(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 July 2008 (17.07.2008)

PCT

(10) International Publication Number
WO 2008/084137 A1

(51) International Patent Classification:
H04B 1/707 (2006.01) H04B 7/216 (2006.01)
H04B 1/69 (2006.01) G06F 7/58 (2006.01)
GO5 13/00 (2006.01)

(21) International Application Number:
PCT/FI2008/000002

(22) International Filing Date: 11 January 2008 (11.01.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
20070031 12 January 2007 (12.01.2007) FI

(71) Applicant (for all designated States except US):
EIGNER OY [FTTH]; Kuusisaarenpolku 5 B, FI-00340 Helsinki (FI).

(72) Inventor; and
(75) Inventor/Applicant (for US only): LEHTINEN, Markku
[FI/FI]; Taitelantie 48 as 21, FI-99600 Sodankyla (FI).

(74) Agent: BERGGREN OY AB; P.O. Box 16 (Antinkatu 3 C), FI-00101 Helsinki (FI).

(54) Title: COMMUNICATION AND REMOTE SENSING APPLICATIONS WITH SIGNALS CODED WITH PERFECT CODES

(57) Abstract: A code for signal transmission is a so-called perfect code with zero sidelobes in its autocorrelation function if and only if the modulus of the Fourier transform of said code is equal to 1. A communication device equipped for communications with perfect codes is one, a code source of which outputs a perfect code. A remote sensing device equipped for remote sensing with perfect codes is one, a code source of which outputs a perfect code.
Communication and remote sensing applications with signals coded with perfect codes

TECHNICAL FIELD

The invention concerns generally the use of a coded electromagnetic signal for the purpose of conveying information between a transmitter and a receiver. In general the invention concerns the use of codes that are perfect in the sense that their autocorrelation functions are free of sidelobes.

BACKGROUND OF THE INVENTION

There are many reasons for applying a code to an electromagnetic signal to be transmitted. Fig. 1 illustrates a communications application, in which a transmitter 101 generates a time-discrete signal \( S = s(t) \) to be transmitted to the receiver 111. The transmission will take place through a wireless channel, where effects like interference, fading and multipath propagation will cause distortions. Additionally it may be desirable to keep unauthorised third parties from receiving the signal. In order to compensate for distorting effects and/or to make unauthorised reception difficult, an encoding unit 102 of the transmitter 101 employs a code \( C \) to produce an encoded signal \( C(S) \). A decoding unit 112 of the receiver employs an inverse code \( C^{-1} \) to cancel the effect of coding, so that as a result the original signal \( S \) is produced. In many cases of digital coding the inverse is the same as the original code, i.e. \( C^{-1} = C \), or a time inverse of the original code, i.e. \( C^{-1}(f) = C(-f) \).

Fig. 2 illustrates a remote sensing application, in which the purpose is to detect some features of a target 221 by using a transmitter 201 to transmit electromagnetic radiation towards the target 221 and a receiver 211 to receive a reflection from the target 221. For example in many radar applications the target 221 is actually a distribution of reflectivity values at different distances. Range resolution could be enhanced by only transmitting a very short radar pulse and by observing the arrival of echoes from different distances. This would lead to uneconomical use of the radar duty cycle and decrease the mean received power. A better solution is to use a coding filter 202 in the transmitter and a corresponding decoding filter 212 in the receiver to apply a phase-modulated pulse code to the transmitted radar signal and to derive the distribution of reflectivity values as a function of dis-
tance by convolving the received signal with the impulse response of a suitable reception filter.

From the viewpoint of signal generation there are significant differences between communication and remote sensing applications. In communications, the contents of the original signal to be transmitted cannot be predicted beforehand: the original payload signal resembles more a random bit stream. The coded transmission signal is produced as a convolution of said bit stream with the code. Although in remote sensing applications a similar approach is possible, in which the original signal to be transmitted is thought to consist of an elementary pulse (or a sequence of elementary pulses) and the coded transmission signal is produced by convolving the elementary pulse(s) with the code, it is more common not to think about elementary pulses and coding filters separately but to take the coded transmission signal as such, as a given, unitary entity.

An important similarity in communication and remote sensing applications is the reliance on the orthogonal properties of the code. In communications, random noise and transmissions coded with different codes will not correlate with the coded form of the transmission signal, so the inverse coding operation at the receiver will cause them to cancel out. In range-specific remote sensing, when a particular distance is considered, reflections of the same coded transmission signal from different distances will similarly cancel out due to code orthogonality, if the time difference between them is sufficiently long in the received composite signal.

The goodness of a code for purposes like those illustrated above is measured by its autocorrelation properties. As an example, let us consider the so-called Barker codes originally introduced in Barker, R. H.: "Group Synchronizing of Binary Digital Systems", in: Communications Theory, edited by Jackson.W. Academic Press, New York, 273-287, 1953. Known Barker codes are finite bit sequences that have been shown to be the most efficient binary codes using single code sequences. The autocorrelation function of each Barker code has a central peak and a number of sidelobes on each side of it. The peak-to-sidelobe ratio equals the length of the Barker code in bits.

The use of a matched filter is a known technique for receiving coded transmissions. A matched filter is one that maximises the output signal-to-noise ratio in the presence of additive stochastic noise. The matched filter that corresponds to a known binary code has an impulse response that is a time-reversed and conju-
gated version of the transmitted coded signal. In digital signal processing the use of a matched filter has the advantages that the filter has a finite length and filtering is readily implemented with a number of logical operations; on the other hand the effect of sidelobes in the autocorrelation function just has to be accepted.

A technique known as mismatched filtering has been described in publications like Luke, H. D.: "Mismatched filtering of periodic quadriphase and 8-phase sequences", IEEE Transactions on Communications, Vol. 51, NoJ, July 2003, and Damtie, B., Lehtinen, M., Orispaa, M., Vierinen, J.: "Mismatched filtering of aperiodic quadriphase codes", IEEE Transactions on Aerospace and Electronic Systems, Vol. XX, No. Y, submitted in August 2006. The convolution of the impulse response of the mismatched filter with the transmission code produces a weight function with no sidelobes. A mismatched filter can be calculated for practically any code, with the merely theoretical exception that the Fourier transform of the code must not have any zeroes. Another merely theoretical drawback of mismatched filters is that the impulse response of a properly calculated mismatched filter has an infinite number of nonzero elements. This is not serious in practice, because only a finite number of elements in an "active range" of the impulse response will be large enough to have significance; the others are negligible and can be approximated with neutral values (zeroes). A yet another drawback of mismatched filters is that they do not achieve the optimal signal-to-noise ratio (SNR) of matched filters. The effect of this drawback can be minimised by examining a large number of possible codes and their corresponding mismatched filters, and selecting those associated with the smallest penalty in SNR (see the last-mentioned prior art publication).

Despite of the many advantages of mismatched filtering, there remains the ultimate aim of presenting perfect codes, the autocorrelation functions of which would exhibit zero sidelobes.

SUMMARY OF THE INVENTION

An objective of the present invention is to present codes for digital signal processing that would have arbitrarily small sidelobes in their autocorrelation functions. Another objective of the invention is to present devices for communication and remote sensing that are capable of utilising such codes.
The objectives of the invention are achieved by employing codes that can be derived from near-perfect codes by applying a certain calculational criterion.

An electronic device for outputting a code according to the invention is characterised in that it is configured to output a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

A communication device adapted to implement coded communications with at least one other communication device according to the invention is characterized in that it comprises:
- at least one of a transmitter and a receiver, and
- a code source configured to provide said at least one of a transmitter and a receiver with a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

A remote sensing device adapted to transmit a coded transmission signal towards a target according to the invention comprises:
- a transmitter, and
- a transmission signal source configured to provide said transmitter with a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

A method for transmitting an encoded signal, a method for receiving a decoding an encoded signal, and a method for remotely sensing a target according to the invention are all characterised by using as the code a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.
A method for producing a code according to the invention is characterized in that the method comprises producing a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

A computer program product according to the invention is characterized in that it comprises machine-readable instructions which, when executed on a computer, cause the computer to produce a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

Mismatched filters, with their capacity of completely eliminating the sidelobes of the weight function, were found by loosening some theoretical constraints; for example the theoretical drawback of the infinite length of the impulse response is accepted, because it can be shown to have no practical significance. In the present invention it was found that by allowing some similar degrees of freedom, it is indeed possible to derive codes that have no sidelobes in their autocorrelation function (or the sidelobes of which may be made arbitrarily small). The perfect codes too have the theoretical drawback of infinite length, but again it can be shown that only a relatively small number of elements in the code will differ enough from zero to give them practical significance.

The exemplary embodiments of the invention presented in this patent application are not to be interpreted to pose limitations to the applicability of the appended claims. The verb "to comprise" is used in this patent application as an open limitation that does not exclude the existence of also unrecited features. The features recited in depending claims are mutually freely combinable unless otherwise explicitly stated.

The novel features which are considered as characteristic of the invention are set forth in particular in the appended claims. The invention itself, however, both as to its construction and its method of operation, together with additional objects and advantages thereof, will be best understood from the following description of specific embodiments when read in connection with the accompanying drawings.
BRIEF DESCRIPTION OF DRAWINGS

Fig. 1 illustrates a known application of coding in communications,
fig. 2 illustrates a known application of coding in remote sensing,
fig. 3a illustrates element values of a 13-bit Barker code,
fig. 3b illustrates element values of a so-called perfect code,
fig. 3c illustrates an autocorrelation function of the code of fig. 3a,
fig. 3d illustrates an autocorrelation function of the code of fig. 3b,
fig. 4 illustrates a programmable code source,
fig. 5 illustrates a communication device utilising perfect codes,
fig. 6 illustrates a remote sensing device utilising perfect codes,
fig. 7 illustrates producing a perfect code according to an embodiment of the invention, and
fig. 8 illustrates producing a perfect code according to another embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION AND EMBODIMENTS

Let us examine a sequence $\epsilon(n)$, $n = 1, 2, 3, \ldots, N$. Each element $\epsilon(n)$ has a phase and an amplitude. If the sequence represents an ordinary binary phase code, the amplitude of each element is constant (equal to 1) and the possible phase values are $0$ and $\pi$ radians. In the case of an ordinary quadriphase code the amplitudes are again constant but the allowable phase values are $0, \pi/2, \pi,$ and $3\pi/2$ radians. These are only examples: at this stage we do not need to set any conditions to the phase and amplitude values of the elements in the sequence. Also, even if above we have assumed that the sequence is finite and has only $N$ elements, we may also think about the sequence as having infinitely many elements $(n \in [-\infty, \infty])$, of which we know that at least those outside the range $n = 1, 2, 3, \ldots, N$ are all zeroes.

The discrete Fourier transform $\tilde{\epsilon}(\omega)$ of the sequence $\epsilon(n)$ is

$$\tilde{\epsilon}(\omega) = \sum_{n=-\infty}^{\infty} \epsilon(n)e^{-i\omega n}, \quad \omega \in [0, 2\pi) \tag{1}$$
and the original sequence \( \varepsilon(n) \) can be written as the inverse transform of its Fourier transform \( \tilde{\varepsilon}(\omega) \) as

\[
\varepsilon(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{\varepsilon}(\omega) e^{i\omega n} d\omega .
\] (2)

If the sequence \( \varepsilon(n) \) represents the transmission code, it is conventional to define that its matched filter is the one, the impulse response \( h_m(n) \) of which is equal to a time-inverted and conjugated copy of the code itself, i.e.

\[
h_m(n) = \overline{\varepsilon(-n)} .
\] (3)

Assuming that a coded transmission sequence \( \varepsilon(n) \) was transmitted and the matched filter \( h_m(n) \) was used in reception, the time series observed at the receiver is the same as would be the response to a single uncoded pulse convolved with the range ambiguity function \( w_m(n) \):

\[
w_m(n) = \varepsilon(n)^* h_m(n) = \varepsilon(n)^* \overline{\varepsilon(-n)} .
\] (4)

Here the asterisk \( ^* \) means convolution. Using a matched filter leads to \( w_m(n) \) having a central peak of height \( N \) and a set of sidelobes on both sides. If the sequence \( \varepsilon(n) \) was a Barker code, the height of the sidelobes is equal to 1.

The known mismatched filter for the sequence \( \varepsilon(n) \) is the one, the Fourier transform of which is equal to \( N/\tilde{\varepsilon}(\omega) \). Using the inverse Fourier transform representation we may write the impulse response \( \lambda(n) \) of the mismatched filter as

\[
\lambda(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} N \overline{\tilde{\varepsilon}(\omega)} e^{in\omega} d\omega .
\] (5)

The theoretical drawbacks and their practical remedies concerning mismatched filters were discussed briefly earlier in the description of prior art.

The features of matched and mismatched filters that were recapitulated above are all known as such. In order to understand certain features of the present invention, let us first write the autocorrelation function \( a_{\varepsilon}(n) \) of a matched filter and the autocorrelation function \( a_{\lambda}(n) \) of a mismatched filter as follows:
The convolution of these two autocorrelation functions is a single peak at the origin, as can be checked by the following calculation:

\[
\sum_{k} a_{\epsilon}(k)a_{\lambda}(i-k) = \sum_{k} \sum_{m} \sum_{m'} \epsilon(m+k)\lambda(m')\delta(m'+i-k) \\
= \sum_{m} \sum_{m'} \epsilon(jm)\delta(m'-i) \\
= \sum_{m} \epsilon jm\lambda(m - i) \\
= N_{b} S(i).
\]

Here \( N_{b} \) is the height of the central peak, and equals the number of elements in the original sequence \( \epsilon(n) \) for sequences with pulses of unit amplitude. If the amplitudes of the pulses of the original sequence do not equal 1, \( N_{b} \) denotes the total energy of the sequence: \( N_{b} = \sum_{m} |\epsilon(m)|^2 \). The shown relation between said two autocorrelation functions means that - at least if the sidelobes of the autocorrelation functions are small compared to the main peak - the sidelobes associated with the mismatched filter are approximately equal to the negated sidelobes of the matched filter:

\[
a_{\lambda}(n) \approx - a_{\epsilon}(n) \quad \text{when } n \neq 0.
\]

Let us now designate as \( \epsilon_{1}(n) \) the average of the original sequence \( \epsilon(n) \) and its inverted mismatched filter \( \lambda(n) \):

\[
\epsilon_{1}(n) = \frac{1}{2}(\epsilon(n) + \lambda(-n)).
\]

The autocorrelation function \( a_{\epsilon_{1}}(n) \) of the first-order averaged sequence \( \epsilon_{1}(n) \) is given as
\[ a^n = -i(2^{* \lambda_{ln} + a_s(n)} + a_\ell(n)) \]  

(11)

Utilising formula (9) we see that the two last terms in the parentheses essentially cancel. This means that the sidelobes of \( a^{\ell}_s(n) \) are clearly lower than those of \( a_\ell(n) \). Thus, \( \varepsilon_1(n) \) is a better code than \( \varepsilon(n) \) in the autocorrelation sense. On the other hand, \( \varepsilon_1(n) \) is just another code sequence, for which a mismatched filter with an impulse response \( \lambda_1(n) \) can be calculated using the known techniques. Then yet another code \( \varepsilon_2(n) \) can be defined, analogously with formula (10), as

\[ \varepsilon_2(n) = \frac{1}{2}(\varepsilon_1(n) + \lambda_1(-n)) \]  

(12)

and the process can be continued an arbitrary number of times, expressed iteratively as

\[ \varepsilon_{g+1}(n) = \frac{1}{2}(\varepsilon_g(n) + \lambda_g(-n)) \]  

(13)

with the sidelobes of the autocorrelation function getting lower at every iteration step.

Fig. 3a illustrates a sequence \( \varepsilon(n), n \in [-64, 64] \), where the elements \( \varepsilon(1) \ldots \varepsilon(13) \) are the known elements of the 13-bit Barker code \( (1, 1, 1, 1, 1, -1, -1, 1, 1, -1, 1, -1, 1) \) and the rest of the elements are zeroes. Fig. 3b illustrates a sequence \( \varepsilon_q(n), n \in [-64, 64] \), which has been produced by starting from sequence \( \varepsilon(n) \) and executing so many iteration rounds of the kind illustrated above in formula (13) that the difference between \( \varepsilon_{q+1}(n) \) and \( \varepsilon_q(n) \) becomes smaller than the numerical accuracy of the calculating device used. It is easy to see that many of the element values in sequence \( \varepsilon_q(n) \) are not exactly 1, -1 or 0, but have real-numbered values that differ by 0.10-0.15 at most from said integer values. With integer values less than about -30 or greater than +40 the elements of the sequence \( \varepsilon_q(n) \) appear in the graphical representation to be essentially zeroes.

If we print out the elements of the sequence \( \varepsilon_q(n) \) with a numerical accuracy of five decimals, we note that with this accuracy there are non-zero elements only in the range of index values from -57 and +71; see the following table.
Table 1: selected elements of $e_q(n)$ derived from 13-bit Barker code

<table>
<thead>
<tr>
<th>n</th>
<th>$e_q(n)$</th>
<th>n</th>
<th>$e_q(n)$</th>
<th>n</th>
<th>$e_q(n)$</th>
<th>n</th>
<th>$e_q(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-60</td>
<td>0.00000</td>
<td>-26</td>
<td>-0.00192</td>
<td>8</td>
<td>1.03877</td>
<td>42</td>
<td>0.00105</td>
</tr>
<tr>
<td>-59</td>
<td>0.00000</td>
<td>-25</td>
<td>-0.00276</td>
<td>9</td>
<td>0.90814</td>
<td>43</td>
<td>-0.00068</td>
</tr>
<tr>
<td>-58</td>
<td>0.00000</td>
<td>-24</td>
<td>-0.00270</td>
<td>10</td>
<td>-1.05241</td>
<td>44</td>
<td>0.00030</td>
</tr>
<tr>
<td>-57</td>
<td>0.00001</td>
<td>-23</td>
<td>-0.00184</td>
<td>11</td>
<td>0.90711</td>
<td>45</td>
<td>-0.00010</td>
</tr>
<tr>
<td>-56</td>
<td>0.00001</td>
<td>-22</td>
<td>-0.00067</td>
<td>12</td>
<td>-1.05632</td>
<td>46</td>
<td>-0.00013</td>
</tr>
<tr>
<td>-55</td>
<td>0.00001</td>
<td>-21</td>
<td>0.00129</td>
<td>13</td>
<td>0.90179</td>
<td>47</td>
<td>0.00034</td>
</tr>
<tr>
<td>-54</td>
<td>0.00001</td>
<td>-20</td>
<td>0.00415</td>
<td>14</td>
<td>-0.01375</td>
<td>48</td>
<td>-0.00041</td>
</tr>
<tr>
<td>-53</td>
<td>0.00001</td>
<td>-19</td>
<td>0.00658</td>
<td>15</td>
<td>-0.10521</td>
<td>49</td>
<td>0.00042</td>
</tr>
<tr>
<td>-52</td>
<td>0.00001</td>
<td>-18</td>
<td>0.00622</td>
<td>16</td>
<td>0.03156</td>
<td>50</td>
<td>-0.00036</td>
</tr>
<tr>
<td>-51</td>
<td>0.00001</td>
<td>-17</td>
<td>0.00855</td>
<td>17</td>
<td>-0.07061</td>
<td>51</td>
<td>0.00029</td>
</tr>
<tr>
<td>-50</td>
<td>-0.00000</td>
<td>-16</td>
<td>0.01093</td>
<td>18</td>
<td>0.07647</td>
<td>52</td>
<td>-0.00017</td>
</tr>
<tr>
<td>-49</td>
<td>-0.00001</td>
<td>-15</td>
<td>0.01262</td>
<td>19</td>
<td>-0.03566</td>
<td>53</td>
<td>0.00009</td>
</tr>
<tr>
<td>-48</td>
<td>-0.00003</td>
<td>-14</td>
<td>0.01276</td>
<td>20</td>
<td>0.03367</td>
<td>54</td>
<td>-0.00003</td>
</tr>
<tr>
<td>-47</td>
<td>-0.00005</td>
<td>-13</td>
<td>0.01874</td>
<td>21</td>
<td>-0.08756</td>
<td>55</td>
<td>-0.00002</td>
</tr>
<tr>
<td>-46</td>
<td>-0.00006</td>
<td>-12</td>
<td>0.01168</td>
<td>22</td>
<td>0.07717</td>
<td>56</td>
<td>0.00005</td>
</tr>
<tr>
<td>-45</td>
<td>-0.00007</td>
<td>-11</td>
<td>0.01654</td>
<td>23</td>
<td>-0.05212</td>
<td>57</td>
<td>-0.00006</td>
</tr>
<tr>
<td>-44</td>
<td>-0.00007</td>
<td>-10</td>
<td>-0.03295</td>
<td>24</td>
<td>0.03295</td>
<td>58</td>
<td>0.00007</td>
</tr>
<tr>
<td>-43</td>
<td>-0.00006</td>
<td>-9</td>
<td>-0.05212</td>
<td>25</td>
<td>-0.1654</td>
<td>59</td>
<td>-0.00007</td>
</tr>
<tr>
<td>-42</td>
<td>-0.00005</td>
<td>-8</td>
<td>-0.07717</td>
<td>26</td>
<td>-0.01168</td>
<td>60</td>
<td>0.00006</td>
</tr>
<tr>
<td>-41</td>
<td>-0.00002</td>
<td>-7</td>
<td>-0.08756</td>
<td>27</td>
<td>0.01874</td>
<td>61</td>
<td>-0.00005</td>
</tr>
<tr>
<td>-40</td>
<td>0.00003</td>
<td>-6</td>
<td>-0.03367</td>
<td>28</td>
<td>-0.01276</td>
<td>62</td>
<td>0.00003</td>
</tr>
<tr>
<td>-39</td>
<td>0.00009</td>
<td>-5</td>
<td>-0.03566</td>
<td>29</td>
<td>0.01262</td>
<td>63</td>
<td>-0.00001</td>
</tr>
<tr>
<td>-38</td>
<td>0.00017</td>
<td>-4</td>
<td>-0.07647</td>
<td>30</td>
<td>-0.01093</td>
<td>64</td>
<td>0.00000</td>
</tr>
<tr>
<td>-37</td>
<td>0.00029</td>
<td>-3</td>
<td>-0.07061</td>
<td>31</td>
<td>0.00855</td>
<td>65</td>
<td>0.00001</td>
</tr>
<tr>
<td>-36</td>
<td>0.00036</td>
<td>-2</td>
<td>-0.03156</td>
<td>32</td>
<td>-0.00622</td>
<td>66</td>
<td>-0.00001</td>
</tr>
<tr>
<td>-35</td>
<td>0.00042</td>
<td>-1</td>
<td>-0.10521</td>
<td>33</td>
<td>0.00658</td>
<td>67</td>
<td>0.00001</td>
</tr>
<tr>
<td>-34</td>
<td>0.00041</td>
<td>0</td>
<td>0.01375</td>
<td>34</td>
<td>-0.00415</td>
<td>68</td>
<td>-0.00001</td>
</tr>
<tr>
<td>-33</td>
<td>0.00034</td>
<td>1</td>
<td>0.90179</td>
<td>35</td>
<td>0.00129</td>
<td>69</td>
<td>0.00001</td>
</tr>
<tr>
<td>-32</td>
<td>0.00013</td>
<td>2</td>
<td>1.05632</td>
<td>36</td>
<td>0.0067</td>
<td>70</td>
<td>-0.00001</td>
</tr>
<tr>
<td>-31</td>
<td>-0.00010</td>
<td>3</td>
<td>0.90711</td>
<td>37</td>
<td>-0.00184</td>
<td>71</td>
<td>0.00001</td>
</tr>
<tr>
<td>-30</td>
<td>-0.00030</td>
<td>4</td>
<td>1.05241</td>
<td>38</td>
<td>0.00270</td>
<td>72</td>
<td>-0.00000</td>
</tr>
<tr>
<td>-29</td>
<td>-0.00068</td>
<td>5</td>
<td>0.90814</td>
<td>39</td>
<td>-0.00276</td>
<td>73</td>
<td>0.00000</td>
</tr>
<tr>
<td>-28</td>
<td>-0.00105</td>
<td>6</td>
<td>-1.03877</td>
<td>40</td>
<td>0.00192</td>
<td>74</td>
<td>0.00000</td>
</tr>
</tbody>
</table>
The sequence \( \varepsilon_q(n) \) above is a representation of a so-called perfect code in the sense that its autocorrelation function has practically zero sidelobes. It is not unique: although starting from the above-mentioned sequence \( \varepsilon(n) \), where the elements \( \varepsilon(1) \ldots \varepsilon(13) \) come from the 13-bit Barker code, causes the iteration to converge on these particular element values, something different would be arrived at if only we started from some other original sequence. As an example, we may consider the five-bit Barker code \((1, 1, 1, -1, 1)\) as a starting point and arrive at the sequence illustrated in the following table.

**Table 2: selected elements of \( \varepsilon_q(n) \) derived from 5-bit Barker code**

<table>
<thead>
<tr>
<th>( n )</th>
<th>( \varepsilon_q(n) )</th>
<th>( n )</th>
<th>( \varepsilon_q(n) )</th>
<th>( n )</th>
<th>( \varepsilon_q(n) )</th>
<th>( n )</th>
<th>( \varepsilon_q(n) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>-26</td>
<td>0.000000</td>
<td>-11</td>
<td>0.00026</td>
<td>4</td>
<td>-1.14257</td>
<td>19</td>
<td>0.00114</td>
</tr>
<tr>
<td>-25</td>
<td>-0.000000</td>
<td>-10</td>
<td>-0.00259</td>
<td>5</td>
<td>0.86775</td>
<td>20</td>
<td>-0.00025</td>
</tr>
<tr>
<td>-24</td>
<td>-0.00001</td>
<td>-9</td>
<td>-0.00647</td>
<td>6</td>
<td>-0.01781</td>
<td>21</td>
<td>-0.00024</td>
</tr>
<tr>
<td>-23</td>
<td>-0.00002</td>
<td>-8</td>
<td>-0.00560</td>
<td>7</td>
<td>-0.16160</td>
<td>22</td>
<td>0.00032</td>
</tr>
<tr>
<td>-22</td>
<td>-0.00001</td>
<td>-7</td>
<td>0.00460</td>
<td>8</td>
<td>0.13103</td>
<td>23</td>
<td>-0.00018</td>
</tr>
<tr>
<td>-21</td>
<td>0.00002</td>
<td>-6</td>
<td>0.01926</td>
<td>9</td>
<td>-0.06342</td>
<td>24</td>
<td>-0.00001</td>
</tr>
<tr>
<td>-20</td>
<td>0.00006</td>
<td>-5</td>
<td>0.03331</td>
<td>10</td>
<td>-0.01504</td>
<td>25</td>
<td>0.00008</td>
</tr>
<tr>
<td>-19</td>
<td>0.00008</td>
<td>-4</td>
<td>0.01504</td>
<td>11</td>
<td>0.03331</td>
<td>26</td>
<td>-0.00006</td>
</tr>
<tr>
<td>-18</td>
<td>0.00001</td>
<td>-3</td>
<td>-0.06342</td>
<td>12</td>
<td>-0.01926</td>
<td>27</td>
<td>0.00002</td>
</tr>
<tr>
<td>-17</td>
<td>-0.00018</td>
<td>-2</td>
<td>-0.13103</td>
<td>13</td>
<td>0.00460</td>
<td>28</td>
<td>0.00001</td>
</tr>
<tr>
<td>-16</td>
<td>-0.00032</td>
<td>-1</td>
<td>-0.16160</td>
<td>14</td>
<td>0.00560</td>
<td>29</td>
<td>-0.00002</td>
</tr>
<tr>
<td>-15</td>
<td>-0.00024</td>
<td>0</td>
<td>0.01781</td>
<td>15</td>
<td>-0.00647</td>
<td>30</td>
<td>0.00001</td>
</tr>
<tr>
<td>-14</td>
<td>0.00025</td>
<td>1</td>
<td>0.86775</td>
<td>16</td>
<td>0.00259</td>
<td>31</td>
<td>-0.00000</td>
</tr>
<tr>
<td>-13</td>
<td>0.00114</td>
<td>2</td>
<td>1.14257</td>
<td>17</td>
<td>0.00026</td>
<td>32</td>
<td>-0.00000</td>
</tr>
<tr>
<td>-12</td>
<td>0.00146</td>
<td>3</td>
<td>0.88557</td>
<td>18</td>
<td>-0.00146</td>
<td>33</td>
<td>0.00000</td>
</tr>
</tbody>
</table>

The iteration will converge if the original sequence already produced small enough sidelobes. Since Barker codes are known to have this property, they are good starting points, but also other kinds of sequences could be used to start the iteration.

Fig. 3c illustrates the autocorrelation function of the 13-bit Barker code, and fig. 3d illustrates the autocorrelation function of the perfect code illustrated in fig. 3b.
Instead of iteration, a perfect code can be derived also directly. Let us define an operator \( S_k \) that causes a shift of \( k \) elements in a sequence:

\[
(S_k \varepsilon)(n) = \varepsilon(n + k) .
\]  

(14)

Fourier-transforming a shifted sequence gives

\[
\sum_{n=-\infty}^{\infty} (S_k \varepsilon)(n)e^{-in\omega} = \sum_{n=-\infty}^{\infty} \varepsilon(n + k)e^{-in\omega} = \sum_{n=-\infty}^{\infty} \varepsilon(n)e^{-i(n-k)\omega} = e^{ik\omega} \sum_{n=-\infty}^{\infty} \varepsilon(n)e^{-i\omega} = e^{ik\omega} \varepsilon(\omega)
\]  

(15)

In order for a sequence \( \varepsilon_p(n) \) to represent a perfect code, the following condition must be true:

\[
\sum_n \varepsilon_p(n) \overline{(S_k \varepsilon_p)(n)} = \frac{1}{2\pi} \int_0^{2\pi} \left| e^{ik\omega} \overline{\varepsilon_p(\omega)} \right|^2 d\omega = \delta_{k,0} .
\]  

(16)

Here we have utilised the result of (15) and the known property of Fourier transforms that inner products of \( \ell^2 \)-sequences (lowercase-\( L \) squared) are transformed to inner products of complex-valued \( L^2 \)-functions defined on \([0, 2\pi]\). Equation (16) is true if and only if \( |\overline{\varepsilon_p(\omega)}| = 1 \) for almost all \( \omega \), because the only sequence with a constant 1 Fourier transform is known to be the simple sequence \( S_{k,0} \).

Let us now suppose that we have a sequence \( \varepsilon(n) \) that represents a close-to-perfect code. The expression "close-to-perfect" means that if we use \( \varepsilon(n) \) to substitute for \( \varepsilon_p(n) \) in formula (16), the equality is only approximatively true, i.e. in addition to the delta spike at the origin there are certain other non-zero values, but they are small compared to the height of the delta spike. The norm, or modulus of the Fourier transform of the close-to-perfect code is \( |\overline{\varepsilon(\omega)}| \approx 1 \) and we can define a perfect code \( \varepsilon_p(n) \) by

\[
\varepsilon_p(n) = \frac{1}{2\pi} \int_0^{2\pi} e^{in\omega} \overline{\varepsilon(\omega)} \overline{\varepsilon(\omega)} d\omega .
\]  

(17)
Thus, as a starting point we may use any known binary or multiphase code with low sidelobes, such as a Barker code for example, and arrive at a practically perfect code by applying formula (17). The selection of the code to start from is not limited to Barker codes. For example, the codes offered as being most optimal for mismatched filtering in the previously mentioned scientific publication of Damtie et al. are good starting points. Depending on source, the scalar measure $|\tilde{e}(\omega)|$ may be designated either as the "modulus" or the "norm" of the Fourier transform $\tilde{e}(\omega)$; we use the designation "modulus" in this description.

If the length of the starting code $\varepsilon(n)$ is finite, its Fourier transform $\tilde{\varepsilon}(\omega)$ is infinitely differentiable, and so is $\tilde{\varepsilon}(\omega)/|\tilde{\varepsilon}(\omega)|$, because for close-to-perfect codes $|\tilde{e}(\omega)| \approx 1$ and the denominator cannot be zero in $\tilde{e}(\omega)/|\tilde{e}(\omega)|$. It follows from this, that while the sequence $\varepsilon_p(n)$ that defines the perfect code is infinitely long, its elements approach zero faster than any polynomial and we only need to consider a finite number of elements necessary to satisfy any numerical accuracy we should aim at.

Strictly speaking a code would be a perfect code in the sense discussed in this description if and only if the modulus of its Fourier transform is exactly equal to one; in other words the absolute value of the Fourier transform is equal to 1 with almost all values of $\omega$. Only an infinitely long code may fulfill this strict criterion. An infinitely long code (i.e. a sequence having infinitely many elements) is a theoretical concept and not viable for practical use. On the other hand it is important to note that also a vast majority of real numbers only exist in theory, and must be approximated with something more practical in real-life devices: a computer can only handle a grid of discrete numerical values spaced apart by an amount that equals the numerical accuracy that the computer has been programmed to use.

So theoretically the modulus of a Fourier transform can equal one only if the sequence from which the Fourier transform was calculated had infinitely many elements (or if the sequence is a trivial case with only one non-zero element). However, the existence of perfect codes in practice relies on the fact that there exist sequences, the "tails" of which (in both negative and positive directions) only consist of elements the values of which are so close to zero that it would be impossible (or at least impractical) to handle them as anything else than zeroes in the memory of a digital signal processing device. Thus, it is sufficient to only store that
part of the sequence in the memory of said digital signal processing device that includes elements with values that differ enough from zero to have a practical representation. A theoretically calculated Fourier transform of (only) the infinitely long sequence would have a modulus equal to one, but in practice a numerically calculated Fourier transform of also the above-mentioned part of the sequence will have a modulus that is equal to one with the employed numerical accuracy.

A plaintext description of equation (17) above is that since with known close-to-perfect codes said modulus is only approximately \( 1 \), it is possible to derive a perfect code from a close-to-perfect code by dividing the Fourier transform of said close-to-perfect code with the modulus of said Fourier transform and taking the inverse Fourier transform of the result of such division.

A corollary of this thinking is that it is possible to generate perfect codes in general by only requiring that the modulus of a Fourier transform of a code is equal to \( 1 \); in other words, by selecting an arbitrary Fourier transform (which does not have to be the Fourier transform of any known close-to-perfect code) that happens to have the characteristic of its modulus being equal to \( 1 \) (with the required numerical accuracy), and taking its inverse transform. It is likely that such an inverse transform of just some arbitrarily selected Fourier-transform-type starting point would be quite extraordinary in shape and technically maybe not viable for use as a code. On the other hand, taken the infinitely large number of possible starting points, it is perfectly possible that also some very attractive and advantageous code sequences can be found this way.

Even if a known close-to-perfect code was selected as a starting point, it should not be considered mandatory that a perfect code could only be found by dividing the Fourier transform of such a starting point with its modulus. Dividing by modulus is only one way of selecting a modified function of modulus one that is close to but different from the original Fourier transform, and also other selections could be made. Intuitively this means that making good selections and inversely transforming them might lead to a variety of perfect codes, with some advantageous characteristics like e.g. the absolute amplitudes of all main pulses in the code being equal to a constant (and thus scalable to any convenient value, like \( 1 \)) and only phases being more complicated.

Fig. 4 illustrates schematically a programmable code source according to an embodiment of the invention. The code source comprises a processor 401, a code
memory 402, a program memory 403, a data interface 404 and optionally a user interface 405. Said elements do not need to be physically separate entities, but their existence has been separately illustrated in fig. 4 for reasons of graphical clarity. The processor 401 is configured to execute programs stored in the program memory 403. The code memory 402 is configured to store starting point codes (for example Barker codes or other close-to-perfect codes) 421, calculated perfect codes 422 and optionally also code characteristics 423, which describe features of various codes and/or information like the applicability of different codes to different purposes or statistics about code usage. The program memory 403 is configured to store programs for calculating perfect codes from starting point codes. Shown in fig. 4 are a Fourier transform based program 431 that implements formula (17) above, and an iteration based algorithm 432 that implements formula (13) above. Alternatively or additionally the program memory 403 might store other calculation programs that implement the generation of perfect codes in other ways; such alternative ways have been discussed in the two immediately preceding paragraphs above. The program memory 403 is also configured to store operating routines 433 that the processor 401 executes e.g. to respond to inputs from the user and data interfaces, to implement memory management, and the like.

The data interface 404 is the means for the processor 401 to communicate with other electronic devices, and the user interface 405 is the means for the processor 401 to exchange information with a human user. If the programmable code source has been implemented as a part of a larger computer arrangement, it can use the interfaces of that computer arrangement and does not need any of its own.

Using perfect codes for communication or remote sensing applications could naturally be accomplished also without the immediate presence of a programmable code source, for example so that a programmable code source has previously calculated a required number of perfect codes and stored them into a memory, from which a communication or remote sensing device can read them and take them into use according to need without e.g. having to know, from which starting point code some particular perfect code has been obtained and through which calculational process. Schematically, one could then consider the "Calculated Perfect Codes" part 422 as a kind of "read-only" type code source.

Fig. 5 illustrates schematically a bidirectional communications device according to an embodiment of the invention. A transceiver 501 is configured to transmit and receive digital encoded information over a transmission channel. Data to be en-
coded and transmitted comes from a data source included in block 502, and received and decoded data goes to a data sink in the same block. For encoding and decoding operations the transceiver 501 uses codes obtained from a code source 503, which may be for example a programmable code source of the kind illustrated in fig. 4 or a read-only type code source. The transceiver 501 may be for example a transceiver of a cellular radio system that utilises spread spectrum technology for encoding transmissions over the wireless interface between base stations and mobile terminals, in which case it uses the codes read from the code source 503 as the pseudorandom sequence needed for spreading and despreading operations.

The invention does not require the communications device to be capable of bidirectional communications. For example in conventional broadcasting applications a broadcasting station may well be equipped for encoding and transmission only, and terminal stations may be equipped for reception and decoding only.

Fig. 6 shows a radar system, which is an example of a remote sensing system according to an embodiment of the invention. In this example, an antenna 600 functions as both transmission and reception antenna and the signal is directed from the transmitter 601 to the antenna or from the antenna to the receiver 603 by means of a duplexing switch 604. The receiver 603 comprises in a known manner an intermediate frequency (IF) mixer, IF amplifier 606 and an analog-to-digital converter 607. Mixing frequency comes to the IF mixer 605 and transmitter 601 from a local oscillator 602. In this exemplary radar set a transmitted pulse-coded signal is measured by a separate receiver 608 which is comprised of corresponding parts: an IF mixer 609, IF amplifier 610 and an analog-to-digital converter 611. Measuring transmitted signals is not necessary in many radar systems where the form and content of the transmitted signal can otherwise be known at the required accuracy.

In the exemplary apparatus the transmitter 601 is controlled and the received data are processed by a computer 612. The shape of the pulse-coded signal to be transmitted comes from a code source 614, which may be for example a programmable code source such as shown above in fig. 4, or a read-only type code source. Signal detection is performed by means of software in an I/Q detection block 615. For the detection of measured transmission pulses the system shown in Fig. 6 has a separate detection block 613. Computation of reflectivity starts with the squaring of the signal in block 619 the result of which is processed so as to
become unambiguous with respect to the range by solving in block 620 the equations representing the response of the measurement. The final radar reflectivity results are produced in block 621. When the reflectivity values for the various ranges have been calculated, the final results can be produced and stored and/or displayed to the user in a known manner. In the case of a weather radar, a typical end result is a pattern in which the reflectivity values are interpreted as meteorological phenomena (such as rain of different intensities) and presented graphically using color codes. The invention does not limit the way in which the final results are generated or presented.

Computation of delay values is a little more complicated. The cross products of the samples are computed in block 616 whereafter the equations depending on the range of the delays are solved in block 617. The coefficients of these equations depend on the contents of the transmitted encoded signal. Therefore, data representing the transmitted signal measured by receiver 608, and/or data representing the original form of the transmission code from the code source 614 are included in the solution of the equations in block 617. The final velocity and Doppler spectrum products are produced in block 618 which may also use the reflectivity data of block 620. As regards the generation and storing and/or presentation of the final results, refer to what was stated above in connection with the final results representing reflectivity.

The use of perfect codes according to an embodiment of the invention and the arithmetic blocks described above are preferably realized in the equipment shown in Fig. 6 in such a manner that the computer 612 comprises at least one program memory in which a stored program controls the operation of the computer 612 so that the method according to the invention is part of said program. Programming the matrix operations, computation of delayed products and other arithmetic operations described above into processes executable by a computer is familiar to a person skilled in the art. The computer 612 may comprise one or more processors to realize the method according to the invention and other tasks relating to the control of the operation of the transmitter 601 and receivers 603 and 608. Preferably the computer 612 also comprises means, which is known per se, for controlling the movement of the antenna 600; for simplicity, said means is not shown in Fig. 6. The system of Fig. 6 may be located in a fixed manner at a ground station or on a vehicle, such as airplane, or it may be realized so as to be portable.
Figs. 7 and 8 are schematic illustrations of both a method aspect and a computer program product aspect that constitute embodiments of the present invention. Fig. 7 corresponds to producing a code by using the direct approach discussed above in association with formula (17). At step 701, a starting point code is read. Its discrete Fourier transform is calculated at step 702, and the modulus of the discrete Fourier transform is calculated at step 703. At step 704 formula (17) is applied to calculate a sequence $\epsilon_p(n)$, and its elements are investigated to identify that range of indexes $n$ that includes elements, the values of which differ from zero with the predefined numerical accuracy that is used. Those elements will constitute the finite sequence of elements that is eventually output as the perfect code at step 704.

In fig. 8 the starting point code is read at step 801. In the iteration, the starting point code constitutes the sequence of the 1:st order. The corresponding mismatched filter is produced at step 802 using formula (5). Step 803 represents the iteration, i.e. repeated application of formula (13) until a predetermined ending condition is fulfilled. One exemplary ending condition is the requirement that two sequences of consecutive order must differ from each other less than a certain limit; e.g. the sequences must be same with the predefined numerical accuracy. At step 804 the non-zero elements (with the predefined numerical accuracy) of the final sequence are output as the desired perfect code.

Mathematically, the sequences that are handled according to the calculational formulas mentioned above should be infinitely long, which is impossible to implement in a real-life computer. A sufficiently good approximation of infinite sequences can be made by making the calculations with sequences that are much longer than any known practical sequence needed to represent a perfect code. In the examples above the perfect codes have somewhat more or somewhat less than one hundred elements, so something like several hundreds or several thousands elements should be enough for the sequences used in calculations.

The embodiments of the invention discussed above are exemplary only and do not limit the invention. In particular, it is possible to make small changes to calculated perfect codes, for example in order to round some calculationally obtained values to closest practical values that can be expressed in a conveniently short number of bits or in order to achieve some other comparable objectives - as long as such small changes only cause very small changes to the autocorrelation function of a truly perfect code, the resulting practical code is still a perfect code in the sense of
the present invention. There is typically a trade-off between using a convenient representation of a code and obtaining the best possible autocorrelation characteristics. How the trade-off decision is made depends on a variety of reasons, including but not being limited to the computing and storage capacity of the equipment that is used, and the resolution at which the element values of a code can be represented in an eventual signal to be transmitted. For example, if the element values of the code should be reflected in amplitude values of a signal to be transmitted, and the signal to be transmitted is generated in a D/A converter that operates with a certain quantization accuracy, it is not practical to calculate and store the values of the elements in the code with an accuracy that would be greater than what the D/A converted can implement in practice.

Similar trade-off considerations apply also to other changes that one could make to the element values or other characteristics of a perfect code. For example, changing the value of one of the five-decimal elements in Table 1 above by the amount of 0.00001 would cause only a very small change to the autocorrelation characteristics of the code. Changing it by 0.00002, or changing two element values each by 0.00001 would cause a slightly bigger change, and so on - the more changes in element values, the bigger changes in autocorrelation, until at some point the autocorrelation function cannot be said to have zero sidelobes any more. However, all such "tampered" perfect codes are still perfect codes in the sense of this description, because making changes to the element values is essentially the same as just taking a coarser predetermined numerical accuracy.

The numerical accuracy that is considered does not need to be a power of ten or a power of two, but any convenient numerical accuracy can be used.
Claims
1. An electronic device for outputting a code, characterized in that the electronic device is configured to output a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

2. An electronic device according to claim 1, characterized in that as said finite sequence of elements it is configured to output a finite subset of an infinite sequence of at least one of the following kinds:

\[ \varepsilon_p(n) = \frac{1}{2\pi} \int_{\pi}^{2\pi} \frac{\varepsilon(\omega)}{|\tilde{\varepsilon}(\omega)|} d\omega \quad \text{or} \]

\[ \varepsilon_{p+1}(n) = \frac{1}{2}(\varepsilon_p(n) + \lambda_p(-n)) \]

where \( \varepsilon_p(n) \) is the n:th element of a sequence of the p:th order, \( \varepsilon_{p+1}(\pi) \) is the n:th element of a sequence of the (p+1):th order, \( n \) is an integer, \( p \) is a positive integer, \( e \) is the neper base, \( i \) is the imaginary unit, \( \omega \) is an integration variable, \( \tilde{\varepsilon}(\omega) \) is a discrete Fourier transform of a starting point code with low sidelobes in its autocorrelation function, and \( \lambda_p(-n) \) is the (-/i):th element of the impulse response of a mismatched filter corresponding to said sequence of the p:th order, and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

3. An electronic device according to claim 1, characterized in that it is a code source configured for use in a communication device that implements coded communications with other communication devices.
4. An electronic device according to claim 1, characterized in that it is a signal source configured for use in a remote sensing device that transmits a coded transmission signal towards a target.

5. A communication device adapted to implement coded communications with at least one other communication device, characterized in that it comprises:
   - at least one of a transmitter and a receiver configured to use a code to implement coded communications, and
   - a code source configured to provide said at least one of a transmitter and a receiver with a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

6. A communication device according to claim 5, characterized in that as said finite sequence of elements said code source is configured to provide said at least one of a transmitter and a receiver with a finite subset of an infinite sequence of at least one of the following kinds:

\[
\varepsilon_p(n) = \frac{1}{2\pi} \int_0^{2\pi} \frac{\xi(\omega) e^{in\omega}}{|\xi(\omega)|} d\omega \quad \text{or} \quad \varepsilon_{p+1}(n) = \frac{1}{2} (\varepsilon_p(n) + \lambda_p(-n))
\]

where
- \(\varepsilon_p(n)\) is the \(n\):th element of a sequence of the \(p\):th order,
- \(\varepsilon_{p+1}(n)\) is the \(n\):th element of a sequence of the \((p+1)\):th order,
- \(n\) is an integer,
- \(p\) is a positive integer,
- \(e\) is the neper base,
- \(i\) is the imaginary unit,
- \(\omega\) is an integration variable,
- \(\xi\{\omega\}\) is a discrete Fourier transform of a starting point code with low sidelobes in its autocorrelation function, and
- \(\lambda_p(-n)\) is the \((-n)\):th element of the impulse response of a mismatched filter corresponding to said sequence of the \(p\):th order,
and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

7. A remote sensing device adapted to transmit a coded transmission signal towards a target, characterized in that it comprises:
   - a transmitter, and
   - a transmission signal source configured to provide said transmitter with a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1;
   wherein said transmitter is configured to use said code to transmit a coded transmission signal.

8. A remote sensing device according to claim 7, characterized in that as said finite sequence of elements said transmission signal source is configured to provide said transmitter with a finite subset of an infinite sequence of at least one of the following kinds:

\[
e_p(n) = \frac{1}{2\pi} \int_{0}^{2\pi} e^{in\omega} \frac{\mathcal{E}(\omega)}{|\mathcal{E}(\omega)|} d\omega \quad \text{or} \quad \epsilon_{p+1}(n) = \frac{1}{2}(\epsilon_p(n) + \lambda_p(-n)),
\]

where \(\epsilon_p(n)\) is the \(n\):th element of a sequence of the \(p\):th order, \(\epsilon_{p+1}(n)\) is the \(n\):th element of a sequence of the \((p+1)\):th order, \(n\) is an integer, \(p\) is a positive integer, \(e\) is the neper base, \(i\) is the imaginary unit, \(\omega\) is an integration variable, \(\mathcal{E}(\omega)\) is a discrete Fourier transform of a starting point code with low sidelobes in its autocorrelation function, and \(\lambda_p(-n)\) is the \((-n)\):th element of the impulse response of a mismatched filter corresponding to said sequence of the \(p\):th order,
and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

9. A method for producing a code, characterized in that the method comprises producing a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

10. A method according to claim 9, characterized in that the method comprises:
- producing a discrete Fourier transform of a starting point code,
- producing a modulus of said discrete Fourier transform, and
- producing said code as a finite subset of an infinite sequence of elements of the following kind:

$$\varepsilon_p(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\bar{\mathcal{E}}(\omega)}{|\mathcal{E}(\omega)|} d\omega,$$

where $\varepsilon_p(n)$ is the n:th element of a sequence of the p:th order, $n$ is an integer, $p$ is a positive integer, $e$ is the neper base, $i$ is the imaginary unit, $\omega$ is an integration variable, and $\mathcal{E}(\omega)$ is said discrete Fourier transform of a starting point code,

and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

11. A method according to claim 9, characterized in that the method comprises:
- producing a starting point code, and
- producing said code as a finite subset of an infinite sequence of elements by performing a number of iterations of the following kind:

$$\varepsilon_p = \frac{1}{2} (\varepsilon_p(n)^+ \lambda_p(-n)),$$
where $\varepsilon_p(n)$ is the n:th element of a sequence of the p:th order, and said starting point code constitutes the sequence of the 1:st order, $\varepsilon_{p+1}(n)$ is the n:th element of a sequence of the (p+1):th order, $n$ is an integer, $p$ is a positive integer, and $\lambda_p(-n)$ is the (-n):th element of the impulse response of a mismatched filter corresponding to said sequence of the p:th order, and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

12. A method for transmitting an encoded message, characterized in that it comprises:
- representing a message to be transmitted in the form of a sequence of symbols,
- encoding said sequence of symbols by convolving it with a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1, to produce an encoded message, and
- transmitting said encoded message.

13. A method according to claim 12, characterized in that said convolving comprises convolving said sequence of symbols with a finite subset of an infinite sequence of elements of at least one of the following kinds:

$$
\varepsilon_p(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{i\omega n} \left| \overline{\hat{e}(\omega)} \right| d\omega \quad \text{or}
$$

$$
\varepsilon_{p+1}(n) = \frac{1}{2} (\varepsilon_p(n) + \lambda_p(-n))
$$

where $\varepsilon_p(n)$ is the n:th element of a sequence of the p:th order, $\varepsilon_{p+1}(n)$ is the n:th element of a sequence of the (p+1):th order, $n$ is an integer, $p$ is a positive integer, $e$ is the neper base, $\pi$ is the imaginary unit,
ω is an integration variable, 
\( \mathcal{E}(\omega) \) is a discrete Fourier transform of a starting point code with low sidelobes in its autocorrelation function, and
\( \lambda_p(-n) \) is the \((-n):th\) element of the impulse response of a mismatched filter corresponding to said sequence of the \( p:th \) order,
and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

14. A method for receiving and decoding an encoded message, characterized in that it comprises:
- receiving an encoded message in the form of a sequence of encoded symbols,
- decoding said sequence of encoded symbols by convolving it with a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

15. A method according to claim 14, characterized in that said convolving comprises convolving said sequence of encoded symbols with a finite subset of an infinite sequence of elements of at least one of the following kinds:

\[
\mathcal{E}_p(n) = \frac{1}{2\pi} \int_0^{2\pi} e^{i\omega n} \mathcal{M}(\omega) d\omega
\]

or

\[
\mathcal{E}_p(n) \approx \frac{1}{2} \left( \mathcal{E}_p(n) + \lambda_p(-0) \right)
\]

where \( \mathcal{E}_p(n) \) is the \( n:th \) element of a sequence of the \( p:th \) order, \( \mathcal{E}_{p+1}(\omega) \) is the \( n:th \) element of a sequence of the \( (p+1):th \) order, \( n \) is an integer, \( p \) is a positive integer, \( e \) is the neper base, \( i \) is the imaginary unit, \( \omega \) is an integration variable, \( \mathcal{E}(\omega) \) is a discrete Fourier transform of a starting point code with low sidelobes in its autocorrelation function, and
26

\( \lambda_p(-n) \) is the (-n):th element of the impulse response of a mismatched filter corresponding to said sequence of the p:th order,

and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.

16. A method for remotely sensing a target, characterized in that it comprises transmitting a coded transmission signal towards a target, which coded transmission signal comprises a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.

17. A method according to claim 16, characterized in that said finite sequence of elements is a finite subset of an infinite sequence of at least one of the following kinds:

\[
\xi_p(n) = \frac{1}{2\pi} \int_0^{2\pi} \frac{\xi(\omega)}{|\xi(\omega)|} d\omega \quad \text{or}
\]

\[
\xi_{p+1}(n) = \frac{1}{2} \left( \xi_p(n) + \lambda_p(-n) \right),
\]

where \( \xi_p(n) \) is the n:th element of a sequence of the p:th order,

\( \xi_{p+1}(n) \) is the n:th element of a sequence of the (p+1):th order,

n is an integer,

p is a positive integer,

e is the neper base,

\( / \) is the imaginary unit,

\( \omega \) is an integration variable,

\( \xi(\omega) \) is a discrete Fourier transform of a starting point code with low sidelobes in its autocorrelation function, and

\( \lambda_p(-n) \) is the (-n):th element of the impulse response of a mismatched filter corresponding to said sequence of the p:th order,

and where said finite subset comprises those elements of the infinite sequence that differ from zero with said predetermined numerical accuracy.
18. A computer program product for producing a code, characterized in that the computer program product comprises machine-readable instructions which, when executed on a computer, cause the computer to produce a finite sequence of elements, which finite sequence of elements constitutes a code that with a predefined numerical accuracy is equal to an infinite sequence of elements that has the characteristic that the modulus of the Fourier transform of said infinite sequence of elements is equal to 1.
Fig. 1
PRIOR ART

Fig. 2
PRIOR ART

Fig. 3a

Fig. 3b
A. CLASSIFICATION OF SUBJECT MATTER

See extra sheet

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC8: H03M, H04B, G01S, G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

FI, SE, NO, DK

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI, XP3E

C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No</th>
</tr>
</thead>
</table>

Further documents are listed in the continuation of Box C

See patent family annex.

Date of the actual completion of the international search 07 April 2008 (07.04.2008)

Date of mailing of the international search report 06 May 2008 (06.05.2008)

Name and mailing address of the ISA/FI

National Board of Patents and Registration of Finland
PO Box 1160, FI-00101 HELSINKI, Finland
Facsimile No. +358 9 6939 5328

Authorized officer
Timo Laakso
Telephone No. +358 9 6939 500
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family members(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td>US 6778603 B1</td>
<td>17/08/2004</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>US 6393047 B1</td>
<td>21/05/2002</td>
<td>DE 69933029 T D1</td>
<td>12/04/2007</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 69933029D D1</td>
<td>12/10/2006</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ZA 200007304 A</td>
<td>11/10/2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TW 448644B B</td>
<td>01/08/2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WO 9966645 A</td>
<td>23/12/1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RU 2226035 C2</td>
<td>20/03/2004</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 2002518928T T</td>
<td>25/06/2002</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EP 1088416 A2</td>
<td>04/04/2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CN 1316142 A</td>
<td>03/10/2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CA 2335018 A1</td>
<td>23/12/1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>BR 991 1341 A</td>
<td>13/03/2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AU 4812799 A</td>
<td>05/01/2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AU 756686B B2</td>
<td>23/01/2003</td>
</tr>
<tr>
<td>US 5847677 A</td>
<td>08/12/1998</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

Form PCT/ISA/210 (patent family annex) (April 2007)
<table>
<thead>
<tr>
<th>Classification</th>
<th>Code</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Int.CI.</td>
<td>H04B 1/707</td>
<td>2006.01</td>
</tr>
<tr>
<td></td>
<td>H04B 1/69</td>
<td>2006.01</td>
</tr>
<tr>
<td></td>
<td>G01S 13/00</td>
<td>2006.01</td>
</tr>
<tr>
<td></td>
<td>H04B 7/276</td>
<td>2006.01</td>
</tr>
<tr>
<td></td>
<td>G06F 7/58</td>
<td>2006.01</td>
</tr>
</tbody>
</table>