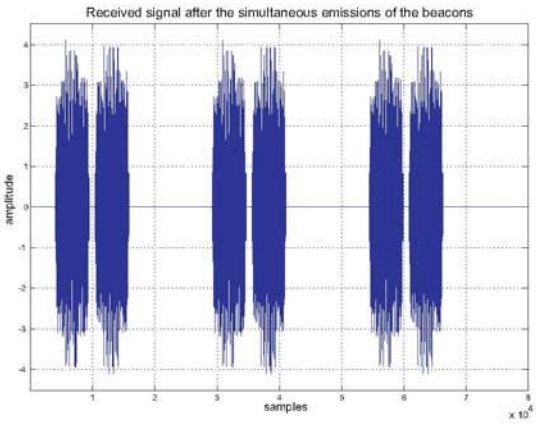


Fig. 9. SIGNAL RECEIVED WITH NO NOISE.

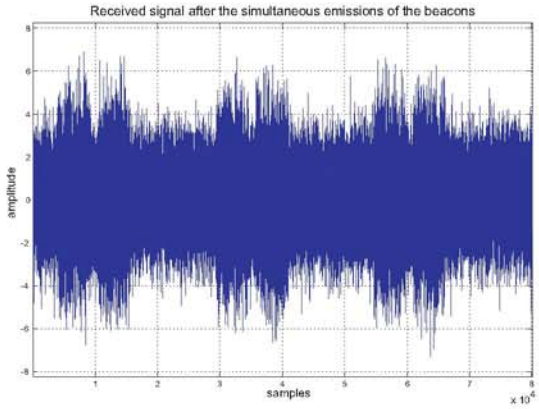


Fig. 12. SIGNAL RECEIVED WITH A SNR= 0 dB.

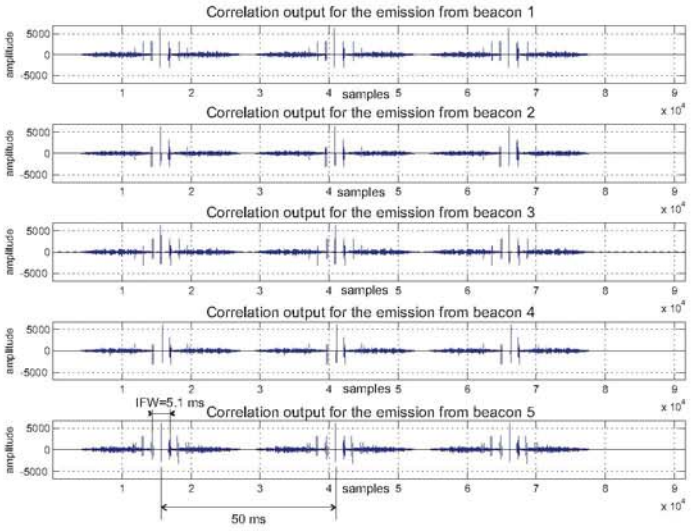


Fig. 10. CORRELATION OUTPUT FOR THE EMISSION OF EVERY BEACON IN CASE OF NO NOISE.

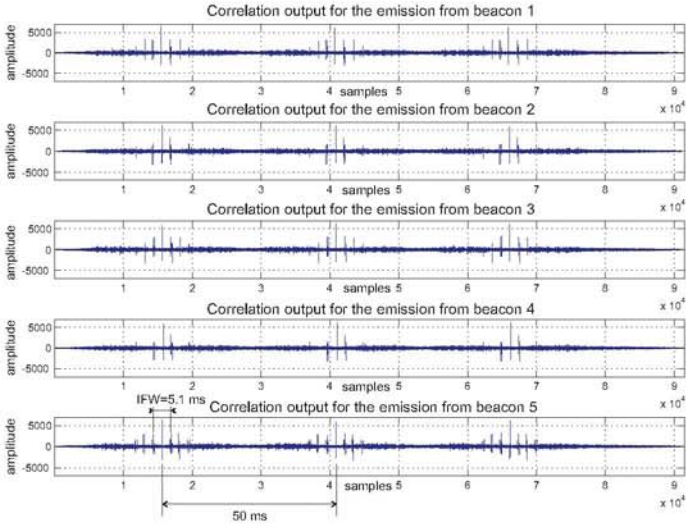


Fig. 13. CORRELATION OUTPUT FOR THE EMISSION OF EVERY BEACON IN CASE OF A SNR= 0 dB.

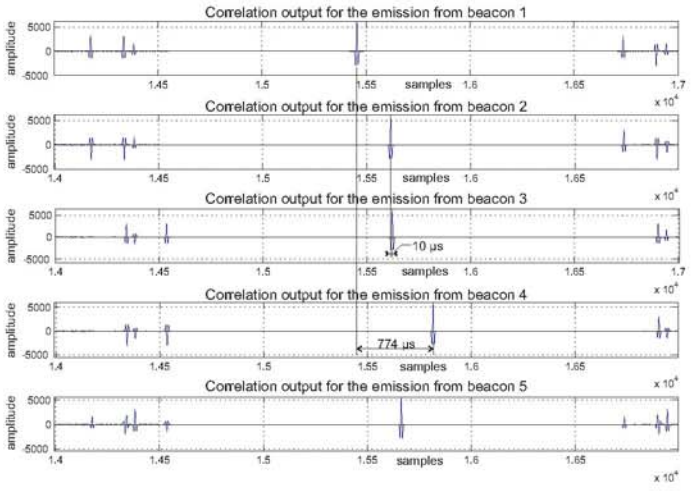


Fig. 11. ENLARGEMENT OF FIG. 10.

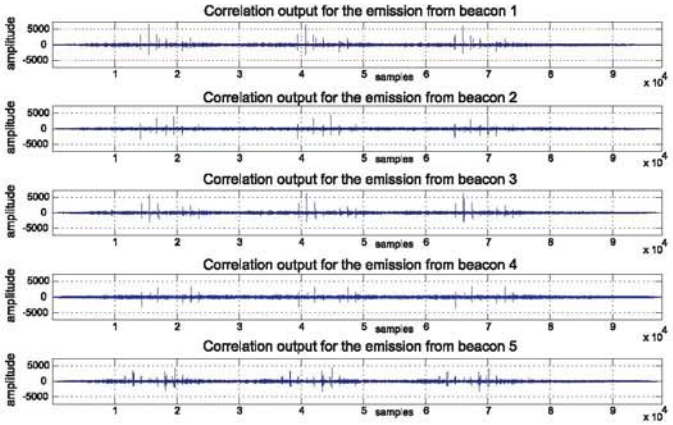


Fig. 14. CORRELATION OUTPUTS IN A SYSTEM WHERE THE TIME-OFFSET BETWEEN SEQUENCES IS LARGER THAN THE IFW.

19th INTERNATIONAL CONGRESS ON ACOUSTICS
MADRID, 2-7 SEPTEMBER 2007

SEPARATION OF CONCURRENT ECHOES DEPENDING ON THE EMITTING SOURCE USING DS-CDMA

PACS: 43.28.Gq, 43.28.Tc

Ureña, J.[1]; Pérez, M C.[1]; Ochoa, A.[1]; Hernández, A.[1]; De Marziani, C. [1]; Álvarez, F. J.[2]; García, J. J. [1]; Jiménez, A.[1]; Jiménez, J. A. [1]
[1]Deparment of Electronics, University of Alcalá; Escuela Politécnica, 28805 Alcalá de Henares. Madrid, Spain; {urena,carmen,alberto,alvaro,marziani,jesus,ajimenez,jimenez}@depeca.uah.es
[2]Department of Electrical Engineering, Electronics and Automatics; University of Extremadura; Escuela Politécnica. 10071 Cáceres, Spain; fafranco@unex.es

ABSTRACT
Many ultrasonic systems based on time-of-flight measurements use signals encoded with binary sequences with similar properties to Gaussian noise. Afterwards, the corresponding echoes are detected correlating the received signal with the original sequence. Thus, an improvement of temporal precision, spatial resolution and robustness to noise is obtained, in comparison with methods based on threshold detection. Furthermore, it could be interesting to have several transducers emitting ultrasound waves over the same spatial zone, and at the same time, allowing measurements from different locations simultaneously. These ultrasonic systems need an asynchronous detection as the echoes could arrive at any moment depending on the transducer and reflector positions. In such case sequences (one in every emitter) with a high Auto-Correlation function and low sidelobes should be used, as well as with low aperiodic Cross-Correlation values among them. As a result, it is possible to detect, and separate according with the original source of the emitted signal, all the echoes even in the case of high noise or attenuation. In this work, three different systems are presented based on Kasami sequences, Complementary Sets of sequences and Loosely Synchronized (LS) sequences. All of them were originally intended for the use in ultrasonic Local Positioning Systems.

INTRODUCTION
The incorporation of the encoding and processing techniques used in radar theory into airborne sonars is a major step in the evolution of these systems, as it supposes a notable improvement in features such as their temporal precision, spatial resolution and robustness to noise. On the other hand, the capability of simultaneous emission through an appropriate selection of the codes to be emitted is very useful in those systems that require a set of measurements made under similar conditions, and where these conditions change very fast, as is the case of a local positioning system where the mobile robot or object to be localised is normally moving. Furthermore, it is important to have efficient algorithms to perform the correlations of the acquired signals, so they can be easily implemented on reconfigurable hardware to achieve real-time operation.

The effectiveness of the DS-CDMA technique depends on the properties of the used spreading codes, requiring ideal properties of autocorrelation (AC) and cross-correlation (CC). Many algorithms have been designed to generate codes, but not all of them meet these ideal properties simultaneously. Examples are the Pseudo-Random sequences [1], Barker codes [2], Golay sequences [3], sets of complementary sequences [4], etc. However these systems are interference limited. There exists Inter-Symbol-Interference (ISI) due to the non zero auto-correlation (AC) sidelobes of the used sequences. They also have Multiple-Access-Interference (MAI) due to the non zero cross-correlations (CC) values. To avoid the effects of both ISI and MAI, the sidelobes of the AC and the CC values should be as small as possible.
Barker codes [2] have been widely used due to their good aperiodic AC. However, their maximal length is limited, so it is not possible to detect them in case of high noise. Furthermore,

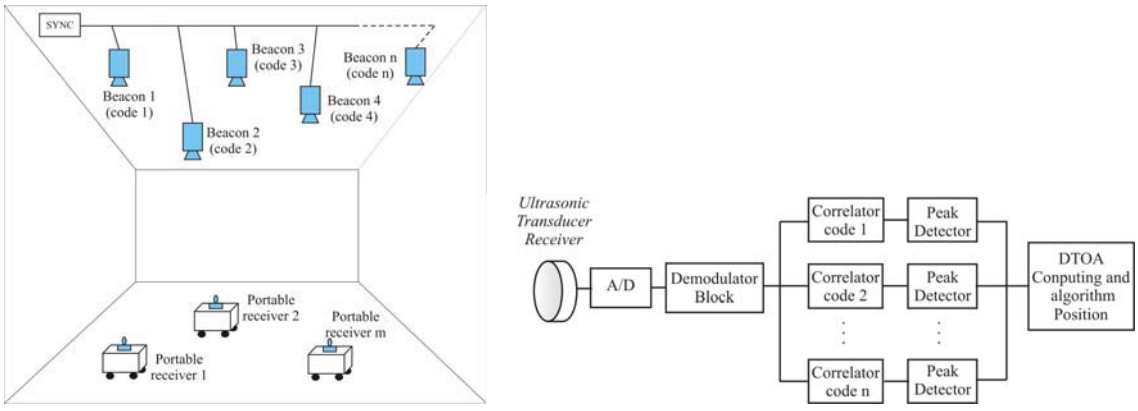
there are no Barker codes with low CC among them. Pseudo-random sequences such as m-sequences, Gold codes or Kasami codes, exhibit non zero off-peak AC and CC values in the case of asynchronous transmissions. Another possibility is Golay pairs, which have good aperiodic correlation properties, and their length is not limited. Nevertheless, they provide only two mutually orthogonal pairs, which is not useful for multi-user environments. Complementary sets of M sequences (M-CSS) are a generalization of Golay codes containing more than two sequences. The elimination of the constraint in the number of sequences of a set yields on a high process gain and also M mutually orthogonal sets. To achieve these goals, it is necessary to add the AC functions of the sequences of the set, or the CC functions of the corresponding sequences in the M sets, respectively. The problem is that, in many systems, transducers have limited bandwidths, so it is not possible to transmit or receive the M sequences of the set simultaneously. Unfortunately, it implies an undesired increase of both ISI and MAI. The detection of Golay pairs or M-CSS can be performed by means of efficient correlation algorithms which notably decrease the computational load and hardware complexity [10]. Loosely Synchronized (LS) codes [6] exhibit an Interference Free Window (IFW), where the aperiodic AC sidelobes and CC values become zero. Consequently, ISI and MAI are completely reduced if the maximum transmission delay is less than the length of the IFW.

As the application of DS-CDMA in airborne ultrasonic systems deals with band-limited transducers and asynchronous detection, in this work, three kinds of sequences with the besr performance are presented: Kasami sequences, Complementary Sets of sequences and Loosely Synchronized (LS) sequences. All of them are here intended for the use in advanced ultrasonic sensors or in ultrasonic Local Positioning Systems.

The paper is organized as follows. Section II introduces the general structure and block diagram of the detection method. In Section III the aforementioned sequences for asynchronous detection are presented. Section IV shows how these sequences are used in LPS applications and, finally, some conclusions are outlined in Section V.

GLOBAL STRUCTURE OF THE ULTRASONIC SYSTEM

Fig. 1 shows a schematic representation of the global structure of a Local Positioning System (LPS). A set of hardware synchronized beacons are placed at known positions of the environment, and all of them cover a determined area by emitting periodically a coded sequence (a different code each beacon). A non-limited number of portable receivers compute their own positions from the measurement of Differences of Times Of Arrival (DTOAs), according to an hyperbolic triangulation algorithm (all the signals are asynchronously detected). At every position, the specific portable receiver has to detect at least 4 beacons for 2D positioning, or 5 beacons for 3D positioning.



a) b)

Figure 1.- LPS configuration (a) and signal processing in a receiver (b).

The acquisition system converts the signal received by the ultrasonic transducer into a digital signal, which is demodulated to extract the transmitted information from the received signal. As there is no temporal reference in the receiver, the non-coherent demodulation is carried out by

digital correlation with the modulation symbol. Later, K correlators simultaneously correlate the demodulated signal with the K ideal emitted sequences. In the precise moment at which the last sample of a received sequence is processed, a peak in the correlator for this sequence is obtained. A peak detector confirms the maximum values exceeding a threshold, assuming that there is not a higher peak in the neighborhood.

SEQUENCES FOR ASYNCHRONOUS DETECTION

The purpose of this section is to evaluate the performance of Kasami sequences, Complementary Sets of M Sequences (M-CSS) and Loosely Synchronized (LS) sequences as encoding sequences for an ultrasonic beacon-based Local Positioning System (LPS). As several transducers will emit ultrasound waves simultaneously over the same zone, the encoding sequences should bear very low cross-correlation (CC) values to provide adequate discrimination among users. This is critical to control the multiple-access-interferences (MAI) in situations where there are large differences in the relative power level of the received signals. Furthermore, the sequences should have sharp auto-correlation (AC) functions, with very low sidelobes, to minimise the inter-symbol-interference (ISI) and provide higher processing gains.

Small set of Kasami sequences

Small set of Kasami sequences were first discussed in [5] to improve the cross-correlation (CC) properties of other pseudo-random codes. Basically, the set of Kasami sequences consists of $2^{n/2}$ sequences with length $L=2^n-1$, being n an even integer, whose AC sidelobes and CC periodic functions take values from the set $\{-1, -2^{n/2}-1, 2^{n/2}-1\}$. Both the periodic and aperiodic maximum AC value is L; nevertheless, the other aperiodic correlation values are not within this set. The aperiodic correlation properties of Kasami codes are analyzed in [6]: the maximum odd CC value for any pair of Kasami sequences is $2^{n-1}+2^{(n-2)/2}+1$, whereas the maximum periodic CC value (also called even CC value) is $2^{n/2}+1$.

The correlation of Kasami sequences is carried out by means of straight-forward matched filter implementations, in case of short sequences, or by Fast-Fourier-Transforms (FFTs) in case of larger ones [8]. These implementations provide large processing cost; and in case of large amount of data to be processed, real-time operation is difficult to achieve.

Complementary Sets of *M* Sequences

M-CSS are sets of M sequences with length $L=M^N$ whose elements are either $\{+1,-1\}$, where $M=2^m$, being m and N any natural number different from zero. The addition of the aperiodic AC functions of the M sequences from that set is zero everywhere, except for the zero shift where has a maximum value of $M \bullet L$. Furthermore, it is possible to obtain M mutually orthogonal sets.

Nevertheless, since more than one sequence encodes a user, a bit emission order has to be used. This implies a degradation of the ideal AC and CC properties: neither the AC sidelobes, nor the CC values for two orthogonal sets are null anymore. As it is desirable that all the sequences in the set are equally affected by changes in the environment, a transmission method based on interleave their bits is carried out. In [9] this method it is compared with linking consecutively the sequences of the set; also a post-processing algorithm is proposed to reduce the degradation in the AC and CC values.

The detection of M-CSS can be done with efficient correlation algorithms that notably decrease the computational load and hardware complexity, in comparison with straight-forward implementations. The hardware implementation of this efficient correlator adapted to the transmission scheme by interleaving is analyzed in [10]. It has been checked that, for equally long interleaved sequences, less hardware resources are required for cases generated from M-CSS with low number M of sequences.

Loosely Synchronized Codes

LS codes exhibit an Interference Free Window (IFW), in a certain vicinity of the cero shift, where the aperiodic AC sidelobes and CC values become zero. Consequently, if the time offsets between the codes, expressed in terms of number of chips intervals, are within the IFW, both ISI and MAI can be eliminated thoroughly.

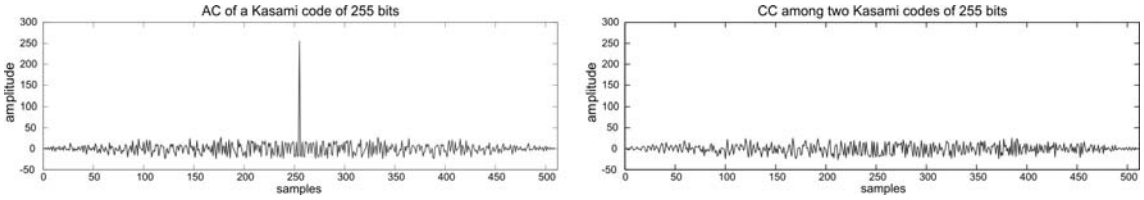
These codes exploit the properties of orthogonal 2-CSS (better known as Golay pairs) [11]. The notation *LS(N,K,W_0)* has been adopted, where *N* is the length of the Golay pairs; *K* is the number of available codes with orthogonal properties in the IFW ($K=2^n$, n Є |N); and W_0 is the number of zeros to insert in the middle of the sequence during its construction (usually $W_0=N-1$). The total length of LS codes is given by $L=KN+W_0$, whereas the length of the IFW is given by $W=min\{2N-1,2W_0+1\}$. The AC peak of a LS sequence is equal to $K \bullet N$, while the AC sidelobes and CC values within the IFW are zero. However, the AC and CC functions exhibit higher values outside the IFW than Kasami and M-CSS. Therefore, to minimize ISI and MAI the maximum dispersion of the channel (denoted by T_d) should satisfy $Td \leq W_0$.

The correlation of these codes can be carried out by means of an Efficient LS Correlator (ELSC) [12] which significantly reduces the total number of operations performed. Also, this correlator can be easily implemented in hardware configurable to achieve real-time operation.

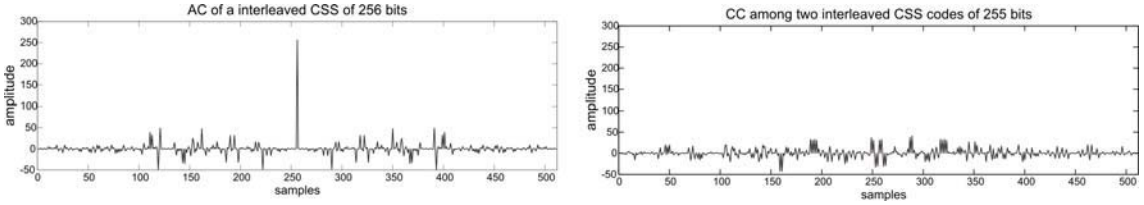
Sequences comparison

A direct comparison of the interferences caused by AC sidelobes and CC values in the three types of codes indicates that LS codes are more adequate, in case that the received transmissions arrive within the IFW. Nevertheless, if this cannot be assured, more serious MAI and multipath interferences will be encountered. Kasami codes are a good solution, as they posses low correlation interferences within all the correlation function (M-CSS's interferences are higher). However, when longer sequences have to be processed in real-time, the increasing in the computational load demands the use of efficient correlators, which does not exist for Kasami codes, so M-CSS have to be selected. Hence, it is inconclusively which of the three codes are better suited; they have to be chosen depending on the particular application characteristics.

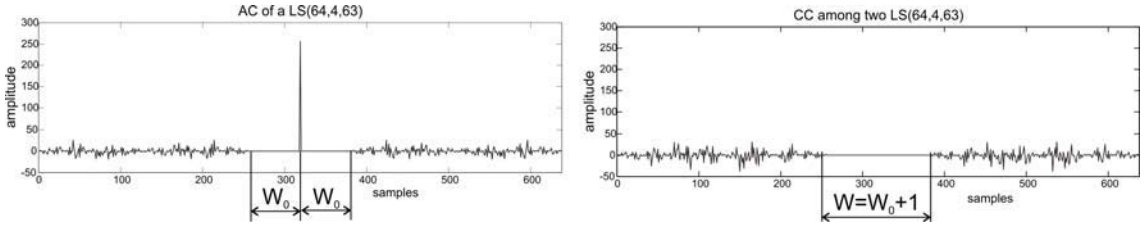
Fig. 2 shows the correlation properties for the case of Kasami sequences with length 255 bits (a); of interleaved CSS sequences with length 256 bits and generated from sets of four complementary pairs (b); and of LS(64,4,63), that is, of LS sequences with length 319 bits and an IFW of 129 bits (c). For each family of sequences, those who have the better performance in case of four simultaneous users have been chosen. Notice that the AC peak of Kasami and interleaved CSS is equal to their length, but the AC peak of LS sequences it is equal to $K \bullet N$.



a) AC and CC functions for Kasami sequences with length 255 bits.



b) AC and CC functions for Kasami sequences with length 255 bits.

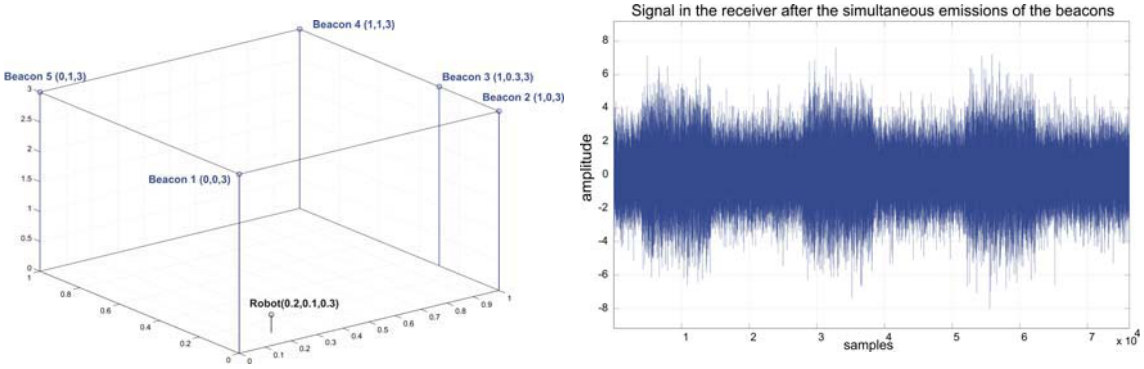


c) AC and CC functions for LS(64,4,63).

Figure 2.- Correlation functions for Kasami, M-CSS and LS codes.

LOCAL POSITIONING SYSTEM

Some simulations have been achieved to evaluate the feasibility of the previous codes in a LPS. The system is composed by a group of five beacons, which can be codified with Kasami, M-CSS or LS codes. These codes have been BPSK modulated by using a symbol with one period of a 50KHz square signal. The five beacons are placed over the floor with a height of 3m, and into a 1m x 1m surface, whereas the robot has been located at the coordinates (x=0.2m, y=0.1m, z=0.3m), as can be seen in Fig. 3. All the beacons are synchronized by a common clock and they periodically emit their corresponding code every 50ms. The emissions are digitalized with a sampling frequency of 500 kHz, demodulated and the corresponding correlations are carried out. A signal-to-noise ratio of SNR= 0dB has been assumed.



a) b)

Figure 3.- a) Scheme of the beacons distribution. b) Kasami signals received with a SNR=0dB.

Firstly, the beacons are encoded with five pseudo-orthogonal Kasami codes with length 1023 bits, thus the duration of the signals is 1023•20μs=20.46ms. Fig. 4.a) depicts the correlator output for the emission from beacon 2. The maximum values indicating the TOAs of every emission can be clearly observed. A similar detection has been carried out with interleaved 32-CSS (the final length of the emitted macro-sequence is of 1024 bits). Fig. 5 shows the correlation results from beacon 2.

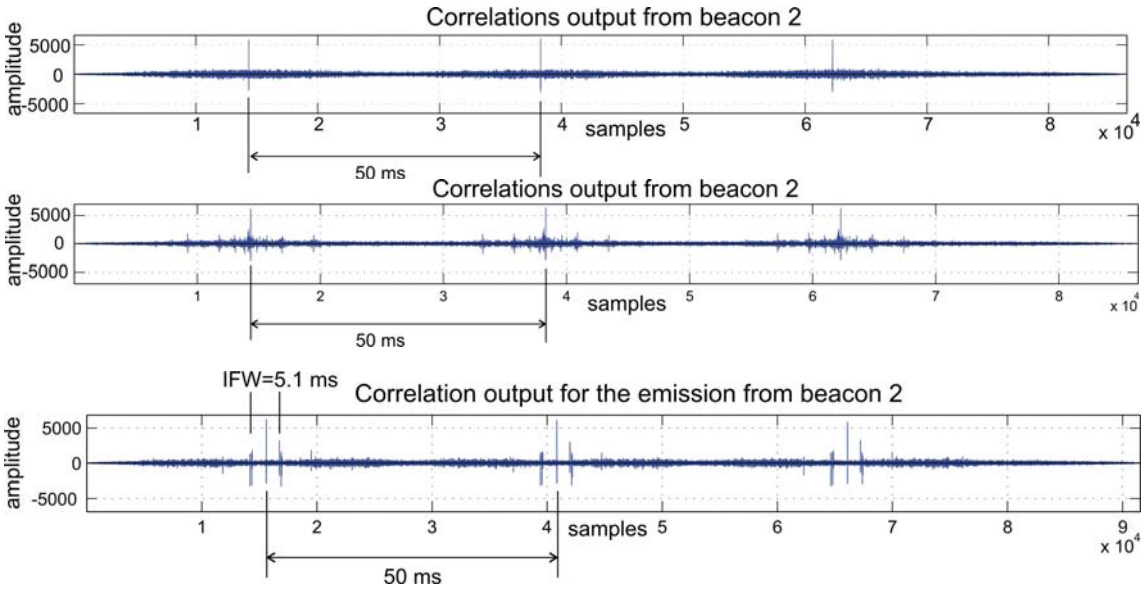


Figure 4.- Correlation output from beacon 2, in case of a SNR=0dB: a) 1023-Kasami codes, b) Macro-sequence from 32-CSS and c) LS(128,8,127).

Finally, every beacon has been encoded with a LS(128,8,127): a 1151-bit LS code with an IFW of 255 bits. Thus, the duration of the signal is 23.02ms, and the maximum delay among the

arrival of the diferent emissions is T_{dmax}=127•20μs=2.54ms. Fig. 4.c) depicts the output of correlator 2, where the IFW can be observed.

CONCLUSIONS

A local positioning system based on the use of Kasami, interleaved M-CSS and LS sequences has been presented. It consists of using simultaneous emission from ultrasonic beacons. Every beacon is encoded with a particular sequence.

The differences between sequences have been pointed out: all of them perform very well in asynchronous detection. Kasami and interleaved M-CSS have good properties in all theirs autocorrelation and crosscorrelation functions. On the other hand LS codes have the particularity of having a perfect auto-correlation and cross-correlation functions in the IFW.

The mobile receiver does not require knowing the emission instant, and therefore, there is not necessary a synchronism trigger signal between the receiver and the beacons. The system allows a non-limited number of mobile receivers working in the same environment.

It is important to remark that interleaved M-CSS and LS codes admit the use of efficient correlators, and then real-time operation is easier to be achieved.

Finally, the performance of the system has been tested by simulation.

ACKNOWLEDGMENTS

This work has been possible thanks to the Madrid Community (INCUBUS: CAM-UAH2006/016), from the Spanish Ministry of Science and Technology (RESELAI: TIN2006-14986-CO2-01) and Spanish Ministry of Promotion (VIATOR: ref 70025-T05).

References:

[1] Chow, A. “Performance of Spreading Codes for Direct Sequence Code Division Multiple Access (DS-CDMA)”, Technical Report, Stanford University , 5 December 2003,.

[2] Golomb, S., “Generalized Barker Sequences”, IEEE Trans. Inf. Theory, IT-11, (4), 1965, pp. 533-537

[3] Golay, M.J.E., “Complementary series”, IRE Trans. Inf. Theory, IT-7, 1961, pp. 82–87.

[4] Tseng, C., “Complementary sets of sequences”, IEEE Trans. Inf. Theory, IT-18, (5), 1972, pp. 644–652.

[5] T. Kasami, “Weight distribution formula for some class of cyclic codes”, Combinational Mathematics and its Appl., Chapel Hill, N.C.: University of North Carolina Press, 1969.

[6] Ch. Zhang, X. Lin, S. Yamada, M. Hatori, “General Method to construct LS Codes by Complete Complementary Sequences. IEICE Trans. Commun. Vol. E88-B, No8, pp. 3484-3487, August 2005.

[7] D. Sarwate and M. Pursley, “Crosscorrelation Properties of Pseudorandom and Related Sequences,” Proc. IEEE, vol. 68, no. 5, pp. 593-619, May 1980.

[8] M. C. Pérez, A. Hernández, J. Ureña, C. De Marziani, A. Jiménez, “FPGA-based Implementation of a Correlator for Kasami Sequences”, Proc. 11th IEEE International Conference on Emerging Technologies and Factory Automation ETFA’06, pp. 1141-1144, 2006.

[9] C. Marziani, J. Ureña, A. Hernández, M. Mazo, J. J. García, A. Jiménez, J. M. Villadangos, M. C. Pérez, F. Álvarez, “Inter-Symbol Interference Reduction on Macro-Sequences Generated form Complementary Sets of Sequences”, IEEE Proc. 32nd Annual Conference of the IEEE Industrial Electronics Society, Nov. 2006.

[10] M. C. Pérez, J. Ureña, A. Hernández, C. De Marziani, A. Jiménez, W. P. Marnane, “Hardware Implementation of an Efficient Correlator for Interleaved Complementary Sets of Sequences”, Journal of Universal Computer Science, vol. 13, no. 3, pp. 388-406, March 2007.

[11] S. Stanzaq, H. Boche, “Are LAS-codes a miracle?”, IEEE Proc. Global Telecommunications Conf. GLOBECOM’01, San Antonio, Texas, vol. 1, pp. 589-593, Nov. 2001.

[12] M. C. Pérez, J. Ureña, A. Hernández, A. Jiménez, W. P. Marnane, F. Álvarez, “Efficient Real-Time Correlator for LS Sequences”, IEEE International Symposium on Industrial Electronics, Vigo, Spain, Accepted for its publication, June 2007.

Available online at www.sciencedirect.com
ScienceDirect
Microprocessors and Microsystems 31 (2007) 393–401
MICROPROCESSORS AND MICROSYSTEMS
www.elsevier.com/locate/micpro

Advanced sensorial system for an acoustic LPS

J. Ureña [a], A. Hernández [a,*], A. Jiménez [a], J.M. Villadangos [a], M. Mazo [a], J.C. García [a], J.J. García [a], F.J. Álvarez [b], C. De Marziani [a], M.C. Pérez [a], J.A. Jiménez [a], A.R. Jiménez [c], F. Seco [c]

[a] *Department of Electronics, University of Alcalá, Campus Universitario s/n, 28805 Alcalá de Henares, Madrid, Spain*
[b] *Department of Electronics and Electromechanical Engineering, University of Extremadura, Spain*
[c] *Instituto de Automática Industrial – CSIC, 28500 Arganda del Rey, Madrid, Spain*

Available online 27 February 2007

Abstract

This work presents the development and implementation of a Local Positioning System (LPS) based on the transmission of ultrasonic signals, proposed for indoor positioning of mobile robots. The LPS consists of several ultrasonic beacons located in the environment, all of them emitting periodically and simultaneously. A portable receiver collects all the transmissions and autonomously computes its position. Neither infrared nor radiofrequency are required for synchronization in the localization process, compared to other related works. Direct Sequence Code Division Multiple Access (DS-CDMA) techniques are used to allow simultaneous emission from several beacons; each one transmitting a different 255-bit Kasami code. The portable receiver can detect the time of arrival for the different signals coming from the beacons, by carrying out all the correlations with the emitted codes. Absolute position of the receiver is determined by hyperbolic trilateration, using the Differences in Times-of-Arrival (DTOA) between a reference beacon and the other ones. The main advantage of this proposal is to permit the coexistence of an unlimited number of portable receivers, what is useful in applications like multirobot cooperation.

Keywords: Acoustic LPS; Kasami codes; DS-CDMA; FPGA; Asynchronous detection

1. Introduction

Two- or three-dimensional local positioning systems (LPS) of mobile robots, based on ultrasonic signals, are well-known for robotic applications. They consist of several beacons located at known positions in the environment where the robot is moving. Robot location can be determined by measuring the Times-of-Arrival (TOAs) of the ultrasonic signals. To synchronize the emission instant of every beacon, an additional radiofrequency or a coded infrared signal is often used [1,2], what increases the complexity of the hardware to be implemented.

If the beacons work as receivers and the mobile robot does as an ultrasonic emitter, it is only necessary to transmit an ultrasonic pulse to obtain the TOA for every beacon. The main drawback is that it requires a centralized communication and processing system to compute the TOA obtained in every beacon and to estimate the robot's position [3].

The proposal of this work is the determination of the absolute position of a mobile robot by using the measurement of the Difference in Times-of-Arrival (DTOA) between a reference beacon and the other ones, assuming that all of them emit simultaneously and in a continuous way. To solve the problem of the simultaneous emissions in beacons, DS-CDMA technique is used. A different 255-bit Kasami code is modulated at 50 kHz in every beacon [4], and transmitted in a periodic way. A receiver on board the robot carries out simultaneous correlations with

* Corresponding author. Tel.: +34 918856583; fax: +34 918856591.
E-mail addresses: urena@depeca.uah.es (J. Ureña), alvaro@depeca.uah.es (A. Hernández), ajimenez@depeca.uah.es (A. Jiménez).
URL: http://www.depeca.uah.es (A. Hernández).

0141-9331/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.micpro.2007.02.003

the codes assigned to each beacon, in order to detect the DTOAs between a reference beacon (the nearest one) and the others.

There exist several previous works that have used encoding of ultrasonic signals to implement advanced sensors (with several emitters and/or receivers). They are mainly focused on the detection of obstacles in robotics, by using pseudo-random sequences [5,6], Barker codes [7,8], or Golay codes [9]. About absolute positioning with ultrasounds, Hazas has used 511-bit Gold sequences [10]; also in [11], *m*-sequences are used to encode the acoustic emissions of different sources. Finally, the hyperbolic trilateration technique is widely used to obtain the absolute position of the mobile robot, with DTOAs [12].

This work describes the design and electronic implementation of the ultrasonic beacons and the portable receiver used in the LPS. The processing algorithms are implemented in FPGA-based platforms, whose operation mode can be modified in run-time, so the global system can be adapted to different environment features.

The paper is organized as follows: Section 2 presents the global architecture of the LPS; in Section 3 the acoustic transducers are described; Sections 4 and 5 are dedicated to the implementation of the beacon system and the portable receiver, respectively; some results are shown in Section 6; and, finally, some conclusions are discussed in Section 7.

2. Global structure

The design of the proposed indoor LPS is based on the GPS system. A set of beacons (as satellites) are placed at known positions of the environment, therefore a non-limited number of portable receivers can compute autonomous and independently their position. For that, the reception of transmissions from a minimum number of beacons is required. On the other hand, in this LPS the beacons are wired-synchronized, allowing a simultaneous and periodic emission. Consequently, the portable receiver asynchronously detects all the signals and it computes its position from DTOAs according to a hyperbolic triangulation algorithm.

Fig. 1 shows a schematic representation of the global structure of the LPS. At every position the portable receiver has to detect at least three beacons (for 2D positioning) or four beacons (for 3D positioning). In order to avoid Mutual Access Interference (MAI) among beacons [13], all of them are encoded with pseudo-orthogonal Kasami codes.

3. Sensors

Three commercial devices (A [14], B [15], and C [16] in Table 1) have been tested using a 4939 condenser microphone, by Brüel and Kjaer [17], and a Murata Super Tweeter [18]. A and C types are used for acoustic applications in cellular-phones, laptops, etc.; therefore it has been necessary to evaluate some key features beyond the audible limit

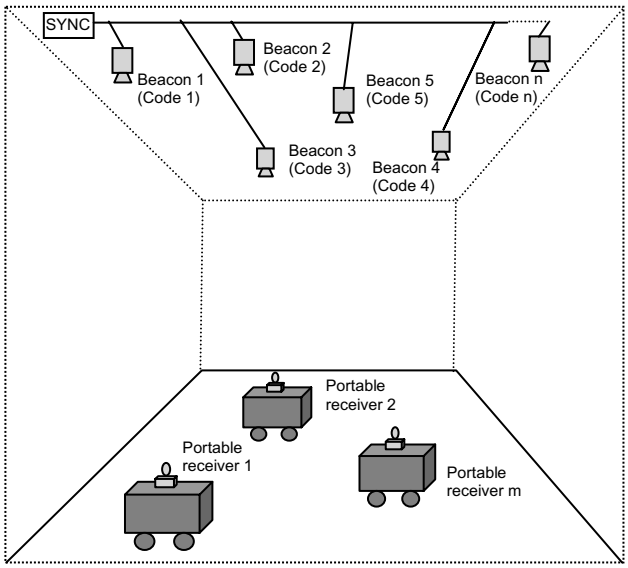


Fig. 1. Schematic representation of the LPS.

(20 kHz). Table 1 not only includes the relevant information supplied by the manufacturers, but also additional data obtained from measurements into an isolated chamber. Transmitter A can emit frequencies up to 50 kHz, with a minimum response at 37 kHz; it provides around 105 dB Sound Pressure Level (SPL) at 30 cm. Receiver C can detect a wide range of frequencies, between 20 Hz and 45 kHz, omni-directionally. The results show that these commercial devices can be used as ultrasonic transducers in LPS applications with the advantages of their small size and reduced cost.

Finally, the device B, which can be used as emitter and as receiver, works with frequencies around 40 kHz with a bandwidth of 15 kHz, and it provides a 105 dB SPL at 38 kHz.

Although some tests have been carried out with all the transducers, experiments and results shown in next sections have been obtained with a Super Tweeter Murata [18] as emitter, and a Panasonic WM-61 transducer [16] as receiver.

4. Beacon system

The ultrasonic beacon system has been placed in the environment to be covered, as can be observed in the general view of the LPS prototype in Fig. 2.

The ultrasonic beacon network is connected through an I2C bus, as shown in Fig. 3. One of the beacons works as master, linked through an enhanced parallel port (EPP) to a central PC where the configuration and operation mode of the system can be modified in real-time. This master beacon controls the I2C bus, whereas the other ones work as slaves. Furthermore, one of the I2C wires provides the synchronization for the simultaneous emission of all the beacons.

Table 1
Commercial and measured parameters of the used transducers: frequency range, sound pressure level (SPL) or sensibility (S), and total beam angle

Transmitters	Freq. range (Hz)	SPL (dB)	Total beam angle (−6 dB)
Super Tweeter Murata [18]	15–100 k	90	60°
(A) Regal (RH-16E) [14]	280–20 k[a]	73	—[b]
(B) Prowave 400WB160 [15]	35–50 k	105 (10 V, 30 cm, 40 kHz)	50° (40 kHz)
Receivers	**Freq. range (Hz)**	**S (dB)**	**Total beam angle (−6 dB)**
Brüel and Kjaer (4939) [17]	20–140 k	1 V/Pa	Omni-directional
(B) Prowave 400WB160 [15]	35–50 k	−78 (0 dB, 10 V/Pa)	50° (40 kHz)
(C) Panasonic (WM-61) [16]	20–20 k[c]	−35 (0 dB, 1 V/Pa)	Omni-directional

[a] Maximum emission frequency tested up to 50 kHz with a minimum value at 37 kHz.
[b] Total beam angle tested at 35 kHz: 120°.
[c] Maximum reception frequency tested up to 40 kHz.

Fig. 2. General view of the developed LPS.

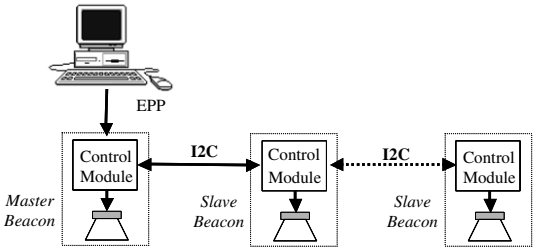


Fig. 3. Block diagram of the beacon system.

As previously mentioned, the determination of the absolute position is based on the measurement of the DTOAs between a reference beacon and the other ones. To solve the problem of the simultaneous emissions in beacons and their interferences in the receiver, a DS-CDMA technique is used, by modulating the ultrasonic signal with different codes for every beacon. The proposed system is flexible enough to allow different codification schemes. However, this work is focused on Kasami sequences, due to their good auto-correlation properties and the suitable cross-correlation results after a modulation process.

Every beacon consists of two main blocks: the control module and the transducer. The selected transducer is a Murata Super Tweeter [18]. Taken into account its frequency response, the transducer should be excited by the Kasami code modulated with a carrier of $f_c = 50$ kHz. A BPSK (Binary Phase Shift Keying) modulation allows the spectrum of the emitted signal to be centred at the desired frequency.

The control module permits to meet the requirements previously mentioned, providing the maximum autonomy and intelligence to every beacon. For that reason, an emitter hardware platform has been developed, where the algorithm to control the emission is implemented.

The features of the emitted signal are determined by the encoding and modulation process. Concerning encoding, it is necessary to define the type of the binary sequence used (Kasami code), and its length L (number of bits). The number p of different Kasami sequences to be considered depends on the number of beacons in the LPS. On the other hand, concerning modulation process, three main parameters are defined: the type of carrier, the number M of samples per carrier period, and the number N_c of carrier periods representing every transmitted bit. The modulation computed by a beacon is given by:

$$m[n] = \sum_{i=0}^{L-1} x[i] \cdot s[n - i \cdot N_c \cdot M] \quad (1)$$

where $x[n]$ is the binary sequence used to encode the emission (Kasami); L is the length of this sequence; and $s[n]$ is a symbol formed by N_c periods of the carrier (at a frequency $f_c = 1/T_c$), with M samples per period. The parameter M also means a ratio between the carrier frequency f_c and the later acquisition system f_S, so $M = f_S/f_c$.

It is possible to simplify the modulation process by using a digital squared carrier. Nevertheless, if it is desired a better shape of the frequency response, a sinusoidal carrier

should be selected. Both options are possible in the system: sinusoidal and squared carrier. The resolution is controlled by the number M of samples, whereas the number N_c allows to modify the transmitted energy, and also the bandwidth of the emission. Thus a high value of N_c implies a long emission interval, so the receiver often becomes more complex. These parameters should be configured depending on the ranging distances, and on the spectral characteristics of the ultrasonic transducer.

The block diagram of the developed computing platform for a beacon is depicted in Fig. 4. It can be divided into three blocks:

- FPGA, which controls the overall system: BPSK modulation, EPP and I2C management.
- D/A conversion block, where the emitting digital signal provided by the FPGA is converted into a suitable excitation for the transducer.
- I2C and EPP/JTAG interfaces, where protocols for both links are implemented.

The control module of the beacon has been designed in a Xilinx XC2S50E FPGA [19]. The main task is the implementation of a BPSK modulator and the management of the communication with a PC through the EPP and with the other beacons through the I2C bus. Fig. 5 shows the block diagram of this design.

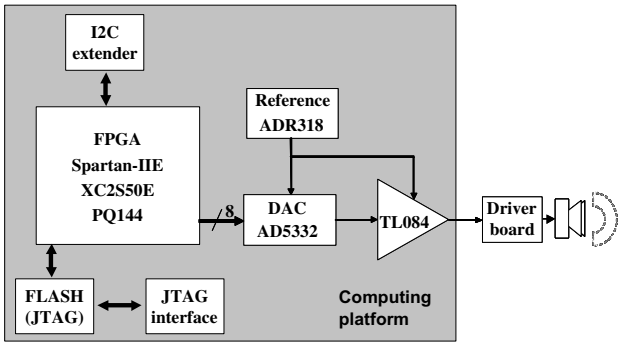


Fig. 4. Block diagram of the beacon computing platform.

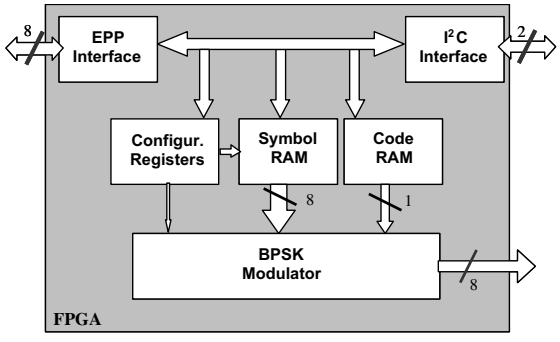


Fig. 5. Block diagram of the implemented design.

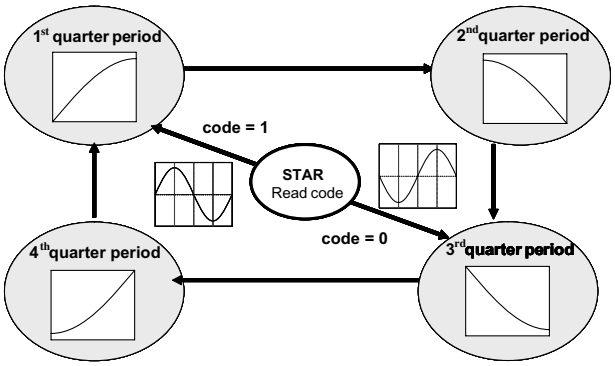


Fig. 6. Finite State Machine developed to manage the modulation process.

Communication interfaces have different operation modes depending on the type of beacon: master or slave. In master mode, the EPP interface is in charge of receiving all the configuration data from the PC; also, it distributes this data to the other slave beacons through the I2C interface. In slave mode, the EPP interface becomes disabled, and the I2C receives data from the master. Both interfaces can modify the configuration registers and memories.

The information, regarding the binary sequence to be emitted and the modulation symbol, is stored in two RAM memories. Both devices are dual-port, since one of them is used by the interfaces to write the carrier and the binary sequence, and the other port by the BPSK modulator to read them. In order to optimize the device utilization, and to exploit the symmetry of the used symbols, the carrier stored in the RAM memory is represented only for one quarter of period.

In the BPSK modulator, the reconstruction of the carrier symbol is managed by a Finite State Machine (FSM). The FSM is based on five states, as Fig. 6 depicts. An initial state indicates the transmission of a new bit code (either '1' or '0'). The other four states generate the corresponding quarters of the carrier symbol to obtain the modulated bit. The cycle is repeated as many times as the N_c configuration register requires it.

Every beacon has to emit its specific encoded signal periodically and continuously. The used master clock is 12.5 MHz since the modulated signal must be centred around 50 kHz and the maximum number M of samples per symbol has been fixed in 256. It is interesting to remark that the modulator also implements the DAC interface.

5. Reception stage

The ultrasonic emission from different beacons covers a determined area, depending on their placement. Inside this area, an ultrasonic receiver can detect the different transmissions and compute its local position. This reception stage is not only suitable for mobile robots, but also for other identification and localization applications (other kind of objects, people, etc.).

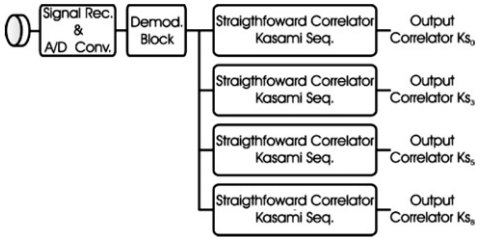


Fig. 7. Block diagram of the reception stage.

Fig. 7 shows the block diagram of the proposed reception stage. It is divided into three main modules, which are detailed next:

- An acquisition system.
- A BPSK demodulation.
- A correlation stage.

The acquisition system converts the signal received by the ultrasonic transducer into digital data. It consists of several stages (see Fig. 8). Firstly, a preamplifier module and an Automatic Control Gain (ACG) block make the signal suitable for the input range of the AD converter. Then, a band-pass filter is included, with cut-off frequencies at 35 and 65 kHz. Finally, the converter samples the input signal at a frequency f_S of 500 kHz, whereas the data width N_b for these samples is 8 bits.

A BPSK demodulator is implemented to extract the transmitted information from the received signal. This is a pre-synthesis configurable module, where some features can be modified according to the application. In that way, it is possible to change the data width N_b of the input, so the output is adjusted in concordance. Also, a sampled period of the carrier should be provided, so the demodulator can carry out the corresponding correlation – see (2).

$$d[n] = \sum_{i=0}^{N_c \cdot M-1} r[i+n] \cdot s[i] \tag{2}$$

where $d[n]$ is the output from the demodulation stage; $r[n]$ is the input signal coming from the acquisition stage; and $s[n]$ is the symbol of the demodulation, formed by N_c periods of the carrier, each one represented by M samples. Note that the parameter M is again $M = f_S/f_c$.

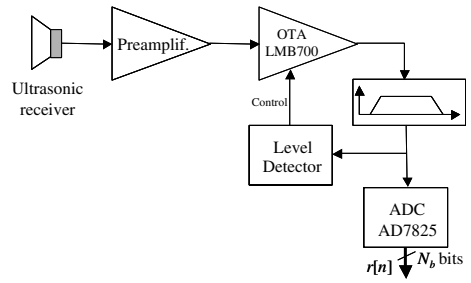


Fig. 8. Ultrasonic transducer driver and AD converter.

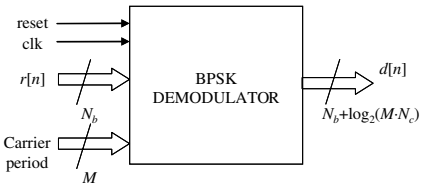


Fig. 9. General scheme of the BPSK demodulator.

Fig. 9 depicts the scheme of this module whereas the internal configuration can be observed in Fig. 10. Here, the described demodulator is particularized for $M = 4$ and $N_c = 2$. The multiplications in the structure shown in Fig. 4 can be reduced to additions and subtractions, by considering binary carriers.

The demodulated output (signal $d[n]$ in Fig. 9 and in Fig. 10) should be correlated in order to search for possible emissions coming from the different beacons existing in the environment. Since every beacon has a Kasami code assigned, as many correlators as beacons have to be implemented. This correlation process is described in (3).

$$t[n] = \sum_{i=0}^{L-1} d[i \cdot (N_c \cdot M) + n] \cdot x[i] \tag{3}$$

where $t[n]$ is the correlation output; $d[n]$ is the demodulation output; $x[n]$ is the Kasami code emitted by a beacon; and L is the length of the mentioned code.

Two approaches can be considered for the development of the correlation stage. Both of them require a buffer where the last $L \cdot N_c \cdot M$ samples coming from the demodulation output $d[n]$ are stored. On one hand, it is possible to organize a sequential operation mode, where the sample buffer is read p times to compute the p correlations with the corresponding Kasami codes (see Fig. 11). This option minimizes the device utilization, although it implies a higher frequency f_{cp} in the computational block. In this case, assuming a sampling input frequency for the buffer f_S, the computational frequency f_{cp} should meet (4).

$$f_{cp} = p \cdot L \cdot f_S \tag{4}$$

where p is the number of Kasami codes existing in the reception; and L is the length of the Kasami codes.

In the second approach, all the correlations are performed in parallel. The computational blocks are duplicated p times, one per each Kasami code, whereas the buffer is shared by all of them. Fig. 12 describes this implementation, where more resources are required but the frequency operation f_{cp} is not so constrained – see (5).

$$f_{cp} = L \cdot f_S \tag{5}$$

where L is the length of the Kasami codes; f_S is the operation frequency of the input buffer; and f_{cp} is the operation frequency in the computational block.

The parallel solution has been implemented in a Xilinx Spartan3e FPGA, assuming two Kasami codes ($p = 2$) with 255 bits ($L = 255$). The emission frequency f_c is 50 kHz; the

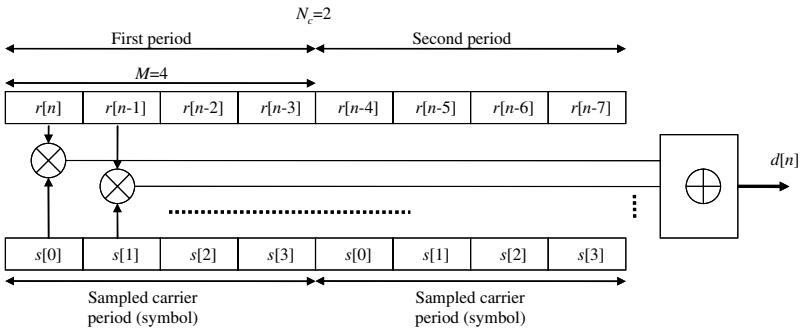


Fig. 10. Internal distribution of the BPSK demodulator.

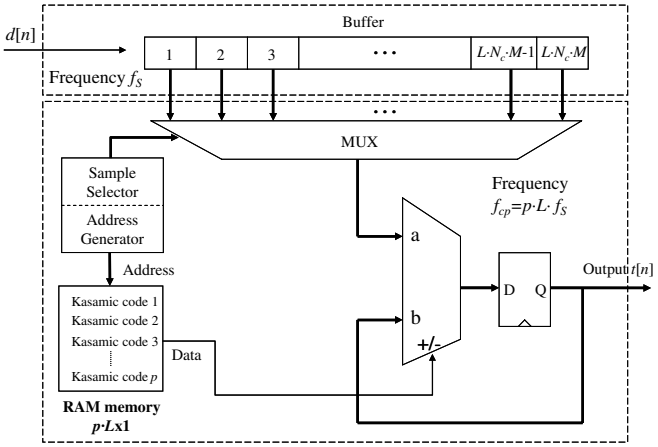


Fig. 11. Sequential implementation of the correlation module.

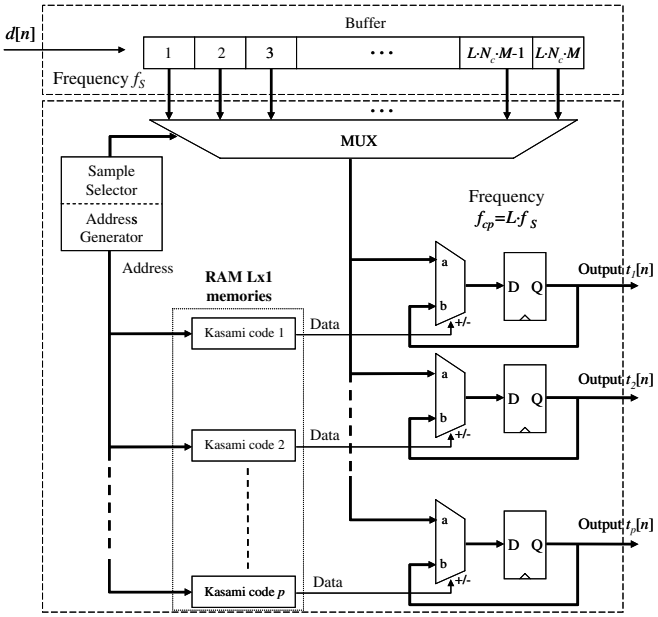


Fig. 12. Parallel implementation of the correlation module.

acquisition module has a sampling frequency f_s of 500 kHz, which implies 10 samples per carrier period ($M = 10$); and a resolution of 8 bits ($N_b = 8$). Also, the modulation has been configured with two periods per bit ($N_c = 2$). With these parameters, a global operating frequency f_S of 500 kHz has been selected for the whole system, except for the computational blocks in the correlation stage. For them, the selected frequency f_{cp} should be higher than 63.5 MHz in the parallel approach, and higher than 127 MHz in the sequential one.

6. Results

The implementation of the whole system, the beacon and the receiver, has been carried out in two different FPGA devices [19]. In the first one, the processing algorithm for emissions from every beacon has been developed in a Xilinx XC2S50E FPGA, achieving the device utilization shown in Table 2. In the second one, a Xilinx XC2S200E has been used for the receiver. Table 3 depicts the obtained results, assuming that the receiver only processes one Kasami code with the parallel approach.

In order to verify the feasibility of the system, some real experiments have been carried out, considering only five beacons. Assuming the beacon distribution shown in Fig. 2, whose dimensions can be observed in Fig. 13, the mobile robot was placed on the floor (approximated coordinates $x = 80$ cm, $y = 76$ cm and $z = -250$ cm). In this situation, with the aforementioned hardware, 100 measurements have been done.

Fig. 14a shows the signal received after one emission. Two transmissions can be identified, since an emission

Table 2
XC2S50E utilization for the implementation of the beacon module

Logic elements	Utilization
Slices	68%
Flip-flop slices	41%
Input LUT	60%
RAM	50%

Table 3
XC2S200E utilization for the implementation of the receiver

Logic elements	Utilization
Slices	99%
Flip-flop slices	6%
Input LUT	62%
RAM	0%

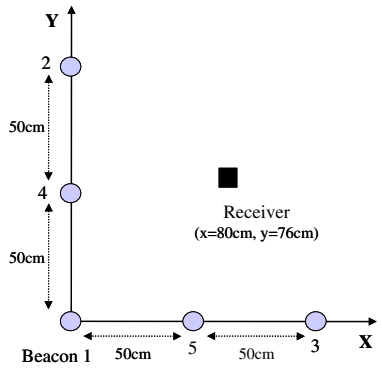


Fig. 13. Projection of the LPS and the approximated position of the mobile receiver.

process is carried out every 80 ms and the duration of the encoded transmission is $L \cdot N_c/f_c = 10.2$ ms (a sampling period T_s of 2 μs is assumed).

Fig. 14b, c and d show the outputs of the corresponding correlators for beacons 1, 2 and 3. Fig. 15 presents the validation and relative position of these peaks, where the DTOAs can be easily computed.

Fig. 16 details the obtained DTOAs assuming that the beacon 1 is the reference, after 100 consecutive emissions and for a static receiver. As can be observed, the variations in the DTOAs are below five samples (10 μs).

6.1. Determination of the absolute position

A hyperbolic trilateration technique is used to obtain the absolute position of the mobile receiver, by means of DTOAs [11]. Considering a group of p beacons ($p = 5$, minimum for 3D positioning), an equation system (6) can be proposed to obtain the absolute position of the mobile robot in 3D (x, y, z). The positions where the beacons are located (x_p, y_p, z_p) and the DTOAs between a reference beacon (for example, beacon 1) and the others are also taken into account.

$$\begin{aligned}
&(x-x_1)^2+(y-y_1)^2+(z-z_1)^2 = d_1^2\\
&(x-x_2)^2+(y-y_2)^2+(z-z_2)^2 = (d_1+c\cdot\Delta T_{12})^2\\
&(x-x_3)^2+(y-y_3)^2+(z-z_3)^2 = (d_1+c\cdot\Delta T_{13})^2\\
&\quad\cdots\qquad\cdots\qquad\cdots\quad=\qquad\cdots\\
&(x-x_P)^2+(y-y_P)^2+(z-z_P)^2 = (d_1+c\cdot\Delta T_{1P})^2
\end{aligned} \tag{6}$$

where d_1 is the distance to beacon 1 (reference one), considered as a variable; and ΔT_{12}, $\Delta T_{13}, \ldots, \Delta T_{1P}$ are the DTOAs measured, between the reference beacon and the others. The propagation speed c of ultrasounds in air has been fixed at 342 m/s. A lineal equation system is obtained, by developing the squared values, subtracting the first equation to the others, and regrouping terms. This system can be solved in a simple way, by the lineal minimum square method, as shown in (7).

$$\mathbf{A}\vec{\mathbf{x}} = \vec{\mathbf{b}} \tag{7}$$

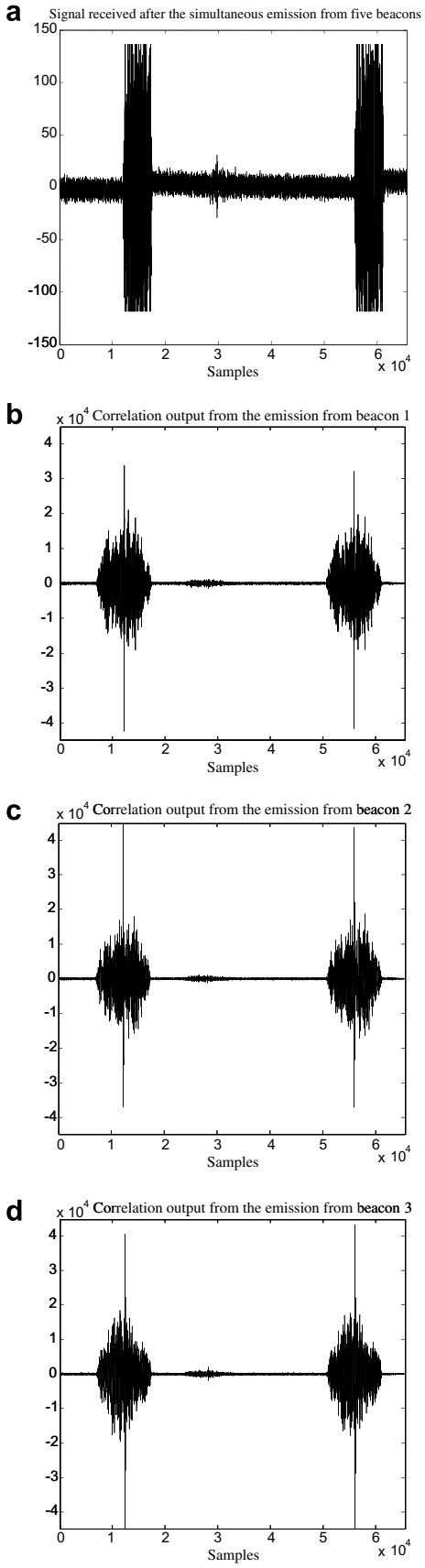


Fig. 14. Received signal $r[n]$ after the simultaneous emission from the five ultrasonic beacons (a); Correlation output $t[n]$ for emissions from beacons 1 (b), 2 (c) and 3 (d) (the others, 4 and 5, are similar).

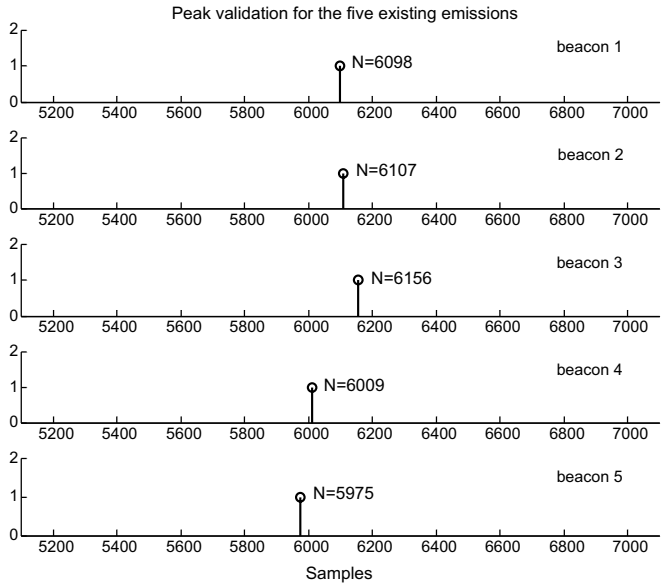


Fig. 15. Relative detection of the five peaks from the correlator Output signals.

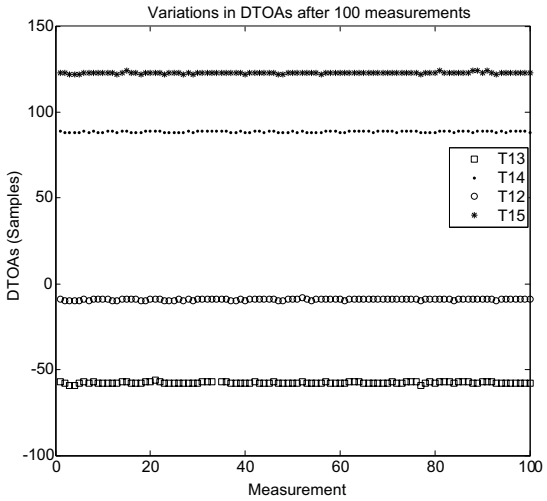


Fig. 16. Variations in DTOAs (beacon 1 considered as reference) after 100 emissions, with a fixed position for the mobile receiver.

where,

$$\mathbf{A} = \begin{bmatrix} 2x_1 - 2x_2 & 2y_1 - 2y_2 & 2z_1 - 2z_2 & -2c\Delta T_{12} \\ 2x_1 - 2x_3 & 2y_1 - 2y_3 & 2z_1 - 2z_3 & -2c\Delta T_{13} \\ \cdots & \cdots & \cdots & \cdots \\ 2x_1 - 2x_P & 2y_1 - 2y_P & 2z_1 - 2z_P & -2c\Delta T_{1P} \end{bmatrix} \quad (8)$$

$$\vec{\mathbf{x}} = \begin{bmatrix} x \\ y \\ z \\ d_1 \end{bmatrix} \quad (9)$$

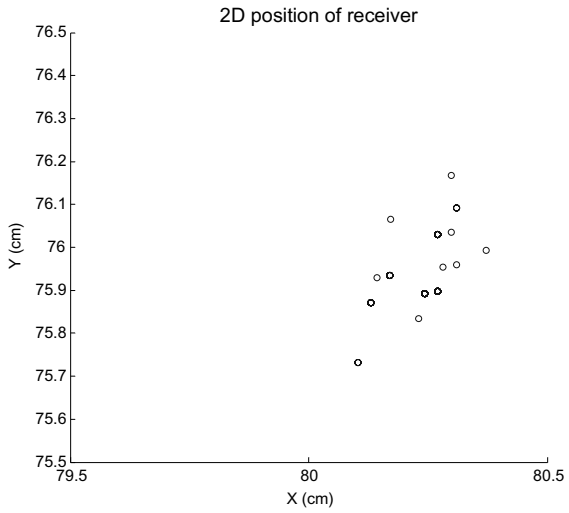


Fig. 17. Position obtained according to DTOAs shown in Fig. 16.

$$\vec{\mathbf{b}} = \begin{bmatrix} c^2\Delta T_{12} + x_1^2 + y_1^2 + z_1^2 - x_2^2 - y_2^2 - z_2^2 \\ c^2\Delta T_{13} + x_1^2 + y_1^2 + z_1^2 - x_3^2 - y_3^2 - z_3^2 \\ \cdots \\ c^2\Delta T_{1P} + x_1^2 + y_1^2 + z_1^2 - x_P^2 - y_P^2 - z_P^2 \end{bmatrix} \quad (10)$$

The solution is shown in (11):

$$\vec{\mathbf{x}} = (\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}}\vec{\mathbf{b}} \quad (11)$$

From the measurements shown in Fig. 16, their corresponding 2D locations have been plotted in Fig. 17, assuming the beacon distribution in Fig. 13. Many results are exactly the same point since their measurements are equal (the system presents a high repeatability). In the case of 2D positioning, only three DTOAs are strictly necessary. The reported position is always suitable (the error of the system is in the range of millimetres), with a small bias. Obviously, the position of the mobile receiver has influence on the precision of the positioning. In this way, Fig. 18

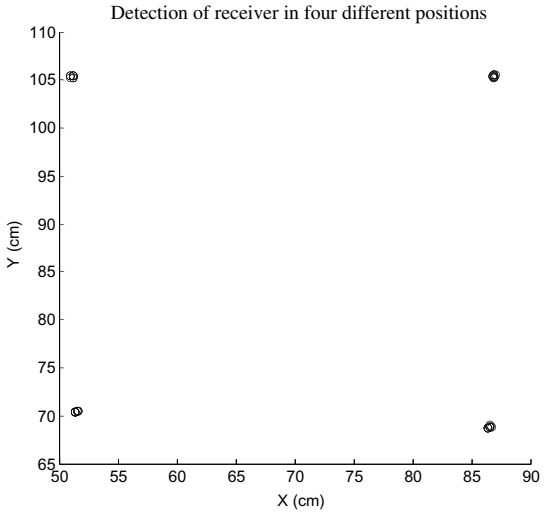


Fig. 18. Dispersion of values for positioning in four different places with the beacon network shown in Fig. 13.

shows the obtained positions for the case of a receiver placed at four different points, with 100 measurements at each one. As can be observed, the error is still in the same range of millimetres.

7. Conclusions

A local positioning system (LPS) for mobile robots, and/or for other objects or people, has been presented. It is based on simultaneous emissions from multiple ultrasonic beacons. In order to solve the problem of the simultaneous emission, the DS-CDMA technique has been applied, by modulating the ultrasonic signal (50 kHz) with a 255-bit Kasami code. Every beacon periodically and continuously transmits the resulting signal. The mobile receiver does not require to know the emission instant, and therefore, it is not necessary a synchronism trigger signal (RF, IR, etc.) between receiver and beacons. The system allows a non-limited number of mobile receivers working in the same environment.

The implementation of all the modules (beacons and mobile receiver) in specific hardware has been also presented, as well as some signals and results obtained from the real system. Next works will consist of installing the receiver on a mobile robot, as well as extending the beacon system to larger spaces.

Acknowledgements

This work has been possible by funding from the Madrid Community (ANESUS project: CAM-UAH2005/016), from the Spanish Ministry of Science and Technology (PARMEI project, ref. DIP2003-08715-C02-01), and from the IMSERSO (ARIADNA project, ref. 212/05).

References

[1] S. Aoyagi, M. Tacaño, H. Noto, Development of Indoor Mobile Robot Navigation System Using Ultrasonic Sensors, in: Proc. of the IFAC Workshop on Intelligent Components for Vehicles (ICV'98), 1998, pp. 345–349.

[2] N. Priyantha, N. Chakraborty, A. Balakrishnan, "The Cricket Location-Support System", Proceedings of 6th Annual International Conference on Mobile Computing and Networking (ACM MOBICOM), Boston, MA, 2000, pp.: 32–43.

[3] M. Addlesee, R. Curwen, S. Hodges, J. Newman, P. Steggles, A. Ward, A. Hopper, Sentient computer, IEEE Computer 37 (2) (2001) 95–97.

[4] T. Kasami, Weight distribution formula for some class of cyclic codes, Combinational Mathematics and its Applications, University of North Carolina Press, Chapel Hill, NC, 1969.

[5] K.W. Jörg, M. Berg, Using Pseudo-random Codes for Mobile Robot Sonar Sensing, in: Proc. of th 3rd IFAC Symposium on Intelligent and Autonomous Vehicles (IAV'98), 1998, pp. 231–236.

[6] A. Heale, L. Kleeman, A Sonar Sensing with Random Double Pulse Coding, in: Proc. of the Australian Conference on Robotics and Automation, 2000, pp. 81–86.

[7] H. Peremans, K. Audenaert, J. Van Campenhout, A high-resolution sensor based on tri-aural perception, IEEE Trans. on Robotics and Automation 9 (1) (1993) 36–48.

[8] J. Ureña, M. Mazo, J.J. García, A. Hernández, E. Bueno, Classification of reflectors with an ultrasonic sensor for mobile robots applications, Robotics and Autonomous Systems 29 (1999) 269–279.

[9] A. Hernández, J. Ureña, J.J. García, V. Díaz, M. Mazo, D. Hernanz, J.P. Dérutin, J. Serot, Ultrasonic signal processing using configurable computing, in: Proc. of the 15th Triennial World Congress of the International Federation of Automatic Control (IFAC'02), 2002.

[10] M. Hazas, A. Ward, A novel broadband ultrasonic location system, in: Proc. of UbiComp 2002: Ubiquitous Computing, 2002, pp. 264–280.

[11] L. Girod, D. Estrin, "Robust Range Estimation Using Acoustic and Multimodal Sensing", Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), vol.: 3, 2001 pp. 1312–1320.

[12] A. Mahajan, M. Walworth, 3D position sensing using the differences in the time-of-flights from a wave source to various receivers, IEEE Transactions on Robotics and Automation 17 (1) (2001) 91–94.

[13] P. Fan, M. Darnell, Sequence Design for Communications Applications, first ed., Research Studies Press LTD, Great Britain, 1996.

[14] Regal Electronics, Inc. High power miniature speakers. Product Specification, www.regalusa.com, 2006.

[15] Pro-Wave Electronic Corp. Air ultrasonic ceramic transducer. Product Specification, www.prowave.com.tw, 2006.

[16] Panasonic Corp. Omnidirectional back electret condenser microphone cartridge. Product Specification, www.panasonic.com/industrial/components, 2006.

[17] Brüel and Kjær Corp., Brüel&Kjær microphones. Product Specification, http://www.bksv.com, 2006.

[18] Murata Manufacturing Co., Ltd. Murata Super Tweeter ESTD01. Product Specification, www.murata.com, 1006.

[19] Xilinx, Inc. Spartan-II 2.5V FPGA family: functional description. Product Specification, San José (CA), 2002.

Efficient Correlator for LS Codes Generated from Orthogonal CSS

Mª Carmen Pérez, *Student Member, IEEE,* Jesús Ureña, *Member, IEEE,* Álvaro Hernández, *Member, IEEE,* Fernando J. Álvarez, *Member, IEEE,* Ana Jiménez, and Carlos De Marziani

***Abstract*—This paper presents a novel method to correlate Loosely Synchronized (LS) codes generated from mutually orthogonal Complementary Sets of Sequences (CSS). The proposed correlator significantly reduces the total number of multiplications and additions to be performed, in comparison with straightforward correlators. This fast correlator, together with the interference free window that LS codes exhibit, make them a good choice for next generation of Wireless communications.**

***Index Terms*—Code division multiaccess, correlation, pulse compression methods, sequences.**

I. Introduction

LS codes exhibit an interference free window (IFW) where the aperiodic auto-correlation and cross-correlation functions become zero. Consequently, the inter-symbol interference, and the multiple-access interference can be completely eliminated if the time dispersion of the channel is within this IFW. Such LS codes can be used in quasi-synchronous code division multiple access systems to mitigate the multipath problem and co-channel interference [1]. In [2] a systematic method to construct LS codes from complementary sets of sequences (CSS) is presented. Usually, the detection of LS codes is carried out by means of straightforward implementations, which require high computational cost that constrains the system throughput. In this paper, an efficient correlator for LS codes generated from CSS is proposed, which notably reduces the number of operations to be carried out, and thus high speeds and long sequences are possible. This method offers the same results as [3] when a family of four LS codes is considered (see [4] for the correspondig efficient correlator of LS codes generated from Golay pairs).

II. LS Codes

A set of K LS codes with length L $\{G = g_k[l];\ 0 \leq k \leq K-1;\ 0 \leq l \leq\ L-1\}$ is a set of sequences composed of three types of elements $g_k[l] \in \{0, +1, -1\}$, whose aperiodic correlation functions are zero in a certain vicinity of the zero shift, that is:

$$\begin{aligned} C_{g_{k_1},g_{k_2}}[\tau] &= \textstyle\sum_{l=0}^{L-1-\tau} g_{k_1}[l]g_{k_2}[l+\tau] = \\ &= \begin{cases} C_0, & for\ \tau = 0, \quad k_1 = k_2 \\ 0, & for\ 1 \leq |\tau| \leq W, \quad k_1 = k_2 \\ 0, & for\ 0 \leq |\tau| \leq W, \quad k_1 \neq k_2 \end{cases} \end{aligned} \tag{1}$$

where $C_{x,y}$ is the aperiodic correlation between x and y, τ is an integer representing the time scale, $C_0 = \sum_{l=0}^{L-1} |g_k[l]|$ is the number of non-zero elements in the LS code, and $2W+1$ denotes the length of the IFW.

Recently, a method to construct LS codes from the well-known CSS [5] has been proposed. Let $\{S_i,\ 0 \leq i \leq N-1\}$ be N orthogonal complementary sets, each one with N sequences of length L_0, i. e., $S_i = \{s_{i,j}[l];\ 0 \leq j \leq N-1;\ 0 \leq l \leq L_0 - 1\}$, where $s_{i,j}[l] \in \{+1, -1\}$, $N = 2^P$ and $L_0 = N^M$, being $P,\ M \in \mathbb{Z}^+$. Then, the algorithm proposed in [2] allows the generation of a set of LS codes as follows:

$$G_{n,m}(z) = \sum_{i=0}^{N-1} h_{m,i} \cdot z^{-iL_0} \cdot [\sum_{j=0}^{N-1} z^{-j(NL_0+W)} S_{\pi_{n,i},j}(z)] \tag{2}$$

where $G_{n,m}(z)$ represents the Z transform of $\{g_{n,m}(\tau);\ 0 \leq n,m \leq N-1\}$, $h_{m,i} \in \{+1, -1\}$, are the elements of a $N \times N$ Hadamard matrix, $\pi_{n,i} = (n,i) \mod N$, and $W \leq L_0 - 1$. Equation (2) shows that LS codes are constructed by concatenation of the $s_{i,j}$ sequences following the order established by $\pi_{n,i}$, and with the polarity indicated by the coefficients $h_{m,i}$ of a matrix whose rows are mutually orthogonal (a Hadamard matrix). Note that $(N-1)$ sets of W zeros appear in determined positions of the LS code to obtain the IFW in the correlation functions. Thus, the final length of the LS code is $L = N^2L_0 + (N-1)W$. For better understanding, consider the example of Fig. 1 where $K = 4$ LS codes with length $L = 39$ are obtained from $N = 2$ CSS of length $L_0 = 8$, assuming $W = 7$.

III. New Correlation Algorithm

An implementation based on a digital filter is proposed for the generation algorithm defined in (2), called efficient LS generator (ELSG) and obtained according to the following steps (see Fig. 2):

Step 1: Generation of $s_{i,j}$, that is, the N orthogonal CSS, each one with N sequences of length L_0. For that purpose N efficient sequence generators (ESSG) can be used (see [6] for details on the structure of the ESSG).

Manuscript received June 27, 2008. The associate editor coordinating the review of this letter and approving it for publication was H.-H. Chen. This work has been possible thanks to the Community of Madrid (FPI grant, INCUBUS project: CAM-UAH2006/016), and to the Spanish Ministry of Science and Technology (RESELAI project: TIN2006-14986-CO2-01).

M. C. Pérez, J. Ureña, A. Hernández, and A. Jiménez are with the Department of Electronics, University of Alcalá, Escuela Politécnica, Campus Universitario s/n, 28805, Alcalá de Henares, Madrid, Spain (e-mail: {carmen, urena, alvaro, ajimenez}@depeca.uah.es).

F. J. Álvarez is with the Department of Electrical Engineering, Electronics and Automatics, University of Extremadura, Facultad de Ciencias, Campus Universitario s/n, 06071, Badajoz, Spain (e-mail: fafranco@unex.es).

C. De Marziani is with the Department of Electronics Engineering, National University of Patagonia San Juan Bosco (Argentina), Ciudad Universitaria, Ruta Provincial nº1 s/n, Km 4, 9000 Comodoro Rivadavia (Chubut), Argentina (e-mail: marziani@unpata.edu.ar).

Digital Object Identifier 10.1109/LCOMM.2008.081000

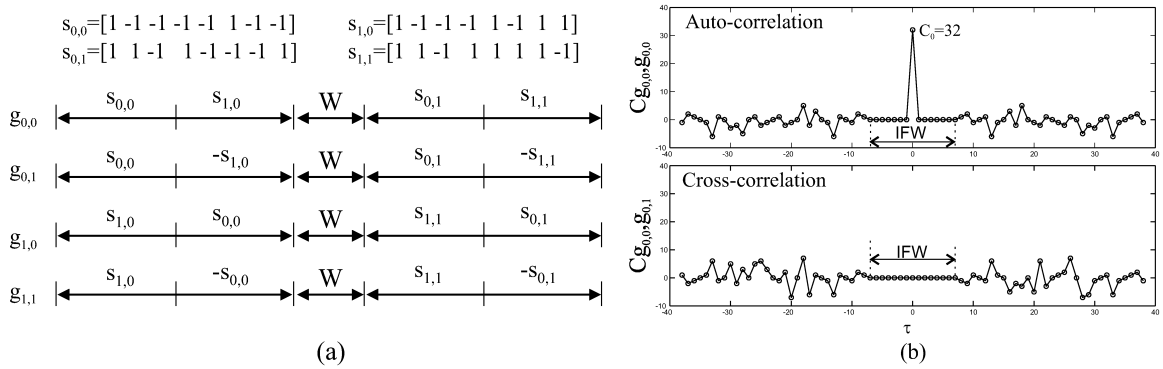


Fig. 1. a) Generation of $K = 4$ LS codes using $N = 2$ orthogonal CSS of length $L_0 = 8$, $W = 7$, $h_{0,0} = 1$, $h_{0,1} = 1$, $h_{1,0} = 1$, $h_{1,1} = -1$, and $\pi_{0,0} = 0$, $\pi_{0,1} = 1$, $\pi_{1,0} = 1$, $\pi_{1,1} = 0$; b) Example of auto-correlation and cross-correlation function of the generated codes.

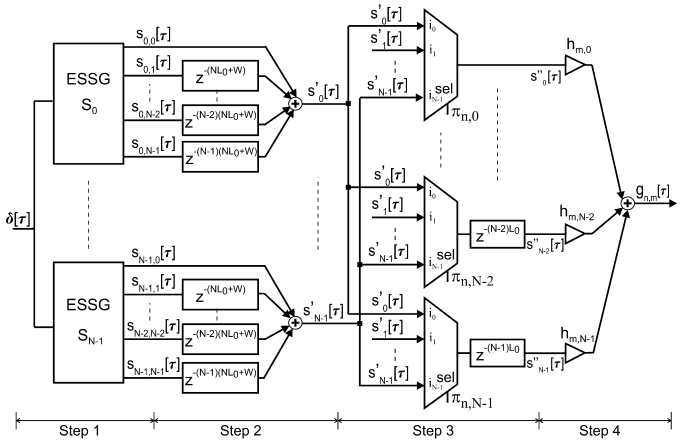


Fig. 2. Block diagram of the Efficient LS code Generator (ELSG).

Step 2: Every ESSG simultaneously generates the N sequences of a set, which are added with different delays. As a result N new sequences are generated, whose Z transforms are:

$$S'_i(z) = \sum_{j=0}^{N-1} z^{-j(NL_0+W)} S_{i,j}(z);\ 0 \le i \le N-1 \qquad (3)$$

Step 3: A set of N multiplexers governed by the elements $\pi_{n,i}$ determines the delay z^{-iL_0} to apply to each one of these sequences S'_i, according to:

$$S''_i(z) = z^{-iL_0} \cdot \sum_{j=0}^{N-1} z^{-j(NL_0+W)} S_{\pi_{n,i},j}(z);\ 0 \le i \le N-1 \qquad (4)$$

Step 4: Finally, every output of step 3 is multiplied by the corresponding element $h_{m,i}$ of the Hadamard matrix and the resulting values are added according to (2) so as to obtain the LS sequence.

Since the input of this filter is a delta pulse and the output is $G_{n,m}(z)$, the transfer function of the filter is $G_{n,m}(z)$ and it behaves as a correlator filter matched to the reversed sequence $g_{n,m}(L-1-\tau)$. Equation (2) can be modified to generate the reversed sequence instead of the direct one, thus obtaining a correlator filter matched to the direct sequence. This modification consists of interchanging the order of the delays in (2) and use the reversed sequences $s_{i,j}(L_0-1-\tau)$. The output of such correlator filter can be expressed as:

$$C_{R,G_{n,m}}(z) = \sum_{i=0}^{N-1} h_{m,i} \cdot z^{-(N-1-i)L_0} \cdot \cdot [\sum_{j=0}^{N-1} z^{-(N-1-j)(NL_0+W)} C_{R,S_{\pi_{n,i},j}}(z)] \qquad (5)$$

where $C_{R,S_{\pi_{n,i},j}}(z)$ represents the Z-transform of the correlation between the input signal $r(\tau)$ and every sequence $s_{\pi_{n,i},j}$ of the N orthogonal CSS S_i. The implementation of the correlation algorithm proposed in (5) can be achieved in four steps as it has been done for the ELSG, thus, obtaining an efficient correlator for LS codes (ELSC). This filter is represented in Fig. 3, and the steps are detailed below.

Step 1: Correlation of the input signal $r(\tau)$ with every one of the N orthogonal CSS. To perform these correlations N efficient set of sequences correlators (ESSC) can be used (see [6] for further information regarding to the ESSC). Every ESSC performs the correlation of the input signal simultaneously with the N sequences of a CSS, so at the end of this step the correlations $C_{r,s_{i,j}}(\tau);\ 0 \le i,j \le N-1$ are available.

Step 2: The former correlations are added with different time shifts, obtaining N partial correlations whose Z-transforms are:

$$C_{R,S'_i}(z) = \sum_{j=0}^{N-1} z^{-(N-1-j)(NL_0+W)} C_{R,S_{i,j}}(z);\ 0 \le i \le N-1 \qquad (6)$$

Step 3: A set of N multiplexers governed by the vector $\pi_{n,i}$ determines the delay $z^{-(N-1-i)L_0}$ to be applied to every partial correlation C_{R,S'_i} as follows:

$$C_{R,S''_i}(z) = \\ = z^{-(N-1-L_0)} \cdot [\sum_{j=0}^{N-1} z^{-(N-1-j)(NL_0+W)} C_{R,S_{\pi_{n,i},j}}(z)];\\ 0 \le i \le N-1 \qquad (7)$$

Step 4: Every output of step 3 is multiplied by the corresponding element $h_{m,i}$ of the Hadamard matrix, and finally they are added according to (5) to get the final correlation result.

A comparison among the number of multiplications and additions to be performed in the ELSC and in the straightforward implementation can be found in Table I.

TABLE I
NUMBER OF OPERATIONS AND MEMORY REQUIREMENTS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED ELSC, AS A FUNCTION OF N, M AND W (THE CSS LENGTH IS $L_0 = N^M$, AND W IS THE LENGTH OF THE SET OF ZEROS THAT APPEAR IN THE LS CODE; THE FINAL LENGTH OF THE LS CODE IS $L = N^2 L_0 + (N-1)W$).

	Straightforward	ELSC	
Products	$N^{(M+2)}$	Step 1	$\frac{M \cdot N^2}{2} log_2(N)$
		Step 4	N
		TOTAL	$N(1 + \frac{M \cdot N}{2} log_2(N))$
Additions	$N^{(M+2)} - 1$	Step 1	$M \cdot N^2 \cdot log_2(N)$
		Step 2	$N(N-1)$
		Step 4	$N - 1$
		TOTAL	$M \cdot N^2 \cdot log_2(N) + N^2 - 1$
Memory Position's requirements	$N^{M+2} + (N-1)W$	Step 1	$\frac{3N^{M+1} - 2N^M - N^2}{2}$
		Step 2	$\frac{(N^{M+1}+W)(N^3 - N^2)}{2}$
		Step 3	$\frac{N^M (N^2 - N)}{2}$
		TOTAL	$\frac{N^{M+4} - N^{M+3} + N^{M+2} + 2N^{M+1} - 2N^M + WN^3 - N^2(1+W)}{2}$

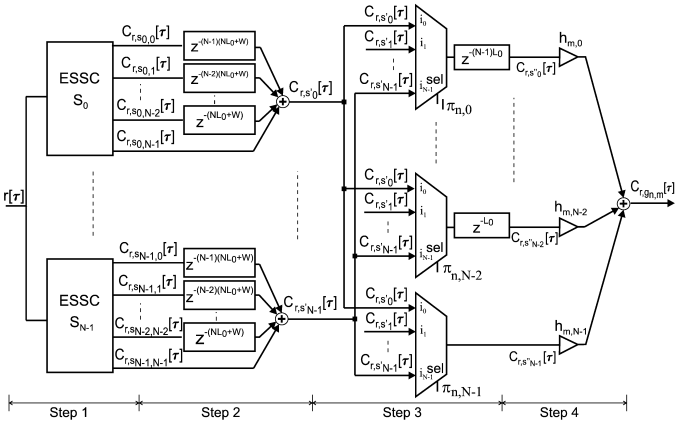


Fig. 3. Block diagram of the Efficient LS code Correlator (ELSC).

Actually, both implementations can be achieved without any multiplication, since LS code coefficients are $\{0, +1, -1\}$, and thus multiplications are reduced to additions and subtractions. In any case, the number of operations is notably reduced with the proposed ELSC. Table I also shows the total number of memory positions necessary to store the data in the straightforward implementation and in the ELSC. The proposed ELSC requires less than $\frac{N^2}{2}$ times the memory needed by the straightforward implementation.

As an example, in case of a family of LS codes with $K = N^2 = 16$ and $L = 4861$, and then generated from $N = 4$ orthogonal CSS (with length $L_0 = 256$ bits), the operations and memory positions required are:

- For ELSC: 211 operations and 33504 memory positions.
- For straightforward implementation: 9721 operations and 4861 memory positions.

The decrease in the number of performed operations involved by the ELSC is more important than the increase in memory requirements in terms of hardware implementation.

IV. CONCLUSION

A new efficient correlator for LS codes has been presented. This implementation achieves a lower computational load than the straightforward implementation. The proposed algorithm takes advantage of the orthogonal CSS properties from which LS codes are derived to reduce the number of operations to be carried out, allowing the implementation in hardware devices as field programmable gate arrays and the use of longer sequences.

REFERENCES

[1] D. Li, "The perspectives of large area synchronous CDMA technology for the fourth-generation mobile radio," *IEEE Commun. Mag.*, vol. 41, no. 3, pp. 114–118, Mar. 2003.
[2] C. Zhang, X. Lin, S. Yamada, and M. Hatori, "General method to construct LS codes by complete complementary sequences," *IEICE Trans. Commun.*, vol. E88-B, no. 8, pp. 3484–3487, Aug. 2005.
[3] S. Stańzak, H. Boche, and M. Haardt, "Are LAS-codes a miracle?" in *Proc. IEEE Global Telecommunications Conf. (GLOBECOM 2001)*, vol. 1, pp. 589–593, San Antonio, USA, Nov. 2001.
[4] M. C. Pérez, J. Ureña, A. Hernández, A. Jiménez, W. P. Marnane, and F. Álvarez, "Efficient real-time correlator for LS sequences," in *Proc. IEEE International Symposium on Industrial Electronics (ISIE 2007)*, vol. 1, pp. 1663–1668, Vigo, Spain, June 2007.
[5] C. C. Tseng and C. L. Liu, "Complementary sets of sequences," *IEEE Trans. Inform. Theory*, vol. IT-18, no. 5, pp. 644–652, Sept. 1972.
[6] C. Marziani, J. Ureña, A. Hernández, M. Mazo, F. Álvarez, J. García, and P. Donato, "Modular architecture for efficient generation and correlation of complementary sets of sequences," *IEEE Trans. Signal Processing*, vol. 55, no. 5, pp. 2323–2337, May 2007.

Efficient Generation and Correlation of Sequence Pairs with Three Zero Correlation Zones

Mª Carmen Pérez*, *Student Member, IEEE,* Jesús Ureña, *Member, IEEE,* Álvaro Hernández, *Member, IEEE,* Ana Jiménez and Carlos De Marziani

***Abstract*—This paper presents three novel methods to generate and correlate efficiently sequence pairs with three zero correlation zones in the sum of their aperiodic correlation functions. These sequences have taken great importance in communications systems when the maximum transmission delay is less than the length of these zones, because the Multiple Access Interference (MAI) and the Inter Symbol Interference (ISI) can be removed. The proposed algorithms are based on the properties of Complementary Sets of Sequences (CSS) and they offer an efficient hardware implementation of the corresponding generators and correlators that reduces the total number of operations to be performed in comparison with straightforward implementations. That makes possible real time operation as well as the use of very long sequences.**

***Index Terms*—Code division multiaccess, sequences, generators, correlators, pulse compression methods.**

EDICS number: DSP-FAST: Fast algorithms for digital signal processing.

I. Introduction

CODES with favorable correlation properties are of substantial interest in applications based on Code Division Multiple Access (CDMA) techniques, such as radar, sonar, cryptography or mobile and wireless communications. In fact, the performance of these applications is largely constrained by the chosen codes, requiring ideal properties of Auto-Correlation (AC) and Cross-Correlation (CC) to avoid both the Inter Symbol Interference (ISI) and the Multiple Access Interference (MAI). Another limitation is imposed by the matched-filter technique used in the reception stage to detect the emitted codes, as it means an increase of signal processing tasks that can put into risk the real-time operation capability of the system. Hence, a lot of efforts are being devoted to find new encoding schemes with good correlation properties together with efficient hardware implementations of the corresponding matched-filter correlators.

Traditional CDMA pseudo-random codes such as m-sequences [1], Gold codes [2] or Kasami sequences [3] exhibit non-zero off-peak AC and CC values and were designed based only on their periodic correlation properties. Walsh-Hadamard sequences [4] and Gold orthogonal codes [5] have perfect orthogonality at zero time delay and they are only useful in case of accurate synchronism. Consequently, these codes limit the achievable performance in many real applications where aperiodic emissions, asynchronous detection, multi-path effects, etc. are quite common.

Since with one single sequence it is impossible to remove completely and simultaneously sidelobes both in the AC and CC functions [6], several authors rely on complementary codes [7]-[9]. Complementary codes were firstly considered by Golay [10] and each one consists of a pair of binary sequences with the same length whose addition of AC and CC functions have perfect properties. Nevertheless, they provide only two mutually uncorrelated pairs and are not suitable for multi-user environments. Complementary sets of sequences (CSS) [11] are a generalization of Golay pairs containing more than two sequences. The elimination of the constrains in the number of sequences yields on a high process gain and also more than two mutually uncorrelated sets. However, the number of uncorrelated sets is limited by the number of sequences for every set. Besides, in many systems it is not possible to simultaneously transmit the sequences of a set, thus emission schemes that arrange the sequences of the set by concatenation or by interleaving are used. These transmission schemes degrade the ideal properties of the sum of the correlation functions, so ISI and MAI appear [12].

Recently, interest in codes with Zero Correlation Zones (ZCZ), as Loosely Synchronous codes, has considerably increased [13]-[16]. Such kind of codes are thought for applications in quasi-synchronous CDMA (QS-CDMA) systems where the relative time delay among different users does not exceed a certain limit. The correlation functions of these codes have zero values around the in-phase shift, providing null ISI and MAI when the length of the ZCZ is larger than the multipath spread and the relative multiple access delays. However, in periodic emissions it is necessary to insert guard intervals (or zero gaps) to maintain its properties [14], [17]. As a solution, in [18] new sequence pairs with ZCZ in the middle and terminal parts of the sum of their aperiodic AC and CC functions have been proposed. These sequence pairs avoid to insert extra guard intervals in case of periodic emissions and allow more simultaneous users than Golay pairs. In practice, if the correlations are computed by means of conventional straightforward correlators [19], which require high computational cost, the system throughput is constrained.

The effort done in the code design process has also involved the research on generators and correlators that allow

This work has been possible thanks to the Spanish Ministry of Science and Technology (RESELAI project: TIN2006-14986-CO2-01).

M. C. Pérez* (corresponding author), J. Ureña, A. Hernández, A. Jiménez are with the Department of Electronics, University of Alcalá, Escuela Politécnica, Campus Universitario s/n, Ctra. Madrid-Barcelona, Km. 33600, 28871, Alcalá de Henares, Madrid, Spain (phone: 0034-918856540; fax: 0034-918856591; e-mail: {carmen, urena, alvaro, ajimenez}@depeca.uah.es).

C. De Marziani is with the Department of Electronics, National University of Patagonia San Juan Bosco (Argentina), Ciudad Universitaria, Ruta Provincial nº1 s/n, Km 4, 9000 Comodoro Rivadavia (Chubut), Argentina (phone: 0054-2974550836; fax: 0054 2974550836; e-mail: marziani@unpata.edu.ar).

Manuscript received November 18, 2008.

an efficient hardware implementation[1]. A solution to reduce the number of operations can be the use of a correlator based on the Fast Fourier Transform (FFT) algorithm [20]-[22]. It implies the implementation of a FFT, a vector multiplication and an inverse FFT. However, the number of operations is reduced at the expense of performing a large number of multiplications (in contrast to the straightforward implementation that requires only additions and subtractions when binary sequences are used). Some authors propose efficient algorithms for specific codes. As an example, in [20] an algorithm based on the Fast Walsh Transform (FWT) is proposed, consisting only in additions and subtractions, for the periodic correlation of m-sequences. A similar proposal is described in [23] for orthogonal Gold codes. In [21] and [24] an Efficient Golay Correlator (EGC) is proposed for the detection of Golay complementary pairs. The EGC reduces the number of additions and subtractions to perform from L (the sequence length) to $2 \cdot log_2(L)$. Recently, this algorithm has been extended for the efficient correlation of complementary sets of any number $M = 2^m$ of sequences with length $L = M^N$, being $m, N \in \mathbb{Z}^+$ [25], and implemented in configurable hardware as shown in [26]. Furthermore, in [27] an efficient correlator for Loosely Synchronous codes has been proposed.

This paper describes different methods to generate Three Zero Correlation Zone (T-ZCZ) codes based on the efficient generation algorithm of CSS proposed in [25]. With these new methods it is possible to obtain lower interferences in the Interference Zone (IZ) and narrower IZ than those obtained with previous methods [18]; and, what is of great importance, an efficient correlator can be derived. The paper is organized as follows. Section II introduces T-ZCZ pairs. Section III describes the way to construct two Generator Matrixes from which T-ZCZ pairs are derived. In Section IV the elements of these matrixes are grouped using three different methods to obtain new T-ZCZ family pairs. Section V deals with efficient generators and correlators for the sequence pairs obtained with every method. Performance for each of the proposed correlators is discussed in Section VI and, finally, in Section VII conclusions are outlined.

II. Definition of T-ZCZ codes

A set of M binary pairs $\{E_k = (e_{k,0}, e_{k,1}); 0 \le k \le M-1\}$ with length L and elements $\{-1, +1\}$, constitutes a T-ZCZ set if the Sum of the aperiodic AC Functions (SACF) and the Sum of the aperiodic CC Functions (SCCF) of the sequences from each pair have the following characteristics, (1) and (2) respectively:

$$\begin{aligned} SACF[\tau] &= \sum_{l=0}^{L-1-\tau} e_{k,0}[l]e_{k,0}[l+\tau] + \\ &\quad + \sum_{l=0}^{L-1-\tau} e_{k,1}[l]e_{k,1}[l+\tau] \\ &= \begin{cases} 2L, & \tau = 0 \\ 0, & 1 \le |\tau| \le Z_C \\ 0, & L - Z_L \le |\tau| \le L \end{cases} \end{aligned} \tag{1}$$

[1]Any implementation that reduces the number of operations to be performed in comparison with the required by a straighforward correlator or generator can be considered *efficient* [23].

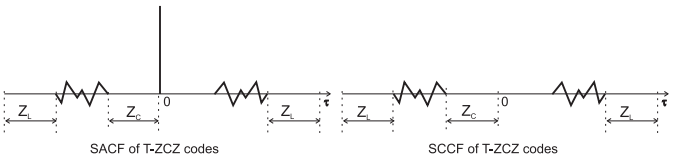


Fig. 1. SACF and SCCF of T-ZCZ pairs.

$$\begin{aligned} SCCF[\tau] &= \sum_{l=0}^{L-1-\tau} e_{k,0}[l]e_{k',0}[l+\tau] + \\ &\quad + \sum_{l=0}^{L-1-\tau} e_{k,1}[l]e_{k',1}[l+\tau] \qquad \forall k \ne k' \\ &= \begin{cases} 0, & 0 \le |\tau| \le Z_C \\ 0, & L - Z_L \le |\tau| \le L \end{cases} \end{aligned} \tag{2}$$

where Z_C is the size of the ZCZ that appears at each side of the origin and Z_L represents the size of the ZCZ in the two terminal parts of the sum of correlation functions, as can be seen in Fig. 1. When $Z_L + Z_C = L - 1$ T-ZCZ turns into Golay pairs.

III. T-ZCZ Generator Matrixes

In [18], T-ZCZ pairs are obtained from a generator matrix Δ. In practice, the iterative method proposed to construct this Δ matrix coincides with the iterative method proposed by Tseng and Liu (see [11], Sec. IV) for generating a matrix Δ containing M mutually Uncorrelated CSS (UCSS). In this paper, new generator matrixes are obtained by using CSS generated from the recursive method proposed in [25]. The advantage of using these other CSS is that an Efficient Generator and an Efficient Correlator for CSS (ESSG and ESSC respectively) are available, and thus, efficient structures for the construction and detection of T-ZCZ are possible.

The following steps summarize the procedure to obtain the recursive equations that provide M uncorrelated CSS $\{S_i; 0 \le i \le M-1\}$; each one with M sequences of length L_0, i. e. $\{S_i = s_{i,j}[l]; 0 \le j \le M-1; 0 \le l \le L_0\}$, where $s_{i,j} \in \{-1, +1\}$, $M = 2^m$ and $L_0 = M^N$, being $m, N \in \mathbb{Z}^+$ (see [25] for further information):

Step 1: Firstly, a matrix with coefficients $w_{i,n} \in \{-1, 1\}$ is initialized.

$$\Lambda_{(n)}^{(1} = \begin{pmatrix} 1 & w_{1,n} \\ 1 & -w_{1,n} \end{pmatrix} \tag{3}$$

Step 2: After m iterations, a new $\Lambda_{(n)}^{(m}$ matrix can be derived as:

$$\Lambda_{(n)}^{(m} = \begin{pmatrix} \Lambda_{(n)}^{(m-1} & \otimes & \left(w_{m,n} \cdot \left(-\Lambda_{(n)}^{(m-1}\right)\right) \\ \Lambda_{(n)}^{(m-1} & \otimes & \left(w_{m,n} \cdot \Lambda_{(n)}^{(m-1}\right) \end{pmatrix}; \quad m = 2, 3, 4 \cdots, \infty \tag{4}$$

Where $\otimes$ denotes bit interleaving. The resulting $\Lambda_{(n)}^{(m}$ matrix shows the polarity with which the bits of iteration $(n-1)$ are going to be linked in the iteration (n).

Step 3: The recursive equations from which the matrix Δ is obtained are:

$$\begin{aligned} \mathbf{S}_{i_{(0)}} &= \delta[\tau] \cdot (1\ 1\ 1\ \cdots\ 1)_{1 \times M} \\ \mathbf{S}_{i_{(n)}} &= \Lambda_{(n)}^{(m} \cdot \mathbf{D}_M \cdot \mathbf{S}_{i_{(n-1)}} \end{aligned} \tag{5}$$

where $\mathbf{S_{i_{(n)}}}$, $\mathbf{S_{i_{(n-1)}}}$ and $\mathbf{D_M}$ are defined in (6); D_n is any permutation of the set $\{M^0, M^1, \cdots, M^{N-1}\}$, $M = 2^m$;

and $N \in \mathbb{Z}^+$ indicates the total number of iterations ($n = 1, 2, \cdots, N$).

$$\mathbf{S}_{i_{(n-1)}} = \begin{pmatrix} s_{i,0_{(n-1)}}[\tau] \\ s_{i,1_{(n-1)}}[\tau] \\ \vdots \\ s_{i,M-1_{(n-1)}}[\tau] \end{pmatrix}; \quad \mathbf{S}_{i_{(n)}} = \begin{pmatrix} s_{i,0_{(n)}}[\tau] \\ s_{i,1_{(n)}}[\tau] \\ \vdots \\ s_{i,M-1_{(n)}}[\tau] \end{pmatrix}$$

$$\mathbf{D}_M = \begin{pmatrix} \delta[\tau] & 0 & 0 & \cdots & 0 \\ 0 & \delta[\tau - D_n] & 0 & \cdots & 0 \\ 0 & 0 & \delta[k - 2 \cdot D_n] & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \delta[\tau - (M-1) \cdot D_n] \end{pmatrix} \tag{6}$$

Step 4: Two different Δ matrixes can be obtained depending on the way in which CSS are combined:

a) $\mathbf{\Delta^{(M,L_0}}$*:* Every row of the $(M \times L_0)$ dimensional generator matrix $\mathbf{\Delta^{(M,L_0}}$ is a sequence of the CSS obtained in step 3, as indicated in (7):

$$\mathbf{\Delta^{(M,L_0}} = S_i^{(M,L_0} = \begin{pmatrix} s_{i,0}[\tau] \\ s_{i,1}[\tau] \\ \vdots \\ s_{i,M-1}[\tau] \end{pmatrix} = \\ = \begin{pmatrix} s_{i,0}[0] & s_{i,0}[1] & \cdots & s_{i,0}[L_0-1] \\ s_{i,1}[0] & s_{i,1}[1] & \cdots & s_{i,1}[L_0-1] \\ \vdots & \vdots & \ddots & \vdots \\ s_{i,M-1}[0] & s_{i,M-1}[1] & \cdots & s_{i,M-1}[L_0-1] \end{pmatrix} \tag{7}$$

b) $\mathbf{\Delta^{(M,M\cdot L_0}}$*:* In this case, M UCSS $\{S_i; 0 \le i \le M-1\}$ are considered for the construction of a new generator matrix $\mathbf{\Delta^{(M,M\cdot L_0}}$ with size $(M \times M \cdot L_0)$, as shown in (8). M different UCSS can be obtained by carrying out all the possible combinations of the coefficients $\{w_{1,1}, w_{2,1}, \cdots, w_{m,1}\}$ at the iteration $n = 1$ in step 3, and by fixing the others to any value. The subscript i, that means the order in which the M CSS are linked, comes from the decimal representation of $\{w_{1,1}, w_{2,1}, \cdots, w_{m,1}\}$, where $w_{1,1}$ is the most significant bit and considering the -1 values as 0 values.

$S_0|S_1$ denotes the concatenation of the sets S_0 and S_1; as a result, the first L_0 columns of $\mathbf{\Delta^{(M,M\cdot L_0}}$ correspond to S_0, the next L_0 columns to S_1, and so on.

For a better understanding, consider an example for the construction of a) $\mathbf{\Delta^{(4,16}}$ and b) $\mathbf{\Delta^{(4,4\cdot 16}}$, obtained from complementary sets of $M = 4$ sequences with length $L_0 = 16$. From steps 1 to 3, the following equations are obtained (9):

$$\begin{aligned} &s_{i,0_{(0)}}[\tau] = s_{i,1_{(0)}}[\tau] = s_{i,2_{(0)}}[\tau] = s_{i,3_{(0)}}[\tau] = \delta[\tau] \\ &s_{i,0_{(n)}}[\tau] = s_{i,0_{(n-1)}}[\tau] - w_{2,n} \cdot s_{i,1_{(n-1)}}[\tau - D_n] + \\ &\quad + w_{1,n} \cdot s_{i,2_{(n-1)}}[\tau - 2 \cdot D_n] + w_{2,n} \cdot w_{1,n} \cdot s_{i,3_{(n-1)}}[\tau - 3 \cdot D_n] \\ &s_{i,1_{(n)}}[\tau] = s_{i,0_{(n-1)}}[\tau] - w_{2,n} \cdot s_{i,1_{(n-1)}}[\tau - D_n] - \\ &\quad - w_{1,n} \cdot s_{i,2_{(n-1)}}[\tau - 2 \cdot D_n] + w_{2,n} \cdot w_{1,n} \cdot s_{i,3_{(n-1)}}[\tau - 3 \cdot D_n] \\ &s_{i,2_{(n)}}[\tau] = s_{i,0_{(n-1)}}[\tau] + w_{2,n} \cdot s_{i,1_{(n-1)}}[\tau - D_n] + \\ &\quad + w_{1,n} \cdot s_{i,2_{(n-1)}}[\tau - 2 \cdot D_n] + w_{2,n} \cdot w_{1,n} \cdot s_{i,3_{(n-1)}}[\tau - 3 \cdot D_n] \\ &s_{i,3_{(n)}}[\tau] = s_{i,0_{(n-1)}}[\tau] + w_{2,n} \cdot s_{i,1_{(n-1)}}[\tau - D_n] - \\ &\quad - w_{1,n} \cdot s_{i,2_{(n-1)}}[\tau - 2 \cdot D_n] - w_{2,n} \cdot w_{1,n} \cdot s_{i,3_{(n-1)}}[\tau - 3 \cdot D_n] \end{aligned} \tag{9}$$

Though $N = 2$ iterations, the length of the four sequences is $L_0 = 16$. When $w_{1,1} = w_{1,2} = w_{2,1} = w_{2,2} = -1$ the generator matrix $\mathbf{\Delta^{(4,16}}$ in (10) is obtained. For the construction of the generator matrix $\mathbf{\Delta^{(4,4\cdot 16}}$ in (11), 4 UCSS are employed. They have been generated from (9) by combining all the possible values of $\{w_{1,1}, w_{2,1}\}$ whereas fixing the other coefficients to $w_{1,2} = w_{2,2} = -1$.

a) $\mathbf{\Delta^{(4,16}} =$

$$\begin{pmatrix} 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 \end{pmatrix} \tag{10}$$

b) $\mathbf{\Delta^{(4,4\cdot 16}} =$

$$= \begin{pmatrix} 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{11}$$

IV. T-ZCZ CODE CONSTRUCTIONS

For each generator matrix three different construction methods of binary T-ZCZ codes are possible, and thus six different families of T-ZCZ codes can be derived. A subscript has been added to indicate the number (1, 2 or 3) of the method used, and an apostrophe to specify that the generator matrix $\mathbf{\Delta^{(M,M\cdot L_0}}$ has been used (the absence of apostrophe corresponds to T-ZCZ pairs obtained from $\mathbf{\Delta^{(M,L_0}}$). For example, T-ZCZ$_1$ represents T-ZCZ codes obtained by the method 1 using $\mathbf{\Delta^{(M,L_0}}$; whereas T-ZCZ$_{1'}$ pairs are also obtained by the method 1, but from the matrix $\mathbf{\Delta^{(M,M\cdot L_0}}$. Furthermore, the notation T-ZCZ$_1(Z_C, Z_L, M, L)$ is used to denote a family of M T-ZCZ pairs with length L generated by the method 1 using $\mathbf{\Delta^{(M,L_0}}$, and with size of the lateral and central ZCZ equal to Z_L and Z_C, respectively. Similar notation is used for the methods 2 and 3.

For brevity, only the constructions with better performance in terms of hardware implementation are considered in this paper. These constructions are: T-ZCZ$_1$, T-ZCZ$_{2'}$ and T-ZCZ$_{3'}$.

A. T-ZCZ$_1$: *Method 1 using* $\mathbf{\Delta^{(M,L_0}}$

The first $\frac{M}{2}$ pairs of a family T-ZCZ$_1$ $\{E_k = (e_{k,0}, e_{k,1}); 0 \le k \le M-1\}$, obtained through the method 1 from only one CSS $S_i = \{s_{i,0}, s_{i,1}, \cdots, s_{i,M-1}\}$ with length $L_0 > M$, are composed by the sequences: $E_0 = (e_{0,0}, e_{0,1}) = (s_{i,0}, s_{i,1})$, $E_1 = (e_{1,0}, e_{1,1}) = (s_{i,2}, s_{i,3})$, $\cdots$, $E_{M/2-1} = (e_{M/2-1,0}, e_{M/2-1,1}) = (s_{i,M-2}, s_{i,M-1})$. The next $\frac{M}{2}$ pairs are obtained from the sequences of a set $S_{i'} = \{\overline{s_{i,1}}, -\overline{s_{i,0}}, \overline{s_{i,2}}, -\overline{s_{i,1}}, \cdots, \overline{s_{i,M-1}}, -\overline{s_{i,M-2}}\}$ uncorrelated with S_i; where $\overline{s_{i,j}}[\tau] = s_{i,j}[L_0 - 1 - \tau]$ imply reversing the order of the elements in $s_{i,j}$ and $-s_{i,j}$ multiplying by -1 the sequence $s_{i,j}$. Thus, $E_{M/2} = (e_{M/2,0}, e_{M/2,1}) = (\overline{s_{i,1}}, -\overline{s_{i,0}})$, $\cdots$, $E_{M-1} = (e_{M-1,0}, e_{M-1,1}) = (\overline{s_{i,M-1}}, -\overline{s_{i,M-2}})$, as shown in (12); where $\lfloor x \rfloor$ represents the largest integer less than or equal to x.

$$\boldsymbol{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0})} = (S_0 \mid S_1 \mid \cdots \mid S_{M-1}) = \begin{pmatrix} s_{0,0} & s_{1,0} & \cdots & s_{M-1,0} \\ s_{0,1} & s_{1,1} & \cdots & s_{M-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{0,M-1} & s_{1,M-1} & \cdots & s_{M-1,M-1} \end{pmatrix} =$$

$$= \begin{pmatrix} s_{0,0}[0] & \cdots & s_{0,0}[L_0-1] & s_{1,0}[0] & \cdots & s_{1,0}[L_0-1] & \cdots & s_{M-1,0}[0] & \cdots & s_{M-1,0}[L_0-1] \\ s_{0,1}[0] & \cdots & s_{0,1}[L_0-1] & s_{1,1}[0] & \cdots & s_{1,1}[L_0-1] & \cdots & s_{M-1,1}[0] & \cdots & s_{M-1,1}[L_0-1] \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ s_{0,M-1}[0] & \cdots & s_{0,M-1}[L_0-1] & s_{1,M-1}[0] & \cdots & s_{1,M-1}[L_0-1] & \cdots & s_{M-1,M-1}[0] & \cdots & s_{M-1,M-1}[L_0-1] \end{pmatrix} \tag{8}$$

TABLE I
T-ZCZ$_1(6,6,4,16)$ OBTAINED WITH COEFFICIENTS $w_{1,1}=w_{2,1}=w_{1,2}=w_{2,2}=-1$

$e_{0,0}$	1 1 –1 –1 1 1 1 1 –1 1 –1 –1 –1 1 –1 1
$e_{0,1}$	1 1 1 1 1 1 –1 –1 –1 1 –1 1 –1 1 1 –1
$e_{1,0}$	1 –1 –1 1 1 –1 1 –1 –1 –1 1 1 –1 –1 –1 –1
$e_{1,1}$	1 –1 1 –1 1 –1 –1 1 –1 –1 –1 –1 –1 –1 1 1
$e_{2,0}$	–1 1 1 –1 1 –1 1 –1 –1 –1 1 1 1 1 1 1
$e_{2,1}$	–1 1 –1 1 1 –1 –1 1 –1 –1 –1 –1 1 1 –1 –1
$e_{3,0}$	1 1 –1 –1 –1 –1 –1 –1 1 –1 –1 1 –1 1 –1 1
$e_{3,1}$	1 1 1 1 –1 –1 1 1 1 –1 1 –1 –1 1 1 –1

$$\begin{cases} e_{\lfloor \frac{j}{2} \rfloor, j \bmod 2} = s_{i,j}, & 0 \le j \le M-1 \\ e_{\lfloor \frac{j+M}{2} \rfloor, (j+1) \bmod 2} = (-1)^{j+1} \cdot \overline{s_{i,j}}, & 0 \le j \le M-1 \end{cases} \tag{12}$$

The length of the T-ZCZ$_1$ codes is equal to the length of the CSS, that is, $L = L_0 = M^N$, where N has to be greater than 1 to obtain ZCZ in the sum of correlation functions. On the other hand, the sizes of these ZCZ for a family of M T-ZCZ$_1$ codes have been computed and are given by (13) and (14).

- When N is even

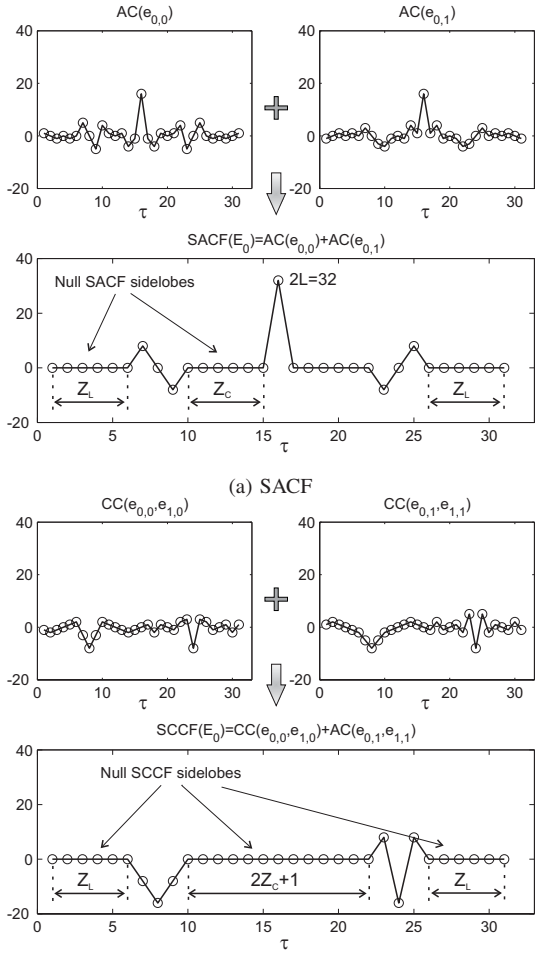
$$Z_C = Z_L = \left\lceil \frac{\frac{M}{2}(M^N-1)}{M^2-1} \right\rceil \cdot 3 \tag{13}$$

- When N is odd and $N > 1$

$$Z_C = \left\lceil \frac{\frac{M}{2}(M^{(N-1)}-1)}{M^2-1} \right\rceil \cdot 3$$
$$Z_L = \left\lceil \frac{\frac{M}{2}(M^{(N-1)}-1)}{M^2-1} \right\rceil \cdot 3 + \frac{M^N}{2} \tag{14}$$

In Table I an example of sequence pairs T-ZCZ$_1(6,6,4,16)$ is shown. Fig. 2 illustrates the SACF and SCCF of the generated T-ZCZ$_1(6,6,4,16)$ codes.

When the number of simultaneous users is less than M, it is possible to choose the combination of T-ZCZ$_1$ pairs that minimizes the number and magnitude of non-zero values in the interference zone (IZ). For example, there are combinations of two T-ZCZ$_1$ pairs with null SCCFs for all shifts. In general cases, in the IZ null values also appear in some positions; and what is more important, the magnitude of the highest interference in the IZ is half the value of the main SACF peak. This means that it could be still possible to detect the arrival of the code, even if one of the receptions comes out of the central ZCZ.

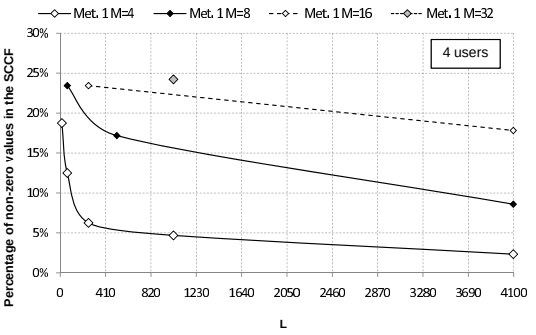


(a) SACF

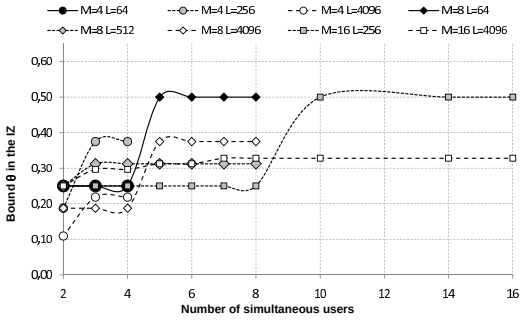
(b) SCCF

Fig. 2. Example of SACF and SCCF of T-ZCZ$_1(6,6,4,16)$ pairs.

Fig. 3 sum up the behavior in the IZ of T-ZCZ$_1$ pairs of different lengths: 3.a corresponds to the percentage of interferences in the SCCF when four simultaneous users are in the system; and 3.b shows the maximum correlation bound θ in the IZ. This bound θ is a commonly used figure of merit to quantify the performance of a family with M codes L-bits long each. It is defined as $\theta = \max(\theta_{AC}, \theta_{CC})$, where θ_{AC} stands for the maximum sidepeak obtained in all the SACFs and θ_{CC} is the maximum value obtained in the SCCFs between all the pairs contained in the family. In Fig. 3.b this bound is divided



(a) Percentage of interferences in the SCCF.



(b) Bound θ in the IZ.

Fig. 3. Performance of T-ZCZ$_1$ pairs in the IZ as depending on the number of M pairs in the family, its length L and the number of simultaneous users.

by the SACF main peak.

B. T-ZCZ$_{2'}$*: Method 2 using* $\mathbf{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0})}$

In order to construct T-ZCZ$_{2'}$ pairs, the generator matrix $\mathbf{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0})}$ is divided into two new submatrixes of the same size: $\mathbf{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0})} = [\mathbf{\Delta}_{\mathbf{L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})} \mid \mathbf{\Delta}_{\mathbf{R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})}]$. The left part $\mathbf{\Delta}_{\mathbf{L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})}$ corresponds to sequences $e_{j,0}$, and the right one $\mathbf{\Delta}_{\mathbf{R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})}$ to sequences $e_{j,1}$; thus the j-th pair is $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{2}-1,j}], [s_{\frac{M}{2},j}|s_{\frac{M}{2}+1,j}|\cdots|s_{M-1,j}])$, having every sequence a length of $L = \frac{M}{2}L_0$. The sequence pairs T-ZCZ$_{2'}(2,2,4,8)$ are shown in Table II as an example.

The sizes of the central and lateral ZCZ have been computed, finding out that $Z_L = Z_C$. For a family with M codes, the ZCZ size can be obtained from (15) depending on the number $n = 1, 2, \cdots, N$ of iterations in the construction of the original CSS.

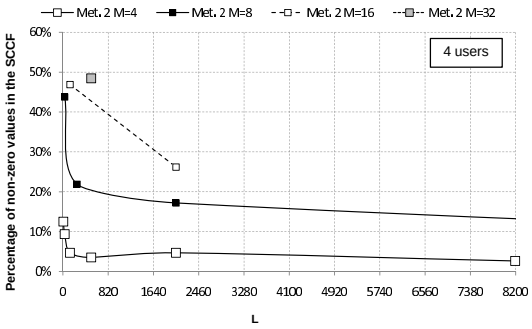
$$\begin{aligned} n = 1 \quad &\rightarrow \quad X_{(1)} = \frac{M}{2} \\ n = 2 \quad &\rightarrow \quad X_{(2)} = \frac{M^2}{2} + 1 \\ n \geq 3 \quad &\rightarrow \quad X_{(n)} = \frac{M^n}{2} + M^{n-2} + X_{(n-2)} \end{aligned} \tag{15}$$

Where $Z_C = Z_L = X_{(N)}$, and $X_{(N)}$ are obtained after $n = N$ iterations.

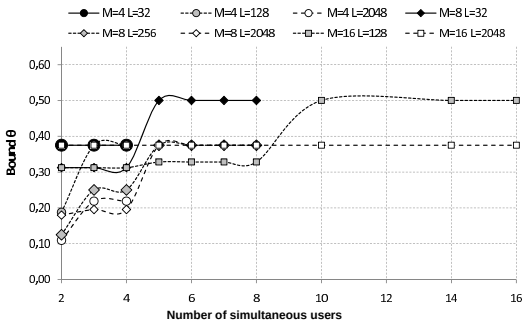
The percentage of interferences in the SACF and SCCF of T-ZCZ$_{2'}$ pairs of length $L = \frac{M^{N+1}}{2}$ is similar to the percentage obtained with T-ZCZ$_1$ pairs of length $L = M^N$. Fig. 4.a shows an example of the percentage of interferences in the

TABLE II
T-ZCZ$_{2'}(2,2,4,8)$ WHERE S_i HAVE BEEN OBTAINED WITH ALL POSSIBLE COMBINATIONS OF $w_{1,1}$ AND $w_{2,1}$.

$e_{0,0}$	1	1	-1	-1	1	-1	-1	1
$e_{0,1}$	1	1	1	1	1	-1	1	-1
$e_{1,0}$	1	1	1	1	1	-1	1	-1
$e_{1,1}$	1	1	-1	-1	1	-1	-1	1
$e_{2,0}$	1	-1	-1	1	1	1	-1	-1
$e_{2,1}$	1	-1	1	-1	1	1	1	1
$e_{3,0}$	1	-1	1	-1	1	1	1	1
$e_{3,1}$	1	-1	-1	1	1	1	-1	-1



(a) Percentage of interferences in the SCCF.



(b) Bound θ in the IZ.

Fig. 4. Performance of T-ZCZ$_2$ pairs in the IZ depending on the number of M pairs in the family, its length L and the number of simultaneous users.

SCCF for four simultaneous T-ZCZ$_{2'}$ pairs. Note that these interferences are enclosed in the IZ. The highest magnitudes of the interferences in the IZ are shown in Fig. 4.b according to the parameter θ, and for different simultaneous users.

C. T-ZCZ$_{3'}$*: Method 3 using* $\mathbf{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0})}$

T-ZCZ$_{3'}$ pairs are obtained by partitioning the submatrixes $\mathbf{\Delta}_{\mathbf{L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})}$ and $\mathbf{\Delta}_{\mathbf{R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})}$ into two equally-sized matrixes, so $\mathbf{\Delta}_{\mathbf{L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})} = [\mathbf{\Delta}_{\mathbf{0_L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{4}}\mathbf{L_0})} \mid \mathbf{\Delta}_{\mathbf{1_L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{4}}\mathbf{L_0})}]$ and $\mathbf{\Delta}_{\mathbf{R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})} = [\mathbf{\Delta}_{\mathbf{0_R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{4}}\mathbf{L_0})} \mid \mathbf{\Delta}_{\mathbf{1_R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{4}}\mathbf{L_0})}]$. A family of T-ZCZ$_{3'}$ pairs, $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{4}-1,j}], [s_{\frac{M}{4},j}|s_{\frac{M}{4}+1,j}|\cdots|s_{\frac{M}{2}-1,j}])$, with $0 \leq j \leq M-1$ and $M > 2$, is obtained from $\mathbf{\Delta}_{\mathbf{L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0})}$. In a similar way, another family $E_j = (e_{j,0}, e_{j,1}) =$

TABLE III
T-ZCZ$_{3'}(6,6,4,16)$ WHERE S_i HAVE BEEN OBTAINED WITH COEFFICIENTS $w_{1,2} = w_{2,2} = -1$ AND ALL POSSIBLE COMBINATIONS OF $w_{1,1}$ AND $w_{2,1}$.

$e_{0,0}$	1	1	−1	−1	1	1	1	1	−1	1	1	−1	−1	1	−1	1
$e_{0,1}$	1	1	−1	−1	−1	−1	−1	−1	−1	1	1	−1	1	−1	1	−1
$e_{1,0}$	1	1	1	1	1	1	−1	−1	−1	1	−1	1	−1	1	1	−1
$e_{1,1}$	1	1	1	1	−1	−1	1	1	−1	1	−1	1	1	−1	−1	1
$e_{2,0}$	1	−1	−1	1	1	−1	1	−1	−1	−1	1	1	−1	−1	−1	−1
$e_{2,1}$	1	−1	−1	1	−1	1	−1	1	−1	−1	1	1	1	1	1	1
$e_{3,0}$	1	−1	1	−1	1	−1	−1	1	−1	−1	−1	−1	−1	−1	1	1
$e_{3,1}$	1	−1	1	−1	−1	1	1	−1	−1	−1	−1	−1	1	1	−1	−1

$([s_{\frac{M}{2},j}|s_{\frac{M}{2}+1,j}|\cdots|s_{\frac{3M}{4}-1,j}],[s_{\frac{3M}{4},j}|s_{\frac{3M}{4}+1,j}|\cdots|s_{M-1,j}])$ is derived from $\mathbf{\Delta}_{\mathbf{R}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{Lo}}$.

The size of the ZCZ has been computed and is specified in (16) where, after $n = 1, 2, \cdots, N$ iterations, $X_{(N)} = Z_C = Z_L$ is obtained.

$$\begin{aligned} n = 1 \quad &\rightarrow \quad X_{(1)} = \tfrac{M}{4} \\ \text{if } n \text{ is even} \quad &\rightarrow \quad X_{(n \bmod 2=0)} = M \cdot X_{(n-1)} + 2 \\ \text{if } n \text{ is odd} \quad &\rightarrow \quad X_{(n \bmod 2=1)} = M \cdot X_{(n-1)} + \tfrac{M}{4} \end{aligned} \tag{16}$$

An illustration of T-ZCZ$_{3'}(6,6,4,16)$ can be found in Table III. On the other hand, Fig. 5 shows the behavior of T-ZCZ$_{3'}$ regarding the interferences in the IZ. It can be observed that the bound θ is lower than 0.5, and thus, it could be still possible to distinguish the main SACF peak from the sidelobes in the IZ.

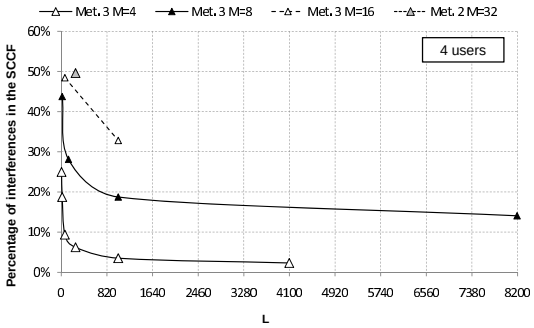
V. EFFICIENT CORRELATION ALGORITHMS

Since the proposed T-ZCZ pairs are based on CSS, it is possible to use the efficient generation and correlation algorithms proposed for CSS. Thus, an effective reduction in the number of operations to be performed in the generation and correlation of T-ZCZ can be obtained if it is compared with a straightforward implementation. In this work, a regular structure is provided for the efficient correlation of T-ZCZ$_1$, T-ZCZ$_{2'}$ and T-ZCZ$_{3'}$ pairs. For brevity, only the efficient generator of T-ZCZ$_1$ codes is explained. The efficient generator filters of T-ZCZ$_{2'}$ and T-ZCZ$_{3'}$ pairs can be easily derived from their corresponding efficient correlators by interchanging the order of the delays as shown in [24].

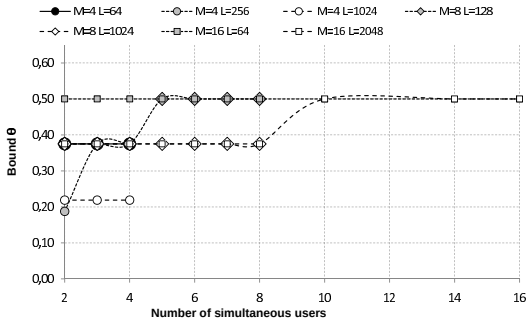
A. *Efficient Generation and Correlation algorithm of* T-ZCZ$_1$ *pairs*

An implementation based on a digital filter is proposed for the generation algorithm defined in (12), called *Efficient Three Zero correlation zones Generator* ETZG$_1$, and shown in Fig. 6. The proposed algorithm uses the Efficient Set of Sequences Generator (ESSG) described in [25] to reduce the number of operations to be carried out.

The first $\frac{M}{2}$ pairs of a set T-ZCZ$_1$ correspond to the sequences of a CSS $S_i = \{s_{i,0}, s_{i,1} \cdots, s_{i,M-1}\}$, and can be



(a) Percentage of interferences in the SCCF.



(b) Bound θ in the IZ.

Fig. 5. Performance of T-ZCZ$_{3'}$ pairs in the IZ depending on the number of M pairs in the family, its length L and the number of simultaneous users.

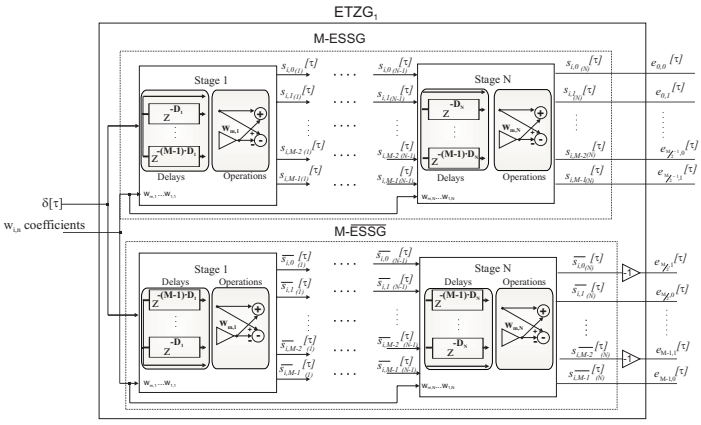


Fig. 6. Block diagram for the Efficient T-ZCZ$_1$ Generator (ETZG$_1$).

obtained by using an ESSG. Next $\frac{M}{2}$ sequence pairs belong to a set $S_{i'} = \{\overline{s_{i,1}}, -\overline{s_{i,0}}, \overline{s_{i,2}}, -\overline{s_{i,1}}, \cdots, \overline{s_{i,M-1}}, -\overline{s_{i,M-2}}\}$. The reversed sequences $\overline{s_{i,j}} = s_{i,j}[L-\tau-1]$ can be obtained by interchanging the order of the coefficients $\{0, 1, \cdots, M-2, M-1\}$, that multiplies delays D_n at every stage of the ESSG, to $\{M-1, M-2, \cdots, 1, 0\}$. The generation algorithm of reversed sequences $\overline{s_{i,j}}$ is represented as $\overline{\text{ESSG}}$ and requires the same resources and number of operations as the original ESSG. As shown in (12), the even outputs of the $\overline{\text{ESSG}}$ are multiplied by -1 to obtain the last $\frac{M}{2}$ T-ZCZ$_1$.

Since the input of the ETZG$_1$ filter is a delta pulse and its outputs are $\{(e_{k,0}, e_{k,1}); 0 \leq k \leq M-1\}$, it behaves as a correlator filter matched to the reversed sequences $\{(\overline{e_{k,0}}, \overline{e_{k,1}}); 0 \leq k \leq M-1\}$. If the order of the delays

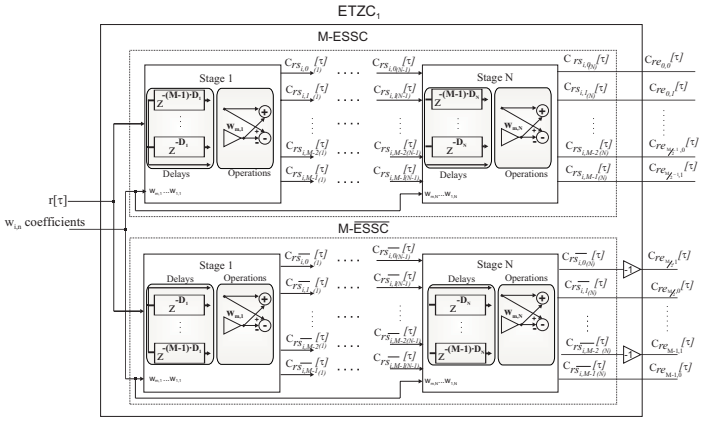


Fig. 7. Block diagram for the Efficient T-ZCZ$_1$ Correlator (ETZC$_1$).

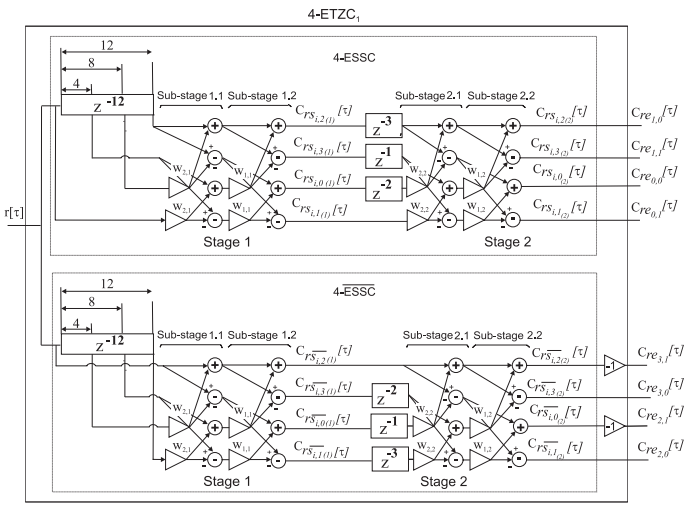


Fig. 8. Example of a 4-ETZC$_1$.

in the ETZG$_1$ is commuted to obtain the reversed sequences $(\overline{e_{k,0}}, \overline{e_{k,1}})$, a correlator with the direct sequences $(e_{k,0}, e_{k,1})$ is obtained [24]. This filter, shown in Fig. 7 and called *Efficient Three Zero correlation zones Correlator* ETZC$_1$, simultaneously performs the correlation of the input signal $r[\tau]$ with the sequences of the M pairs of a family T-ZCZ$_1$.

The simultaneous correlation of the input signal $r[\tau]$ with every one of the M sequences $s_{i,j}$ is carried out by the Efficient Set of Sequences Correlator (ESSC); obtaining $C_{r,s_{i,j}}[\tau]$, and thus, $C_{r,e_{\left\lfloor \frac{j}{2} \right\rfloor, j \bmod 2}} = C_{r,s_{i,j}}$, where $0 \leq j \leq M-1$. By changing the order of the coefficients that multiplies the delays D_n at every stage of the ESSC to $\{0, 1, \cdots, M-2, M-1\}$, a correlator $\overline{\text{ESSC}}$ matched to the set $S_{i'}$ is achieved. Even outputs of the $\overline{\text{ESSC}}$ are negated so as to obtain the correlation with the last $\frac{M}{2}$ pairs $C_{r,e_{\left\lfloor \frac{j+M}{2} \right\rfloor, (j+1) \bmod 2}} = (-1)^{j+1} \cdot C_{r,\overline{s_{i,j}}}$, where $0 \leq j \leq M-1$. Fig. 8 is an example of a ETZC$_1$ for a family of $M = 4$ codes T-ZCZ$_1$ with length $L = 16$. Note that the first delay stage can be shared between the ESSC and the $\overline{\text{ESSC}}$ blocks.

A comparison among the number of multiplications and additions to be performed in the ETZC$_1$ and in the straightforward implementation can be found in Table IV. Actually, both implementations can be achieved without any multiplications, since T-ZCZ code coefficients are $\{-1, 1\}$, and therefore multiplications are reduced to additions and subtractions. Table IV also shows the total number of bits necessary to store the data in the straightforward and in the ETZC$_1$ implementation, assuming an initial data-width equal to 1 (then this value has to be multiplied by the width of the A/D converter that could be in the input). For an easy comparison, Table V shows the computation number and memory bits required by the straightforward and the proposed efficient implementation for families with $M = 4$ and $M = 16$ codes of length $L = M^N$. As can be observed, there is a remarkable improvement when using the ETZC$_1$.

TABLE IV
NUMBER OF OPERATIONS AND MEMORY BITS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED ETZC$_1$, DEPENDING ON THE NUMBER M OF PAIRS IN A T-ZCZ$_1$ FAMILY AND ITS LENGTH $L = M^N$.

Implemen.	Products	Additions	Memory bits
Straight.	$2ML$	$2M \cdot (L-1)$	$M^N + 2M^{N+1} + 2M \cdot (1 + mN)$
ETZC$_1$	$M \cdot log_2 L + \frac{M}{2}$	$2M \cdot log_2 L$	$2(M-1)M^{N-1} + (M^2 - M)\left[\frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}}\right]$

B. *Efficient Correlation algorithm of* T-ZCZ$_{2'}$ *pairs*

According to the generation method of T-ZCZ$_{2'}$, one pair is composed by the sequences $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{2}-1}], [s_{\frac{M}{2},j}|s_{\frac{M}{2}+1,j}|\cdots|s_{M-1,j}])$, where $0 \leq j \leq M-1$. Thus, the AC of sequences $e_{j,0}$ and $e_{j,1}$ can be expressed in terms of correlations with $s_{i,j}$ sequences as shown in (17) and (18).

In general, the correlation of any input signal $r[\tau]$ with the two sequences of a pair E_j is given by (19), where the correlations $C_{r,s_{i,j}}$ can be achieved by means of the ESSC, thus reducing the total number of operations to be carried out.

$$\begin{aligned} &C_{r,e_{j,0}} = \\ &C_{r,s_{0,j}}[\tau] + C_{r,s_{1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{2}-1,j}}[\tau - (\tfrac{M}{2}-1)L_0] \\ &C_{r,e_{j,1}} = \\ &C_{r,s_{\frac{M}{2},j}}[\tau] + C_{r,s_{\frac{M}{2}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{M-1,j}}[\tau - (\tfrac{M}{2}-1)L_0] \end{aligned} \tag{19}$$

The proposed ETZC$_{2'}$ correlator performs the simultaneous correlation of the input signal with the sequences of the M pairs of a T-ZCZ$_{2'}$ family, and can be implemented through the following steps (see Fig. 9): in step 1 the $\frac{M}{2}$ ESSCs placed on top perform the efficient correlations among the input signal $r[\tau]$ and the sequences $\{s_{i,j}; 0 \leq i \leq \frac{M}{2}; 0 \leq j \leq M-1\}$ of $e_{j,0}$. The $\frac{M}{2}$ ESSCs placed on bottom correlate $r[\tau]$ with the sequences $\{s_{i,j}; 0 \leq i \leq \frac{M}{2}; 0 \leq j \leq M-1\}$ of $e_{j,1}$. As a result, at the end of this step, the correlations with the M^2 sequences of the matrix $\Delta^{(M, M \cdot L_0}$ are available. In step 2 the previous correlations are delayed according to (19) obtaining $C_{r,s'_{i,j}} = C_{r,s_{i,j}}[\tau - (\frac{M}{2} - 1 - i)L_0]$ for $0 \leq i \leq \frac{M}{2} - 1$, and $C_{r,s'_{i,j}} = C_{r,s_{i,j}}[\tau - (M - 1 - i)L_0]$ for $\frac{M}{2} \leq i \leq M-1$. Finally, in step 3 the M additions on top allow to obtain the correlation with sequences $e_{j,0}$, that

TABLE V
NUMBER OF OPERATIONS AND MEMORY BITS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED ETZC$_1$ FOR THE CORRELATION OF $M = 4$ AND $M = 8$ PAIRS OF DIFFERENT LENGTHS $L = M^N$.

N	$M=4$					$M=8$				
	L	straightforward		ETZC$_1$		L	straightforward		ETZC$_1$	
		Operations	Memory bits	Operations	Memory bits		Operations	Memory bits	Operations	Memory bits
2	16	248	184	48	60	64	2032	1200	144	336
3	64	1016	632	72	300	512	16 368	8864	216	3080
4	256	4088	2376	96	1284	4096	131 056	69 840	288	25 200
5	1024	16 376	9304	120	5244	32 768	1 048 560	557 312	360	202 328
6	4096	65 528	36 968	144	21 108	262 144	8 388 592	4 456 752	432	1 619 520

$$
\begin{aligned}
&C_{e_{j,0},e_{j,0}}[\tau] = \sum_{l=0}^{L-1-\tau} e_{j,0}[l]e_{j,0}[l+\tau] = \\
&= \sum_{l=0}^{L_0-1-\tau} (s_{0,j}[l]s_{0,j}[l+\tau] + s_{0,j}[l]s_{1,j}[l-L_0+\tau] + \cdots + s_{0,j}[l]s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \\
&+ s_{1,j}[l-L_0]s_{0,j}[l+\tau] + s_{1,j}[l-L_0]s_{1,j}[l-L_0+\tau] + \cdots + s_{1,j}[l-L_0]s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \\
&+ \cdots + s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{0,j}[l+\tau] + s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{1,j}[l-L_0+\tau] + \\
&+ \cdots + s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{\frac{M}{2}-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau]) = \\
&= \sum_{i=0}^{\frac{M}{2}-1} C_{s_{i,j},s_{i,j}}[\tau] + \sum_{i=0}^{\frac{M}{2}-1} \sum_{\substack{h=0 \\ h\neq i}}^{\frac{M}{2}-1} C_{s_{i,j},s_{i,h}}[\tau-(h-i)L_0]
\end{aligned}
\tag{17}
$$

$$
\begin{aligned}
&C_{e_{j,1},e_{j,1}}[\tau] = \sum_{l=0}^{L-1-\tau} e_{j,1}[l]e_{j,1}[l+\tau] = \\
&= \sum_{l=0}^{L_0-1-\tau} (s_{\frac{M}{2},j}[l]s_{\frac{M}{2},j}[l+\tau] + s_{\frac{M}{2},j}[l]s_{\frac{M}{2}+1,j}[l-L_0+\tau] + \cdots + s_{\frac{M}{2},j}[l]s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \\
&+ s_{\frac{M}{2}+1,j}[l-L_0]s_{\frac{M}{2},j}[l+\tau] + s_{\frac{M}{2}+1,j}[l-L_0]s_{\frac{M}{2}+1,j}[l-L_0+\tau] + \cdots + \\
&+ s_{\frac{M}{2}+1,j}[l-L_0]s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau] + \cdots + s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{\frac{M}{2},j}[l+\tau] + \\
&+ s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{\frac{M}{2}+1,j}[l-L_0+\tau] + \cdots + s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0]s_{M-1,j}[l-(\tfrac{M}{2}-1)L_0+\tau]) = \\
&= \sum_{i=\frac{M}{2}}^{M-1} C_{s_{i,j},s_{i,j}}[\tau] + \sum_{i=\frac{M}{2}}^{M-1} \sum_{\substack{h=\frac{M}{2} \\ h\neq i}}^{M-1} C_{s_{i,j},s_{i,h}}[\tau-(h-i)L_0]
\end{aligned}
\tag{18}
$$

is, $\{C_{r,e_{j,0}} = \sum_{i=0}^{\frac{M}{2}-1} C_{r,s'_{i,j}}; 0 \leq j \leq M-1\}$; and the M additions on bottom the correlations with sequences $e_{j,1}$, i. e., $\{C_{r,e_{j,1}} = \sum_{i=\frac{M}{2}}^{M-1} s'_{i,j}; 0 \leq j \leq M-1\}$.

With this proposal, a high reduction in the number of operations is achieved, when compared to the straightforward implementation, at the expense of increasing the memory requirements. If the length of the sequences is $L = \frac{M^{N+1}}{2}$, Table VI shows the computational requirements for both implementations. Regarding the ETZC$_{2'}$, the product operations are performed in step 1 and consist of changes in the sign (multiplications by -1 or $+1$). Furthermore, in step 1 M^2mN additions are carried out, whereas the rest of the additions are computed in step 3. On the other hand, the first term in the expression that sum up the memory requirements of ETZC$_{2'}$ corresponds to the memory necessary in step 1, and the second term to the memory required in step 2. An example of computational requirements of both implementations for different lengths and numbers of pairs is shown in Table VII; where the decrease in the number of operations to be performed when the ETZC$_{2'}$ is used can be observed.

C. Efficient Correlation algorithm of T-ZCZ$_{3'}$ pairs

The architecture of the efficient correlator of T-ZCZ$_{3'}$ pairs (ETZC$_{3'}$) is similar to the architecture of the ETZC$_{2'}$. If the generation submatrix $\mathbf{\Delta}_{\mathbf{L}}^{(\mathbf{M},\frac{\mathbf{M}}{\mathbf{2}}\cdot\mathbf{L_0}}$ is used, each pair is $E_j = (e_{j,0}, e_{j,1}) = ([s_{0,j}|s_{1,j}|\cdots|s_{\frac{M}{4}-1,j}], [s_{\frac{M}{4},j}|s_{\frac{M}{4}+1,j}|\cdots|s_{\frac{M}{2}-1,j}])$, with $M > 2$. Then, the correlation of the input signal $r[\tau]$ with every sequence of a pair T-ZCZ$_{3'}$ can be separated into the correlation of $r[\tau]$ with the corresponding sequences $s_{i,j}$, as shown in (20). The proposed correlator filter is represented in Fig. 10. When the submatrix $\Delta_R^{(M,M\cdot L_0}$ is used to generate the T-ZCZ$_{3'}$ pairs, the correlation between $r[\tau]$ and $(e_{j,0}, e_{j,1})$ is the indicated in (21), and it can be performed with the architecture depicted in Fig. 10 by substituting the $w_{i,n}$ coefficients for those corresponding to sets $\{S_i; \frac{M}{2} \leq i \leq M-1\}$.

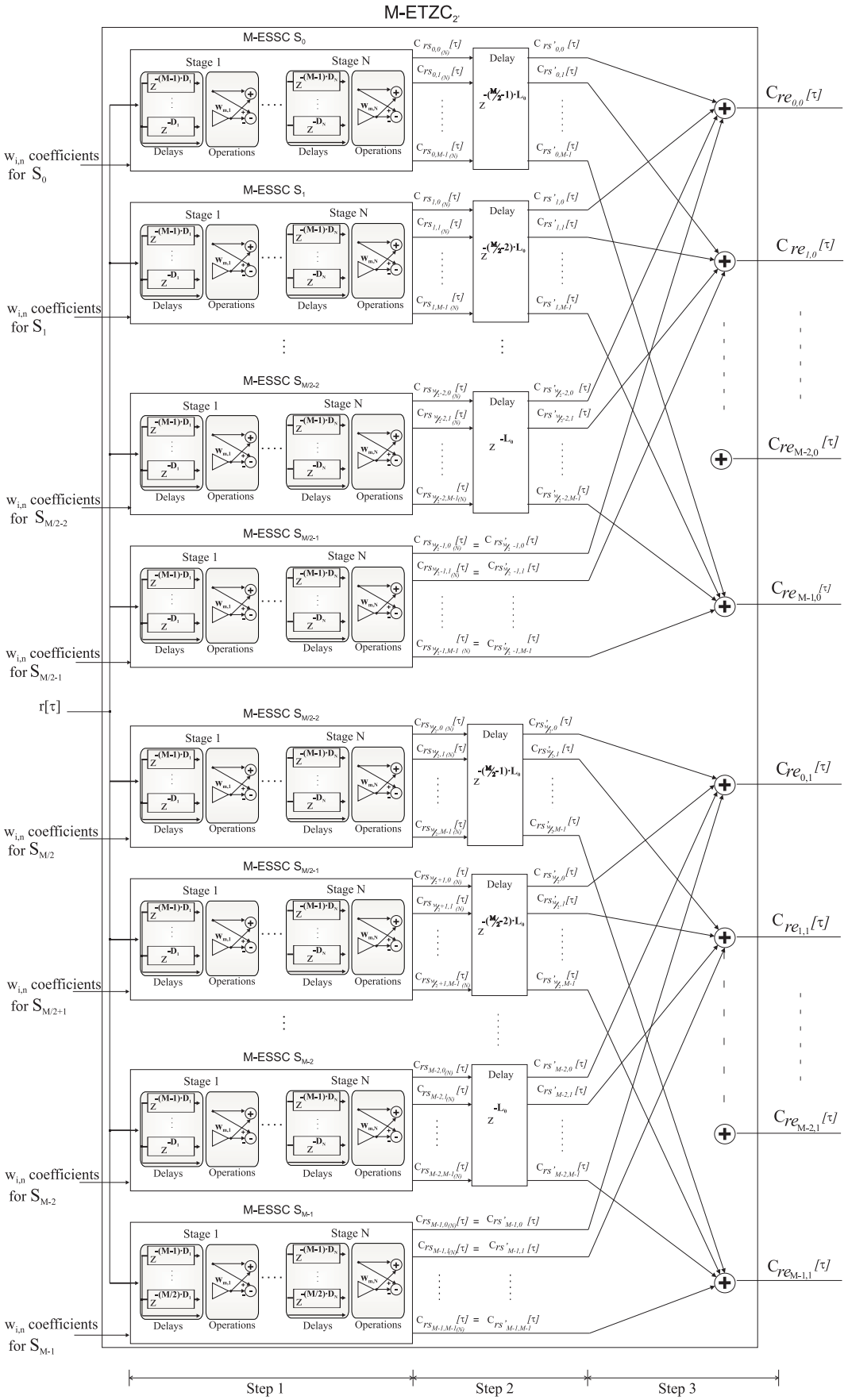


Fig. 9. Block diagram for the Efficient T-ZCZ$_{2'}$ Correlator (ETZC$_{2'}$).

TABLE VI
NUMBER OF OPERATIONS AND MEMORY BITS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED ETZC$_{2'}$, DEPENDING ON THE NUMBER M OF PAIRS IN A T-ZCZ$_{2'}$ FAMILY AND ITS LENGTH $L = \frac{M^{N+1}}{2}$.

Implementation	Products	Additions	Memory bits
Straightforward	M^{N+2}	$2M(\frac{M^{N+1}}{2} - 1)$	$\frac{M^{N+1}}{2} + M^{N+2} + 2M \cdot (1 + mN + m - 1)$
ETZC$_{2'}$	$\frac{M^2}{2}mN$	$M^2mN + 2(\frac{M^2}{2} - M)$	$(M-1)M^N + \frac{(M^3-M^2)}{2}\left[\frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}}\right] + (1 + mN) \cdot \frac{M}{2}(\frac{M}{2} - 1) \cdot M^N$

TABLE VII
NUMBER OF OPERATIONS AND MEMORY BITS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED ETZC$_{2'}$ FOR THE CORRELATION OF $M = 4$ AND $M = 8$ PAIRS OF DIFFERENT LENGTHS $L = \frac{M^{N+1}}{2}$.

N	$M = 4$					$M = 8$				
	L	straightforward		ETZC$_{2'}$		L	straightforward		ETZC$_{2'}$	
		Operations	Memory bits	Operations	Memory bits		Operations	Memory bits	Operations	Memory bits
1	8	120	104	56	36	32	1008	640	336	440
2	32	504	336	104	280	256	8176	4496	624	6720
3	128	2040	1216	152	1496	2048	65 520	35 008	912	73 760
4	512	8184	4688	200	7176	16 384	524 272	278 768	1200	739 776
5	2048	32 760	18 528	248	33 016	131 072	4 194 288	2 228 512	1488	7 100 768

$$
\begin{aligned}
&C_{r,e_{j,0}} = \\
&C_{r,s_{0,j}}[\tau] + C_{r,s_{1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{4}-1,j}}[\tau - (\tfrac{M}{4} - 1)L_0] \\
&C_{r,e_{j,1}} = \\
&C_{r,s_{\frac{M}{4},j}}[\tau] + C_{r,s_{\frac{M}{4}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{M}{2}-1,j}}[\tau - (\tfrac{M}{4} - 1)L_0]
\end{aligned} \tag{20}
$$

$$
\begin{aligned}
&C_{r,e_{j,0}} = \\
&C_{r,s_{\frac{M}{2},j}}[\tau] + C_{r,s_{\frac{M}{2}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{\frac{3M}{4}-1,j}}[\tau - (\tfrac{M}{4} - 1)L_0] \\
&C_{r,e_{j,1}} = \\
&C_{r,s_{\frac{3M}{4},j}}[\tau] + C_{r,s_{\frac{3M}{4}+1,j}}[\tau - L_0] + \cdots + C_{r,s_{M-1,j}}[\tau - (\tfrac{M}{4} - 1)L_0]
\end{aligned} \tag{21}
$$

The total number of operations and memory requirements of the ETZC$_{3'}$ and a straightforward implementation can be observed in Table VIII. Table IX shows the computational requirements for the correlation of family pairs with $M = 4$ and $M = 8$ sequences of different lengths $L = \frac{M^{N+1}}{4}$. A considerable reduction in the number of operations is obtained for the ETZC$_{3'}$.

VI. DISCUSSION

T-ZCZ codes are of great interest for quasisynchronous applications and they can replace CSS when the number of simultaneous users is greater than two (ZCZ are obtained by using only two sequences). The proposed T-ZCZ pairs are derived from a CSS obtained as described in [25], and they offer some advantages in comparison with T-ZCZ pairs proposed in [18]: larger ZCZ for specific code lengths; lower values of interferences in the IZ; and the possibility of using efficient algorithms for its generation and correlation that decreases the computational load.

The election of T-ZCZ generated by method 1 using $\mathbf{\Delta}^{(\mathbf{M},\mathbf{L_0}}$ (T-ZCZ$_1$), method 2 using $\mathbf{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0}}$ (T-ZCZ$_{2'}$) or method 3 using $\mathbf{\Delta}^{(\mathbf{M},\mathbf{M}\cdot\mathbf{L_0}}$ (T-ZCZ$_{3'}$), mainly depends on the desired code length and the size of the ZCZ. T-ZCZ$_1$ codes have reduced ZCZ when N is even, whereas T-ZCZ$_{3'}$ are the ones with larger ZCZ, as can be observed in Fig. 11. On the other hand, considering the hardware implementation of the corresponding correlators, the one that requires less operations and less memory resources is the ETZC$_1$ (see Fig. 12). ETZC$_{2'}$ and ETZC$_{3'}$ reduce the number of operations performed in comparison with a straightforward correlator, however they require more memory. Since the decrease in the number of operations involved in the correlation is more important than the increase in memory requirements in terms of hardware implementation, the ETZC$_{2'}$ and the ETZC$_{3'}$ are a better option than the conventional straightforward implementations.

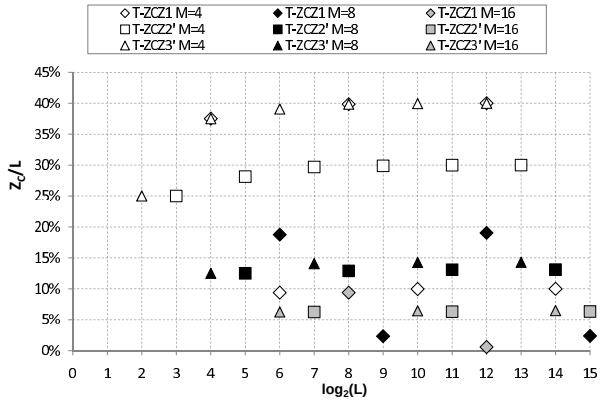


Fig. 11. Size of the central ZCZ expressed in percentage regarding to the total length of the code.

VII. CONCLUSION

Three different mechanism to construct pairs with ZCZ in the central and terminal parts of the sum of their aperiodic correlation functions are proposed, based on complementary

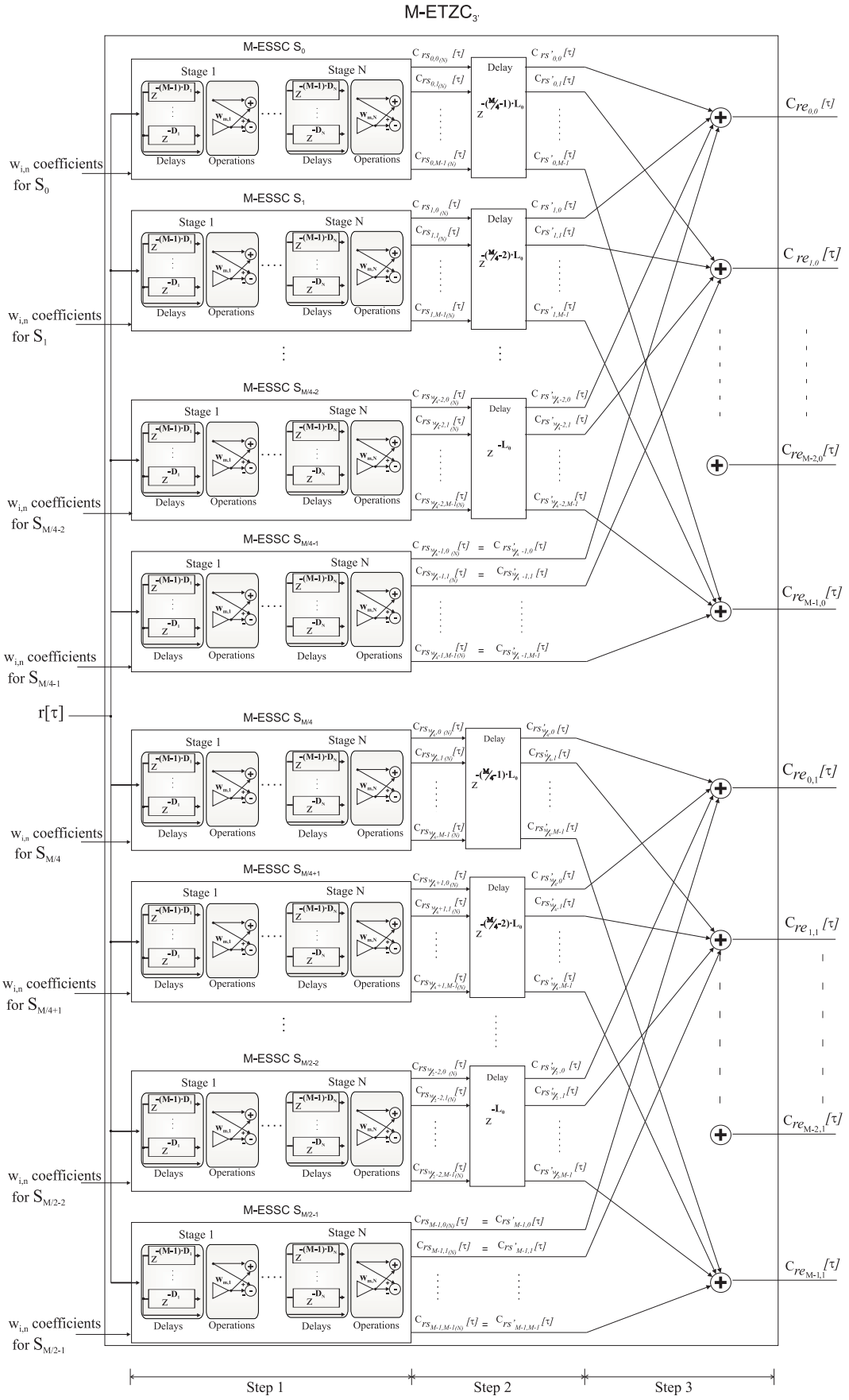


Fig. 10. Block diagram for the Efficient T-ZCZ$_{3'}$ Correlator (ETZC$_{3'}$).

TABLE VIII
NUMBER OF OPERATIONS AND MEMORY BITS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED $\text{ETZC}_{3'}$, DEPENDING ON THE NUMBER M OF PAIRS IN A T-$\text{ZCZ}_{3'}$ FAMILY AND ITS LENGTH $L = \frac{M^{N+1}}{4}$.

Implementation	Products	Additions	Memory bits
Straightforward	$\frac{M^{N+2}}{2}$	$2M(\frac{M^{N+1}}{4} - 1)$	$\frac{M^{N+1}}{4} + \frac{M^{N+2}}{2} + 2M \cdot (1 + mN + m - 2)$
M-ETZC_3	$\frac{M^2}{4}mN$	$\frac{M^2}{2}mN + 2(\frac{M^2}{4} - M)$	$(M-1)\frac{M^N}{2} + \frac{(M^3-M^2)}{4}\left[\frac{M^{N-1}-1}{M-1} + mM^N \sum_{i=1}^{N-1} \frac{i}{M^{i+1}}\right] + (1 + mN) \cdot (\frac{M^2}{16} - \frac{M}{4})M^N$

TABLE IX
NUMBER OF OPERATIONS AND MEMORY BITS IN THE STRAIGHTFORWARD CORRELATOR AND THE PROPOSED $\text{ETZC}_{3'}$ FOR THE CORRELATION OF $M = 4$ AND $M = 8$ PAIRS OF DIFFERENT LENGTHS $L = \frac{M^{N+1}}{4}$.

N	$M = 4$					$M = 8$				
	L	straightforward		$\text{ETZC}_{3'}$		L	straightforward		$\text{ETZC}_{3'}$	
		Operations	Memory bits	Operations	Memory bits		Operations	Memory bits	Operations	Memory bits
1	4	56	144	24	48	16	496	576	160	400
2	16	248	352	48	312	128	4 080	3 312	304	4816
3	64	1016	1136	72	1392	1024	32 752	24 864	448	43 168
4	256	4088	4224	96	5736	8192	262 128	196 944	592	371 824
5	1024	16 376	16 528	120	23 136	65 536	2 097 136	1 573 248	736	3 173 440

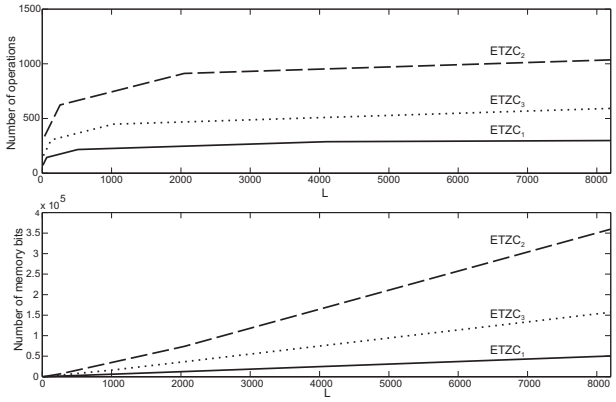


Fig. 12. Comparison among the number of operations performed and memory resources required by correlators ETZC_1, $\text{ETZC}_{2'}$ and $\text{ETZC}_{3'}$, when $M = 8$ pairs of different lengths are considered.

set of sequences. The developed algorithms allow an efficient implementation of the generator and correlator for these pairs, in comparison with straightforward implementations, significantly reducing the total number of calculations. Thus, real-time performance as well as the use of longer sequences are possible. Furthermore, the regular architecture of the proposed generators and correlators allows its implementation in hardware devices as Field Programmable Gate Arrays, reducing the overall design time.

The proposed pulse compressor, together with the ZCZ zones and number of available pairs, makes very attractive the inclusion of T-ZCZ pairs in quasisynchronous CDMA systems.

REFERENCES

[1] S. W. Golomb, *Shift Register Sequences*, Holden-Day, Inc, San Francisco, 1967.

[2] R. Gold, *Optimal Binary Sequences for Spread Spectrum Multiplexing*, IEEE Transactions on Information Theory, IT-13, pp. 619–621, October, 1967.

[3] T. Kasami, *Weight Distribution Formula for some Class of Cyclic Codes*, Coordinated Science Lab. University of Illinois, Report R–285,pp. 1–29 April, 1966.

[4] J. J. Sylvester, *Thoughts on inverse orthogonal matrices, simultaneous sign successions, and tessellated pavements in two or more colours, with applications to Newton's rule, ornamental tile-work, and the theory of numbers.*, Philosophical Magazine, vol. 34, pp. 461–475, 1867.

[5] H. Donelan and T. O'Farrell, *A new method for generating sets of orthogonal sequences for a synchronous CDMA system*, IEEE Electronics Letters, vol. 35, no. 8, pp. 1537–1538, September, 1999.

[6] L. R. Welch, *Lower bounds on the maximum cross-correlation of signals*, IEEE Transactions on Information Theory, vol. 20, no. 3, pp. 397–399, May, 1974.

[7] H. Chen, *Next generation CDMA technologies*, John Wiley & sons, Ltd, West Sussex PO19 8SQ, England, 2007.

[8] F. J. Álvarez and J. Ureña and M. Mazo and A. Hernández and J. J. García and C. De Marziani *High reliability outdoor sonar prototype based on efficient signal coding*, IEEE Transactions on ultrasounds, ferroelectrics and frequency control, vol. 53, no. 10, pp. 1862–1872, october 2006.

[9] S. Wang and A. Abdi, *MIMO ISI channel estimation using uncorrelated Golay complementary sets of polyphase sequences*, IEEE Transactions on vehicular technology, vol. 56, no. 5, pp. 3024–3039, September, 2007.

[10] M. J. Golay, *Complementary Series*, IRE Transactions on Information Theory, vol. IT-7, pp. 82–87, April 1961.

[11] C. C. Tseng C. L. and Liu, *Complementary Sets of Sequences*, IEEE Transactions on Information Theory, vol. IT-18, no. 5, pp. 644–652, September, 1972.

[12] C. De Marziani and J. Ureña and A. Hernández and M. Mazo and J. J. García and J. M. Villadangos and M. C. Péerez and A. Ochoa and F. J. Álvarez, *Inter-Symbol interference reduction on macro-sequences generated from complementary sets of sequences*, Proc. of IEEE 32nd Annual Conference on Industrial Electronics IECON'06, pp. 3367-3372, Paris (France), November, 2006.

[13] F. Pingzhi, *Spreading sequence design and theoretical limits for quasisynchronous CDMA systems*, EURASIP Journal on Wireless Communications and Networking, vol. 2004, no. 1, pp. 19–31, March, 2004.

[14] S. Stańzak and H. Boche and M. Haardt, *Are LAS-codes a miracle?*, Proc. of IEEE Global Telecommunications Conference (GLOBECOM'2001), San Antonio (USA), vol. 1, pp. 589–593, November, 2001.

[15] D. Li, *The perspectives of Large Area Synchronous CDMA technology for the fourth-generation mobile radio*, IEEE Communications Magazine, vol. 41, no. 3, pp. 114–119, March, 2003.

[16] Q. K. Trinh and P. Fan D. Peng and M. Darnell, *Construction of optimal mismatched periodic sequence sets with zero correlation zone*, IEEE Signal processing letters, vol. 15, pp. 341–344, January, 2008.

[17] H. Wei and L. Hanzo, *On the uplink performance of LAS-CDMA*, IEEE Transactions on Wireless Communications, vol. 5, no. 5, pp. 1187–1196, May, 2006.

[18] C. Zhang and X. Lin and M. Hatori, *Novel sequence pair and set with Three Zero Correlation Windows*, IEICE Transactions on Communications, vol. E88-B, No. 4, pp. 1517–1522, April, 2005.

[19] G. J. R. Povey and P. M. Grant, *Simplified matched filter receiver designs for spread spectrum communications applications*, Electronics & Communications Engineering Journal, vol. 5, no. 2, pp. 59–64, April, 1993.

[20] S. Z. Budisin, *Fast PN sequence correlation by FWT*, Proc. of IEEE Integration Research, Industry and Education in Energy and Communication Engineering, MELECON'89,pp. 513–515, April, 1989.

[21] S. Z. Budisin, *Efficient Pulse Compressor for Golay Complementary Sequences*, IEEE Electronics Letters, vol. 27, no. 3, pp. 219–220, January, 1991.

[22] Y. Tsai and G. Zhang and X. Wang, *Polyphase codes for uplink OFDM-CDMA Systems*, IEEE Transactions on Communications, vol. 56, no. 3, pp. 435–444, March, 2008.

[23] B. M. Popovic, *Efficient Despreaders for Multi-code CDMA Systems*, Proc. of IEEE Universal Personal Communication Record, San Diego (USA), vol. 2, pp. 516–520, October, 1997.

[24] B. M. Popovic, *Efficient Golay Correlator*, IEEE Electronics Letters, vol. 35, no. 17, pp. 1427–1428, August, 1999.

[25] C. De Marziani and J. Ureña and A. Hernández and M. Mazo and F. J. Álvarez and J. J. García and P. Donato, *Modular architecture for efficient generation and correlation of complementary set of sequences*, IEEE Transactions on Signal Processing, vol. 55, no. 5, pp. 2323–2337, May, 2007.

[26] M. C. Pérez and J. Ureña and A. Hernández and C. De Marziani and A. Jiménez, *Hardware Implementation of an Efficient Correlator for Interleaved Complementary Sets of Sequences*, Journal of Universal Computer Science, vol. 13, no. 3, pp. 388–406, March, 2007.

[27] M. C. Pérez and J. Ureña and A. Hernández and F. J. Álvarez and A. Jiménez and C. Marziani, *Efficient correlator for LS codes*, IEEE Communications Letters, vol. 12, no. 10, pp. 764–766, October, 2008.