Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE DÉFENSE

Demodulation and Decoding Studies of the 39-tone MIL-STD-188-110A HF Signal

Gerard Duprat

Defence R&D Canada - Ottawa
TECHNICAL MEMORANDUM
DRDC Ottawa TM 2002-082
November 2002

Canada

Demodulation and decoding studies of the 39-tone MIL-STD-188-110A HF signal

Gerard Duprat

Defence R&D Canada – Ottawa

Technical Memorandum

DRDC Ottawa TM 2002-082

November 2002

Abstract

A set of algorithms has been developed to demodulate and decode the 39 tone signal, which is prevalent in the High Frequency (HF) frequency band. This signal, based on the MIL-STD-188-110A Standard, is one of several different types generated by the AN/PRC-138 Harris radio. Defence R&D Canada (DRDC) - Ottawa has two of these radios. Our work focused on the 39 tone fixed frequency mode, although the 39 tone signal is also the underlying modulation used in the AN/PRC-138's frequency hopping mode. Thus, the work described here will also be useful to anyone developing dehopping and demodulation algorithms for the AN/PRC-138 frequency hopping signal. The signals were captured by the Agilent Technologies Blackbird system in a laboratory setting. The objective of the task was to gain a detailed understanding of the sophisticated 39 tone signal and to develop software radio algorithms for demodulating and decoding the signal. The report describes the signal structure, the signal capture equipment, and the steps involved in demodulating and decoding the signal so that the transmitted messages can be read at the receiving end. The report also describes the problems encountered during the algorithm development process. Since the signal captures took place in an ideal setting and *a priori* information of the signal structure was used to assist in the demodulation and decoding process, the current algorithms must be modified slightly to be able to handle the more general situation. The report concludes with ways of generalizing the algorithms.

Résumé

Un ensemble d'algorithmes a été conçu afin de démoduler et de décoder les signaux composés de fréquences multiples, utilisées simultanément (modem à 39 tons, mode d'opération parallèle). Ce type de signal est utilisé de façon prépondérante dans la bande de fréquence HF. Ce signal est défini par le standard MIL-STD-188-110A. Les radios de modèle AN/PRC-138 manufacturées par Harris se servent de ce type de signal, parmi tant d'autres, pour transmettre données et texte. Recherche et Développement pour la Défense Canada (RDDC) - Ottawa possède deux de ces radios. Bien que durant la présente étude l'accent soit mis seulement sur le mode d'opération des radios en fréquence fixe, il doit être mentionné que les signaux à 39 tons sont aussi utilisés par les radios AN/PRC-138 lorsqu'elles opèrent en mode bonds de fréquence. Donc le travail décrit ici sera utile à toute personne qui voudra développer des algorithmes dont le but est d'attacher ensemble les bonds d'une même fréquence ou de démoduler les signaux transmis par les dites radios. Les signaux furent interceptés et recueillis par le système Blackbird conçu par Agilent Technologies. Le but de ce travail consistait à acquérir une connaissance détaillée de ce genre de signal sophistiqué ainsi que de développer un logiciel avec des algorithmes convenant à ce type de signal radio. Le rapport décrit la structure du signal, l'équipement utilisé durant l'interception des signaux, ainsi que les différentes étapes du processus de démodulation et de décodage.

Ce processus a été mené jusqu'à son terme c'est à dire jusqu'au point où un message peut être lu. Le rapport décrit aussi les problèmes rencontrés durant le développement des algorithmes. Il y en eut beaucoup. Il doit être mentionné que les signaux ont été interceptés et recueillis dans des conditions idéales dans le laboratoire et en l'absence de tout bruit de fond. De plus, certaines caractéristiques de la structure du signal furent utilisées *a priori* durant le processus de démodulation et de décodage. En conséquence, ces algorithmes devront être légèrement modifiés afin d'être utilisés dans des conditions plus générales et plus proche de la réalité. Le rapport conclu avec des suggestions qui rendront l'utilisation de ces algorithmes plus générale.

Executive summary

The main objectives of the work described in this report were to gain a good understanding of the signal generated by the 39 tone modem used in the Harris AN/PRC-138 manpack radio, and to reach a stable point in the development of software radio algorithms used to demodulate and decode this signal. These goals were achieved, making it possible to retrieve the content of a message which had been captured in a laboratory environment.

The work focused on the 39 tone fixed frequency mode of operation. However, the 39 tone signal is also the underlying modulation used in the AN/PRC-138's frequency hopping mode. Thus, the work described here will also be extremely useful to anyone developing dehopping and demodulation algorithms for the AN/PRC-138 frequency hopping signal.

The 39 tone signal uses 39 subcarrier tones located in the audio frequency band. Each tone is a Differential Quadrature Phase Shift Keyed (DQPSK) signal. A uniform baud rate of 44.44 symbols per second is used for all possible bit rates, which are 75, 150, 300, 600, 1200, and 2400 bps. With this baud rate, the time duration of each symbol (symbol period or signal element) is $T_{symb} = 22.5$ ms. For bit rates less than 2400 bps, there will be some redundancy bits built into the signal. This is achieved with the time and frequency diversity built into the signal structure. The lower the bit rate, the greater the redundancy.

At the start of a transmission, a preamble is used for synchronizing the receiving and transmitting equipment, and for establishing a phase reference for subsequent signal elements (symbol periods). Periodically throughout the transmission, a synchronization signal is sent; this is accomplished through a framing process. The signal also has a 40^{th} tone added to it, which is unmodulated. This tone, also known as the Doppler or pilot tone, is used to correct frequency offsets introduced either by Doppler shifts in the signal during transmission or by radio equipment instabilities.

The signal captures were carried out in an ideal laboratory environment. To handle a more general case, the current algorithms would have to be enhanced. Presently, time synchronization of the data prior to demodulation is based on use of the preamble tones. As mentioned in the report, a signal reception can occur anywhere during the transmission, i.e., one might miss the preamble. If this is the case, any of the 39 tones can be used for time synchronization purposes. This added capability in the algorithms will require only a slight modification. Another problem is that *a priori* knowledge of some parameters were used for the deinterleaving and decoding activities (i.e., bit rate and interleaving degree); this information may not be readily available and, therefore, will have to be extracted in some way from the signal itself. Bit rate can be easily identified by looking at the redundancy within one signal element (for 2400 and 1200 bps) and from the redundancy patterns existing between subsequent signal elements for bit rates below 1200 bps. Once the bit rate has been identified, the interleaving degree can be deduced by determining the length in bits of the insertion interval between framing sequence bits. Also, the decoding algorithms must be improved to be able to handle the 2400 bps case which uses 56-bit codewords.

Finally, an area requiring more research is the problem of the rotation of the signal constellations associated with each tone. It was speculated (but not proven) that this rotation probably results from an offset in the spacing of the 39 tones.

Duprat, G. 2002. Demodulation and decoding studies of the 39-tone MIL-STD-188-110A HF signal. DRDC Ottawa TM 2002-082. Defence R&D Canada – Ottawa.

Sommaire

L’objectif premier du travail décrit dans ce rapport est d’acquérir une connaissance solide du signal génèré par le modem à 39 tons utilisé dans l’équipement portable radio AN/PRC-138 manufacturé par Harris. Cet objectif inclut aussi la stabilisation des algorithmes utilisés pour démoduler et décoder ce type de signal. Ce but a été atteint. Il fut possible de récupérer le contenu d’un message à partir de données interceptées dans le laboratoire.

Bien que durant la présente étude l’accent soit mis seulement sur le mode d’opération des radios en fréquence fixe, il doit être mentionné que les signaux à 39 tons sont aussi utilisés par les radios AN/PRC-138 lorsqu’elles opèrent en mode bonds de fréquence. Donc le travail décrit ici sera utile à toute personne qui voudra développer des algorithmes dont le but est d’attacher ensemble les bonds d’une même fréquence ou de démoduler les signaux transmis par les dites radios.

Le signal à 39 tons utilise 39 sous-porteuses localisées dans la bande de fréquence acoustique. Chaque ton (ou chaque sous-porteuse) est modulé differentiellement en quadrature par déplacement de phase (DQPSK). Un débit uniforme de 44.44 bauds (ou symboles par seconde) est utilisé pour tous les débits binaires possibles. Avec un tel débit la durée de chaque symbole sera de 22.5 ms. Pour ce type de signal le débit binaire ne peut être que de 75, 150, 300, 600, 1200 ou 2400 bps. Pour les débits inférieurs à 2400 bps il y aura quelques symboles binaires (ou bits) redondant inclus dans le signal. Ceci est accompli grâce à un processus de diversité en temps et en fréquence inclus dans la structure du signal. Plus le débit binaire sera faible, plus il y aura de bits superflus dans le signal.

Au début de chaque transmission, un préambule est utilisé afin de synchroniser les équipements émetteurs et récepteurs, ainsi que pour définir une valeur de référence pour la phase de chaque sous-porteuse. Durant la transmission un signal de synchronisation est envoyé périodiquement. Il existe aussi dans le signal un quarantième ton qui lui n’est pas modulé. Ce ton est connu sous le non de ton Doppler (ou parfois ton pilote). Il est utilisé pour corriger les déviations en fréquence causées dans le signal par l’effet Doppler ou bien par des instabilités propres à l’équipement radio.

Les signaux ont été interceptés et recueillis dans des conditions idéales dans le laboratoire et en l’absence de tout bruit de fond. En conséquence, ces algorithmes devront être modifiés afin d’être utilisés dans des conditions plus générales et plus proche de la réalité. Actuellement la synchronisation en temps des données, nécessaire pour obtenir une démodulation exacte, est accomplie en utilisant les caractéristiques des sous-porteuse durant le préambule. Il est évident qu’un signal peut être intercepté et reçu n’importe quand durant la transmission. Il est donc possible que le préambule ne soit pas intercepté. Dans ce cas, n’importe laquelle des 39 sous-porteuses, existant dans un symbole, peut être utilisée pour synchroniser les données en temps. Modifier les algorithmes en ce sens sera simple. Un autre problème réside dans le fait qu’une connaissance *a priori* de certain paramètres est utilisée notamment pour démêler et

regrouper ainsi que pour décoder les données (par exemple le débit binaire et le degré de fragmentation). Cette information peut ne pas être accessible directement et donc doit être extraite du signal lui-même. Le débit binaire est facilement identifiable en observant le patron (la structure) des bits redondants à l'intérieur de chaque symbole et aussi comment ce patron varie d'un symbole à l'autre. Une fois le débit binaire établi, le degré de fragmentation sera trouvé en déterminant la longueur des données, en bit, existant entre deux séquences consécutives de synchronisation.

Pour finir, la cause de la rotation observée dans la constellation du signal et qui dépend de la fréquence de chaque sous-porteuse devra être investiguée. Il a été suggère (mais non formellement prouvé) que cette rotation était probablement causée par un léger désaccord existant dans le taux d'échantillonnage entre les radios et l'équipement utilisé pour intercepter le signal. Ceci fait que les 39 sous-porteuses ne sont pas exactement séparées par 56.25 Hz tel qu'indiqué dans le standard MIL-STD-188-110A.

Duprat, G. 2002. Demodulation and decoding studies of the 39-tone MIL-STD-188-110A HF signal. DRDC Ottawa TM 2002-082. R&D pour la défense Canada – Ottawa.

Table of contents

List of figures

List of tables

1. INTRODUCTION

A set of algorithms have been developed to demodulate and decode the 39 tone signal, based on the MIL-STD-188-110A standard. The 39 tone signal is found in the HF frequency band and is one of several different types (including Automatic Link Establishment (ALE) and Frequency Hopping) used by two Harris AN/PRC-138 manpack radios purchased by DRDC Ottawa for experimental purposes. The 39 tone signal is the underlying modulation used in the AN/PRC-138's frequency hopping mode. Thus, this work will be very useful to anyone developing dehopping and demodulation algorithms for the AN/PRC-138 frequency hopping signal. Our work, however, focused on the the 39 tone fixed frequency mode.

The signals were captured using an Agilent Blackbird system in a laboratory setting. The objective of the task was to gain a detailed understanding of this complex signal so that demodulation and decoding algorithms could be developed. This report describes the details of the signal structure, the signal capture equipment, and the steps involved in demodulating and decoding the signal. The algorithms have been developed so that the transmitted message can be read at the receiving end. The problems encountered during the algorithm development process are also described. Since the signal captures took place in an ideal setting, and *a priori* information of the signal structure was used to assist in the demodulation and decoding process, the current algorithms would need to be modified slightly to handle a more general situation. Therefore this report concludes with methods for carrying out these generalizations.

2. OVERVIEW OF THE 39-TONE MODEM SIGNAL

This section presents the general concepts associated with the 39 tone signal, the details of the signal structure, issues related to encoding and decoding the data, and finally the steps involved in extracting the data bits from the received signal. The contents of this section are based on information provided in [1].

2.1 General Concepts

The 39 tone signal uses 39 subcarrier tones located in the audio frequency band. Each tone is a Differential Quadrature Phase Shift Keyed (DQPSK) signal. A uniform baud rate of 44.44 symbols per second is used for all possible bit rates, which are 75, 150, 300, 600, 1200, and 2400 bps. With this baud rate, the time duration of each symbol (symbol period or signal element) is $T_{symb} = 22.5$ ms. For bit rates less than 2400 bps, there will be some redundancy bits built into the signal, the redundancy increasing as the bit rate decreases. This will be shown in more detail later.

At the start of a transmission, a preamble is used for synchronizing the receiving and transmitting equipment, and for establishing a phase reference for subsequent signal elements (symbol periods). Periodically throughout the transmission, a synchronization signal is sent; this is accomplished through a framing process. The signal also has a 40^{th} tone added to it, which is unmodulated. This tone, also known as the Doppler or pilot tone, is used to correct frequency offsets introduced either by Doppler shifts in the signal during transmission, or by radio equipment instabilities.

The process of generating the overall signal at baseband is illustrated in Figures 1 and 2. Given a sequence of bits, the signal is constructed in the digital frequency domain. A block of bits corresponding to a symbol period is mapped onto the 39 tones, where each tone (which represents 2 bits during that symbol period) will have a phase corresponding to the 2-bit symbol. Once the signal in the frequency domain is constructed in this fashion for that symbol period, the Doppler tone is then added. At that point, the frequency domain signal is converted to a sampled signal in the time domain using the inverse Fast Fourier Transform (FFT). The sampled signal is then applied to a D/A converter. The resulting analog baseband signal is used to modulate a Single Sideband (SSB) signal at Radio Frequency (RF).

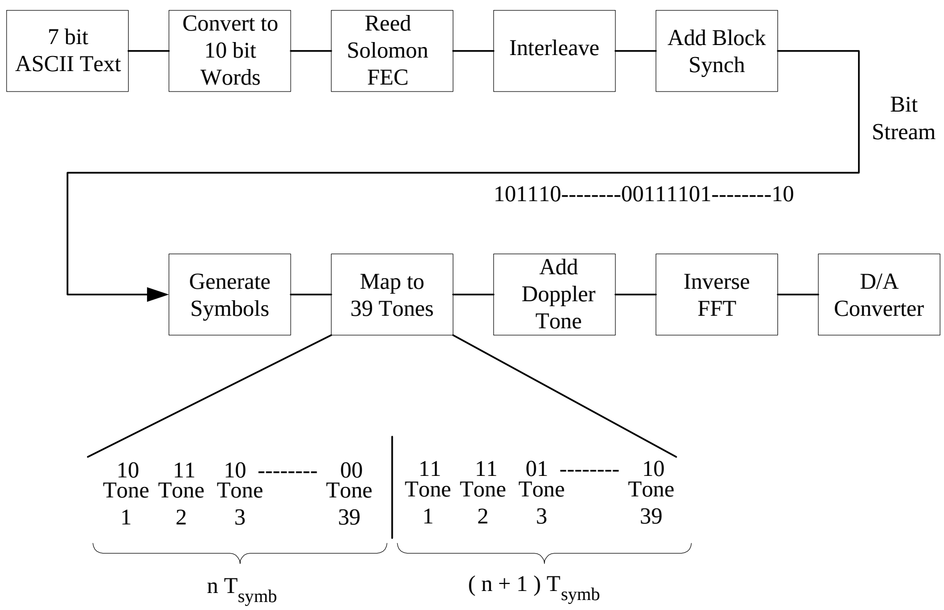


Figure 1: *Message generating process using the 39 tone modem.*

The task of the Electronic Support Measures (ESM) operator, given an SSB 39 tone signal that has been received at RF, is to work backwards. The operator must tune the ESM receiver onto the signal, mix it to baseband where it is digitized, attempt to synchronize to the signal in time and frequency, obtain a sequence of symbols by measuring the phase difference from symbol period to symbol period, convert the symbol sequence into a serial bit stream, and perform bit deinterleaving and Forward Error Correction (FEC) decoding. The process of accomplishing this will be discussed in the subsequent paragraphs.

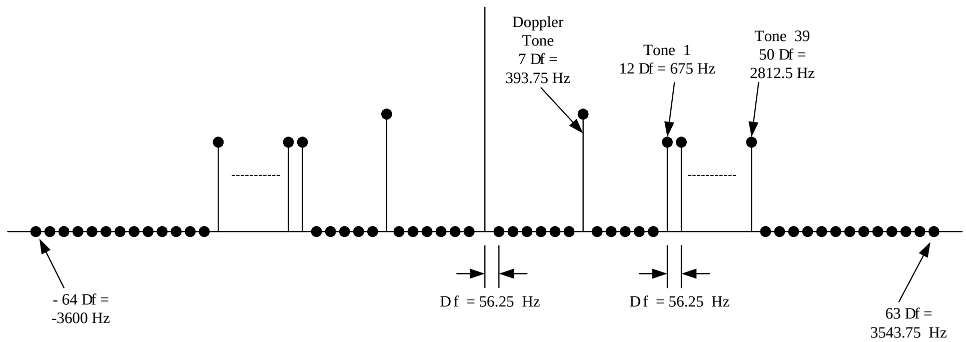


Figure 2: *Spectrum structure of the 39 tone signal.*

2.2 Signal Structure

The sequence of events occurring during the transmission phase is illustrated in Figure 3. From this figure, it can be seen that the signal is composed of a preamble, synchronization blocks, and data blocks.

Prior to transmission of the data, a three-part preamble is transmitted. The first part of the preamble is 14 signal elements (symbol periods) long, and consists of four equal-amplitude unmodulated data tones of 787.5, 1462.5, 2137.5 and 2812.5 Hz. Part two of the preamble is 8 signal elements long and consists of three modulated data tones of 1125.0, 1800.0 and 2475.0 Hz. These tones are advanced by π radians at the beginning of each signal element of the second part of the preamble. The third part of the preamble is one signal element in duration and consists of all 39 tones (56.25 Hz apart) plus the Doppler tone. This part of the preamble establishes the starting phase reference for subsequent signal elements. The tone phases at the onset of each part of the preamble, along with their normalized amplitudes, are described in Table 1.

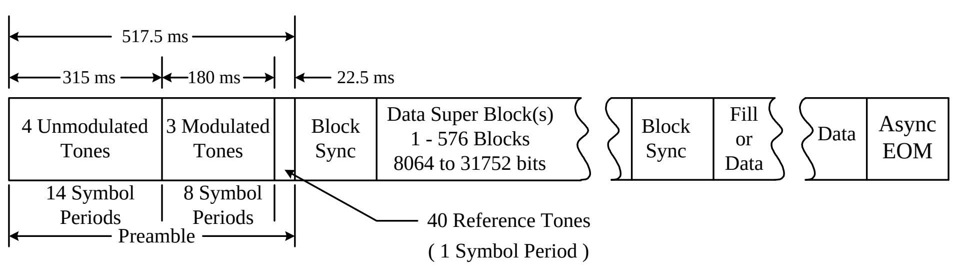


Figure 3: *Signal structure of the 39 tone waveform.*

Table 1: Normalized Amplitudes and Phases for Tones

Preamble Part	Tone Frequency (Hz)	Tone Function	Amplitude	Initial Phase (Deg)
1	787.50	Data Tone 3	3	0.0
1	1462.50	Data Tone 15	3	103.7
1	2137.50	Data Tone 27	3	103.7
1	2812.50	Data Tone 39	3	0.0
2	1125.00	Data Tone 9	4	0.0
2	1800.00	Data Tone 21	4	90.0
2	2475	Data Tone 33	4	0.0
3	393.75	Doppler Tone	2	0.0
3	675.00	Data Tone 1	1	0.0
3	731.25	Data Tone 2	1	5.6
3	787.50	Data Tone3	1	19.7
3	843.75	Data Tone 4	1	42.2
3	900.00	Data Tone 5	1	73.1
3	956.25	Data Tone 6	1	115.3
3	1012.50	Data Tone 7	1	165.9
3	1068.75	Data Tone 8	1	225.0
3	1125.00	Data Tone 9	1	295.3
3	1181.25	Data Tone 10	1	14.1
3	1237.5	Data Tone 11	1	101.3
3	1293.75	Data Tone 12	1	199.7
3	1350.00	Data Tone 13	1	303.8
3	1406.25	Data Tone 14	1	59.1
3	1462.50	Data Tone 15	1	185.6
3	1518.75	Data Tone 16	1	317.8
3	1575.00	Data Tone 17	1	101.3
3	1631.25	Data Tone 18	1	253.1
3	1687.50	Data Tone 19	1	56.3
3	1743.75	Data Tone 20	1	225.0
3	1800.00	Data Tone 21	1	45.0
3	1856.25	Data Tone 22	1	236.3
3	1912.50	Data Tone 23	1	73.1
3	1968.75	Data Tone 24	1	281.3
3	2025.00	Data Tone 25	1	137.8
3	2081.25	Data Tone 26	1	5.6
3	2137.50	Data Tone 27	1	239.1
3	2193.75	Data Tone 28	1	123.8
3	2250.00	Data Tone 29	1	19.7
3	2306.25	Data Tone 30	1	281.3
3	2362.50	Data Tone 31	1	194.1
3	2418.75	Data Tone 32	1	115.3

Preamble Part	Tone Frequency (Hz)	Tone Function	Amplitude	Initial Phase (Deg)
3	2475.00	Data Tone 33	1	45.0
3	2531.25	Data Tone 34	1	345.9
3	2587.50	Data Tone 35	1	295.3
3	2643.75	Data Tone 36	1	253.1
3	2700.00	Data Tone 37	1	222.2
3	2756.25	Data Tone 38	1	199.7
3	2812.50	Data Tone 39	1	185.6

Block synchronization, which controls the framing process, enables the location of the data block (also referred to as superblocks) boundaries. This synchronization must occur before proper deinterleaving and decoding can commence. Framing is established by periodically inserting a known pseudorandom sequence into the encoded data bit stream. The pseudorandom sequence is generated by the primitive polynomial $f(x) = x^9 + x^7 + x^6 + x^4 + 1$. The feedback register to generate the sequence is illustrated in Figure 4. The block framing sequence is inserted every time a specific number of superblocks has been transmitted; the actual number can be found in [1]. The length of the framing sequence is predefined and varies with the data rate and the interleaving degree, also specified in [1]. The insertion interval (the number of bits between framing sequence bits), the framing sequence length and the interleaving degree depend on the transmission mode: synchronous or asynchronous. In the asynchronous case, these parameters also depend on the number of bits used to form the character set. To illustrate this point, Table 2 provides the various combinations of insertion interval, sequence length and interleaving degree for the 10-bit asynchronous character set used by the radio in this study. Other combinations can be found in [1]. It should be noted that the final sequence of bits of the framing sequence is always 1111111110.

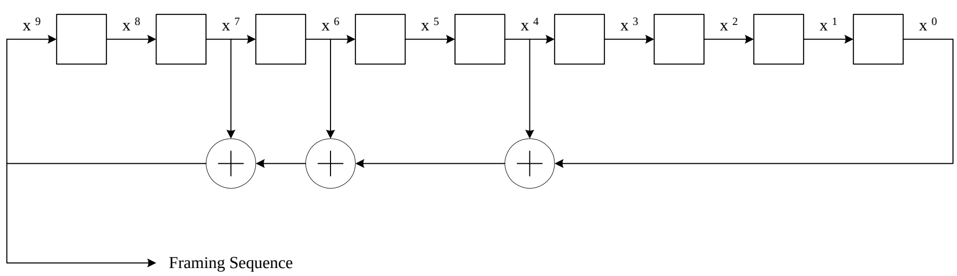


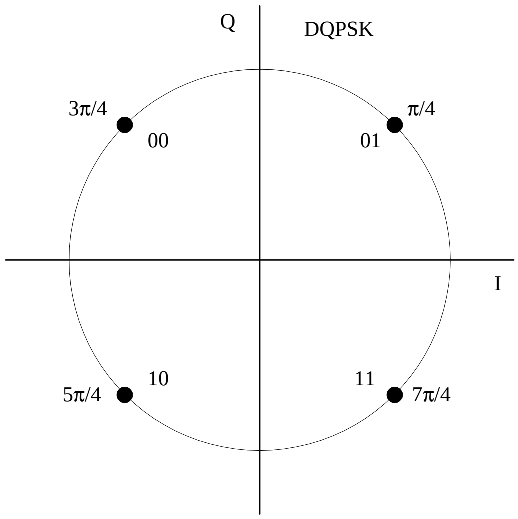
Figure 4: *Shift register configuration.*

Following the block synchronization are the data blocks, or superblocks. The bit stream forming the data blocks results from the demodulation of the 39 tone waveform. Tone frequencies and normalized amplitudes are as indicated in Table 1. All the data tones, in theory, maintain constant amplitude. The amplitude of the Doppler tone is 6 dB higher than the amplitude of the other tones. The 39 tones are simultaneously keyed to produce a 22.5 ms long signal element for each tone. At the onset of each signal element, every

Table 2: *Framing Sequence Insertion Intervals and Lengths*

Data Rate (bps)	Interleaving Degree	Insertion Interval (Superblocks)	Insertion Interval (bits)	Sequence Length (bits)
75	1	585	16380	260
75	4	242	27104	416
75	12	75	25200	400
75	35	18	17640	280
150	1	585	16380	260
150	9	110	27720	440
150	25	36	25200	400
150	75	9	18900	300
300	1	585	16380	260
300	17	49	23324	356
300	47	22	27720	440
300	153	5	21420	340
600	1	585	16380	260
600	33	30	27720	440
600	99	10	27720	440
600	315	2	17640	280
1200	1	585	16380	260
1200	63	15	26460	420
1200	195	6	32760	520
1200	585	1	16380	260
2400	1	144	13440	256
2400	36	7	23520	448
2400	72	3	20160	384
2400	288	1	26880	512

data tone experiences a phase advance relative to its phase at the start of the previous signal element. Four values are allowed for a phase change: 45, 135, 225, or 315 degrees. For each signal element the 39 phase changes are mapped into a 2-bit symbol (also called a di-bit). This results in a stream of 78 bits. The mapping, illustrated in Figure 5, is as follows: $45° \rightarrow 10$, $135° \rightarrow 00$, $225° \rightarrow 01$, and $315° \rightarrow 11$.



Example:
A bit sequence of **01 01 01 01** on a particular tone maps into the symbol sequence:

exp[j(ref + π /4)]
exp[j(ref + π /2)]
exp[j(ref + 3π/4)]
exp[j(ref + 2π)]

where "**ref**" is the initial phase of the tone prior to the transmission of the bit sequence.

Figure 5: *Signal constellation for the DQPSK modulation on each tone.*

There is certain amount of redundancy in the data transmitted within one signal element, which depends on the bit rate. For bit rates less than 2400 bps, information carried on tones 1 through 7 (bits 1 to 14) is also carried on tones 33 through 39 (bits 65 to 78). In addition, the first 64 bits of the 78 bit block are partitioned into a number of data words to be transmitted during each 22.5 ms signal element. The number of data words depends on the bit rate. For all the applicable bit rates, the bit stream for one signal element is organized as follows:

(a) Maximum redundancy exists at 75 bps. Sixteen 4-bit long data words of information are carried in bits 1 to 64, with bits 65 to 78 carrying the information already in bits 1 to 14.

(b) Full redundancy exists at 150 bps. Eight 8-bit long data words of the information are carried in bits 1 to 64, with bits 65 to 78 carrying the information already in bits 1 to 14.

(c) Full redundancy exists at 300 bps. Four 16-bit long data words of information are carried in bits 1 to 64, with bits 65 to 78 carrying the information already in bits 1 to 14.

(d) Full redundancy at 600 bps. Two 32-bit long data words of information are carried in bits 1 to 64, with bits 65 to 78 carrying the information already in bits 1 to 14.

(e) Partial redundancy exists at 1200 bps. One 64-bit data word of information is carried in bits 1 to 64, with bits 65 to 78 carrying the information already in bits 1 to 14.

(f) No redundancy at 2400 bps. One 78-bit long data word carries the information.

Only the first data word in the 78-bit stream contains current information (the amount of current information depends on the bit rate; for example, 150 bps implies data word sizes of 8 bits). The other subsequent data words of the 78-bit stream are copies of data words already transmitted during previous signal elements. Tables 3 to 8 illustrate this time/frequency diversity structure for each of the available bit rates.

Once the tones have been translated into a bit stream, the data is deinterleaved and then decoded. From the ESM perspective, we are only interested in the deinterleaving process. However, to better understand this process, the interleaving at the transmitter will be briefly described.

There are four possible interleaving degrees for bit rates under 2400 bps. At 2400 bps, there are eight interleaving degrees. The various interleaving degrees are listed in Table 2. The interleaver takes the input of an FEC encoded data block and creates a single bit stream by sending the data bits row by row. Figures 6 and 7 illustrate the interleaving process.

Deinterleaving is the reverse operation. The actual deinterleaving process is carried out as follows:

(a) Isolate all the data bits delimited by two synchronization blocks.

(b) Group the data bits into superblocks, where each superblock contains xn bits, where x is the interleaving degree and n is the length of a codeword. The code words are 28 bits long (12 data bits and 16 parity bits) for bit rates below 2400 bps, and 56 bits long (40 data bits and 16 parity bits) for 2400 bps.

(c) Split the superblocks into codewords as illustrated in Figures 6 and 7.

(d) Input the codewords into the decoder. The encoding/decoding process is based on 4-bit long symbols. From Figures 6 and 7 it can be seen that some interleaving occurs in the encoder for the 28-bit long codewords. These codewords are created in pairs. The 4-bit symbols are created two at a time.

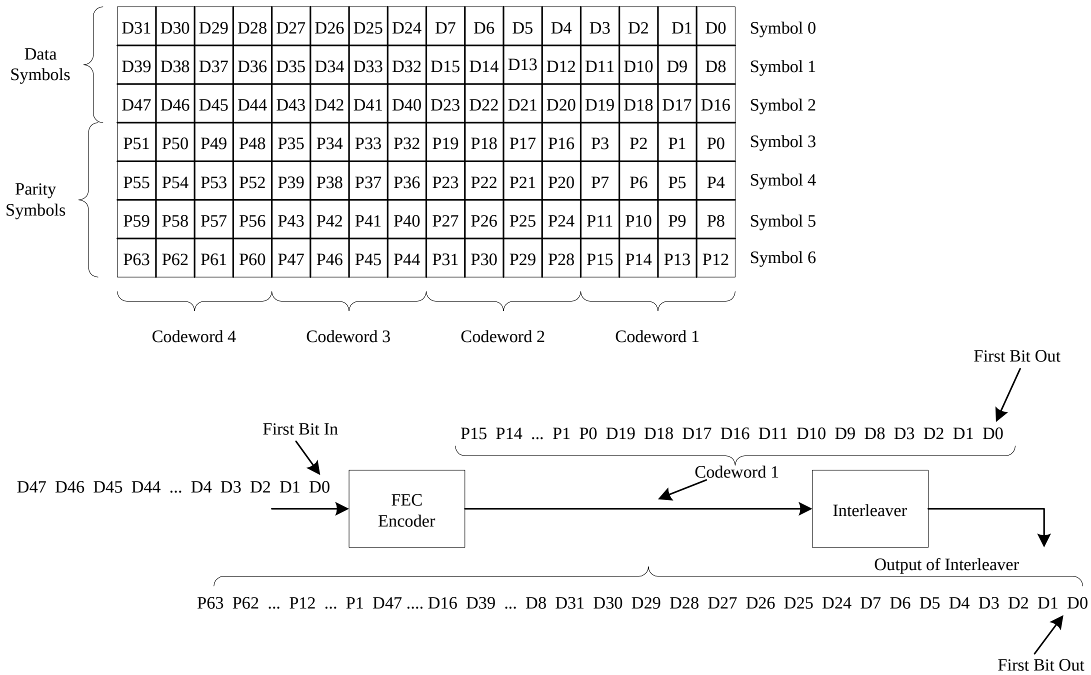


Figure 6: *Interleaver for an even number of code words.*

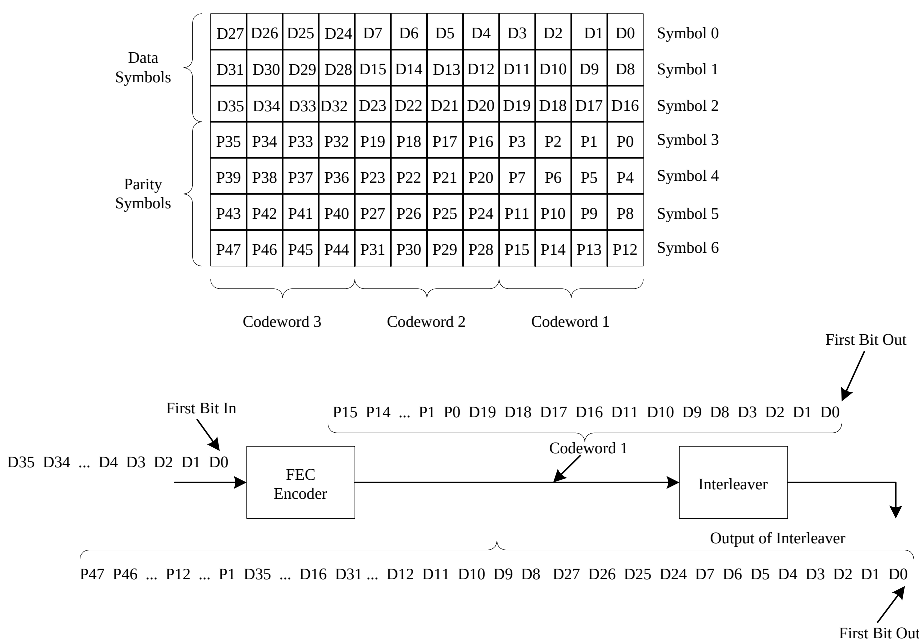


Figure 7: *Interleaver for an odd number of code words.*

The first 4 bits into the encoder form the first symbol in the first codeword. The next 4 bits form the first symbol in the second codeword, etc. After a pair of codewords is created in the encoder, they are stacked together with the other codewords for that sequence and then put into the interleaver.

Table 3: *Time/Frequency Diversity at 75 bps*

Tone Number	Bit Number		Data Word
1 2	1 3	2 4	Actual Word
3 4	1 3	2 4	Copy of word transmitted 1 signal element in the past
5 6	1 3	2 4	Copy of word transmitted 2 signal elements in the past
7 8	1 3	2 4	Copy of word transmitted 3 signal elements in the past
9 10	1 3	2 4	Copy of word transmitted 4 signal elements in the past
11 12	1 3	2 4	Copy of word transmitted 5 signal elements in the past
13 14	1 3	2 4	Copy of word transmitted 6 signal elements in the past
15 16	1 3	2 4	Copy of word transmitted 7 signal elements in the past
17 18	1 3	2 4	Copy of word transmitted 8 signal elements in the past
19 20	1 3	2 4	Copy of word transmitted 9 signal elements in the past
21 22	1 3	2 4	Copy of word transmitted 10 signal elements in the past
23 24	1 3	2 4	Copy of word transmitted 11 signal elements in the past
25 26	1 3	2 4	Copy of word transmitted 12 signal elements in the past
27 28	1 3	2 4	Copy of word transmitted 13 signal elements in the past
29 30	1 3	2 4	Copy of word transmitted 14 signal elements in the past
31 32	1 3	2 4	Copy of word transmitted 15 signal elements in the past
33 34	1 3	2 4	Copy of actual word
35 36	1 3	2 4	Copy of word transmitted 1 signal element in the past
37 38	1 3	2 4	Copy of word transmitted 2 signal elements in the past
39	1	2	Partial copy of word 3 elements in the past

Table 4: *Time/Frequency Diversity at 150 bps*

Tone Number	Bit Number		Data Word
1	1	2	Actual Word
2	3	4	
3	5	6	
4	7	8	
5	1	2	Copy of word transmitted 2 signal elements in the past
6	3	4	
7	5	6	
8	7	8	
9	1	2	Copy of word transmitted 4 signal elements in the past
10	3	4	
11	5	6	
12	7	8	
13	1	2	Copy of word transmitted 6 signal elements in the past
14	3	4	
15	5	6	
16	7	8	
17	1	2	Copy of word transmitted 8 signal elements in the past
18	3	4	
19	5	6	
20	7	8	
21	1	2	Copy of word transmitted 10 signal elements in the past
22	3	4	
23	5	6	
24	7	8	
25	1	2	Copy of word transmitted 12 signal elements in the past
26	3	4	
27	5	6	
28	7	8	
29	1	2	Copy of word transmitted 14 signal elements in the past
30	3	4	
31	5	6	
32	7	8	
33	1	2	Copy of actual word
34	3	4	
35	5	6	
36	7	8	
37	1	2	Partial copy of word transmitted 2 elements in the past
38	3	4	
39	5	6	

Table 5: *Time/Frequency Diversity at 300 bps*

Tone Number	Bit Number		Data Word
1	1	2	Actual Word
2	3	4	
3	5	6	
4	7	8	
5	9	10	
6	11	12	
7	13	14	
8	15	16	
9	1	2	Copy of word transmitted 4 signal elements in the past
10	3	4	
11	5	6	
12	7	8	
13	9	10	
14	11	12	
15	13	14	
16	15	16	
17	1	2	Copy of word transmitted 8 signal elements in the past
18	3	4	
19	5	6	
20	7	8	
21	9	10	
22	11	12	
23	13	14	
24	15	16	
25	1	2	Copy of word transmitted 12 signal elements in the past
26	3	4	
27	5	6	
28	7	8	
29	9	10	
30	11	12	
31	13	14	
32	15	16	
33	1	2	Partial copy of actual word
34	3	4	
35	5	6	
36	7	8	
37	9	10	
38	11	12	
39	13	14	

Table 6: *Time/Frequency Diversity at 600 bps*

Tone Number	Bit Number		Data Word
1	1	2	Actual Word
2	3	4	
3	5	6	
4	7	8	
5	9	10	
6	11	12	
7	13	14	
8	15	16	
9	17	18	
10	19	20	
11	21	22	
12	23	24	
13	25	26	
14	27	28	
15	29	30	
16	31	32	
17	1	2	Copy of word transmitted 8 signal elements in the past
18	3	4	
19	5	6	
20	7	8	
21	9	10	
22	11	12	
23	13	14	
24	15	16	
25	17	18	
26	19	20	
27	21	22	
28	23	24	
29	25	26	
30	27	28	
31	29	30	
32	31	32	
33	1	2	Partial copy of actual word
34	3	4	
35	5	6	
36	7	8	
37	9	10	
38	11	12	
39	13	14	

Table 7: *Time/Frequency Diversity at 1200 bps*

Tone Number	Bit Number		Data Word
1	1	2	Actual Word
2	3	4	
3	5	6	
4	7	8	
5	9	10	
6	11	12	
7	13	14	
8	15	16	
9	17	18	
10	19	20	
11	21	22	
12	23	24	
13	25	26	
14	27	28	
15	29	30	
16	31	32	
17	33	34	
18	35	36	
19	37	38	
20	39	40	
21	41	42	
22	43	44	
23	45	46	
24	47	48	
25	49	50	
26	51	52	
27	53	54	
28	55	56	
29	57	58	
30	59	60	
31	61	62	
32	63	64	
33	1	2	Partial copy of actual word
34	3	4	
35	5	6	
36	7	8	
37	9	10	
38	11	12	
39	13	14	

Table 8: *Time/Frequency Diversity at 2400 bps*

Tone Number	Bit Number		Data Word
1	1	2	Actual Word
2	3	4	
3	5	6	
4	7	8	
5	9	10	
6	11	12	
7	13	14	
8	15	16	
9	17	18	
10	19	20	
11	21	22	
12	23	24	
13	25	26	
14	27	28	
15	29	30	
16	31	32	
17	33	34	
18	35	36	
19	37	38	
20	39	40	
21	41	42	
22	43	44	
23	45	46	
24	47	48	
25	49	50	
26	51	52	
27	53	54	
28	55	56	
29	57	58	
30	59	60	
31	61	62	
32	63	64	
33	65	66	
34	67	68	
35	69	70	
36	71	72	
37	73	74	
38	75	76	
39	77	78	

2.3 Encoding and Decoding

2.3.1 Reed-Solomon Codes

Coding consists of adding redundant bits to the data to be transmitted prior to interleaving and modulation in order to correct the errors introduced during transmission. For the 39 tone signal, the added bits (referred to here as parity bits) are computed using a shortened Reed-Solomon(5,11) block code ($RS(5, 11)$), whose generator polynomial is

$$g(x) = x^4 + a^{13}x^3 + a^6x^2 + a^3x + a^{10} \tag{1}$$

The coefficients a^i are elements of the binary Galois field of order 2^4 (i.e., $GF(2^4)$). They are formed as the field of polynomials over $GF(2^4)$ $modulo$ $x^4 + x + 1$ where $x^4 + x + 1$ is a primitive polynomial. Each 4-bit long symbol mentioned earlier is mapped to an element of $GF(2^4)$.

RS codes are block-based error correcting codes. An RS code is specified as $RS(n, k)$ with s-bit symbols ($s = 4$ for the 39 tone signal). The term “symbol”used here is different from the 2-bit symbol related to mapping data bits to a particular phase. This means that the encoder takes k symbols of s-bits each and adds parity symbols to create an n-symbol long codeword. There are $n - k$ parity symbols of s-bits each. An RS decoder can correct up to t symbols of s-bits each that contain errors in a codeword, where $2t = n - k$. The 39 tone signal performs encoding using $RS(7, 3)$ for bit rates less than 2400 bps, and $RS(14, 10)$ at 2400 bps. Thus, the RS code can correct up to 2 symbol errors. This means that up to 8 bit errors can occur per codeword (either 28 or 56 bits long). More information on RS codes can be found in [2] and [3].

The next several paragraphs provide a brief summary of the Galois theory relevant to the calculations for the 39 tone signal.

2.3.2 Galois Fields

A finite field $GF(q)$ has at least one primitive element a whose order is $(q - 1)$. The nonzero elements of the Galois field are all powers of the primitive element a. The elements of a Galois field are closed under $modulo$ addition and multiplication. This means that any two elements of the Galois field can be multiplied or added to obtain another element of the field.

In the case of the 39 tone signal, the Galois field is $GF(2^4)$. There are the 16 nonzero elements in the Galois field, i.e.,

$$GF(2^4) = \{0, a^0, a^1, a^2, a^3, a^4, a^5, a^6, a^7, a^8, a^9, a^{10}, a^{11}, a^{12}, a^{13}, a^{14}, a^{15}\}$$

Table 9: *Elements of $GF(2^4)$*

Element of GF	a^i $modulo$ (a^4+a+1)	$GF(2^4)$	Binary Representation $(a^3,\ a^2,\ a^1,\ a^0)$
0	0	0	0 0 0 0
a^0	1	1	0 0 0 1
a^1	a	a	0 0 1 0
a^2	a^2	a^2	0 1 0 0
a^3	a^3	a^3	1 0 0 0
a^4	a^4	$a+1$	0 0 1 1
a^5	a^2+a	a^2+a	0 1 1 0
a^6	a^3+a^2	a^3+a^2	1 1 0 0
a^7	a^4+a^3	a^3+a+1	1 0 1 1
a^8	a^4+a^2+1	a^2+1	0 1 0 1
a^9	a^3+a	a^3+a	1 0 1 0
a^{10}	a^4+a^2	a^2+a+1	0 1 1 1
a^{11}	a^3+a^2+a	a^3+a^2+a	1 1 1 0
a^{12}	$a^4+a^3+a^2$	a^3+a^2+a+1	1 1 1 1
a^{13}	$a^4+a^3+a^2+a$	a^3+a^2+1	1 1 0 1
a^{14}	a^4+a^3+a	a^3+1	1 0 0 1
a^{15}	a^4+a	1	0 0 0 1

$GF(2^4)$ is a binary field, so the coefficients of the elements are 0 and 1. Since all arithmetic on the coefficients is $modulo$ 2, negative signs can be ignored. The elements of the Galois field are defined by taking three initial terms, 0, 1 and a, and a primitive polynomial of order 4, (i.e., $q(a) = a^4 + a + 1$). Starting with 0, 1 and a, the elements of $GF(2^4)$ can be found by taking the last element of GF, multiplying it by the primitive element a, and dividing it into $q(a)$. The process is repeated until the result of $a^i\ mod(q(a)) = 1$. Table 9 provides the elements of $GF(2^4)$.

All operations on the elements of the Galois field are carried out using $modulo$ arithmetic, in this case $mod(2^4 - 1)$. Multiplication is performed as follows:

$$a^{13}a^9 = a^{(13+9)mod(2^4-1)} = a^{22mod(15)} = a^{22-15} = a^7$$

Division is performed by multiplying the numerator by the inverse of the denominator using the $modulo$ operation:

$$a^2/a^4 = a^2a^{15-4} = a^{2+11} = a^{13}$$

Addition is more complex. Addition is performed by converting the element to its base representation and adding common terms using $modulo$ 2 addition, and then converting the result back to a simple representative element. For example,

$$a^5 + a^{12} \quad = \quad (a^2 + a) + (a^3 + a^2 + a + 1)$$

$$\begin{aligned} &= a^3 + a^2 + a^2 + a + a + 1 \\ &= a^3 + 1 \\ &= a^{14} \end{aligned}$$

Addition of elements of GF can also be understood by immediately taking the equivalent binary representation, performing $modulo$ 2 addition, and then mapping the result to the GF equivalent. For example,

$$\begin{aligned} a^5 + a^{12} &= (0110) + (1111) \\ &= (1001) \\ &= a^{14} \end{aligned}$$

2.3.3 RS Encoding

The generator polynomial of an RS code over a Galois field is derived from the equation

$$g(x) = (x + a)(x + a^2) \cdots (x + a^{2t}). \qquad (2)$$

For $2t = n - k$ (recall that $t = 2$ for the 39 tone signal), we have

$$g(x) = (x + a)(x + a^2)(x + a^3)(x + a^4). \qquad (3)$$

Using $GF(2^4)$ arithmetic,

$$g(x) = x^4 + a^{13}x^3 + a^6x^2 + a^3x + a^{10}. \qquad (4)$$

An $RS(n, k)$ code takes k symbols and adds $(n - k)$ parity symbols to create an $n-$symbol long message. Consider, for example, the $RS(7, 3)$ code that is used in the 39 tone signal for bit rates below 2400 bps. Let $m(x)$ be the message to encode. The message will be

$$t(x) = x^{2t}m(x) + r(x) \qquad (5)$$

where $r(x) = x^{2t}m(x)\ modulo(g(x))$. The term x^{2t} acts as a shift register, placing the original message in the upper bits of the message and adding the parity bits at the end. The result is a coded message that contains the original message. For example, let

$$m(x) = a^1 + a^8x^1 + a^3x^2$$

$$x^{2t}m(x) = a^1x^4 + a^8x^5 + a^3x^6$$

and

$$r(x) = (a^1x^4 + a^8x^5 + a^3x^6)\ modulo\ (x^4 + a^13x^3 + a^6x^2 + a^3x + a^{10})$$

Using Galois arithmetic,

$$r(x) = a^4x^2 + a^2x + a^8$$

Then,

$$t(x) = a^3x^6 + a^8x^5 + a^1x^4 + 0x^3 + a^4x^2 + a^2x + a^8$$

Recall that each symbol a^i represents a 4 bit binary value. Therefore the $(7, 3)$ code in the example above takes $3 \times 4 = 12$ message bits, adds $4 \times 4 = 16$ parity bits, and transmits the resulting 28-bit encoded message. Mapping the GF elements to their binary values (see Table 9) produces the 28-bit long transmitted message

$$1000 \quad 0101 \quad 0010 \quad 0000 \quad 0011 \quad 0100 \quad 0101$$

Similarly, an $RS(14, 10)$ code takes $10 \times 4 = 40$ bits and adds $4 \times 4 = 16$ parity bits, resulting in a 56 bit long message. The 39-tone parallel modem interleaving function is based on 28-bit codewords. Therefore, a codeword using the $RS(7, 3)$ code represents one coded message, while the $RS(14, 10)$ code requires two codewords to represent a single coded message.

2.3.4 Decoding RS codes

If $v(x)$ is the received message and $t(x)$ the transmitted message, then

$$e(x) = v(x) - t(x) \tag{6}$$

where $e(x)$ represents the error polynomial, a function of the bit errors introduced by the transmission medium. Decoding Reed Solomon codes consists of determining this error polynomial, and then correcting the received word to recover the originally transmitted codeword.

The first step in determining the error polynomial is to determine the syndromes, the bit patterns used for error-detection and correction [9] , of the received message. The syndromes are obtained by reapplying the encoding rules to the received word. To do this, evaluate $v(x)$ at the prescribed roots of the generator polynomial $\{a, a^2, a^3, \cdots, a^{2t}\}$:

$$S_j = v(a^j), \qquad j = 1, \cdots, 2t. \tag{7}$$

Since $t = 2$ for the $RS(7, 3)$ and $RS(14, 10)$ used in the 39-tone parallel mode, there will be four syndromes, i.e., $S_1,\ S_2,\ S_3$ and S_4.

The second step consists of determining the error locator polynomial $L(x)$ from the syndrome values where

$$L(x) = 1 + L_1x^1 + L_2x^2 + \cdots + L_tx^t. \tag{8}$$

Therefore, for $t = 2$,

$$L(x) = 1 + L_1x^1 + L_2x^2. \tag{9}$$

There are two coefficients, L_1 and L_2, which, according to [2] and [3], are related to the syndrome values through the relation

$$\begin{pmatrix} S_1 & S_2 \\ S_2 & S_3 \end{pmatrix} \begin{pmatrix} L_2 \\ L_1 \end{pmatrix} = \begin{pmatrix} -S3 \\ -S4 \end{pmatrix}. \quad (10)$$

From Eq. 10, we obtain

$$L_2 = (S_3^2 - S_2S_4)/(S_2^2 - S_1S_3) \quad (11)$$
$$L_1 = (-S_2^2S_3 + S_1S_2S_4)/(S_2^3 - S_1S_2S_3). \quad (12)$$

The above equations refer to the situation where two errors are present. For this case, the four syndrome values are nonzero. When only one error is present,

$$L_1 = S_2/S_1. \quad (13)$$

When the four syndromes are null, no bit errors are present and the received code word is identical to the one that had been transmitted. The method used to find the error locator polynomial is Berlekamp's iterative algorithm described in [3].

Once the coefficients have been determined, the error locator polynomial must be formed and its roots calculated. The roots of $L(x)$ are the inverses of the error locators $\{X_i\}, i = 1, \cdots, t$. To find the roots of $L(x)$, calculate $L(a^j)$ for $j = 1, \cdots, m-1$, where $m = 2^{(n-k)} - 1$. If $L(a^j) = 0$, then a^j is a root of $L(x)$, and $X_i = 1/a^j$, where $i \in \{1, \cdots, t\}$. For the $RS(7,3)$ and $RS(14,10)$ codes, there are only two error locators, since $t = 2$.

After the error locators have been calculated, the last step of the decoding process, calculating the error polynomial, can be initiated. For that purpose, an infinite degree syndrome polynomial is defined as

$$S(x) = S_1x + S_2x^2 + \cdots + S_{2t}x^{2t} + \cdots \quad (14)$$

The error magnitude polynomial is defined as

$$O(x) = L(x)(1 + S(x)) \;\; modulo \;\; (x^{2t} + 1). \quad (15)$$

The error magnitude polynomial is calculated using the Forney algorithm described in [3]. For the $RS(7,3)$ and $RS(14,10)$ codes, the error magnitude polynomial is

$$O(x) = (1 + L_1x^1 + L_2x^2)(1 + S_1x^1 + S_2x^2 + S_3x^3 + S_4x^4) \;\; modulo \;\; (x^5).$$

Given the error locators X_1 and X_2, the error magnitudes are found to be

$$E_{j1} = O(X_1^{-1})/(1 + X_2X_1^{-1}) \quad (16)$$

and

$$E_{j2} = O(X_2^{-1})/(a + X_1 X_2 X^{-1}). \quad (17)$$

The error polynomial is then

$$e(x) = E_{j1}x^{j1} + E_{j2}x^{j2} \quad (18)$$

where $j1$ and $j2$ are the exponents of the error locators X_1 and X_2. Finally, the corrected message, defined as $t(x) = v(x) + e(x)$, should be equal to the transmitted message, provided no more than t errors were incurred during transmission.

2.4 Extracting the Data Bits

In the asynchronous mode of operation, character sets can range between 7 bits and 12 bits long. Each of the characters starts with a Start Bit (St) and ends with one or two Stop Bits (Sp), depending on the character set being used. In addition, the character may contain a Parity Bit (P). The remaining bits in the character are Data Bits (Da). For example, the 10-bit character set used in the Harris AN/PRC-138 radio would be structured as follows:

St	Da	Da	Da	Da	Da	Da	Da	P	Sp

Prior to American Standard Code for Information Interchange (ASCII) conversion, the data bits are stripped from the 10-bit word, leaving the 7-bit ASCII character. This character is then converted to the appropriate symbol using the ASCII conversion table.

As a final point, it should be noted that in synchronous operation the character sets contain only data bits, i.e., there are no start, stop and parity bits.

3. EXPERIMENTAL SET-UP

This section briefly describes the experimental set-up used to collect data samples of the 39 tone signal from the Harris AN/PRC-138 manpack radio. The radios were directly connected through coaxial cables to the Blackbird system (HP E3238S) ([4, 5]). Data collection was performed in the laboratory in a noise and interference free environment. Care was taken to avoid saturation of the A/D converter (HP 1437A). To prevent saturation, an attenuation of 80 dB was applied to the signal delivered by the Harris radio prior to entering the HP E6500A VXI tuner system of Blackbird. The final sampling rate of the collected signal was 40 kHz, achieved through the use of the digital drop receiver in Blackbird. The data was stored in 32 bit complex format. The complex baseband frequency range was -20 kHz to +20 kHz. Using Matlab, the captured data file, obtained with a sampling rate of 40 kHz, was resampled to provide a new sampling rate of 14.4 kHz (upsampled by a factor of 9, and downsampled by a factor of 25). Justification for this new sampling rate is discussed later. The resulting data prior to demodulation and decoding were in the complex baseband frequency range of -7200 Hz and +7200 Hz.

For the collections carried out for this study, the radios were operating at a frequency of 22 MHz, the upper part of the HF band. As mentioned in [6], the 39 tone signal is supported in the Harris radio in both bit-synchronous and asynchronous modes. Bit-synchronous operation can be done only through the use of what is called the Universal Data Terminal (UDT) [7]. For the study reported on here, only the asynchronous mode was used.

4. DEMODULATION

This section discusses issues related to synchronization, phase demodulation, and the bit mapping process.

4.1 Synchronization

Prior to continuing with any demodulation processing, it is essential that the signal under analysis be synchronized in frequency and in time.

The demodulation process is based on the determination of the phase discontinuity occurring every symbol period between two consecutive signal elements (see Section 2.2). Phase calculation is the core of the present analysis. For each of the signal elements under analysis, 39 phase values are simultaneously determined. Therefore, a direct phase determination from the imaginary and real part of the time domain signal is not feasible. The signal is decomposed into its frequency components prior to performing any phase calculation. Consequently, analysis is performed in the frequency domain. During the analysis, each of the spectral line exactly corresponds to a unique tone among the tones composing the signal, that is, frequency synchronization is achieved. In other words, a spectral line does not overlap two different tones.

Accurate frequency decomposition of the signal into its components is not enough to ensure useful phase demodulation. The determination of each of the phase values used in the simultaneous demodulation process necessitates spectral analysis over many data points of the signal under consideration. To obtain uncorrupted phase values it is essential that the spectral analysis be performed on data sets pertaining to a single signal element, that is, the data must be time synchronized. Data sets which overlap contiguous signal elements will provide phase values unusable for any DQPSK processing.

For the present task, the analysis of the signal starts with the beginning of the transmission. Figure 8 is a time domain representation of the signal. The left side of the plot represents thermal noise occurring before the transmission starts. The right side of the plot represents the beginning of the transmission. Parts I and II of the signal preamble are then available for analysis. It is convenient to use information about the preamble structure and characteristic to achieve frequency and time synchronization, as detailed next.

The frequency spectrum of the beginning of the capture is plotted in Figure 9 and shows four unmodulated tones, the first part of the preamble (see Figure 3). The spectrum was calculated using an 8192 point FFT, and therefore 1.66 Hz bin resolution, with no windowing, over a data segment corresponding to a signal element duration of 22.5 ms. Since a sampling rate of 14.4 kHz yields 324 data samples in 22.5 ms, zero-padding was used to increase the number of samples. Once the starting point of the transmission has been established, 14 signal elements must be skipped to get to the

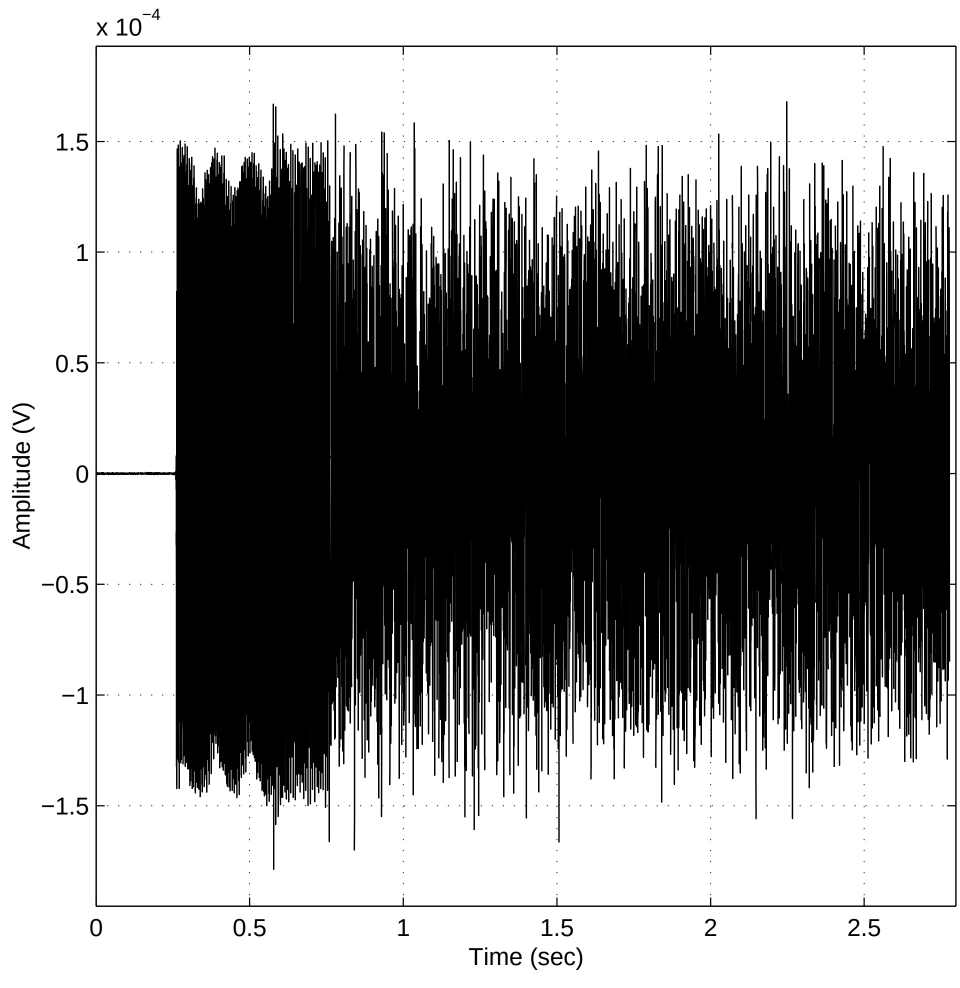


Figure 8: *Portion of the time domain 39 tone signal.*

second part of the preamble, which is characterized by 3 modulated tones. During this second part, the phase of the 3 modulated tones is advanced by π radians from signal element to signal element. This property was used in our analysis to synchronize the data in time as follows.

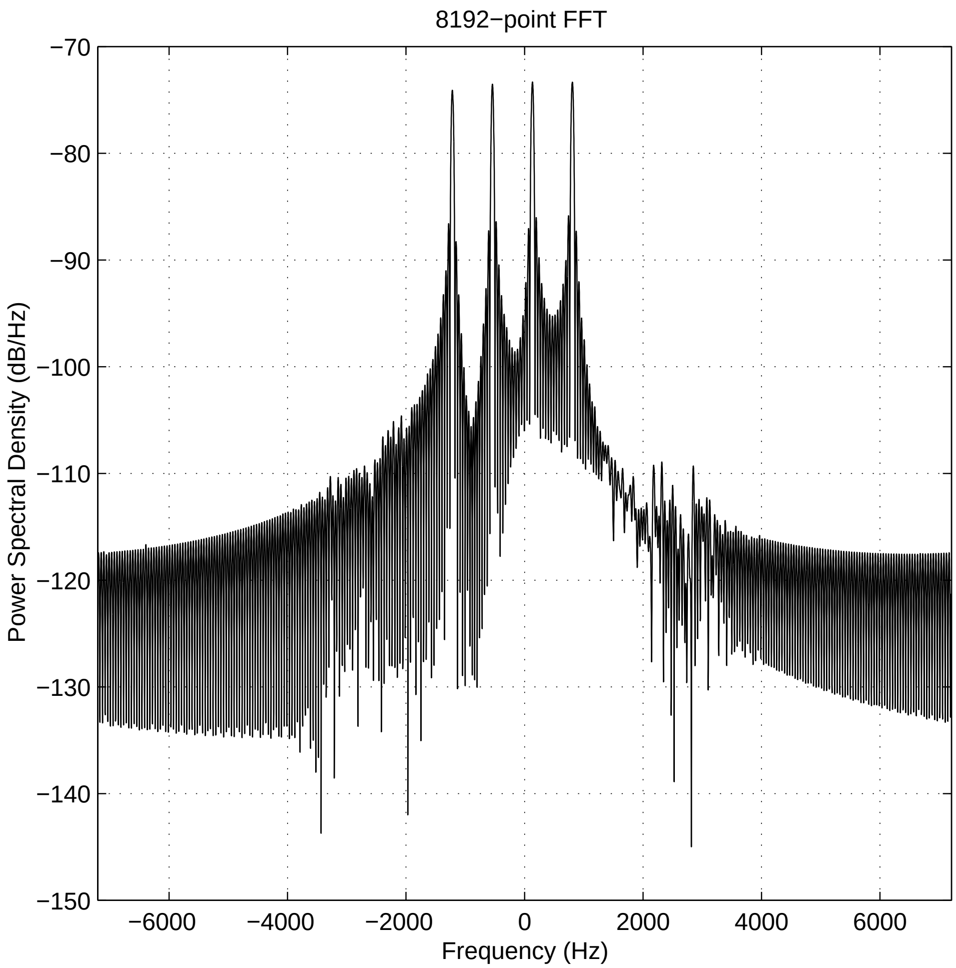


Figure 9: *Power spectral density of part one of the preamble.*

The first step of the process consists of identifying the frequencies of the preamble tones. This uses a fine resolution FFT of 1.66 Hz, as discussed above. An example of the frequency spectra results for the second part of the preamble is shown in Figure 10. To obtain an accuracy better than 1.66 Hz, a spectral line power- weighted average was used around the peak positions. The measured frequencies were then compared to the nominal values of the standard [1]. The difference frequency Δf was used to complex mix the captured data to the nominal frequencies.

The second step of the process consists of performing a series of pairs of 256 point FFT's over two consecutive signal elements. For a signal sampled at 14.4 kHz, a 256

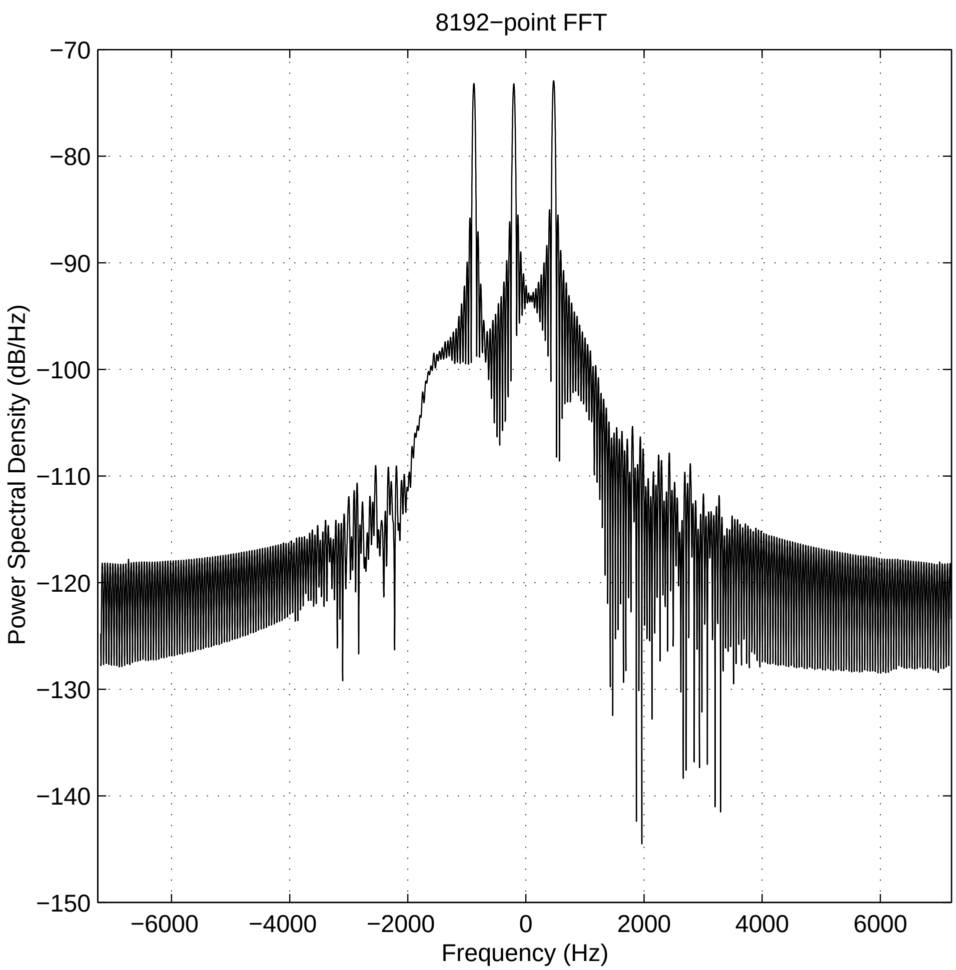


Figure 10: *Power spectral density of part two of the preamble.*

point FFT has a resolution of 56.25 Hz which corresponds to the frequency spacing of the tones indicated in the standard [1]. If the mixing process has been properly done, each tone in the spectrum will have only one spectral line as shown in Figure 11.

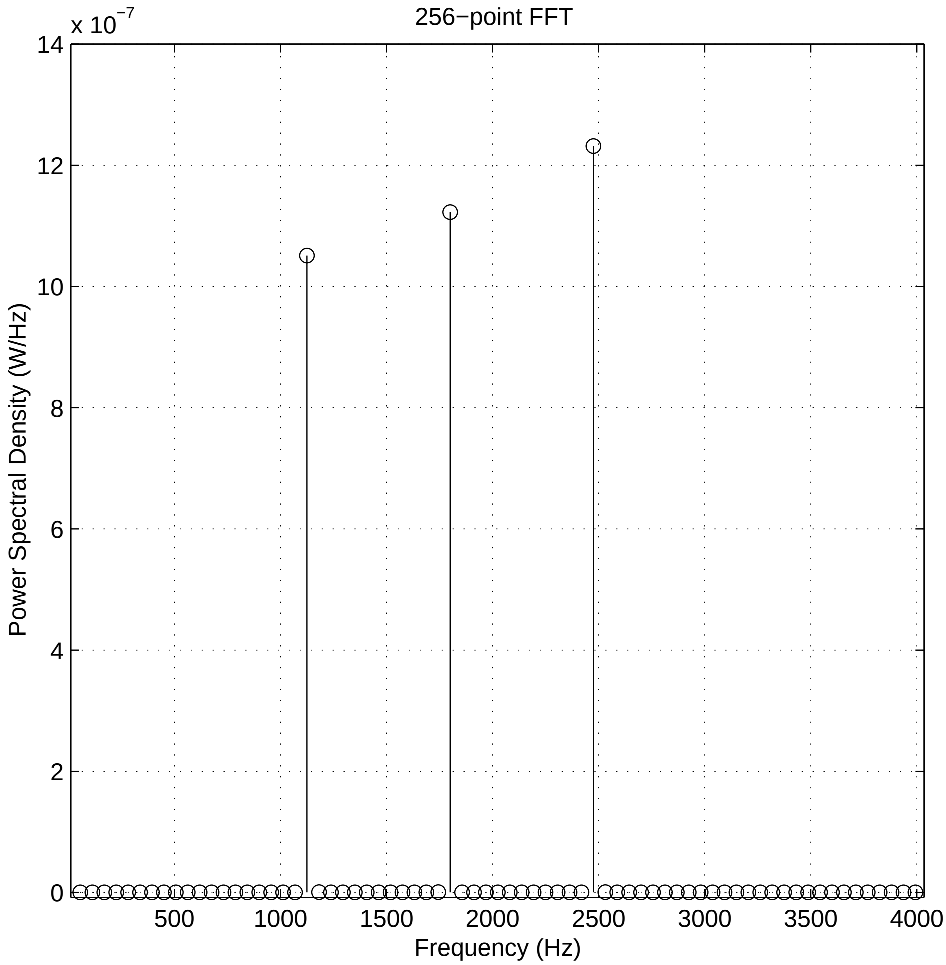


Figure 11: *Power spectral density of part one of the preamble after frequency synchronization.*

In the final step of the process, the real and imaginary parts of the spectral lines are used to calculate a phase value for each 22.5 ms signal element. The resultant phase calculations for the tones of the first signal element are used to predict what the phases of the tones should be in the next, or adjacent, signal element. If properly synchronized in time, the difference between predicted and calculated values should be π. The predicted phase value for the tones of the second signal element is

$$\phi_{n+1} = \phi_n + 2\pi f T, \tag{19}$$

where f is the nominal frequency of one of the tones, and T is the duration of the signal element (22.5 ms).

The above discussion is based on the ideal situation where the first data point of the 256 point FFT block is aligned with the signal element boundary. This will typically not be the case. More likely the initial 324 point segment to be processed will straddle the symbol boundary. The steps to be taken to synchronize in time are illustrated in Figure 12. As mentioned above, for a sampling rate of 14.4 kHz, signal elements are 324 points long; however, only 256 points of the signal element are used for the FFT. This implies that 68 FFTs must be done by sliding the processed segment position across the signal element, one point at a time. For the FFTs associated with blocks within the symbol period, the phase difference between predicted and calculated values should be tightly distributed around π radians. For the others, where the 256 point FFTs straddle the symbol boundary, the phase information should be significantly different from π. This effect is illustrated in Figure 13. From this figure, it can be seen that the data are synchronized in the time range from 0.015 to 0.018 secs. The point in the figure where the oscillation is maximum and seems to change polarity corresponds to the case where one half of the points (i.e., 128) of the FFT block belong to one symbol period, and the other half to the adjacent symbol period. These observations are corroborated by the spectra illustrated in Figures 14, 15 and 16. Figure 14 corresponds to the case when time synchronization has been achieved, and the other two when it has not.

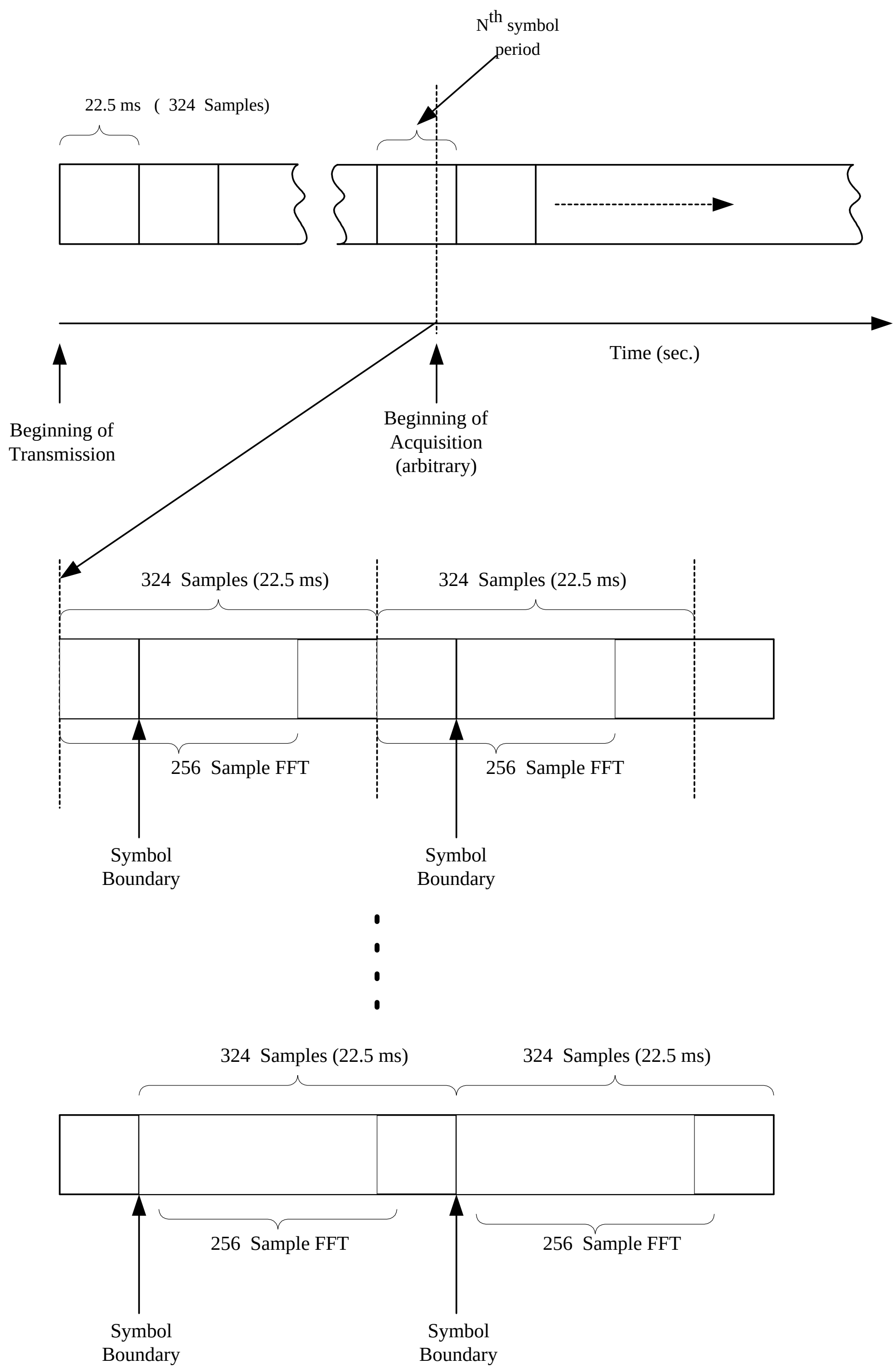


Figure 12: *Time synchronization process.*

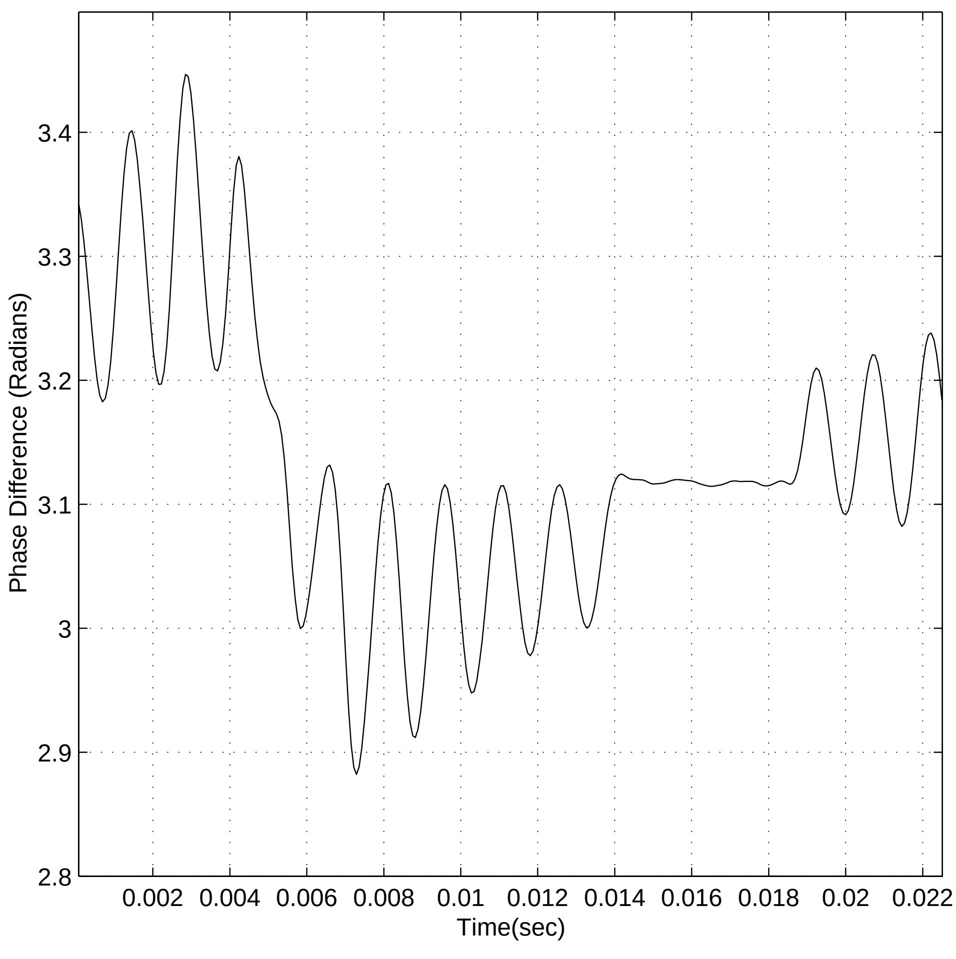


Figure 13: *Phase difference during time synchronization process.*

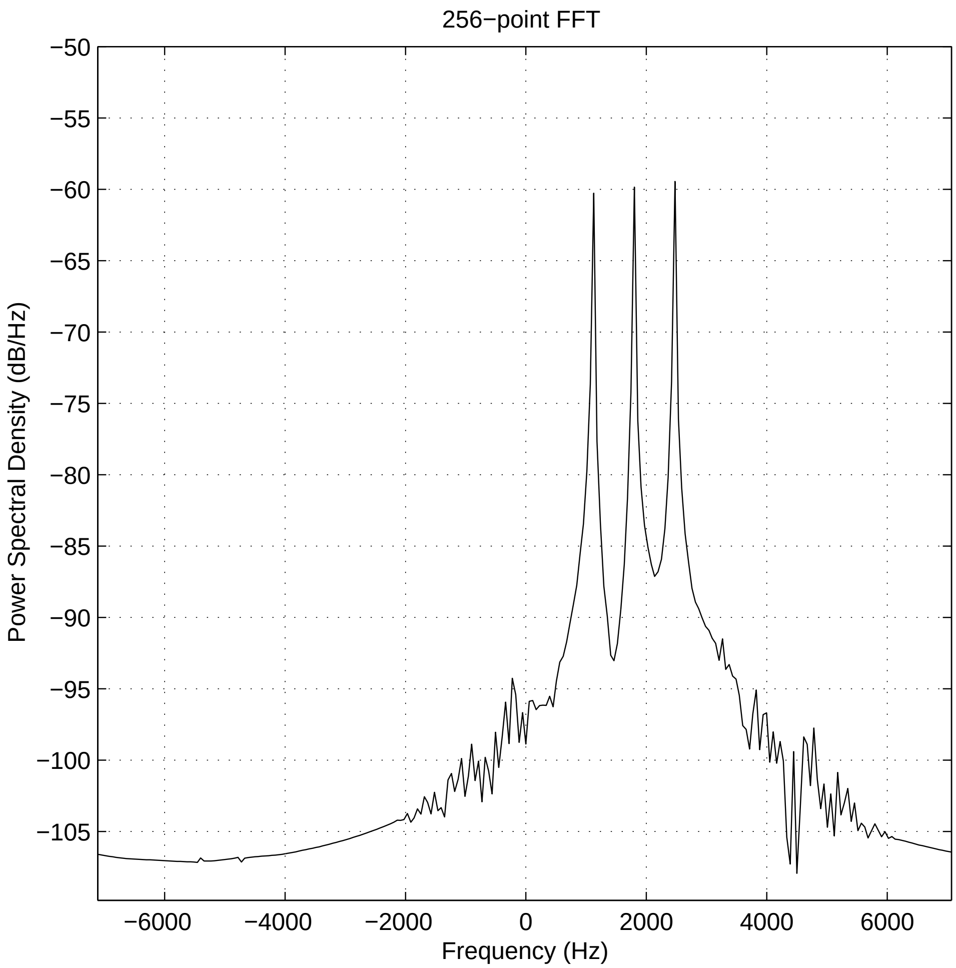


Figure 14: *Spectrum when data are synchronized in time.*

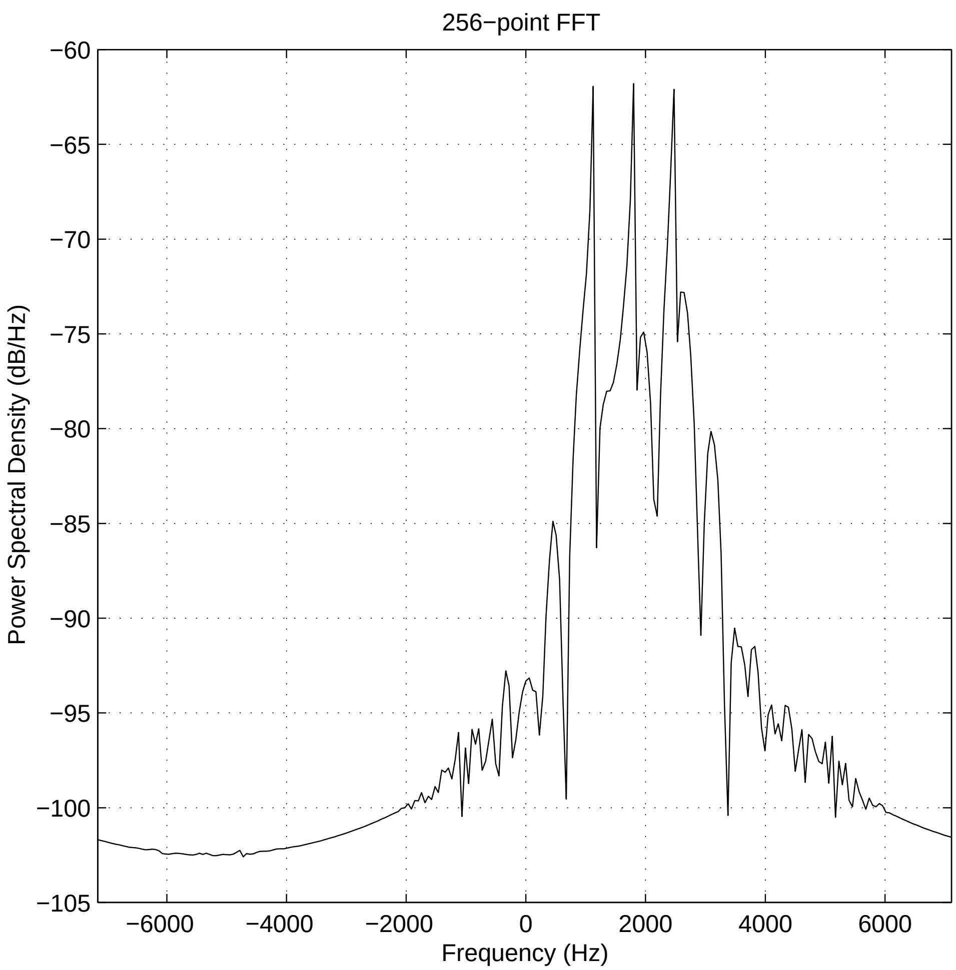


Figure 15: *Spectrum when time synchronization is not perfectly achieved.*

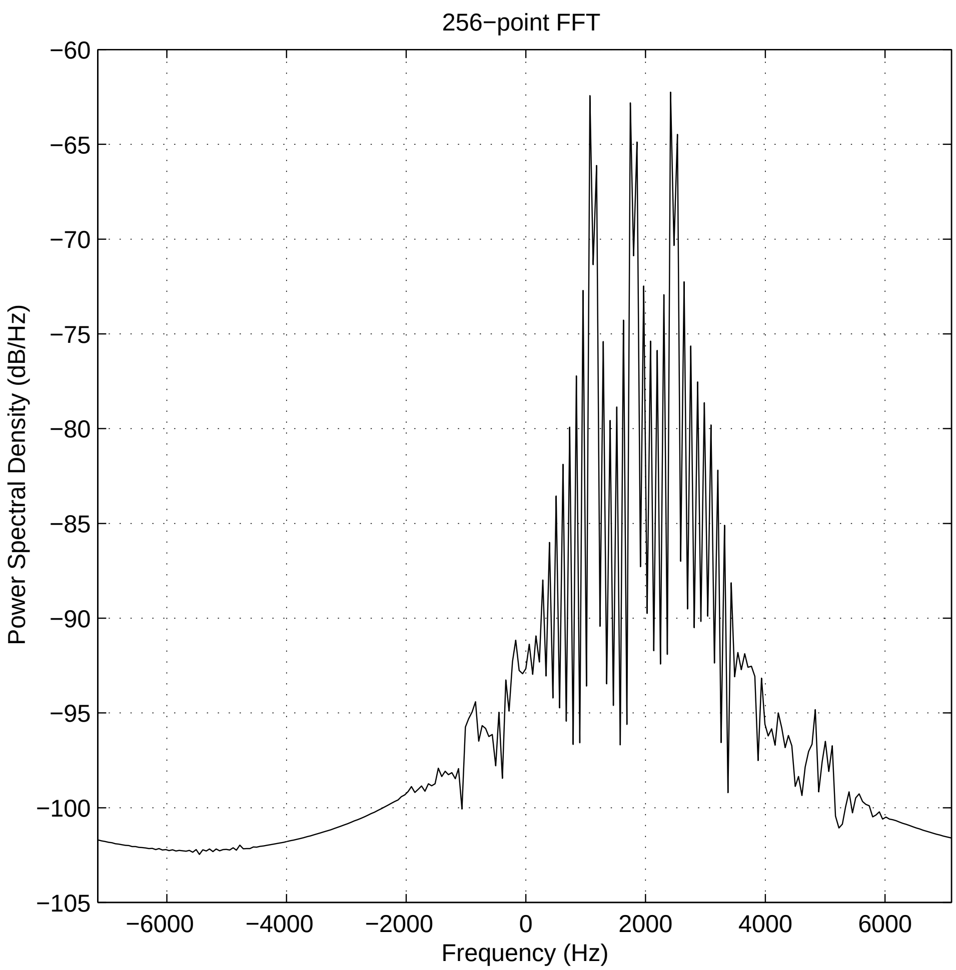


Figure 16: *Spectrum when there is no time synchronization whatsoever.*

4.2 Phase Demodulation

Once synchronization has been achieved, the phase demodulation process can begin. The data signal elements corresponding to the first two parts of the preamble can be skipped, and demodulation can start at the block synchronization part of the frame (see Figure 3).

Before carrying out any processing, the frequencies of the tones must be checked and, if necessary, corrected of any shift that may have occurred during the transmission. The Doppler tone is monitored for this purpose. The frequency synchronization process described above is repeated here, using the 8192 point FFT with zero padding over one signal element. An example of the spectrum is illustrated in Figure 17. An accurate position of the Doppler peak is determined by a power-weighted average and the frequency difference between this value and the nominal value from the standard (393.75 Hz) is calculated. This difference is used in a complex mixing operation to shift the time-synchronized complex data to the nominal value.

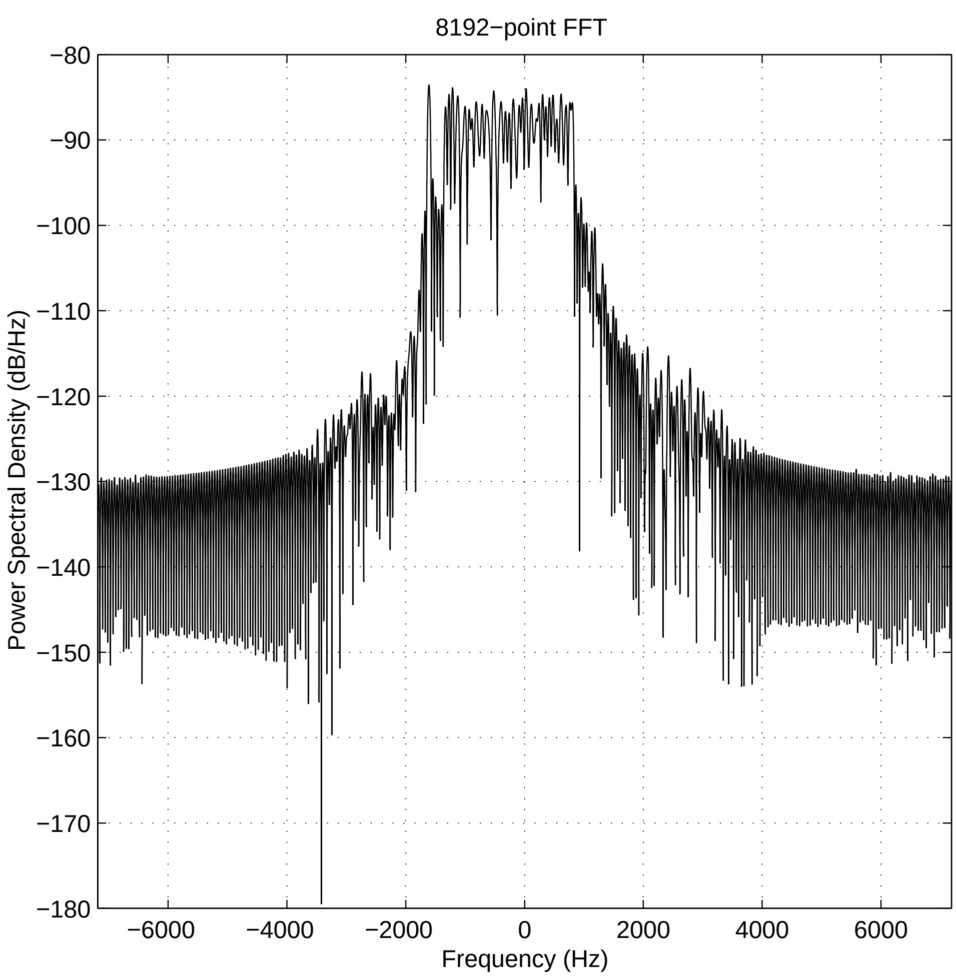


Figure 17: *Spectrum of 39 tone waveform.*

The Harris radios generate their 39 tone signals in the digital domain, prior to D/A conversion, using a 7.2 kHz sampling rate. At the receiving end, they sample the incoming signal at 14.4 kHz. Doubling the sampling rate on reception produces two interleaved blocks of 128 points each at a sampling rate of 7.2 kHz. Performing 128-point FFT's (56.25 Hz resolution) on each of the data blocks provides a snapshot of the phase behavior of the tones during a sampling interval. This additional information is used to refine the DQPSK demodulation process.

After frequency and time synchronization have been achieved, a 128-point FFT of a deinterleaved subblock of the 256 block is performed within a signal element. The FFT results are shown in Figure 18. Each of the 39 tones (not including the Doppler tone) corresponds to only one spectral line, each having an amplitude and phase. The amplitudes of the tones are within the tolerance indicated in the standard. Also, the separation of the spectral lines allows comparison of phases from one symbol period to another. From the measurements that were carried out, it was observed that the signal from the radio oscillated in frequency, with a range of ± 8 Hz over a duration of approximately 1 second. The Doppler tone was monitored, and its frequency used to correct for any frequency and phase offsets from tone to tone. Finally, by calculating two deinterleaved 128-point FFTs per signal element, the phase variation from signal element to signal element for each of the 39 tones could be monitored and used to make refinements. Figure 19 shows a bar chart of the phase differences observed for tone number one over time. There are four distinct average phase levels. These levels, however, do not correspond to the values mentioned in the standard.

Figure 20 shows the phase differences for tone one, over 703 signal elements, presented in the form of a polar plot or constellation diagram. The phases cluster into four areas 90 degrees apart, with each cluster spread over 30 to 40 degrees around a centroid. The constellation for tone number 10 is illustrated in Figure 21. Its centroid is offset from that for tone number one. It was found that the clusters for the various tones were offset from one another in a progressive manner by about -5.5 degrees, suggesting that there was a frequency rotation occurring across the tones. It was hypothesized that this rotation was due to an offset in the spacing of the 39 tones. This offset may have originated from a lack of performance in the transmitter while maintaining the tone frequencies within the values specified in [1]. It may also have been created by the sampling rate of the Blackbird equipment. A combination of these circumstances could also have caused the offset. Unfortunately, time did not permit looking for ways to remedy this offset. However, it was still possible to demodulate the signal and convert the symbols into di-bits, albeit with some effort.

First, the constellations must be rotated into place so that the clusters are centered at 45, 135, 225, and 315 degrees. Rotating the constellation for a particular tone involves an ambiguity of 90, 180, or 270 degrees. This is the case for each tone. To determine the correct rotational value, di-bits were generated for various rotational values. Then the bit pattern in the block synchronization part of the frame and the redundancy characteristics in the message were sought. Two of the 39 correctly rotated

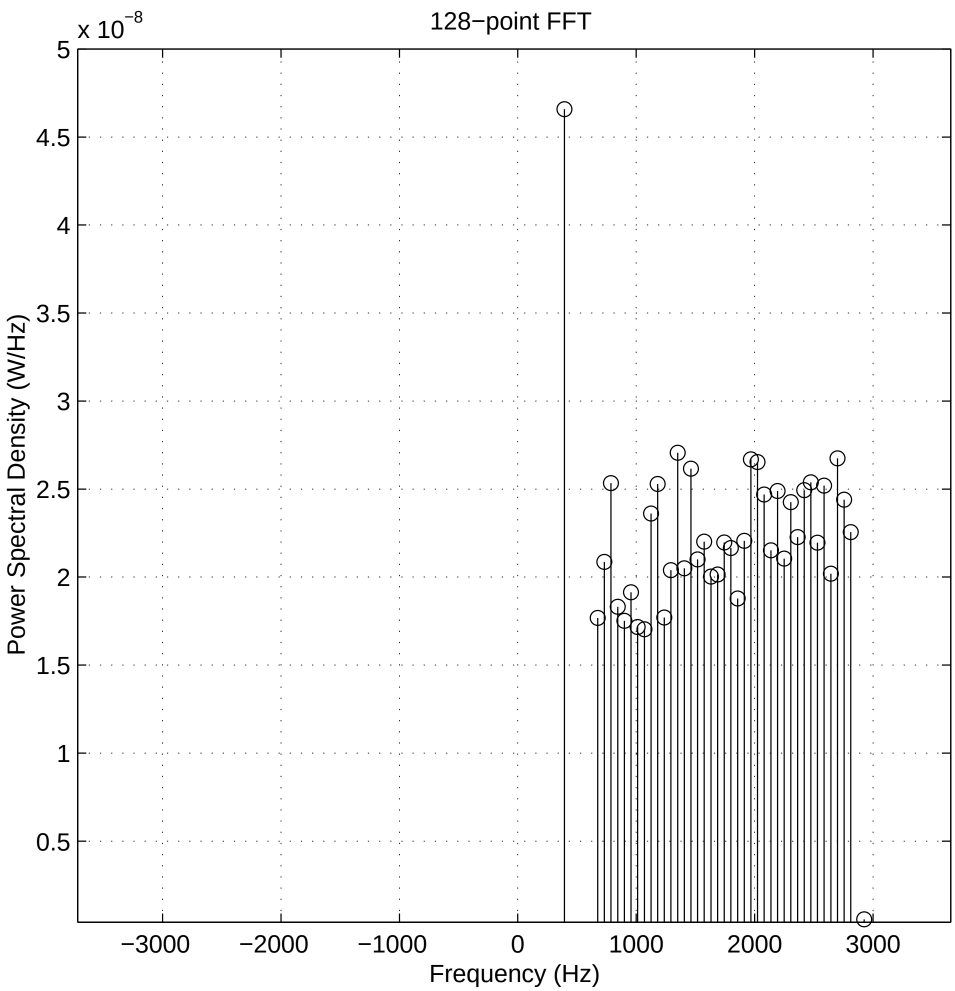


Figure 18: *Spectrum of 39 tone waveform after time and frequency synchronization.*

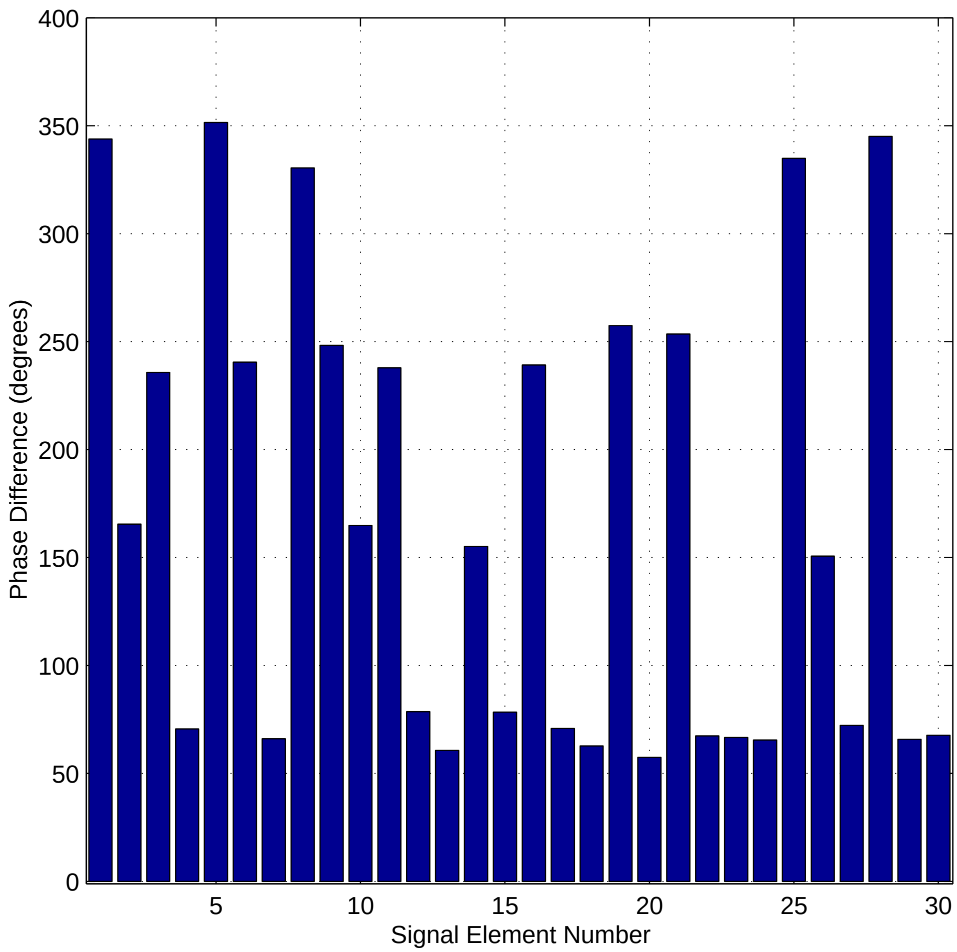


Figure 19: *Successive phase differences from symbol period to symbol period for tone number 1 over a duration of 30 symbol periods.*

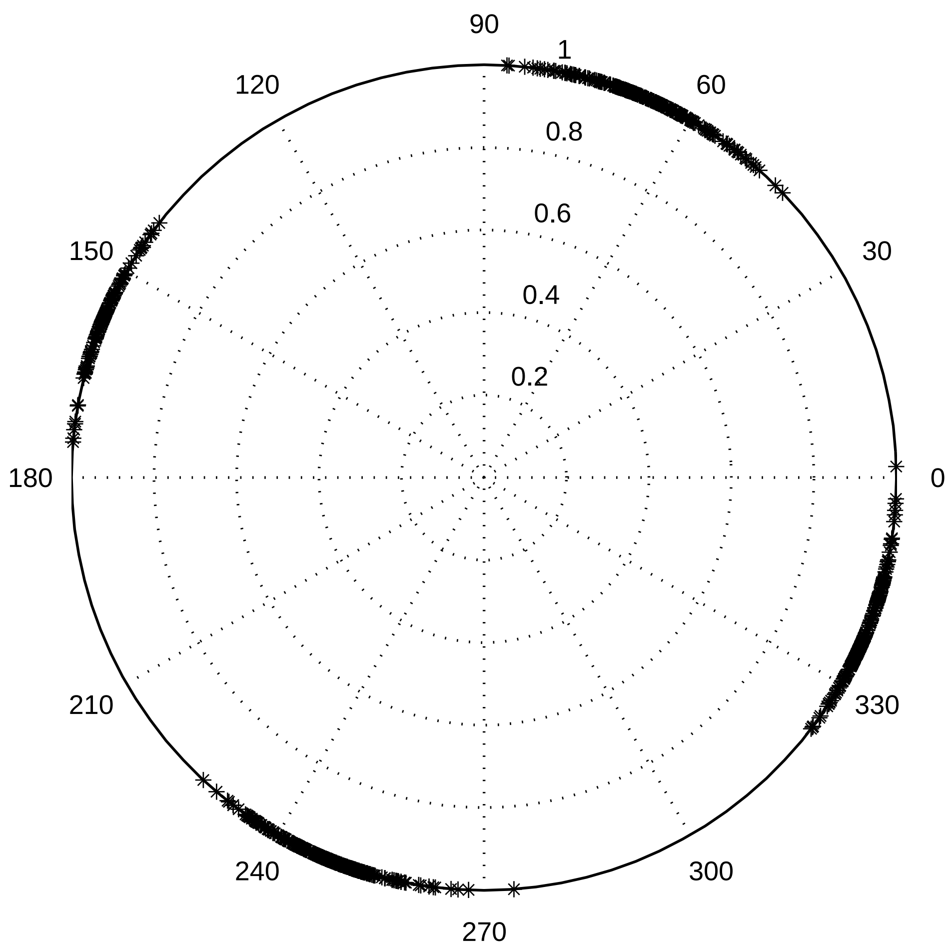


Figure 20: *Signal constellation for tone number one over 703 symbol periods.*

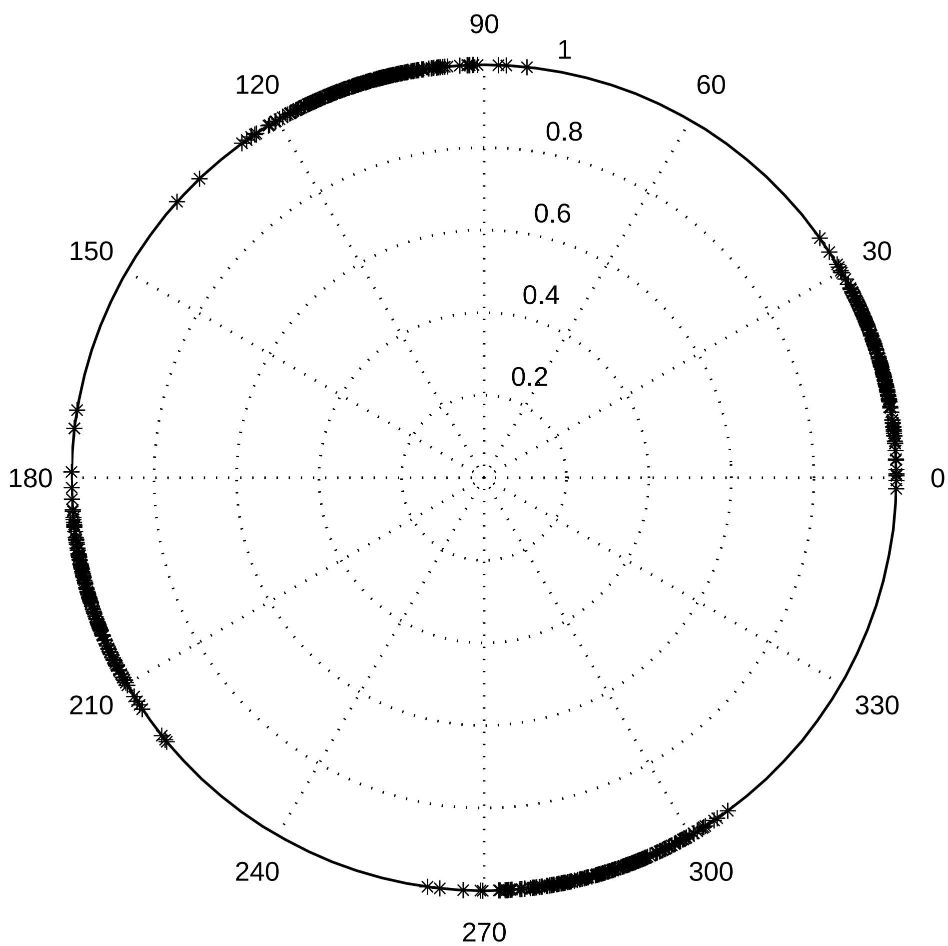


Figure 21: *Signal constellation for tone number 10 over 703 symbol periods.*

constellations are illustrated in Figures 22 and 23.

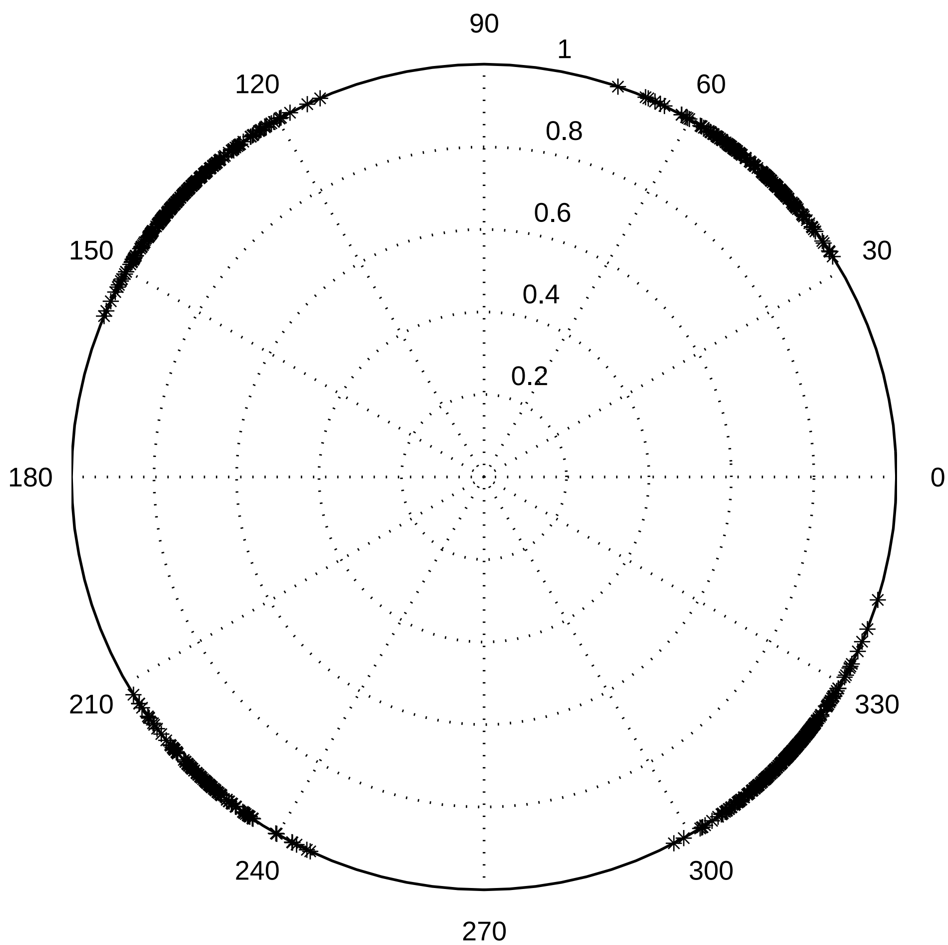


Figure 22: *Signal constellation for tone number one over 703 symbol periods after constellation rotation.*

Once rotated, the phases could be mapped to di-bits according to the values shown in Figure 5. For each signal element, a 78-bit long stream is built. To verify the validity of the bit stream obtained, the redundancy patterns mentioned in Tables 3 to 7 were identified. The first pattern occurs when bits 65 to 78 are redundant with bits 1 to 14. The second redundancy pattern applies to bits of consecutive signal elements for data rates below 1200 bps. For example, for 75 bps, bits 1 to 4 constitute the actual data words, while bits 5 to 8 are redundant with the data word transmitted one signal element in the past. Once the redundancy is successfully identified, the redundant bits are removed from the bit stream in order to obtain the decodable bit stream. A final check consists of identifying the final bits of the framing sequence which, according to [1], is 1111111110 (see Figure 24). The sequence repeats according to the insertion

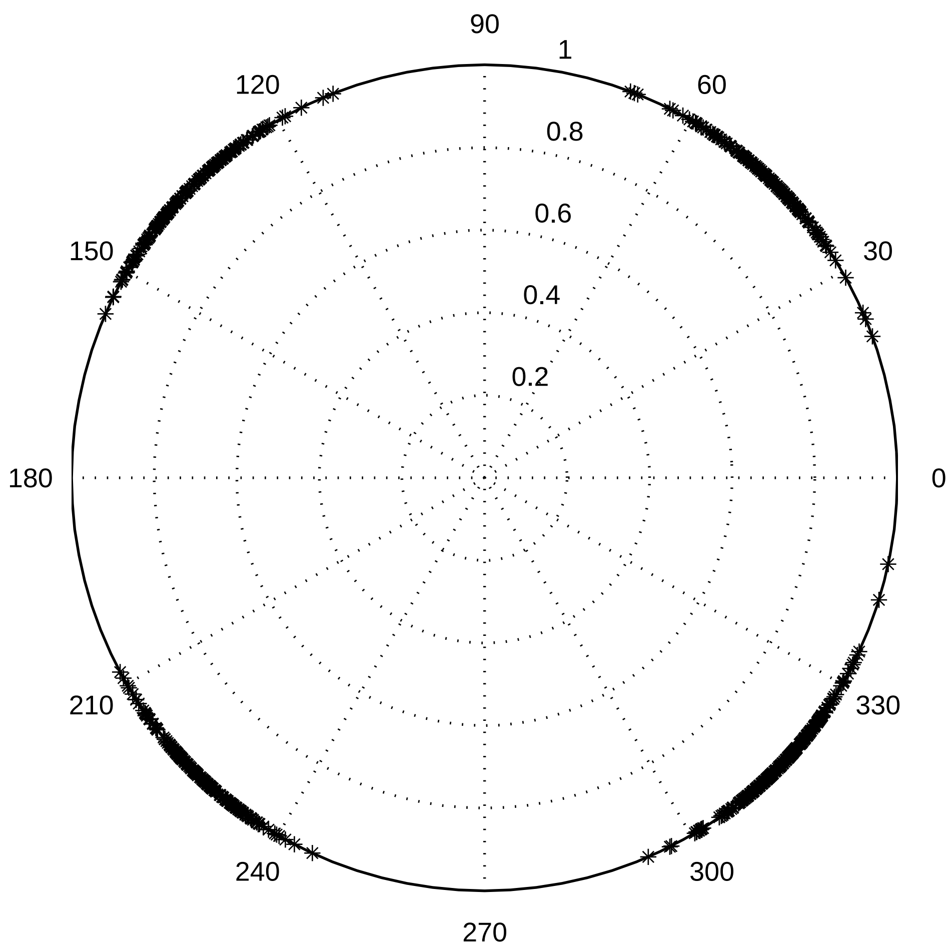


Figure 23: *Signal constellation for tone number 10 over 703 symbol periods after constellation rotation.*

interval and framing sequence length values indicated in Table 2. For the 10-bit character set used in this study and a long interleaving degree (e.g. 585, see Table 2), a 16380-bit long insertion interval and a 260-bit long framing sequence were found to be used. Therefore, the 1111111110 sequence repeats every 16640 bits. This result was found while analyzing the 1200 bit rate mode.

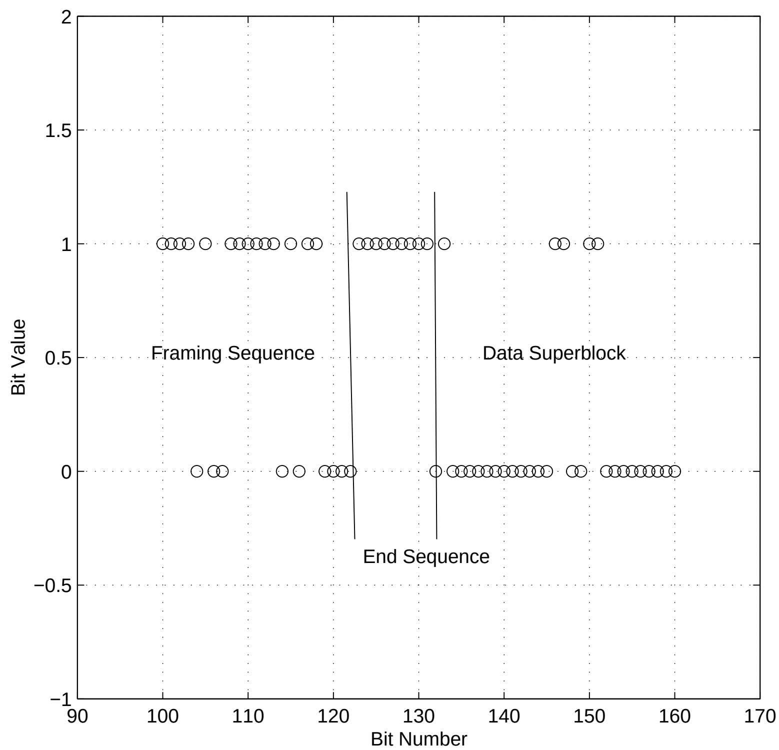


Figure 24: *End sequence for first framing sequence*

5. DEINTERLEAVING AND DECODING

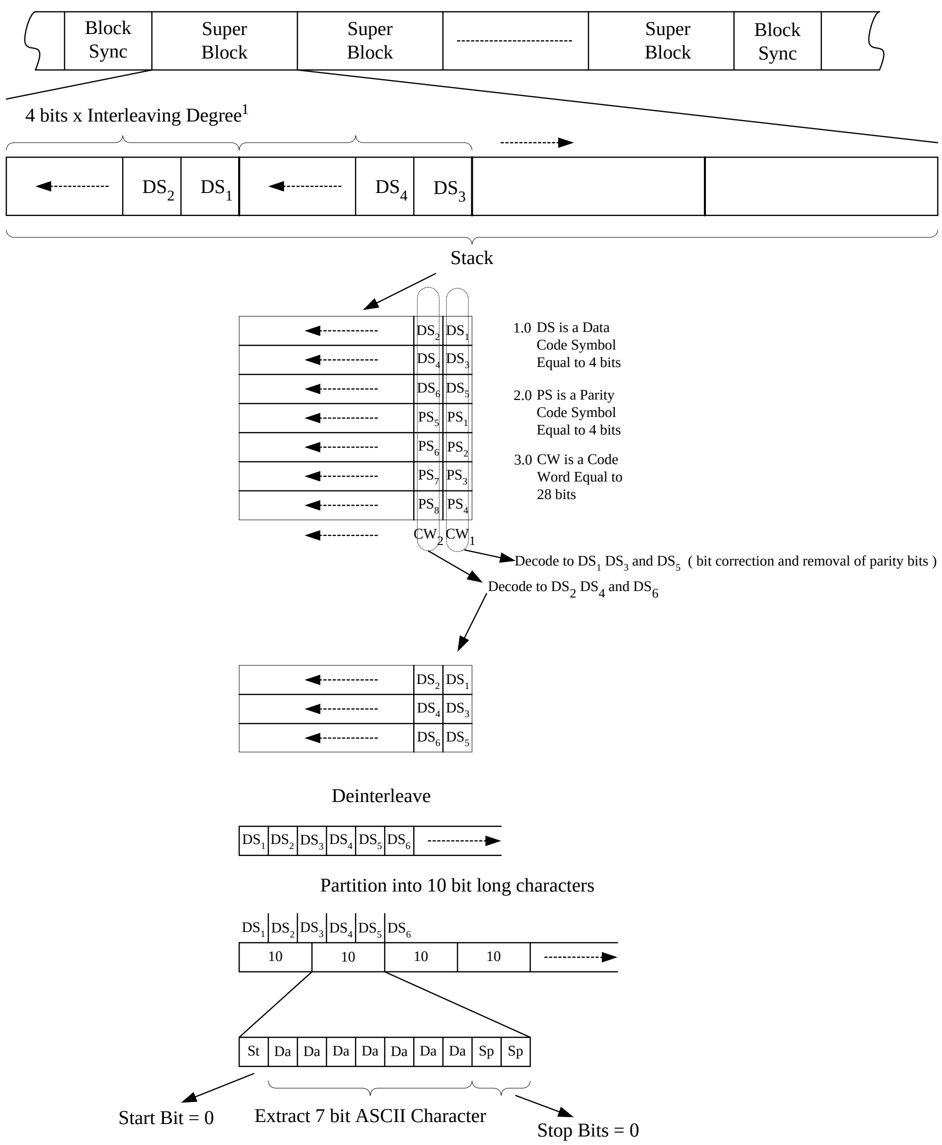
The deinterleaving and decoding process is summarized in Figure 25. The deinterleaving process is based on a series of Matlab developed algorithms. The use of these algorithms requires input of *a priori* knowledge of both the bit rate and interleaving degree. The bit rate can be deduced from analyzing the redundancy levels described earlier. Thus, for a given bit rate, the deinterleaving degree can be obtained through the tracing of two consecutive framing sequences. The number of bits between the two 1111111110 sequences can be determined and the interleaving degree looked up in a table. The deinterleaving process is carried out as described in the following paragraphs.

First, based on tabulated values extracted from [1], the synchronization sequence length (framing sequence) in numbers of bits and insertion interval in numbers of superblocks, is determined for a given bit rate and interleaving degree.

Second, based on the identification of the final bits of two consecutive framing sequences, the data superblocks are identified and extracted from the decodable bit stream. The identification of the framing sequence is accomplished by matching 10-bit long sequences extracted from the decodable bit stream with the end sequence 1111111110. The end sequence indicates where the framing sequence ends and where the data superblocks start. After the first end sequence is found, its repeat interval is verified against intervals specified in [1]. The interval, in numbers of bits, between two consecutive end sequences corresponds to the number of bits contained in all the superblocks. For the 10-bit character and the 1200 bps asynchronous mode discussed here, the number of bits is 16380, with a framing sequence of 260 bits. Thus, two consecutive end sequences are 16640 bits apart in the decodable bit stream. When the two conditions described above are satisfied, the data superblocks are extracted from the decodable bit stream.

Third, the deinterleaving operation is initiated. According to [1], deinterleaving is performed on each individual superblock, and there can be several superblocks between the framing sequences. Therefore, the number of superblocks must be known. This information is also in [1] and depends on the bit rate and interleaving degree. The number of codewords in a superblock is also needed to perform deinterleaving. This number depends on the interleaving degree, and, from Table 2, the interleaving degree could be understood to be the number of codewords in a superblock.

Recall that a codeword is either 28 or 56 bits long. The portion of the bit stream corresponding to the insertion interval is partitioned in segments whose length, given in bits, is 4 times the number of codewords (recall that each symbol is 4 bits long). The number of segments so obtained corresponds to the number of symbols composing the codewords (7 for bits rates below 2400 bps and 14 for 2400 bps). The segments are then stacked in a 7 or 14 row matrix as illustrated in Figures 6 and 7. This stacking operation allows isolating each of the codewords.



Note: Interleaving degree is equivalent to # of Codewords

Figure 25: *Deinterleaving and decoding process.*

The final operation consists of reorganizing the bit stream so that all the bits of each codeword are in continuous sequence, starting with the first codeword of the superblock. The deinterleaving process is illustrated in Figure 25 for the 1200 bps data rate, with an interleaving degree of 585 (1 superblock per insertion interval, 585 codewords per superblock, 12 data bits and 16 parity bits per codeword). This deinterleaving process is repeated for all the superblocks in the insertion interval.

Decoding is performed, in sequence, over the deinterleaved bit stream of each of the codewords pertaining to the same superblock. That is, decode the bit stream of codeword 1, then of codeword 2, and so on.

The decoding is done by calls to a series of Matlab functions. One function calculates the syndromes of the codeword, or received polynomial. Another, based on Berlekamp's iterative algorithm, determines the error locator polynomial from the syndrome values. This is followed by a function that calculates the roots of the error locator polynomial to obtain the error locators. The last Matlab function, based on Forney's algorithm, calculates the error magnitude polynomial.

The above operations determine the error pattern in the received codeword and, therefore, allow one to retrieve the transmitted codeword. This operation is repeated for all codewords of the same superblock.

The resulting bit stream now contains only data bits, all parity bits having been removed. Since during the encoding process codewords were created in pairs and therefore interleaved, another deinterleaving operation must be carried out. (Refer to Figures 6 and 7 for odd and even numbers of codewords.) In the case of an odd number of codewords in the superblock, the last codeword in the superblock does not belong to a pair of codewords and, therefore, has not been interleaved.

After this last deinterleaving operation has been carried out, the original transmitted bit stream is recovered. For synchronous transmission, the decoded bit stream is split into 7-bit long characters and ASCII conversion is performed. For asynchronous transmission, the decoded bit stream is split into segments corresponding to the number of bits forming the characters (10-bit characters for this study). The start bit, stop bit and parity bit are removed ([1] indicates that these are zero or null bits) and the 7-bit long ASCII characters are formed. This is illustrated in the following example:

Step 1: Extract the decoded bit stream.
000010110000000010000001101100011110110000111011000111001100

Step 2: Split the bit stream into 10-bit long characters.
0000101100 0000001000 0001101100 0111101100 0011101100 0111001100

Step 3: Remove the start bit at the beginning of each character, as well as the stop bit and parity bit at the end, to obtain the 7-bit ASCII characters.

0001011
0000010
0011011
1111011
0111011
1110011

Step 4: Perform ASCII conversion, as in Table 10. The bits must be reversed inside each 7-bit long character before performing ASCII conversion.

Table 10: *Example of Decoded Bit Stream*

Binary String	Decimal Value	Character
1101111	111	o
1110110	118	v
1100101	101	e
1110010	114	r
0001101	13	carriage return
0001010	10	line feed
1110011	115	s
1110101	117	u
1100011	99	c
1101000	104	h
0100000	32	space
1101100	108	l
1101111	111	o
1101110	110	n
1100111	103	g

6. CONCLUSIONS

The work presented in this report summarizes the familiarization activity in receiving, demodulating and decoding a 39 tone signal. This work consisted of obtaining a strong understanding of this type of signal and in reaching a stable point in the development of the algorithms. It was possible to retrieve the content of a message intercepted in a laboratory environment. However, the problem of the rotation in the constellations requires more research. It was speculated (but not proven) that this rotation probably resulted from the tones not being exactly spaced 56.25 Hz apart as specified in [1].

The signal captures were carried out in an ideal laboratory situation. The current algorithms must be enhanced to be able to handle the more general situation. Presently, time synchronization of the data, prior to demodulation, is based on the preamble tones. As mentioned in the report, a signal reception can occur anywhere during the transmission. If this is the case, any of the 39 tones can be used for time synchronization purposes. This added capability in the algorithms will require only a slight modification. Another problem with the general intercept case is that *a priori* knowledge of some parameters were used for the deinterleaving and decoding activities (i.e., bit rate and interleaving degree). Bit rate can be easily identified by looking at the redundancy within one signal element (for 2400 and 1200 bps) and from the redundancy pattern existing between subsequent signal elements for bit rates below 1200 bps. Once the bit rate has been identified, the interleaving degree can be deduced by determining the length in bits of the insertion interval (between framing sequence bits 1111111110). Finally, the decoding algorithms must improved to be able to handle the 2400 bps case, which uses 56-bit codewords.

References

1. Department of Defense, USA,“Interoperability and Performance Standards for Data Modem MIL-STD-188-110A”, September, 1991

2. Plank, J. S., “A Tutorial on Reed-Solomon Coding for Fault-Tolerance in RAID-like Systems”, Technical Report UT-CS-96-332, Department of Computer Science, University of Tennessee, July, 1996

3. Michelson, A. M., Levesque, A. H., “Error Control Techniques for Digital Communication”, John Wiley and Sons Inc., 1985

4. Hewlett Packard, “HP E6500A VXI Tuner User’s Guide”, Hewlett Packard Publication E6500-90001, November, 1997

5. Hewlett Packard, “HP E6501A/E6502A/E6503A VXI Receiver User’s Guide”, Hewlett Packard Publication E6500-90006, September, 1998

6. Harris, “Falcon Series Tactical Communications System AN/PRC-138 Manpack Installation and Operation Manual”, Harris Publication 10372-0008-01, November, 1993

7. Harris, “RF-6710 Universal Data Terminal (UDT) Software User’s Guide”, Harris Publication 10518-2020-01, February, 1997

8. Harris, “RT-1694 Receiver-Transmitter Intermediate Maintenance Manual”, Harris Publication 10515-0007-4300, Rev. A, December, 1995

9. “Comprehensive Dictionary of Electrical Engineering”, P.A. Laylante, editor, CRC Press, IEEE Press, 1999

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

   DEFENCE R&D CANADA - OTTAWA
   OTTAWA, ONTARIO, K1A 0Z4

2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable)

   UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

   DEMODULATION AND DECODING STUDIES OF THE 39-TONE MIL-STD-188-110A HF SIGNAL (U)

4. AUTHORS (Last name, first name, middle initial)

   DUPRAT, GERARD.

5. DATE OF PUBLICATION (month and year of publication of document)

   NOVEMBER 2002

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)

   49

6b. NO. OF REFS (total cited in document)

   9

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

   DRDC OTTAWA TECHNICAL MEMORANDUM

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

   DEFENCE R&D CANADA - OTTAWA
   OTTAWA, ONTARIO, K1A 0Z4

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

   15bb30

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

   DRDC Ottawa TM 2002-082

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

   (x) Unlimited distribution
   ( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
   ( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
   ( ) Distribution limited to government departments and agencies; further distribution only as approved
   ( ) Distribution limited to defence departments; further distribution only as approved
   ( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

   Unlimited.

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

(U) A set of algorithms has been developed to demodulate and decode the 39 tone signal, which is prevalent in the High Frequency (HF) frequency band. This signal, based on the MIL-STD-188-110A Standard, is one of several different types generated by the AN/PRC-138 Harris radio. Defence R&D Canada (DRDC) - Ottawa has two of these radios. The work focused on the 39 tone fixed frequency mode, although the 39 tone signal is also the underlying modulation used in the AN/PRC-138's frequency hopping mode. Thus, the work described here will also be useful to anyone developing dehopping and demodulation algorithms for the AN/PRC-138 frequency hopping signal. The signals were captured by the Agilent Technologies Blackbird system in a laboratory setting. The objective of the task was to gain a detailed understanding of the sophisticated 39 tone signal and to develop software radio algorithms for demodulating and decoding the signal. The report describes the signal structure, the signal capture equipment, and the steps involved in demodulating and decoding the signal so that the transmitted messages can be read at the receiving end. The report also describes the problems encountered during the algorithm development process. Since the signal captures took place in an ideal setting and a priori information of the signal structure was used to assist in the demodulation and decoding process, the current algorithms must be modified slightly to be able to handle the more general situation. The report concludes with ways of generalizing the algorithms.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

39 TONE
MIL-STD-188-110A
RADIO FREQUENCY
COMMUNICATIONS
HF
DQPSK
REED-SOLOMON
GALOIS FIELD
ENCODING/DECODING
SIGNAL DEMODULATION

Defence R&D Canada

Canada's leader in defence
and national security R&D

R & D pour la défense Canada

Chef de file au Canada en R & D
pour la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca