

Universidad de Alcalá
Escuela Politécnica Superior

GRADO EN ELECTRÓNICA Y AUTOMÁTICA INDUSTRIAL



Trabajo Fin de Grado

“Herramienta de apoyo para comparación de esquemas de codificación en sistemas de localización basados en señales ultrasónicas”

ESCUELA POLITECNICA
SUPERIOR

Autor: Andrea Pérez Alonso

Tutor/es: M.^a Carmen Pérez Rubio y Santiago Murano

Junio 2019



Universidad
de Alcalá

Escuela Politécnica Superior
Departamento de Electrónica

GRADO EN INGENIERÍA ELECTRÓNICA Y AUTOMÁTICA
INDUSTRIAL

Trabajo Fin de Grado

**“Herramienta de apoyo para comparación de
esquemas de codificación en sistemas de localización
basados en señales ultrasónicas”.**

Autora: Andrea Pérez Alonso

Tutora: M^ª Carmen Pérez Rubio

Cotutor: Santiago Murano

TRIBUNAL

Presidente: José Manuel Villadangos Carrizo

Vocal 1º: Francisco Javier Meca Meca

Vocal 2º: M.^ª Carmen Pérez Rubio

Fecha: Julio 2019

Agradecimientos

Quisiera agradecer este proyecto a la gente que siempre me está apoyando.

A toda mi familia: en especial a mis padres, Alicia y Juan Carlos, ya que sin vosotros esto no sería posible. Por luchar constantemente conmigo por el futuro que yo elijo y ser un apoyo incondicional. No puedo pedir más.

A mi novio, Daniel, por aguantar todas esas curvas repentinas de ánimo, por hacerme ver que cada paso que doy es para avanzar y no retroceder. Porque el cierre de una etapa es el comienzo de otra, nuestra etapa, nuestra vida.

A todos mis grandes amigos, los que han vivido mis encerronas de estudio y no han dejado nunca de animarme.

A todos esos compañeros, y ahora también amigos, que me han acompañado durante estos años de carrera. Ellos más que nadie, saben lo que es vivir esto y juntos hemos podido con todo. Sé que os llevo para siempre.

A mis tutores, M.^a Carmen y Santiago, por vuestras horas incontables de dedicación y por vuestra gran motivación para este proyecto que ha hecho que todo resulte sencillo y satisfactorio.

Índice general

Agradecimientos.....	2
Índice general	4
Índice de figuras	6
Índice de tablas	14
Glosario de acrónimos y símbolos.....	15
Acrónimos.....	15
Símbolos	15
Resumen	18
Resumen en castellano.....	18
Abstract.....	19
Resumen extendido	20
Memoria	22
1. INTRODUCCIÓN.....	22
1.1. Antecedentes y estado actual del tema.....	22
1.2. Objetivos del TFG.....	23
1.3. Contexto	23
1.4. Estructura del documento.....	23
2. ESQUEMAS DE CODIFICACIÓN Y MODULACIÓN ANALIZADOS.....	25
2.1. Conceptos básicos	25
2.2. Revisión de códigos	30
2.3. Modulación.....	44
2.4. Conclusiones	48
3. MODELADO DE EFECTOS TÍPICOS SOBRE LA SEÑAL ULTRASÓNICA	
50	
3.1. Respuesta del transductor.....	50
3.2. Transmisión de la señal	51
3.3. Efectos del canal.....	51
3.4. Conclusiones	53
4. DISEÑO DE LA PLATAFORMA INTERACTIVA PARA LA PRUEBA DE	
ESQUEMAS DE CODIFICACIÓN EN SISTEMAS ULTRASÓNICOS.....	57
4.1. INTRODUCCIÓN	57

4.2. INTERFAZ GRÁFICA: PANTALLA INICIO, ELECCIÓN CODIFICACIÓN	58
4.3. CONFIGURACIÓN EMISIÓN	59
4.4. CONFIGURACIÓN EFECTOS	78
4.5. VISUALIZACIÓN DE RESULTADOS	84
4.6. VENTANA DE SALIDA.....	117
5. CONCLUSIONES Y LÍNEAS FUTURAS	118
5.1. Recopilación de logros del trabajo	118
5.2. Trabajos futuros.....	118
6. PRESUPUESTO.....	119
6.1. Presupuesto de material y licencias.....	119
6.2. Presupuesto de las horas de desarrollo	119
6.3. Presupuesto global.....	120
7. PLIEGO DE CONDICIONES	121
8. MANUAL DE USUARIO	122
8.1. Introducción	122
8.2. Ventana Inicio	123
8.3. Ventana de selección del lenguaje de codificación.....	124
8.4. Ventana de Configuración de la emisión	125
8.5. Ventana de Configuración de efectos añadidos	150
8.6. Ventana de Visualización de resultados.....	157
9. BIBLIOGRAFÍA	190
10. ANEXO: CÓDIGOS.....	192
10.1. Cálculo de la Cota AC para secuencias Kasami.....	192
10.2. Cálculo de la Cota AC para las secuencias CSS	194
10.3. Cálculo de la Cota AC para secuencias LS	197
10.4. Cálculo de la Cota AC para secuencias Zadoff-Chu	198
10.5. Cálculo de la Cota CC de las secuencias Kasami.....	200
10.6. Cálculo de la Cota CC de las secuencias CSS.....	202
10.7. Cálculo de la Cota CC de las secuencias LS	205
10.8. Cálculo de la Cota CC de las secuencias Zadoff-Chu	207
10.9. Calcular SMR	209
10.10. Simulación paramétrica del ruido.....	210

Índice de figuras

Fig. 1. Esquema general de la interfaz.....	21
Fig. 2. Funciones de correlación según tipo de códigos	28
Fig. 3. Secuencias Kasami 63 bits	31
Fig. 4. AC y CC Secuencia Kasami 63 bits periódica.....	32
Fig. 5. AC y CC Secuencia Kasami 63 bits aperiódica	32
Fig. 6. Propiedades de auto-correlación parejas Golay del código C1	33
Fig. 7. Propiedades de correlación cruzada entre parejas Golay del código 1 con el código 2	34
Fig. 8. AC y SACF para código 1 secuencia M-CSS M=4 y L=64	35
Fig. 9. CC y SCCF para código 1 con código 4 secuencia M-CSS M=4 y L=64	36
Fig. 10. Concatenación código CSS	37
Fig. 11. Entrelazado código CSS.....	37
Fig. 12. Estructura de M=4 códigos LS caso periódico [Pér09]	38
Fig. 13. Códigos LS con M=4, Lo=16 aperiódico.....	39
Fig. 14. AC códigos LS M=4, Lo=16.....	40
Fig. 15. CC Códigos LS M=4, Lo=16	40
Fig. 16. C1,C2,C3,C4 y C5 en valor absoluto	41
Fig. 17. AC Código 1 y CC Código 1-Código 2 Secuencias Zadoff-Chu L=61 periódica	42
Fig. 18. AC Código 1 y CC Código 1-Código 2 Secuencias Zadoff-Chu L=61 aperiódica	43
Fig. 19. FFT Código 1 Zadoff Chu L=61, Nemo=5	43
Fig. 20. Modulador BPSK [GPÁ+15]	44
Fig. 21. Código Kasami L=63, Nemo=4	45
Fig. 22. Señal portadora BPSK $f_c=41,667\text{kHz}$, $f_{se}=500\text{kHz}$ y $N_c=2$	45
Fig. 23. Señal Kasami modulada resultante	46
Fig. 24. Modulador QPSK [GPÁ+15].....	47
Fig. 25. Códigos Zadoff-Chu L=61, Nemo=4	47
Fig. 26. Señal portadora QPSK $f_c=41,667\text{kHz}$, $f_{se}=500\text{kHz}$ y $N_c=2$	47
Fig. 27. Señal Zadoff.Chu modulada resultante	48
Fig. 28. Correlaciones banda base VS modulada con M=4, L=64 CSS modulados con $N_c=2$ ciclos de portadora y factor de sobremuestreo $O_f=12$	48
Fig. 29. Pico de correlación ampliado	49
Fig. 30. Respuesta del transductor	50
Fig. 31. Señal CSS con efecto del transductor	50
Fig. 32. Imagen orientativa de las configuraciones de emisión	51
Fig. 33. Señal Kasami modulada en BPSK y con efecto del transductor.....	53
Fig. 34. Señal Kasami modulada en BPSK, con efecto del transductor y con Ruido Gaussiano de -10dB.....	53

Fig. 35. Señal Kasami modulada en BPSK, con efecto del transductor y con Ruido Impulsivo de -20dB en la mitad del código con 10% de duración.....	54
Fig. 36. Señal Kasami modulada en BPSK, con efecto del transductor y cerca-lejos de ganancia 1,0.6,3,1.3 y 0.2 para cada código respectivamente.....	54
Fig. 37. Señal Kasami modulada en BPSK, con efecto del transductor y con Multicamino con retardo 6, 0, 7.2, 15 y 3 ms, y ganancia 1, 0.6, 3, 1.3 y 0.2, para cada código respectivamente.....	55
Fig. 38. Señal Kasami modulada en BPSK, con efecto del transductor y Doppler en el aire con $v_r = -15\text{m/s}$	55
Fig. 39. Cálculo de RMSE entre -20dB y 10 dB.....	56
Fig. 40. Ventana de diseño GUIDE.....	57
Fig. 41. TFG: Introducción.....	58
Fig. 42. Ventana 1 TFG: Elección de codificación.....	58
Fig. 43. Ventana 2 TFG: Configuración emisión Kasami.....	60
Fig. 44. Ejemplo Ventana 2 Kasami: Configuración.....	61
Fig. 45. Ejemplo ventana 2 Kasami: banda base.....	62
Fig. 46. Ejemplo ventana 2 Kasami: portadora.....	62
Fig. 47. Ejemplo ventana 2 Kasami: modulada.....	63
Fig. 48. Ejemplo ventana 2 Kasami: transductor.....	63
Fig. 49. Ejemplo ventana 2 Kasami: emisión.....	64
Fig. 50. Ventana 2 TFG: Configuración de la emisión CSS.....	65
Fig. 51. Ejemplo Ventana 2 CSS: Configuración.....	66
Fig. 52. Ejemplo ventana 2 CSS: banda base.....	67
Fig. 53. Ejemplo ventana 2 CSS: portadora.....	67
Fig. 54. Ejemplo ventana 2 CSS: modulada.....	68
Fig. 55. Ejemplo ventana 2 CSS: transductor.....	68
Fig. 56. Ejemplo ventana 2 CSS: emisión.....	69
Fig. 57. Ventana 2 TFG: Configuración emisión LS.....	70
Fig. 58. Ejemplo Ventana 2 LS: Configuración.....	71
Fig. 59. Ejemplo ventana 2 LS: banda base.....	71
Fig. 60. Ejemplo ventana 2 LS: portadora.....	72
Fig. 61. Ejemplo ventana 2 LS: modulada.....	72
Fig. 62. Ejemplo ventana 2 LS: transductor.....	73
Fig. 63. Ejemplo ventana 2 LS: emisión.....	73
Fig. 64. Ventana 2 TFG: Configuración emisión Zadoff-Chu.....	74
Fig. 65. Ventana 2 Zadoff-Chu: Configuración.....	75
Fig. 66. Ejemplo ventana 2 Zadoff-Chu: banda base.....	75
Fig. 67. Ejemplo ventana 2 Zadoff-Chu: portadora.....	76
Fig. 68. Ejemplo ventana 2 Zadoff-Chu: modulada.....	76
Fig. 69. Ejemplo ventana 2 Zadoff-Chu: transductor.....	77
Fig. 70. Ejemplo ventana 2 Zadoff-Chu: emisión.....	77
Fig. 71. Ventana 3 TFG: Configuración de efectos (1).....	78
Fig. 72. Ventana 3 TFG: Configuración de efectos (2).....	79

Fig. 73. Ventana 3: Ruido Blanco Gaussiano.....	79
Fig. 74. Ventana 3: Ruido Impulsivo: Ejemplo.....	80
Fig. 75. Ventana 3: Efecto cerca-lejos: Ejemplo	81
Fig. 76. Función cálculo multicamino modo aleatorio.....	82
Fig. 77. Ventana 3: Multicamino: Ejemplo	82
Fig. 78. Efecto Doppler	83
Fig. 79. Simulación paramétrica.....	83
Fig. 80. Ventana 4 TFG: Resultados caso 1	85
Fig. 81. Ventana 4: Resultados caso 1, Representación aperiódica	86
Fig. 82. Resultados caso 1, Representación periódica.....	86
Fig. 83. Ventana 4: Resultados caso 1, Representación AC.....	87
Fig. 84. Ventana 4: Resultados caso 1, Representación CC.....	87
Fig. 85. Ventana 4: Resultados caso 1, Representación FFT	88
Fig. 86. Ventana 4 TFG: Resultados caso 2	89
Fig. 87. Ventana 4: Resultados caso 2, Visualización CC con señal emitida	90
Fig. 88. Ventana 4: Resultados caso 2, Visualización correlaciones al cuadrado normalizadas.....	90
Fig. 89. Ventana 4: Resultados caso 2, Visualización envolvente de correlación	91
Fig. 90. Ventana 4 TFG: Resultados caso 3	92
Fig. 91. Ventana 4 Kasami: banda base aperiódica.....	93
Fig. 92. Ventana 4 Kasami: AC banda base aperiódica	94
Fig. 93. Ventana 4 Kasami: CC banda base aperiódica	94
Fig. 94. Ventana 4 Kasami: FFT banda base.....	95
Fig. 95. Ventana 4 Kasami: banda base periódica.....	95
Fig. 96. Ventana 4 Kasami: AC banda base periódica	96
Fig. 97. Ventana 4 Kasami: CC banda base periódica	96
Fig. 98. Ventana 4 Kasami: modulación aperiódica.....	97
Fig. 99. Ventana 4 Kasami: AC modulada aperiódica	97
Fig. 100. Ventana 4 Kasami: CC modulada aperiódica	98
Fig. 101. Ventana 4 Kasami: FFT modulada	98
Fig. 102. Ventana 4 Kasami: modulada periódica	99
Fig. 103. Ventana 4 Kasami: AC modulada periódica	99
Fig. 104. Ventana 4 Kasami: CC modulada periódica	100
Fig. 105. Ventana 4 Kasami: transductor aperiódica.....	100
Fig. 106. Ventana 4 Kasami: AC transductor aperiódica.....	101
Fig. 107. Ventana 4 Kasami: CC transductor aperiódica	101
Fig. 108. Ventana 4 Kasami: FFT transductor	102
Fig. 109. Ventana 4 Kasami: transductor periódica	102
Fig. 110. Ventana 4 Kasami: AC transductor periódica.....	103
Fig. 111. Ventana 4 Kasami: CC transductor periódica.....	103
Fig. 112. Ventana 4 Kasami: señal recibida ideal	104
Fig. 113. Ventana 4 Kasami: CC Señal recibida ideal	104

Fig. 114. Ventana 4 Kasami: correlaciones normalizadas al cuadrado señal recibida ideal	105
Fig. 115. Ventana 4 Kasami: envolvente correlación señal recibida ideal.....	105
Fig. 116. Ventana 4 Kasami: SMR para diferente ventana de exclusión	106
Fig. 117. Ventana 4 Kasami: CC normalizadas al cuadrado Señal recibida ideal con distinta exclusión	106
Fig. 118. Ventana 4 Kasami: Señal recibida con efectos	107
Fig. 119. Ventana 4 Kasami: CC Señal recibida con efectos	107
Fig. 120. Ventana 4 Kasami: CC normalizadas al cuadrado señal recibida con efectos	108
Fig. 121. Ventana 4 Kasami: envolvente de correlación señal recibida con efectos....	108
Fig. 122. Ventana 4 Kasami: Simulación paramétrica	109
Fig. 123. Ventana 4 CSS: Banda base ideal	111
Fig. 124. Ventana 4 CSS: AC de las M secuencias y SACF del código 1 banda base ideal	111
Fig. 125. Ventana 4 CSS: CC de las M secuencias y SCCF del código 1 banda base ideal	112
Fig. 126. Ventana 4 CSS: FFT de las M secuencias del código 1 banda base ideal	112
Fig. 127. Ventana 4 CSS: señal recibida sin efectos añadidos	113
Fig. 128. Ventana 4 LS: banda base	114
Fig. 129. Comparación del resultado de cotas dentro y fuera de la ZCZ	114
Fig. 130. Ventana 4 LS: señal recibida sin efectos añadidos	115
Fig. 131. Ventana 4 Zadoff-Chu: FFT banda base	116
Fig. 132. Ventana 5 TFG: Locate-US	117
Fig. 133. Manual de Usuario: Inicio de la interfaz	122
Fig. 134. Manual de Usuario: Ventana de Inicio.....	123
Fig. 135. Manual de Usuario: Selección del lenguaje de codificación.....	124
Fig. 136. Manual de Usuario: Ventana de Configuración de la emisión de secuencias Kasami.....	125
Fig. 137. Imágen representativa	127
Fig. 138. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Datos.....	128
Fig. 139. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización banda base.....	128
Fig. 140. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la portadora	129
Fig. 141. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la señal modulada.....	129
Fig. 142. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la señal con efecto del transductor	130
Fig. 143. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la señal a emitir	130

Fig. 144. Manual de Usuario: Ventana de Configuración de la emisión de secuencias CSS	131
Fig. 145. Imágen representativa	134
Fig. 146. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Datos.....	135
Fig. 147. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización banda base.....	135
Fig. 148. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la portadora	136
Fig. 149. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la señal modulada.....	136
Fig. 150. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la señal con efecto del transductor.....	137
Fig. 151. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la señal a emitir	137
Fig. 152. Manual de Usuario: Ventana de Configuración de la emisión de secuencias CSS	138
Fig. 153. Imágen representativa	140
Fig. 154. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Datos.....	141
Fig. 155. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización banda base.....	141
Fig. 156. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la portadora	142
Fig. 157. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la señal modulada.....	142
Fig. 158. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la señal con efecto del transductor	143
Fig. 159. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la señal a emitir	143
Fig. 160. Manual de Usuario: Ventana de Configuración de la emisión de secuencias Zadoff-Chu	144
Fig. 161. Imágen representativa	146
Fig. 162. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Datos	147
Fig. 163. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización banda base	147
Fig. 164. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la portadora.....	148
Fig. 165. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la señal modulada.....	148
Fig. 166. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la señal con efecto del transductor	149

Fig. 167. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la señal a emitir	149
Fig. 168. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables	151
Fig. 169. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Ruido Blanco Gaussiano.....	151
Fig. 170. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Ruido Impulsivo: Ejemplo.....	152
Fig. 171. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Ruido Impulsivo: Ejemplo (2).....	152
Fig. 172. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Efecto cerca-lejos: Ejemplo.....	153
Fig. 173. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Efecto cerca-lejos: Ejemplo (2)	153
Fig. 174. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Función cálculo multicamino modo aleatorio	154
Fig. 175. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Multicamino: Ejemplo	154
Fig. 176. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Multicamino: Ejemplo (2)	155
Fig. 177. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Efecto Doppler.....	155
Fig. 178. Manual de Usuario: Ventana de Configuración de efectos añadidos: Simulación paramétrica	156
Fig. 179. Manual de Usuario – Ventana de Configuración de efectos añadidos: Simulación paramétrica de ruido.....	156
Fig. 180. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Configuración de la emisión.....	164
Fig. 181. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Configuración de efectos añadidos.....	164
Fig. 182. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: banda base aperiódica	165
Fig. 183. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC banda base aperiódica.....	165
Fig. 184. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC banda base aperiódica.....	166
Fig. 185. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: FFT banda base.....	166
Fig. 186. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: banda base periódica	167
Fig. 187. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC banda base periódica.....	167

Fig. 188. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC banda base periódica.....	168
Fig. 189. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada aperiódica.....	168
Fig. 190. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada aperiódica.....	169
Fig. 191. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada aperiódica.....	169
Fig. 192. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: FFT modulada	170
Fig. 193. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada periódica.....	170
Fig. 194. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada periódica.....	171
Fig. 195. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada periódica.....	171
Fig. 196. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada con efecto transductor aperiódica	172
Fig. 197. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada con efecto transductor aperiódica	172
Fig. 198. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada con efecto transductor aperiódica	173
Fig. 199. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: FFT modulada con efecto transductor.....	173
Fig. 200. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada con efecto transductor periódica	174
Fig. 201. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada con efecto transductor periódica	174
Fig. 202. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada con efecto transductor periódica	175
Fig. 203. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Señal recibida sin efectos canal.....	175
Fig. 204. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC Señal recibida sin efectos canal	176
Fig. 205. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC Normalizada al cuadrado Señal recibida sin efectos canal.....	176
Fig. 206. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Envoltente Señal recibida sin efectos canal.....	177
Fig. 207. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Señal recibida con efectos canal.....	177
Fig. 208. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC Señal recibida con efectos canal.....	178

Fig. 209. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Normalizada al cuadrado Señal recibida con efectos canal.....	178
Fig. 210. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Envolvente Señal recibida con efectos canal.....	179
Fig. 211. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Configuración efectos añadidos (2).....	179
Fig. 212. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Simulación paramétrica del ruido.....	180
Fig. 213. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: Configuración de la emisión.....	183
Fig. 214. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: Banda base ideal	183
Fig. 215. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: AC Banda base ideal	184
Fig. 216. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: CC Banda base ideal	184
Fig. 217. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: FFT Banda base ideal	185
Fig. 218. Manual de Usuario: Ventana de resultados secuencias LS: Ejemplo: Configuración de la emisión.....	186
Fig. 219. Manual de Usuario: Ventana de resultados secuencias LS: Ejemplo: Banda base dentro de la ZCZ.....	187
Fig. 220. Manual de Usuario: Ventana de resultados secuencias LS: Ejemplo: banda base fuera de la ZCZ.....	187
Fig. 221. Manual de Usuario: Ventana de resultados secuencias Zadoff-Chu: Ejemplo: FFT banda base.....	188
Fig. 222. Ventana 5 TFG: Locate-US	189

Índice de tablas

Tabla 1. Resumen de las distintas ventanas de la pantalla de resultados	84
Tabla 2. Presupuesto de materiales y licencias.....	119
Tabla 3. Presupuesto de las horas de desarrollo	119
Tabla 4. Presupuesto global.....	120

Glosario de acrónimos y símbolos

Acrónimos

AC \equiv *Auto-correlation function* (Función de autocorrelación)

BPSK \equiv *Binary Phase Shift Keying* (Modulación por Desplazamiento de Fase Binario)

BW \equiv *Band Width* (Ancho de banda)

CC \equiv *Cross-correlation function* (Función de correlación cruzada)

CDMA \equiv *Code Division Multiple Access* (Acceso Múltiple por División de Códigos)

CSS \equiv *Complementary Set of Sequences* (Conjunto de Secuencias Complementarias)

FFT \equiv *Fast Fourier Transform* (Transformada rápida de Fourier)

LFSR \equiv *Linear Feedback Shift Register* (Registro de Desplazamiento Realimentado Lineal)

LS \equiv *Loosely Synchronous codes* (Códigos Débilmente Sincronizados)

QPSK \equiv *Quadrature Phase Shift Keying* (Modulación por Desplazamiento de Fase Cuadrada)

RMSE \equiv *Root-Mean-Square Error* (Raíz del Error Cuadrático Medio)

SNR \equiv *Signal-to-Noise Ratio* (Relación señal-ruido)

SMR \equiv *Sidelobe to Mainlobe Ratio* (Relación pico lateral con pico principal)

SACF \equiv *Sum of Autocorrelation Functions* (Suma de funciones de autocorrelación)

SCCF \equiv *Sum of Cross Correlation Functions* (Suma de funciones de correlación cruzada)

ZCZ \equiv *Zero Correlation Zone* (Zona de correlación cero)

Símbolos

Θ \equiv *Bound* o valor máximo de correlación

Θ_{ac} \equiv Cota de Autocorrelación o lóbulo lateral máximo de auto-correlación

Θ_{cc} \equiv Cota de Correlación Cruzada o valor máximo de correlación cruzada

η \equiv Eficiencia energética

ϵ \equiv Umbral aceptable en las secuencias cuasi-ortogonales

$AC C_{iSk} \equiv$ Representación de la AC del Código C_i de secuencia Sk

$AC C_i \equiv$ Representación de la AC del Código C_i

$c \equiv$ Velocidad del sonido en el aire a una temperatura de 20°C y una presión de 1 atm

$C_{a,b} \equiv$ Función de correlación cruzada aperiódica entre a y b

$C_i \equiv$ Código i del conjunto

$CC C_i, C_{jSk} \equiv$ Representación de la CC del Código C_i con el código C_j de secuencia Sk

$CC C_i, C_j \equiv$ Representación de la CC del Código C_i con el código C_j

$f_c \equiv$ frecuencia de la portadora

$f_{se} \equiv$ frecuencia de muestreo

$L \equiv$ Longitud del código

$L_o \equiv$ Longitud de parejas Golay Base

$M \equiv$ Tamaño de Familia

$N_c \equiv$ Número de ciclos de la portadora

$N_{emo} \equiv$ Número de emisores simultáneos

$N_{rep} \equiv$ Número de repeticiones consecutivas del código

$q \equiv$ factor de diezmado

$R_{a,b} \equiv$ Función de correlación cruzada periódica entre a y b

$Sk \equiv$ Secuencia k del código C_i del conjunto

$T_e \equiv$ Tiempo de emisión de un código

$T_{gap} \equiv$ Espaciado entre emisiones consecutivas

$T_{shift} \equiv$ Retardo entre emisiones

$W \equiv$ Tamaño de la semi-ventana de ZCZ

$W_q \equiv$ Zona reducida de interferencias

Resumen

Resumen en castellano

En este Trabajo Fin de Grado (TFG) se ha desarrollado una plataforma interactiva mediante GUIDE, herramienta de Matlab, que permite comparar distintos esquemas de codificación y modulación y analizar su desempeño en sistemas de posicionamiento ultrasónicos. Así, mediante dicha herramienta se pueden realizar pruebas con varios códigos (Kasami, Conjuntos de Secuencias Complementarias, Zadoff-Chu y LS), configurando parámetros como la longitud, número de emisores simultáneos o solapamiento entre emisiones. Además, pueden configurarse diversos parámetros de la modulación y añadir efectos típicos sobre la señal ultrasónica (ancho de banda del transductor, ruido ambiente, multicamino, efecto cerca-lejos y Doppler, entre otros) para su evaluación. La herramienta proporciona gráficas de resultados de correlación, espectro en frecuencia y otros valores de interés como las cotas de correlación o el SMR (*Sidelobe to Mainlobe Ratio*). Además, permite exportar la configuración del esquema de codificación-modulación elegido para la posterior realización de pruebas reales.

La herramienta desarrollada es versátil, permite múltiples configuraciones para la evaluación de distintos efectos típicos sobre la señal ultrasónica, así como el análisis comparativo y elección de la codificación más adecuada según la aplicación final.

Palabras clave: ultrasonidos, codificación, modulación, efectos reales, aplicación.

Abstract

In this Final Degree Project an interactive platform has been developed through GUIDE, tool of Matlab, which allows to compare coding and modulation schemes and analyze their performance in ultrasonic positioning systems. Thus, by means of which it is possible to carry out tests with several codes (Kasami, Complementary Sequence Sets, Zadoff-Chu and LS), configuring the parameters such as the length, the number of simultaneous emitters or the overlap between the emissions. In addition, various parameters of the modulation and the typical effects of the ultrasonic signal (width of the transducer band, ambient noise, multipath, near-far effect and Doppler, among others) can be configured for its evaluation. Graphical tools for correlation results, frequency spectrum and other values of interest such as correlation fees in the SMR (Sidelobe to Mainlobe Ratio). In addition, it allows exporting the configuration of the coding-modulation scheme chosen for the realization of real tests.

The developed tool is versatile, allows multiple configurations for the evaluation of the typical effects on the ultrasonic signal, as well as the comparative analysis and the choice of the most appropriate coding according to the final application.

Keywords: ultrasound, coding, modulation, real effects, application.

Resumen extendido

El proyecto que se expone en esta memoria pretende facilitar el estudio y análisis de distintos esquemas de codificación/modulación habituales en sistemas de posicionamiento ultrasónicos de modo que se pueda elegir aquel que mejor se ajuste a las demandas de la aplicación final. Para ello, se desarrolla una interfaz interactiva mediante GUIDE, herramienta de Matlab, que compara dichos esquemas realizando pruebas y ofreciendo resultados para cada configuración determinada de emisión.

En la fig.1. se muestra, a grandes rasgos, el recorrido que sigue la aplicación. En ella se diferencian varios apartados:

- Generación de la emisión:
 - o Selección del código a emitir: Kasami, CSS (*Complementary Set of Sequences*), Zadoff-Chu, LS (*Loosely Synchronous codes*).
 - o Configuración de propiedades: L (longitud del código), M (tamaño de familia), N_{rep} (número de repeticiones del código), N_{emo} (número de usuarios simultáneos), entre muchas otras.
- Modulación: BPSK (*Binary Phase Shift Keying*) o QPSK (*Quadrature Phase Shift Keying*) en función de la naturaleza de los códigos a emitir
 - o Permite añadir o no el efecto del transductor.
 - o Almacenamiento de los códigos en banda base o modulados para su posible aplicación en pruebas reales.
- Canal de transmisión:
 - o Configuración de efectos: ruido Gaussiano, cerca-lejos, atenuación, etc.
- Resultados y visualizaciones en sus distintos procesos (banda base, modulada o emitida):
 - o θ_{ac} (Cotas de autocorrelación)
 - o θ_{cc} (Cotas de correlación cruzada)
 - o SMR (*Sidelobe to Mainlobe Ratio*) o relación pico principal a lóbulo lateral.
 - o Correlaciones y FFT (*Fast Fourier Transform*).
 - o Simulación paramétrica del ruido:
 - Cálculo del RMSE (*Root-Mean-Square Error*) en la señal emitida en función de ruido Gaussiano.
- Además, la aplicación permite enlazar con un desarrollo previo del grupo de investigación GEINTRA-USRF para la realización de pruebas reales en un sistema de posicionamiento local ultrasónico (U-LPS) a partir de los códigos modulados generados con la herramienta desarrollada.

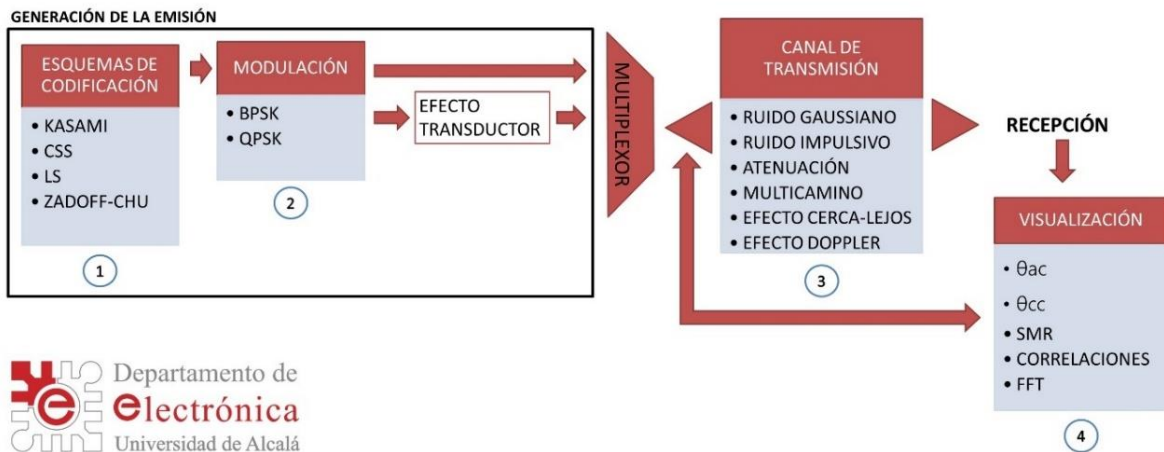


Fig. 1. Esquema general de la interfaz

Memoria

1. INTRODUCCIÓN

1.1. Antecedentes y estado actual del tema

La técnica de localización ultrasónica frente a otras alternativas, tales como las basadas en radio o visión artificial, ofrece una precisión de posicionamiento alta gracias a la baja velocidad del sonido en el aire. Además, tiene la ventaja de utilizar transductores ultrasónicos pequeños y de bajo costo. [MPU⁺18]

La incorporación de técnicas de codificación en este tipo de sistemas presenta una serie de ventajas tales como una mayor robustez ante el ruido, al efecto cerca-lejos o al efecto Doppler. Por otro lado, para que sea posible realizar varias emisiones simultáneas sin producir interferencias entre ellas, se deben elegir secuencias de modo que el valor de correlación cruzada entre ellas sea mínimo. Si se usan esquemas de codificación adecuados no resulta necesario incluir algoritmos de cancelación de interferencias más complejos y costosos computacionalmente. Esto, además de permitir un aumento de la frecuencia de operación, resulta de gran utilidad en sistemas que requieren conjuntos de medidas obtenidas en condiciones similares y en donde dichas condiciones cambian rápidamente. Este es el caso, por ejemplo, de los sistemas de localización, en donde el robot u objeto a localizar suele estar en movimiento. [Pér09]

Cabe la posibilidad de introducir estos sistemas en otras aplicaciones basadas en técnicas CDMA (Acceso Múltiple por División de Códigos) como pueden ser las comunicaciones inalámbricas y la sincronización.

Para ajustar los códigos a emitir al ancho de banda del transductor, es necesaria la modulación. Esto implica una pérdida de las propiedades iniciales de los códigos utilizados.

El modelado y simulación de estos parámetros (tipo de código, longitud, modulación, efectos del canal sobre la señal ultrasónica), permite anticipar inconvenientes y la elección de aquellos esquemas de codificación y modulación que ofrezcan una mayor robustez de cara a la aplicación final.

1.2. Objetivos del TFG

El objetivo de este trabajo es proporcionar una plataforma que permita de una forma sencilla y versátil generar distintos códigos y modulaciones para su uso en aplicaciones prácticas y, al mismo tiempo, sirva como herramienta de simulación para comprobar los efectos más comunes sobre la señal ultrasónica codificada.

Este proyecto tiene gran utilidad para investigadores de esta área facilitando el análisis del diseño de señal a emitir, dado su importante impacto sobre el desempeño del sistema final en aplicaciones basadas en medidas de tiempos de vuelo o diferencias de tiempos de vuelo de señales ultrasónicas. A su vez, también es útil a nivel didáctico para alumnos que se inician en el tema. El trabajo puede fácilmente extrapolarse a otras áreas que demanden estrategias de codificación eficientes, como comunicaciones o radar.

1.3. Contexto

Este TFG se sitúa dentro del proyecto de investigación MICROCEBUS (RTI2018-095168-B-C51) y del proyecto de innovación docente “MEJORA DEL PROCESO DE ENSEÑANZA-APRENDIZAJE A TRAVÉS DE MATERIAL DE APOYO Y AUTO-ESTUDIO PARTICULARIZADO A LAS NECESIDADES FORMATIVAS DEL ESTUDIANTE” (ref. UAHEV1047).

1.4. Estructura del documento

El documento se divide en los siguientes capítulos:

El Capítulo 1 recoge el resumen del TFG, sus correspondientes objetivos y una visión global del trabajo.

El Capítulo 2 revisa las propiedades principales de los esquemas de codificación y modulación analizados, así como los conceptos básicos para la evaluación de los mismos.

El Capítulo 3 aborda el diseño y desarrollo de la interfaz generada en este proyecto, ofreciendo resultados y conclusiones finales.

El Capítulo 4 cuantifica el coste de desarrollo de este TFG mediante un presupuesto.

El Capítulo 5 corresponde al pliego de condiciones necesarias para el uso de la aplicación.

El Capítulo 6 contiene el manual de usuario de la interfaz desarrollada.

En el Capítulo 7 aparecen las referencias bibliográficas utilizadas durante el trabajo.

Por último, en el Capítulo 8 se adjuntan como anexo los códigos de las funciones generadas más relevantes.

2. ESQUEMAS DE CODIFICACIÓN Y MODULACIÓN ANALIZADOS

2.1. Conceptos básicos

Antes de revisar los códigos que permite analizar la herramienta, se repasarán algunos conceptos a los que se hará referencia a lo largo del TFG. Puede ampliarse información en [Pér09].

2.1.1. *Correlación periódica (AC y CC)*

La correlación entre dos señales se calcula con el fin de medir su grado de similitud. Por lo tanto, se puede decir que la correlación es una medida que establece la dependencia de una señal con otra (correlación cruzada) o consigo misma (autocorrelación).

Dado un conjunto de M (tamaño de familia) secuencias binarias de longitud L (longitud del código) siendo $\{A = am[\ell] \in \{-1, 1\}; 0 \leq m \leq M-1; 0 \leq \ell \leq L-1\}$, se define su AC (función de autocorrelación) discreta periódica cuando $m = s$ y su CC (función de correlación cruzada) periódica cuando $m \neq s$:

$$R_{am,as}[\tau] = \sum_{l=0}^{L-1} a_m[l]a_s[l + \tau] \quad (1)$$

Las secuencias son periódicas ya que $\{am\} = (\dots, am[0], am[1], \dots, am[L-1], am[0], am[1], \dots, am[L-1], \dots)$ y la suma $\ell + \tau$ se realiza en módulo L .

2.1.2. Correlación aperiódica (AC y CC)

La expresión de correlación aperiódica viene dada como:

$$C_{am,as}[\tau] = \begin{cases} \sum_{l=0}^{L-1-\tau} a_m[l]a_s[l+\tau], & 0 \leq \tau \leq L-1 \\ \sum_{l=0}^{L-1+\tau} a_m[l-\tau]a_s[l], & 1-L \leq \tau \leq 0 \\ 0, & |\tau| \geq L \end{cases} \quad (2)$$

Cuando $m = s$ se obtiene la AC aperiódica, y cuando $m \neq s$ la CC aperiódica. Con objeto de simplificar, sólo se considera la función en el rango $0 \leq \tau \leq L-1$.

2.1.3. Códigos ortogonales, códigos ortogonales generalizados y códigos incorrelados

Se dice que dos secuencias son ortogonales si cumplen:

$$R_{am,as}[\tau] = \begin{cases} L, & \tau = 0, \quad m = s \\ 0, & \tau = 0, \quad m \neq s \end{cases} \quad (3)$$

Según la expresión anterior, la CC es nula únicamente para el desplazamiento cero. Sin embargo, es difícil mantener sincronizados emisores y receptores. Posibles reflexiones (efecto de multicamino) provocan la recepción de la señal emitida en distintos instantes de tiempo y la pérdida de la ortogonalidad entre secuencias.

Como solución a este problema, en entornos que no requieren un sincronismo estricto (cuasi-síncronos), se añaden dos conceptos: ortogonalidad generalizada y cuasi-ortogonalidad generalizada.

Dos secuencias son ortogonales generalizadas si tienen las siguientes características:

$$R_{am,as}[\tau] = \begin{cases} \eta L, & \tau = 0 & m = s \\ 0, & 1 < |\tau| < ZCZ & m = s \\ 0, & 0 < |\tau| < ZCZ & m \neq s \end{cases} \quad (4)$$

$$\eta = \frac{1}{L} \sum_{l=0}^{L-1} |a_m[l]| \leq 1 \quad (5)$$

Existe una ZCZ (zona de correlación cero) alrededor del origen donde las funciones de correlación tienen valor cero. Las secuencias presentan mejores características cuanto mayor es dicha zona. Por el contrario, si $ZCZ = 0$, se obtienen secuencias ortogonales convencionales. En definitiva, la ZCZ indica el grado de ortogonalidad generalizada de las secuencias.

Cuando $am \in \{-1, 1\}$, la eficiencia energética η valdrá 1 y en el caso de que haya valores nulos en la secuencia, esto es $am \in \{-1, 0, 1\}$, será menor que la unidad.

En las secuencias cuasi-ortogonales generalizadas existe una zona reducida de interferencias (Wq) alrededor del origen en las que las interferencias no son cero, sino que su valor está por debajo de un umbral ϵ aceptable por el sistema. Gracias a esto, se consigue aumentar el número de secuencias que cumplen esta condición.

El sistema que define las secuencias cuasi-ortogonales es el siguiente:

$$R_{am,as}[\tau] = \begin{cases} \eta L, & \tau = 0, & m = s \\ \leq \epsilon, & 1 < |\tau| < Wq, & m = s \\ \leq \epsilon, & 0 < |\tau| < Wq, & m \neq s \end{cases} \quad (6)$$

Dentro de la aplicación desarrollada, estas propiedades de ortogonalidad generalizada se aprecian en las secuencias LS.

Se dice que las secuencias se encuentran incorreladas cuando la CC es nula para cualquier desplazamiento. No obstante, a día de hoy, no existen familias de códigos unitarios que cumplan con estas dos propiedades simultáneamente [GPÁ⁺15] [Wel74].

La Fig. 2. resume los conceptos de correlación explicados anteriormente.

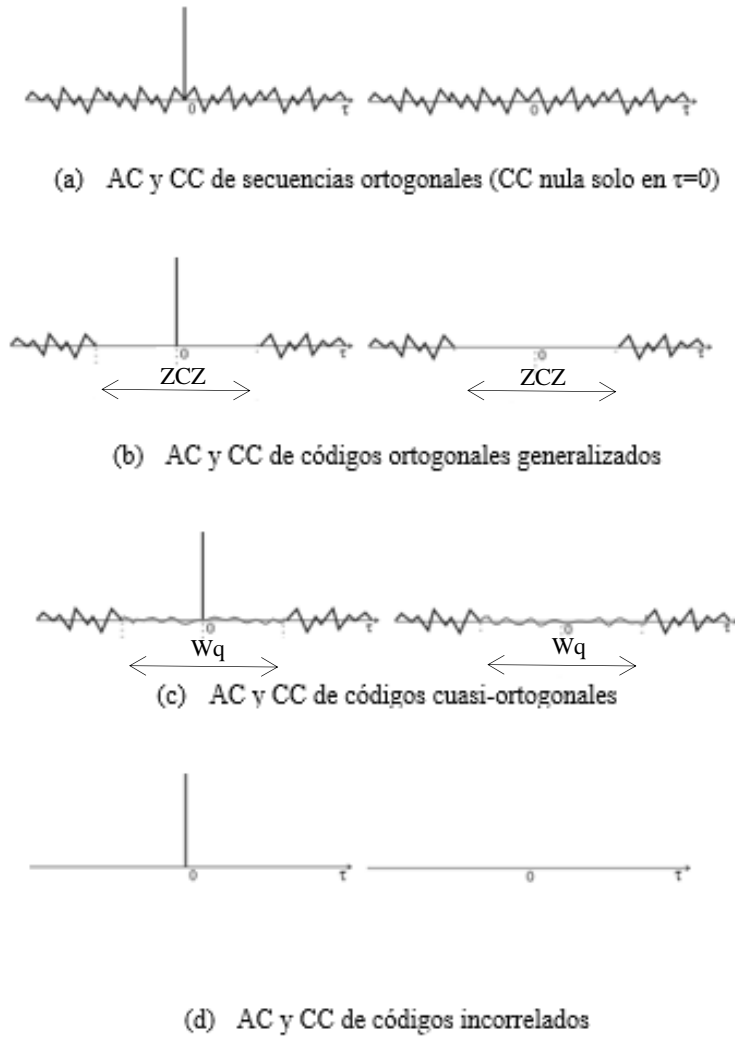


Fig. 2. Funciones de correlación según tipo de códigos

2.1.4. Cotas de correlación AC (Θ_{ac}) y CC (Θ_{cc})

Tanto en correlación periódica como en aperiódica, el valor del pico principal de autocorrelación es el mismo. Esto se debe a que, para el desfase cero, el valor de las funciones de correlación periódica y aperiódica coinciden ($C_{am,as}(0) = R_{am,as}(0)$ y $C_{am}(0) = R_{am}(0) = \sum_{l=0}^{L-1} a_m[l]^2$).

El cálculo del valor máximo de interferencia en la correlación o *bound* θ se emplea para evaluar la bondad de una familia de códigos.

$$\theta = \max\{\theta_{AC}, \theta_{CC}\} \quad (7)$$

θ_{ac} representa el valor del máximo pico lateral obtenido de entre todas las autocorrelaciones de las M secuencias de la familia y θ_{cc} el pico máximo de correlación cruzada entre ellas.

$$\theta_{AC} = \max \left\{ \frac{|C_{am,am}[\tau]|}{C_{am,am}[0]}; \forall m \in [0, \dots, M-1]; \forall \tau \neq 0 \right\} \quad (8)$$

$$\theta_{CC} = \max \left\{ \frac{|C_{am,as}[\tau]|}{C_{am,am}[0]}; \forall m, s \in [0, \dots, M-1]; \forall \tau; m \neq s \right\} \quad (9)$$

Tanto la ecuación (8) como la (9) sirven para el caso periódico y aperiódico, cambiando $R_{am,am}$ por $C_{am,am}$ y $R_{am,as}$ por $C_{am,as}$.

2.1.5. Sidelobe to Mainlobe Ratio (SMR)

Este concepto es similar al cálculo de la cota θ_{ac} pero en la señal se tienen en cuenta los efectos de la modulación, del transductor empleado y los efectos del canal de transmisión.

2.2. Revisión de códigos

2.2.1. Secuencias Kasami

Son secuencias binarias que presentan un comportamiento similar al de una secuencia completamente aleatoria, aunque son generadas de forma determinista (pseudo-aleatorias) [Pér09] [Kas68]. El grado de aleatoriedad lo marcan los siguientes criterios:

- La AC periódica en cualquier instante $\tau \neq 0$ debe ser una constante c de bajo valor:

$$R_A[\tau] = \begin{cases} L, & \tau = 0 \\ c, & \tau \neq 0 \end{cases} \quad (10)$$

En donde $\{A = a[\ell] \in \{-1, 1\}; 0 \leq \ell \leq L-1\}$.

- En cada período el número de 1 no debe exceder en más de una unidad al número de -1 , esto es, $|\sum_{l=0}^{L-1} a[l]| \leq 1$.
- En cada período la mitad de los 1 ó -1 consecutivos tienen longitud 1, un cuarto tienen longitud 2, un octavo longitud 3, etc. Además, hay el mismo número de 1 consecutivos que de -1 .

Generación códigos

Estas secuencias tienen un mecanismo de generación muy sencillo. Se parte de una secuencia- m [Pér09] m_1 de longitud $L=2^N-1$ obtenida a partir de un LFSR (Registro de Desplazamiento Realimentado Lineal) siendo N un número par. Se diezma la secuencia m_1 por un factor $q = 2^{\frac{N}{2}} + 1$ generando una segunda secuencia de período $2^{\frac{N}{2}} - 1$. Concatenando q veces la secuencia diezmada se obtiene una nueva secuencia m_2 de longitud $L=2^N-1$. El conjunto de secuencias Kasami está formado por m_1 y la suma módulo-2 de m_1 con cualquier versión desfasada de m_2 :

$$K = \{m_1, m_1 \oplus m_2, m_1 \oplus Dm_2, m_1 \oplus D^2m_2, \dots, m_1 \oplus D^{L'-1}m_2\} \quad (11)$$

Donde $L' = 2^{\frac{N}{2}} - 1$ representa la longitud de la secuencia m_1 diezmada por el factor q . Por lo tanto, el conjunto contiene $M = 2^{\frac{N}{2}}$ secuencias.

En la Fig. 3. se ofrece un ejemplo de las secuencias Kasami generadas para una $L=63$ resultando una $M=8$.

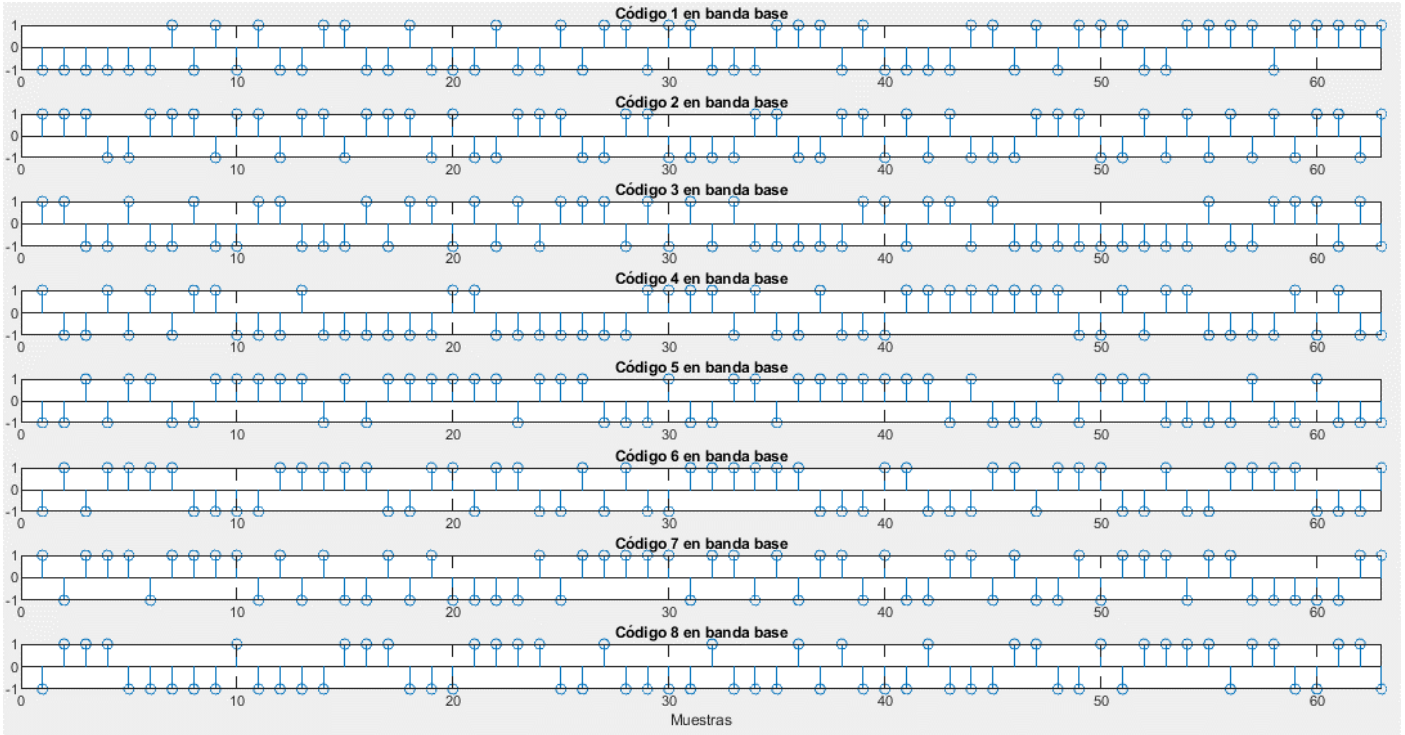


Fig. 3. Secuencias Kasami 63 bits

Propiedades de correlación periódica

Las AC y CC periódicas de estas secuencias toman valores en el rango:

$$R_{k_1, k_2}[\tau] = \begin{cases} L, & \tau = 0, & k_1 = k_2 \\ \{-1, -s(N), s(N) - 2\}, & \tau \neq 0, & k_1 = k_2 \\ \{-1, -s(N), s(N) - 2\}, & \forall \tau, & k_1 \neq k_2 \end{cases} \quad (12)$$

En donde $s(N) = -1 + 2^{\frac{N}{2}}$.

El siguiente ejemplo muestra la AC y la CC de una secuencia Kasami de 63 bits generada desde la aplicación desarrollada:

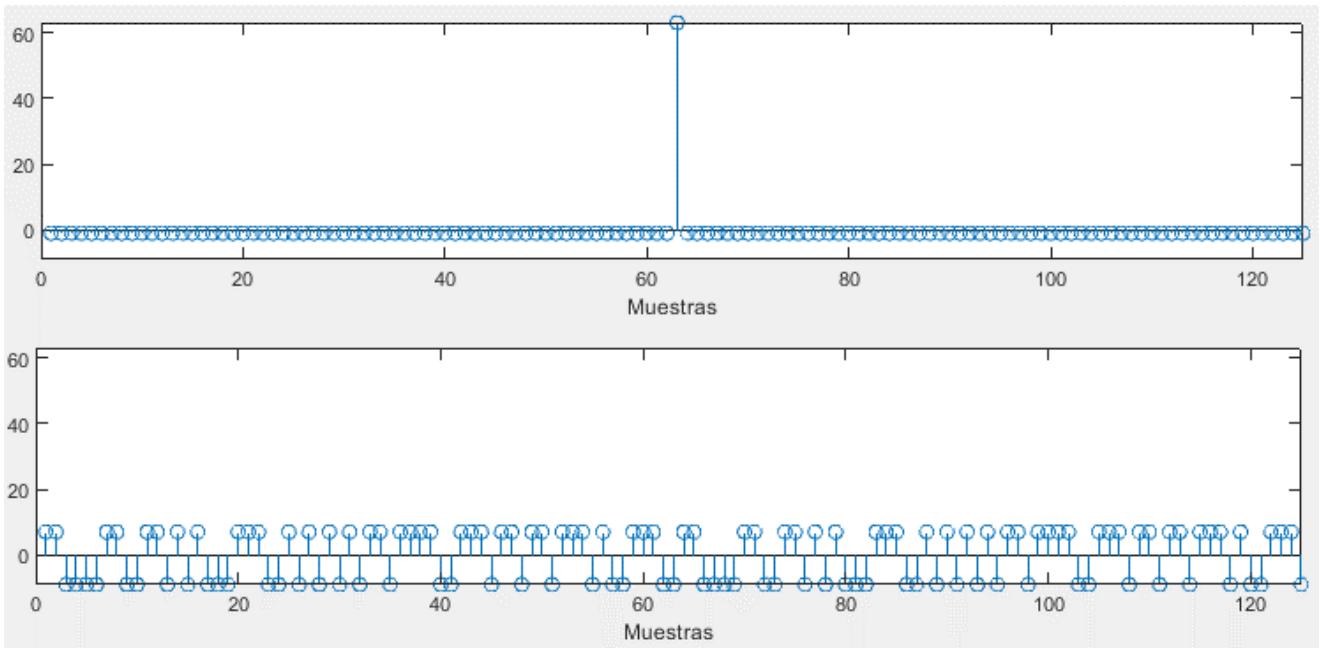


Fig. 4. AC y CC Secuencia Kasami 63 bits periódica

Comportamiento en aperiódico

Al igual que ocurre con otras secuencias pseudoaleatorias, cuando la emisión se realiza de modo aperiódico las propiedades de correlación se degradan y los valores obtenidos no se ciñen a los de la expresión (12).

En la Fig. 4. aparece el ejemplo anterior aplicado al caso aperiódico.

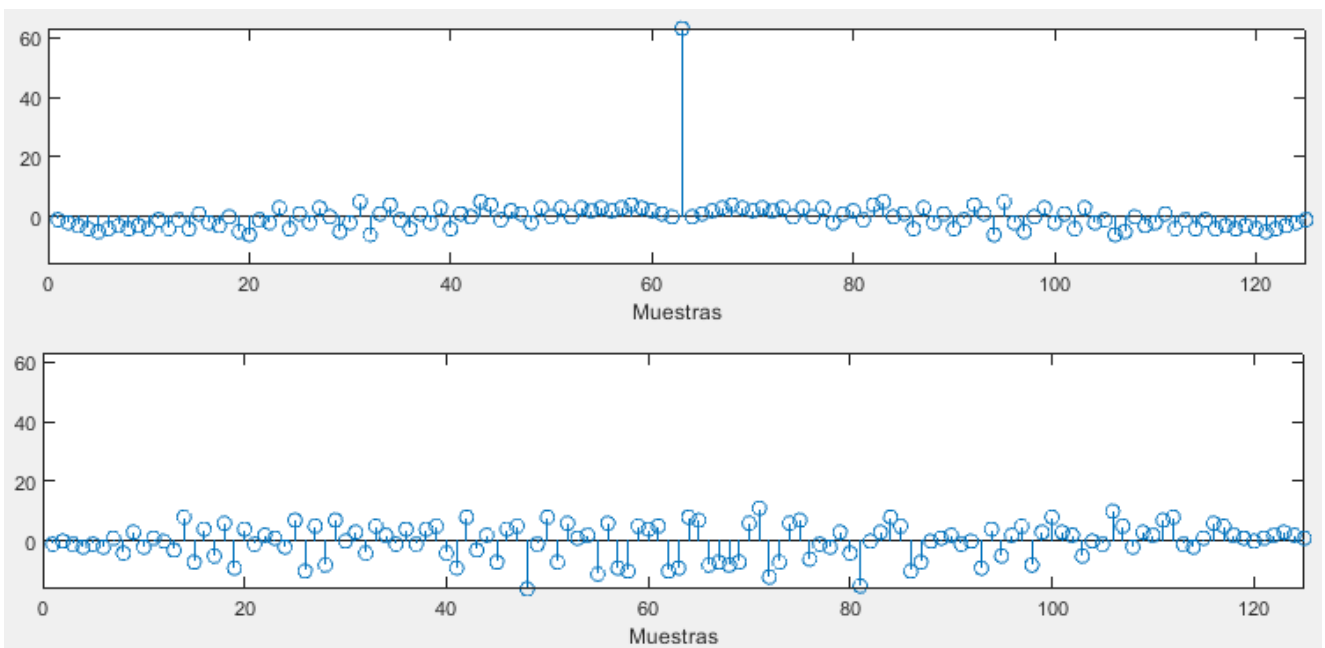


Fig. 5. AC y CC Secuencia Kasami 63 bits aperiódica

2.2.2. Conjuntos de secuencias complementarias

Usando códigos pseudo-aleatorios es posible tener bajas correlaciones cruzadas. Frente a ello, los conjuntos de M -CSS (M Secuencias Complementarias) logran correlaciones nulas sumando las funciones de correlación de las secuencias que componen dicho conjunto.

Las parejas Golay se obtienen cuando $M=2$ de longitud $L = 2^a$ con $a \in \mathbb{N} - \{0\}$. Si dos parejas de códigos Golay son completamente incorreladas entre sí, es decir, que la suma de las correlaciones entre las correspondientes secuencias de cada par es cero para cualquier desplazamiento, sería posible la emisión simultánea de dos señales sin que haya interferencias entre ellas [Pér09] [TL72].

Se llama SACF y SCCF a las funciones de autocorrelación y correlación cruzada, respectivamente, resultantes de sumar las distintas correlaciones que componen el código.

En la Fig. 6 y 7 se muestran las propiedades de parejas Golay con una $L=32$, donde $AC C_i s_k$ hace referencia a la función AC de la secuencia s_k del código C_i ; y $CC C_i, C_j s_k$ es la función de CC de las secuencias s_k de los códigos C_i y C_j (con $i \neq j$).*

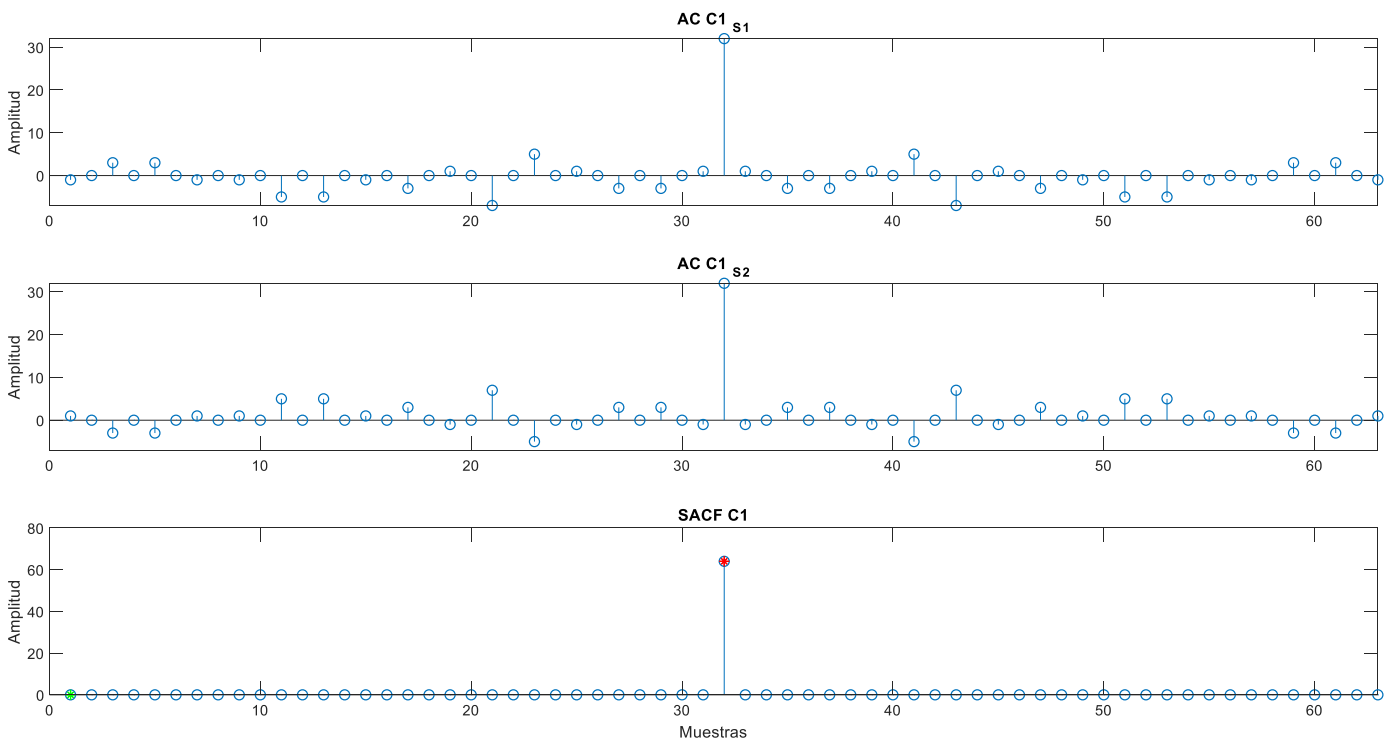


Fig. 6. Propiedades de auto-correlación parejas Golay del código C1

* Esta nomenclatura es la utilizada en la aplicación desarrollada para la representación de los distintos códigos, por lo que, a partir de aquí, se sobreentiende su significado. Sólo los conjuntos CSS estarán compuestos por códigos con secuencias, el resto aparecerá de la forma $AC C_i$ para la auto-correlación y $CC C_i, C_j$ para la correlación-cruzada.

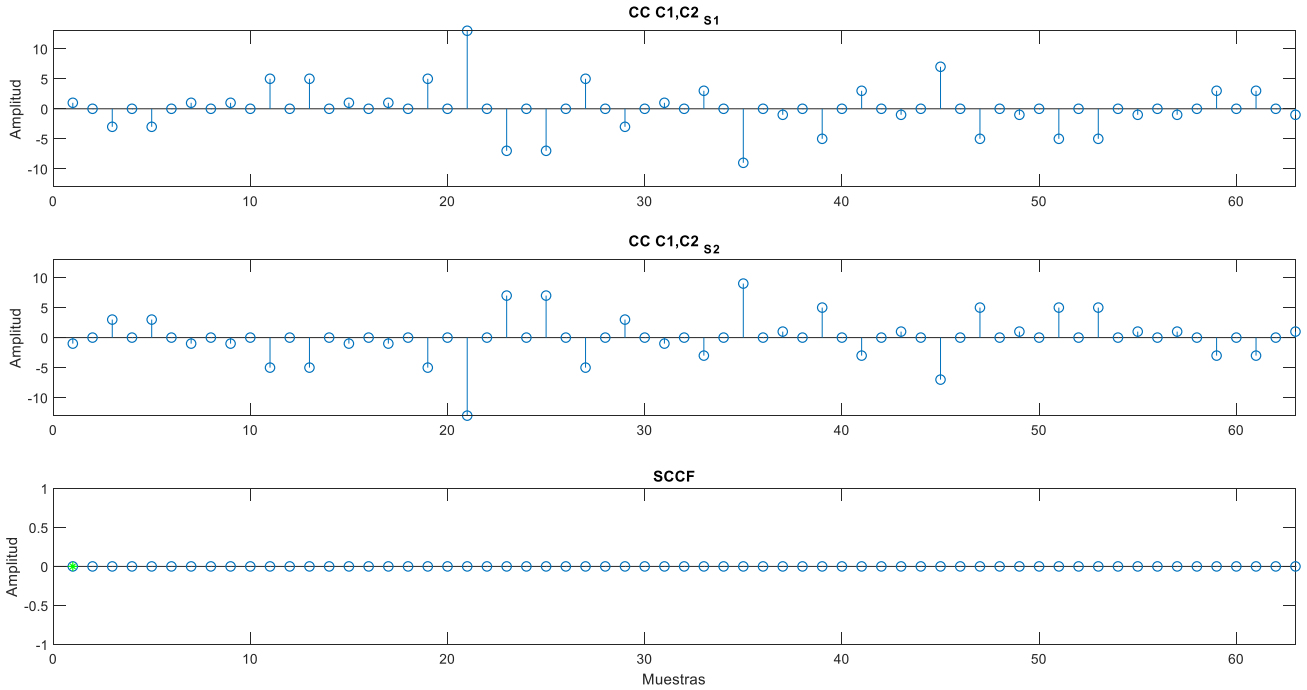


Fig. 7. Propiedades de correlación cruzada entre parejas Golay del código 1 con el código 2

Generación de códigos

Los conjuntos M -CSS constituyen una generalización de las parejas Golay, pudiendo componerse con un número de secuencias M mayor de dos. Esto provoca que la gran ventaja frente a las Golay es que las CSS disponen de hasta M conjuntos mutuamente incorrelados entre sí [Pér09].

Un conjunto C_i con M secuencias binarias s_j de longitud L , $C_{i,s_j}[l] \in \{-1,1\}$; $0 \leq l \leq L-1$; $1 \leq i,j \leq M$, es complementario si la suma de su SACF es nula para todos los desplazamientos distintos de cero, y tiene un valor máximo de $M \cdot L$ para el caso de un desplazamiento nulo.

$$SACF_i = \sum_{j=1}^M AC_{C_{i,s_j}}[\tau] = M \cdot L \cdot \delta[\tau] = M^{N+1} \cdot \delta[\tau] \quad (13)$$

En donde $\delta[\tau]$ es la función delta de Kronecker, y $L = M^N$ la longitud de las $M = 2^m$ secuencias que componen el conjunto, pudiendo ser m y N cualquier valor natural a excepción del cero.

Dados dos CSS, C_i y C_k , se dice que uno es compañero del otro si se cumplen las siguientes propiedades:

- Ambos conjuntos tienen el mismo número de secuencias, M en este caso, y todas tienen la misma longitud L .
- La SCCF de las secuencias de ambos conjuntos es nula para cualquier desplazamiento:

$$SCCF_i = \sum_{j=1}^M CC_{C_i, C_k, s_j}[\tau] = 0 \quad \forall \quad (14)$$

Propiedades de correlación aperiódica

En la siguiente figura se visualizan las AC y SACF para el código $C1$ de una M -CSS configurada para $M=4$ y $L=64$.

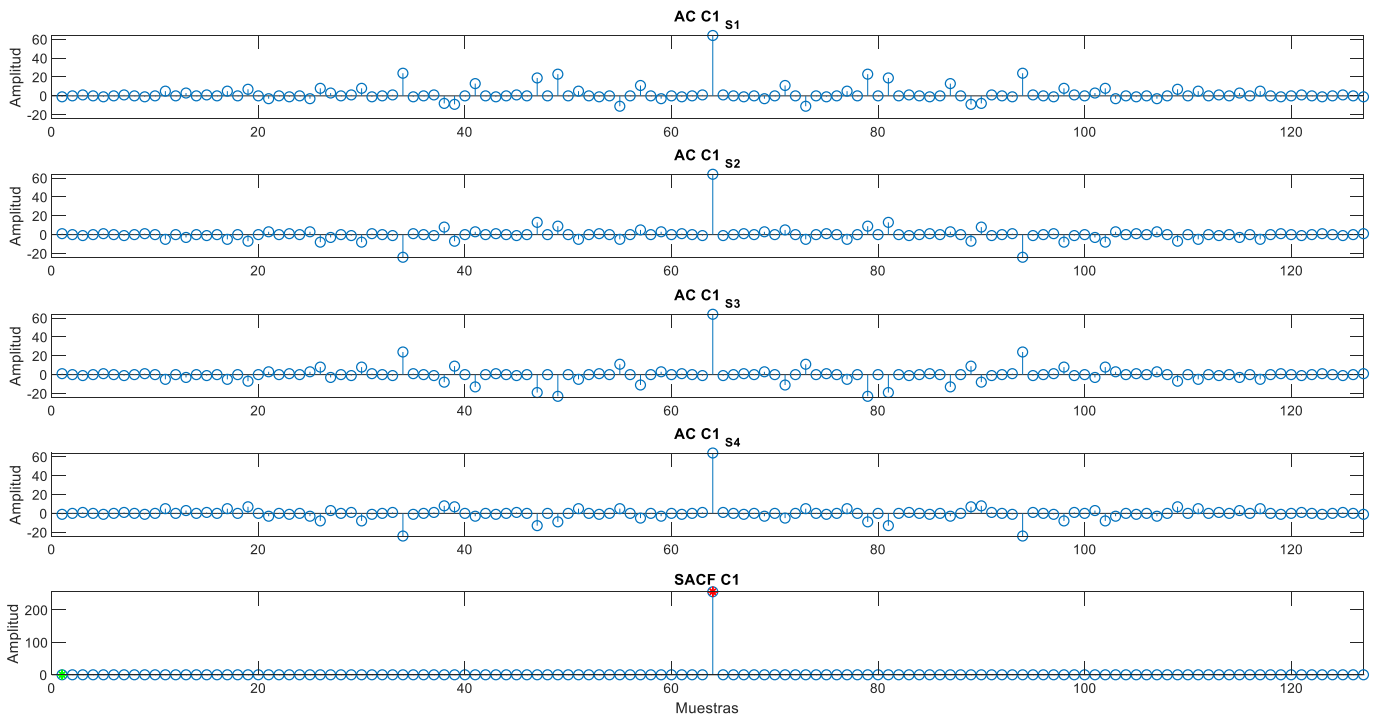


Fig. 8. AC y SACF para código 1 secuencia M -CSS $M=4$ y $L=64$

La Fig. 9 se compone del caso particular de las CC y SCCF del código $C1$ con el código $C4$, considerando dos CSS compañeros generados con $M=4$ y $L=64$.

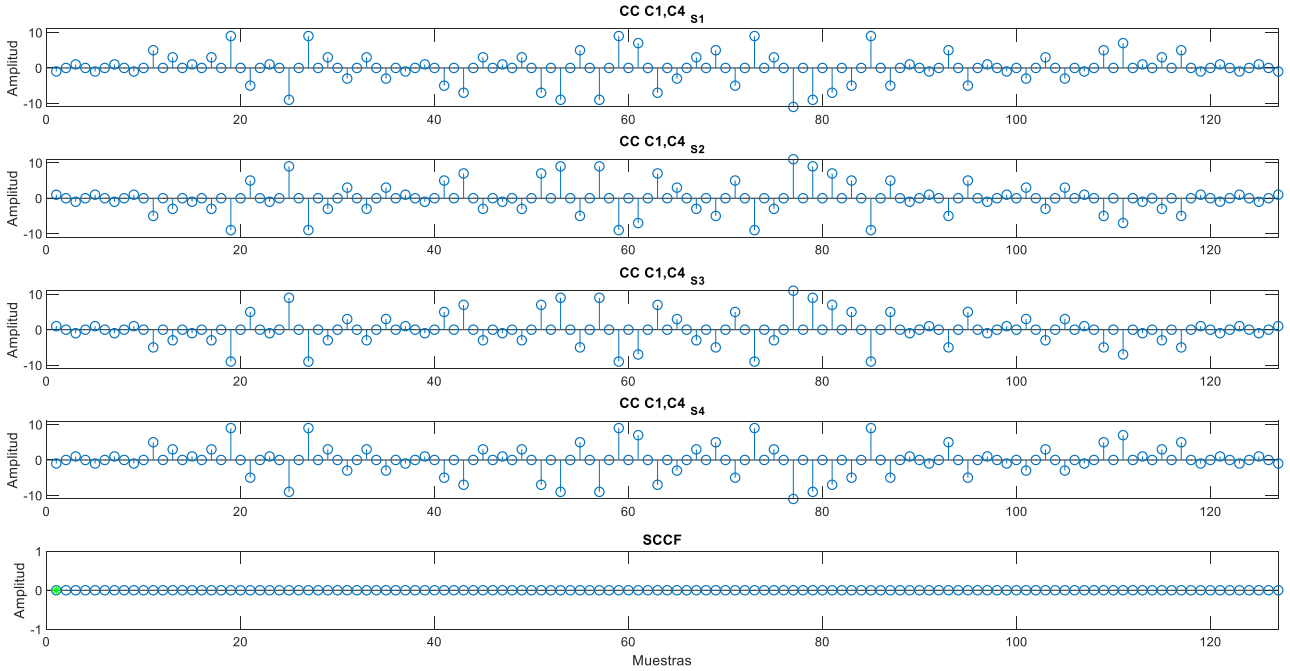


Fig. 9. CC y SCCF para código 1 con código 4 secuencia M-CSS M=4 y L=64

Mecanismos de ordenación de bits para su transmisión: pérdida de propiedades

Para generar la transmisión de la señal, se necesita unificar las M secuencias que forman cada código de M-CSS. Por ello, la aplicación consta de dos mecanismos de ordenación de bits: concatenación y entrelazado.

El proceso de concatenación conlleva la organización de las secuencias una tras otra. De esta manera la señal correspondiente al código 1 estará compuesta de su secuencia 1, seguida de su secuencia 2 hasta llegar a la secuencia M . La Fig. 10 muestra un ejemplo para una $M=2$ y $L=8$.

Concatenación: (15)

$$\{C_{i,s1}[0] \dots C_{i,s1}[L-1] \quad C_{i,s2}[0] \dots C_{i,s2}[L-1] \quad \dots \quad C_{i,sM}[0] \dots C_{i,sM}[L-1]\}$$

El entrelazado consiste en generar la señal del código 1 añadiendo el primer valor de todas sus secuencias por orden, después su segundo valor y así hasta llegar al último valor. En la Fig. 11 se refleja, para la misma configuración anterior, este mecanismo de ordenación.

Entrelazado: (16)

$$\{C_{i,s1}[0] \quad C_{i,s2}[0] \quad \dots \quad C_{i,sM}[0] \quad \dots \quad C_{i,s1}[L-1] \quad C_{i,s2}[L-1] \quad \dots \quad C_{i,sM}[L-1]\}$$

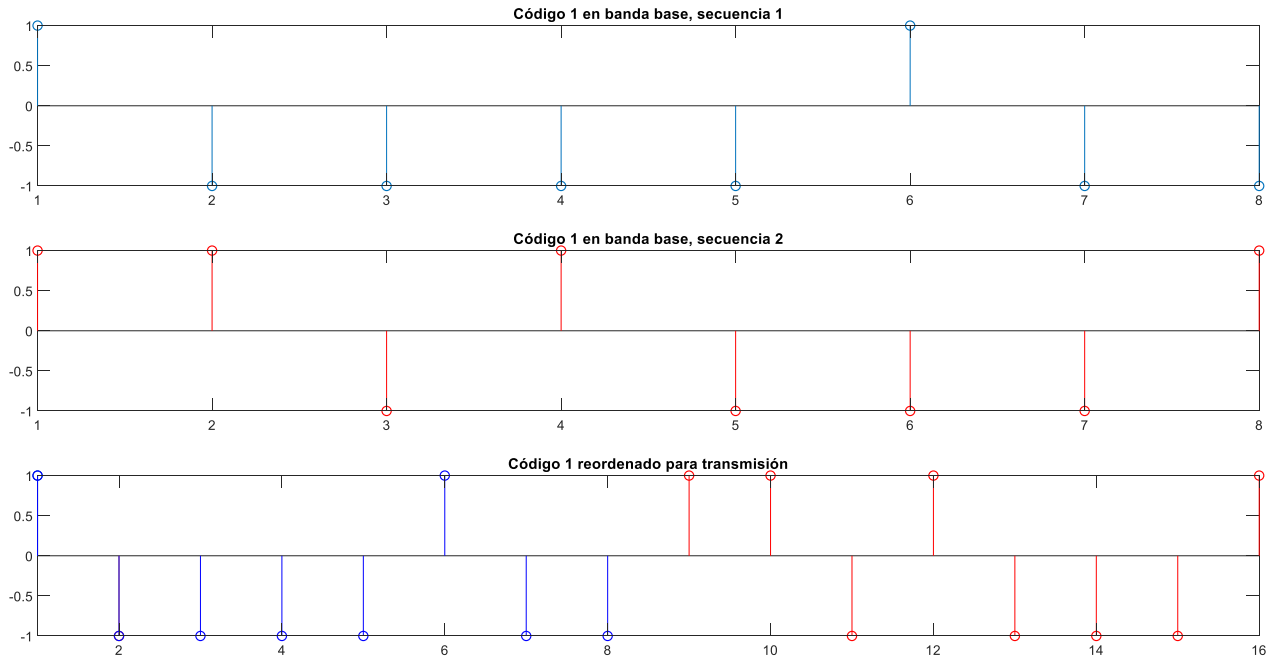


Fig. 10. Concatenación código CSS

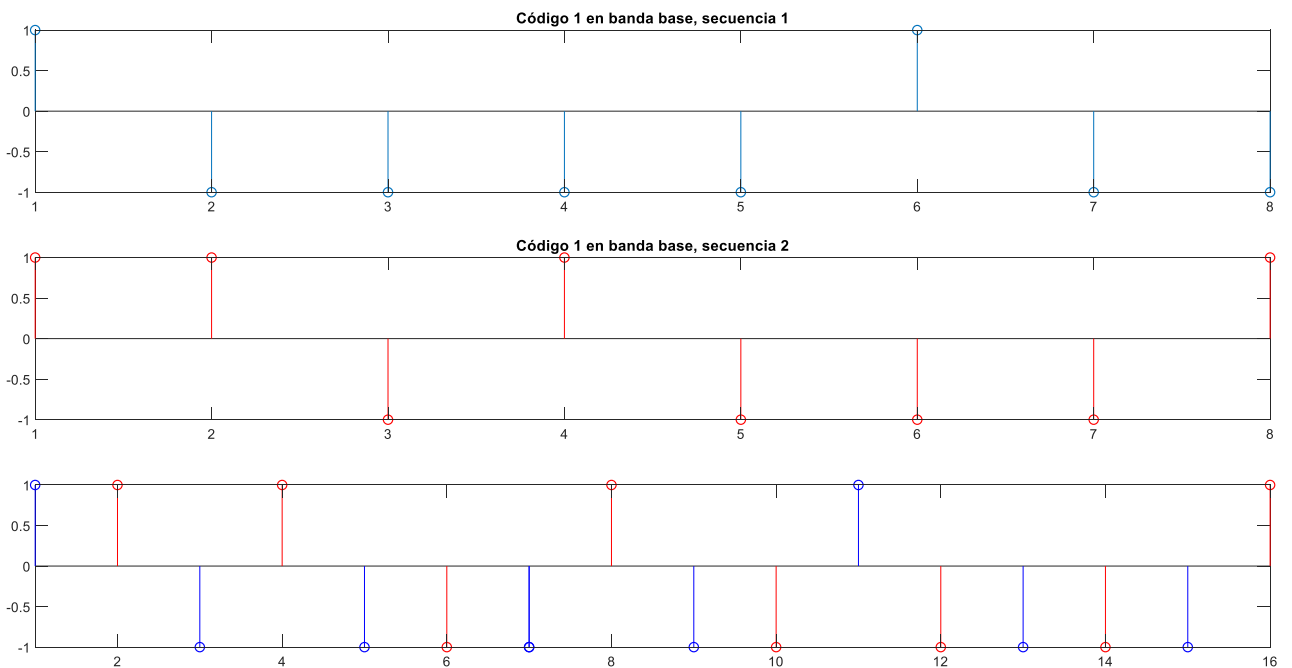


Fig. 11. Entrelazado código CSS

Cuando las distintas secuencias se unifican, estos códigos pierden sus propiedades. Sin embargo, se han incluido en la aplicación desarrollada ya que son la base de generación de muchos códigos con ventanas libres de interferencias (ortogonales generalizados), como por ejemplo los LS estudiados en este trabajo.

2.2.3. Códigos LS

Los LS son códigos ortogonales generalizados. Esto es que, en sus funciones de correlación, tienen una ZCZ alrededor del origen.

Generación de códigos a partir de las CSS

Un conjunto de $M = 2^n$, $n \in \mathbb{N}$, códigos LS de longitud L $\{C = c_m[l]; 0 \leq m \leq M-1; 0 \leq l \leq L-1\}$ compuesto por elementos $\{-1,0,1\}$ puede generarse a partir de dos parejas Golay $G_1 = (G_{1,s_1}, G_{1,s_2})$ y $G_2 = (G_{2,s_1}, G_{2,s_2})$ de longitud L_0 e incorreladas entre sí, según la siguiente ecuación:

$$C[z] = \sum_{i=1}^{M/2} h_{m,i} \cdot z^{-(i-1)L_0} \cdot [s_{\pi_i,0}(z) + z^{-\left(\frac{M}{2}L_0+W\right)} s_{\pi_i,1}(z)] \quad (17)$$

$C_m(z)$ representa la transformada Z de $c_m[\tau]$; $h_{m,i} \in \{-1,1\}$ son los elementos de una matriz de Hadamard de tamaño $(M/2 \times M/2)$; el vector $\pi = [\pi_1, \pi_2, \dots, \pi_{M/2}] \in \{0,1\}$ denota la representación binaria de un número natural cualquiera n , $0 \leq n < 2^{M/2}$, es decir, $n = \sum_i \pi_i 2^i$ y, finalmente, $W \leq L_0 - 1$ es el tamaño de la semi-ventana de ZCZ (normalmente se usa $W = L_0 - 1$). Los primeros $M/2$ códigos LS se generan concatenando las parejas Golay con la polaridad indicada por los coeficientes $h_{m,i}$ de la matriz de Hadamard, y siguiendo el orden establecido por π_i . Para obtener los siguientes $M/2$ códigos, es necesario sustituir en la ecuación (15) el vector π por su complementario $\pi^* = [\pi^*_1, \pi^*_2, \dots, \pi^*_{M/2}]$, $\pi^*_m = \pi_m + 1 \pmod{2}$; $1 \leq m \leq M/2$. Los códigos LS obtenidos tienen una longitud $L = ML_0 + W$; aunque, si las emisiones son periódicas, es necesario insertar un intervalo de guarda de, al menos, W ceros. En estos casos $L = ML_0 + 2W$, como queda ilustrado en la Fig. 12.

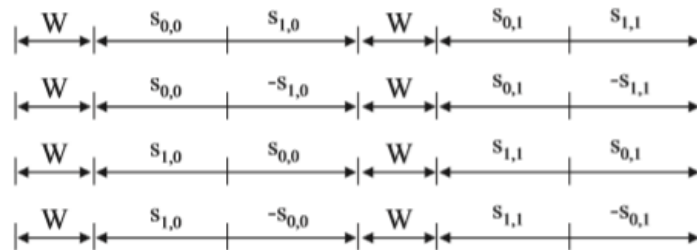


Fig. 12. Estructura de $M=4$ códigos LS caso periódico [Pér09]

La Fig. 13 representa los códigos generados desde la aplicación para el caso de $M=4$, $L_o=16$ ($W=15$) aperiódico.

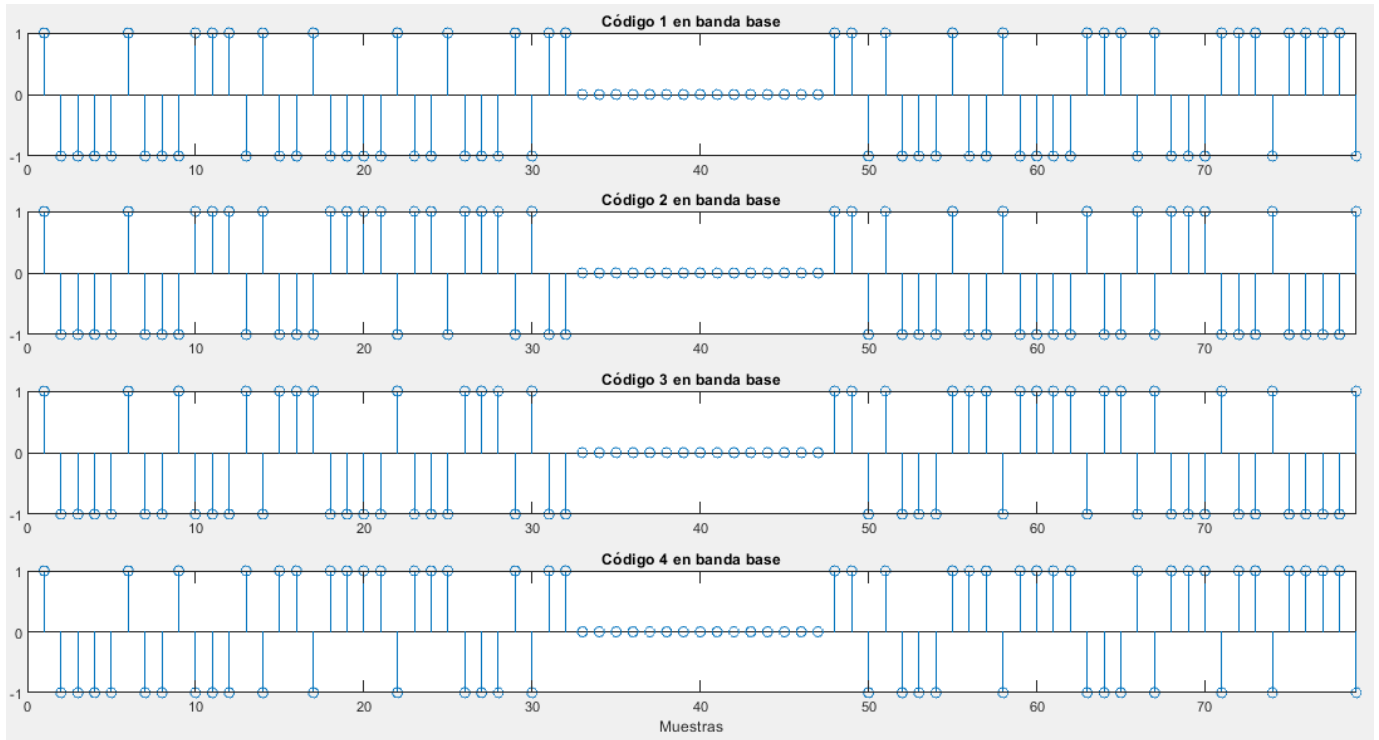


Fig. 13. Códigos LS con $M=4$, $L_o=16$ aperiódico

Propiedades de correlación

Los valores de correlación obtenidos fuera de la ZCZ de los códigos LS son mayores a los obtenidos en los CSS. De ahí, la importancia de asegurar que la diferencia de tiempos máxima entre recepciones sea menor que W .

En las Fig. 14 y 15 se visualizan las AC y las CC respectivamente de los C_i ($1 \leq i \leq M$) códigos LS generados para la Fig. 13. El valor del pico máximo de auto-correlación es $M \cdot L_o$.

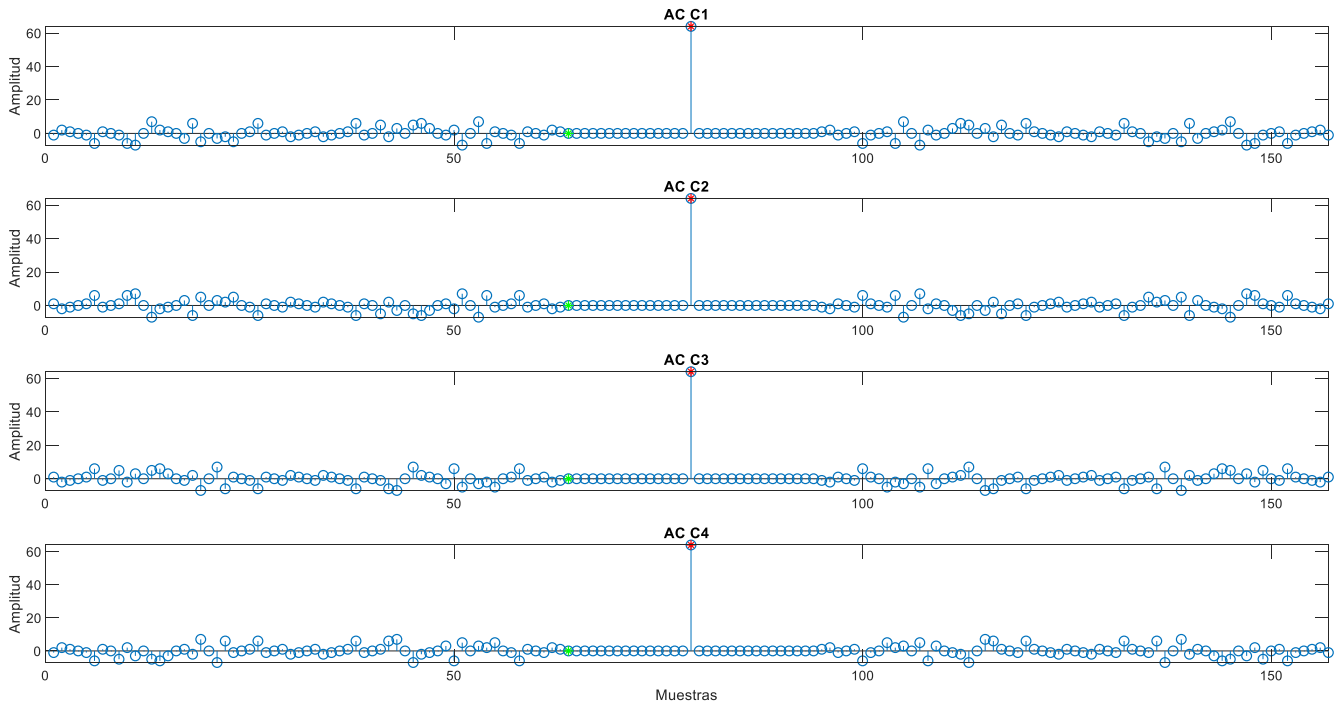


Fig. 14. AC códigos LS $M=4$, $L_o=16$

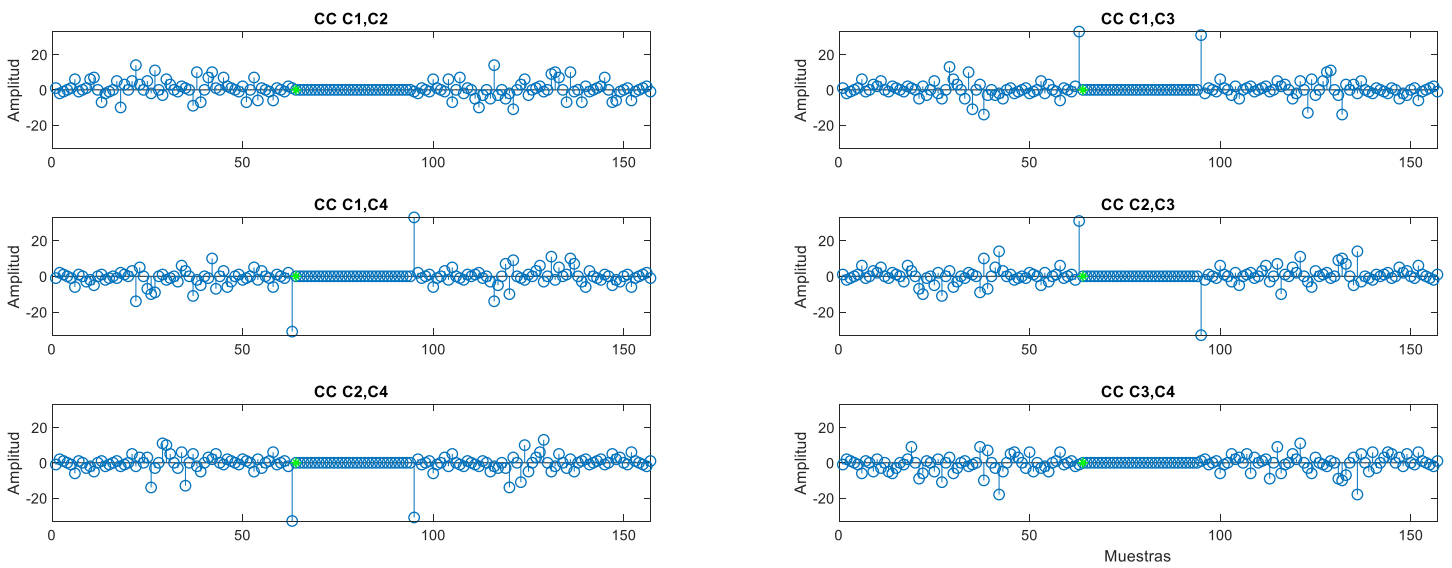


Fig. 15. CC Códigos LS $M=4$, $L_o=16$

2.2.4. Códigos Zadoff-Chu

Las secuencias Zadoff-Chu, son códigos polifásicos y complejos de módulo unitario [MPU⁺18] [SBH01] usados para la sincronización de símbolos.

La principal diferencia frente a los otros códigos analizados es que son complejos en lugar de binarios.

Generación de códigos (semilla)

Dicha secuencia Z_r de longitud L se puede definir de la siguiente manera:

$$Z_r[l] = e^{-i\frac{2\pi r l(l+1)}{2L}}, \quad 0 \leq l \leq L - 1 \quad (18)$$

La longitud del código L es un número impar y r es un número entero conocido como índice raíz o semilla.

La secuencia va a tener un total de $L-1$ semillas. En función del *Nemo* (Número de Emisores Simultáneos) que se escojan a través de la interfaz, se seleccionarán por orden ascendente aquellas semillas que tengan el pico de correlación principal menor.

La Fig. 16 representa, para una $L=61$ y $Nemo=5$, el valor absoluto de las 5 semillas complejas con mejores propiedades de correlación.

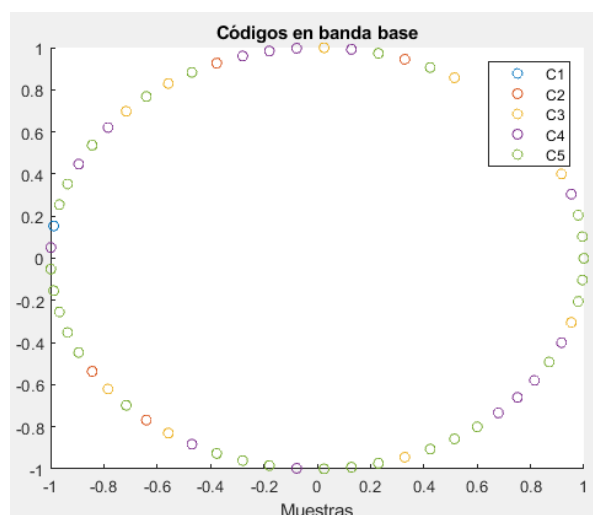


Fig. 16. C1,C2,C3,C4 y C5 en valor absoluto

Propiedades de correlación

La AC periódica no presenta lóbulos laterales, por lo que viene definida como:

$$R_{Z_{rk}}[\tau] = \begin{cases} L, & \tau \bmod L = 0 \\ 0, & \tau \bmod L \neq 0 \end{cases} \quad (19)$$

Si L es un número impar, la CC presentará valores bajos. Un estudio previo de estos códigos propone que L sea un número primo para obtener mejores propiedades de correlación [MPU⁺18].

En las siguientes figuras se muestra un ejemplo de las propiedades de correlación de los códigos Zadoff-Chu para una $L=61$ y $Nemo=5$ (Semilla de $r=60$).

Al igual que pasa en los códigos Kasami, en el caso aperiódico no se mantienen las propiedades de correlación indicadas.

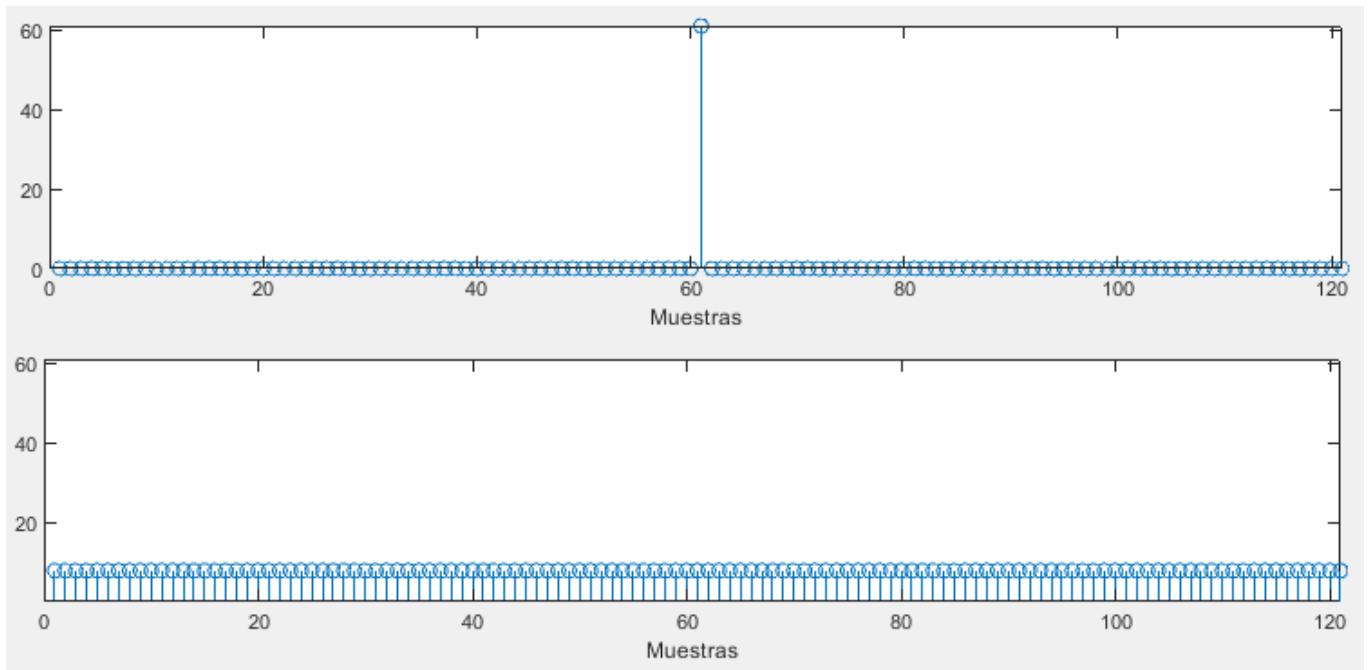


Fig. 17. AC Código 1 y CC Código 1-Código 2 Secuencias Zadoff-Chu $L=61$ periódica

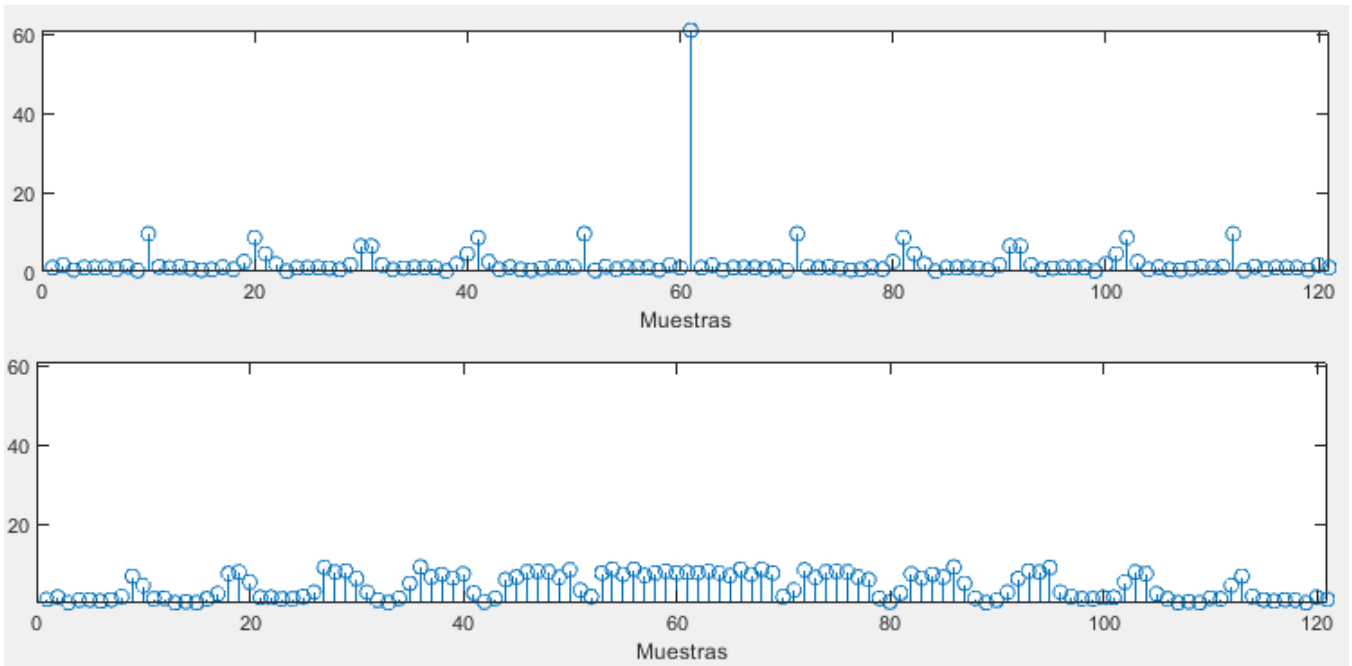


Fig. 18. AC Código 1 y CC Código 1-Código 2 Secuencias Zadoff-Chu $L=61$ aperiódica

FFT

Una peculiaridad de estos códigos es que, al aplicar la FFT (*Fast Fourier Transform*) sobre la señal en banda base, el resultado obtenido es otra secuencia Zadoff-Chu compleja pero con módulo distinto a uno.

Para el caso anterior, la siguiente imagen resulta de aplicar la FFT al código 1. Se puede observar que el módulo ha cambiado a un valor de aproximadamente 7,5.

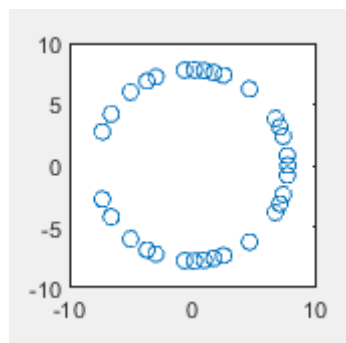


Fig. 19. FFT Código 1 Zadoff Chu $L=61$, $Nemo=5$

2.3. Modulación

Para ajustar la señal al ancho de banda del transductor, es necesaria la modulación de los códigos. Se utilizan dos técnicas distintas en función su naturaleza.

2.3.1. BPSK

Esta modulación es aplicada en los códigos Kasami, CSS y LS por su composición binaria. La señal portadora toma dos posibles fases: 0° y 180° .

Dicha técnica cumple la siguiente función de modulación:

$$x(t) = \begin{cases} A \sin(2\pi fct), & \text{Binario "1"} \\ A \sin(2\pi fct + \pi), & \text{Binario "0"} \end{cases} \quad (20)$$

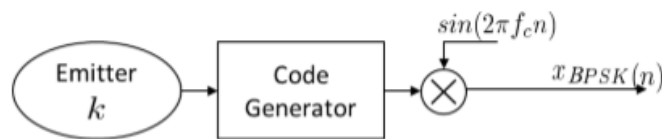


Fig. 20. Modulador BPSK [GPÁ+15]

Para un código Kasami de $L=63$ y $Nemo=4$, se configura la señal portadora con una $f_c=41,667kHz$ (frecuencia de portadora), $f_{se}=500kHz$ (frecuencia de muestreo) y $N_c=2$ (número de ciclos de la portadora). Como resultado, se obtiene la señal modulada que aparece en la Fig. 23.

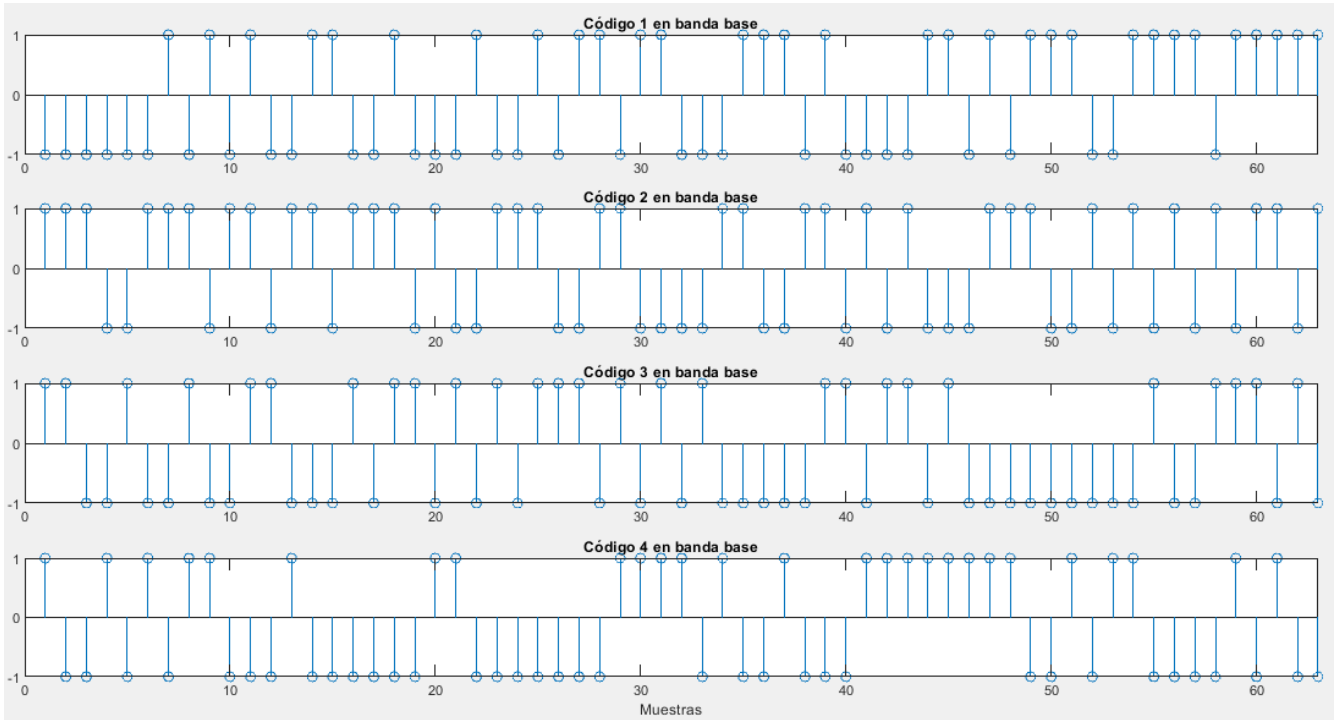


Fig. 21. Código Kasami $L=63$, $Nemo=4$

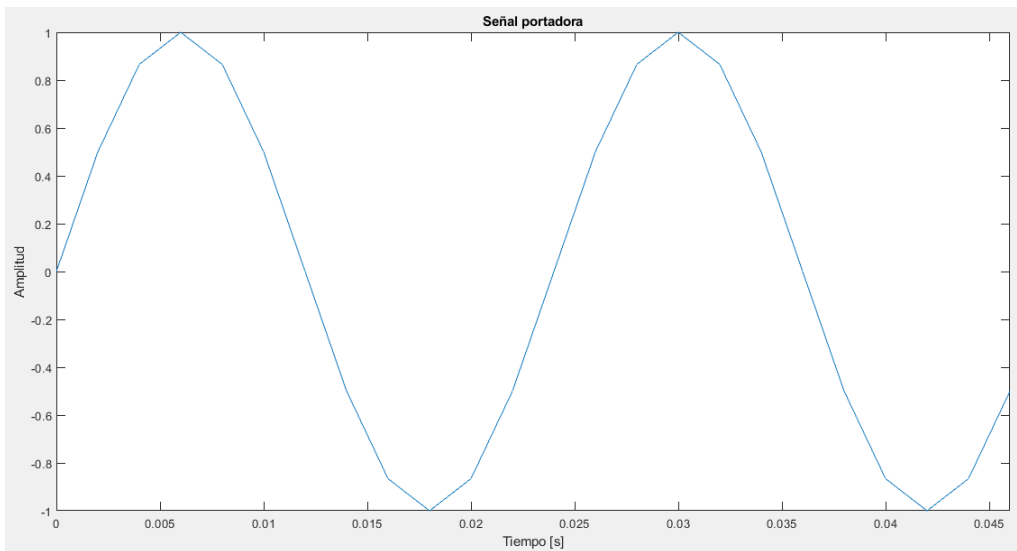


Fig. 22. Señal portadora BPSK $f_c=41,667\text{kHz}$, $f_{se}=500\text{kHz}$ y $N_c=2$

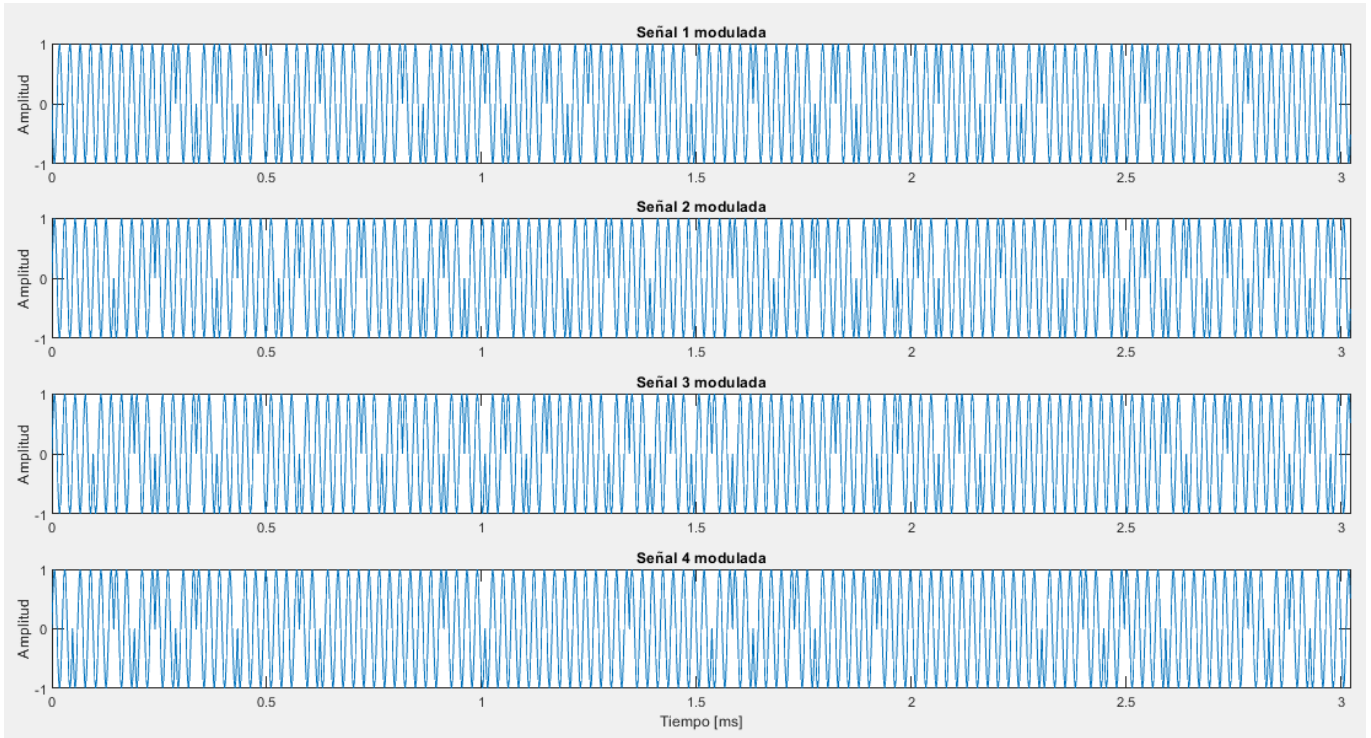


Fig. 23. Señal Kasami modulada resultante

2.3.2. QPSK

Es la técnica usada en los códigos Zadoff-Chu al estar compuestos por números complejos. Los datos de entrada se combinan en dos grupos de dos bits [GPÁ⁺15]. Cada código dos bits genera una de las cuatro posibles fases de salida del portador: +45°, 135°, -45° y -135°.

El siguiente sistema recoge sus posibles soluciones:

$$x(t) = \begin{cases} A \sin(2\pi fct + \frac{\pi}{4}), & \text{Binario "11"} \\ A \sin(2\pi fct + \frac{3\pi}{4}), & \text{Binario "01"} \\ A \sin(2\pi fct - \frac{3\pi}{4}), & \text{Binario "00"} \\ A \sin(2\pi fct - \frac{\pi}{4}), & \text{Binario "10"} \end{cases} \quad (21)$$

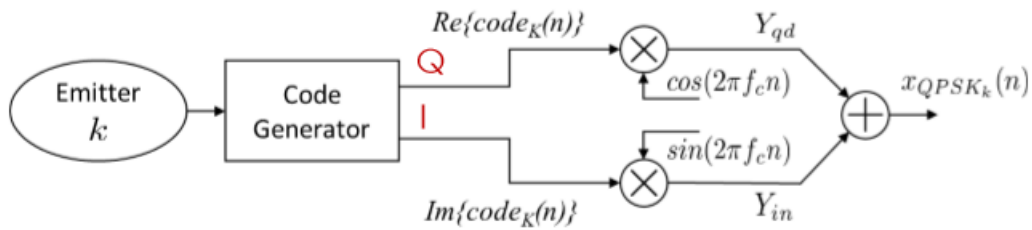


Fig. 24. Modulador QPSK [GPÁ+15]

Para un código Zadoff-Chu de $L=61$ y $Nemo=4$, se configura la señal portadora con una $f_c=41,667kHz$ (frecuencia de portadora), $f_{se}=500kHz$ (frecuencia de muestreo) y $Nc=2$ (número de ciclos de la portadora). Como resultado, se obtiene la señal modulada que aparece en la Fig. 27.

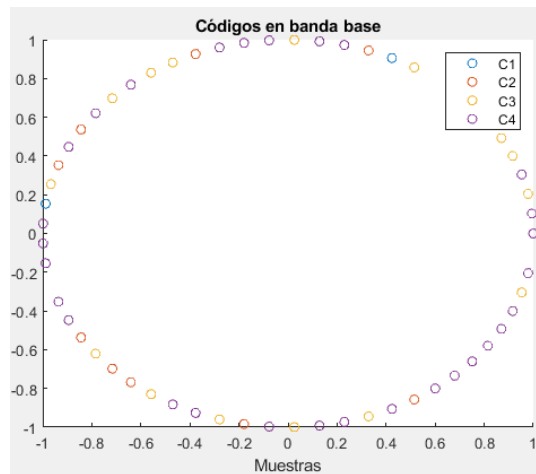


Fig. 25. Códigos Zadoff-Chu $L=61$, $Nemo=4$

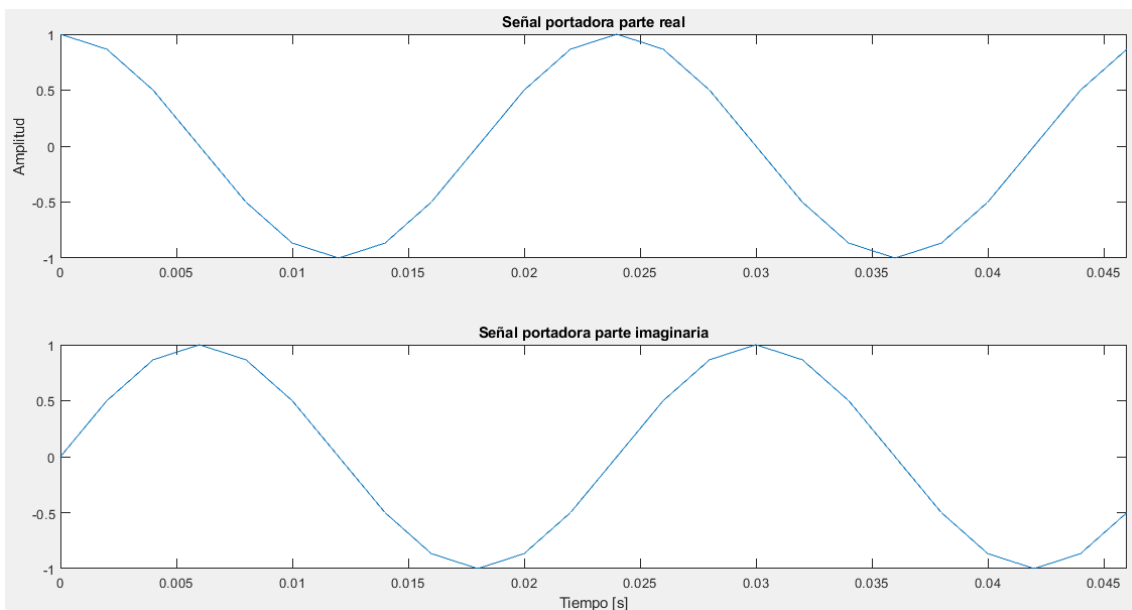


Fig. 26. Señal portadora QPSK $f_c=41,667kHz$, $f_{se}=500kHz$ y $Nc=2$

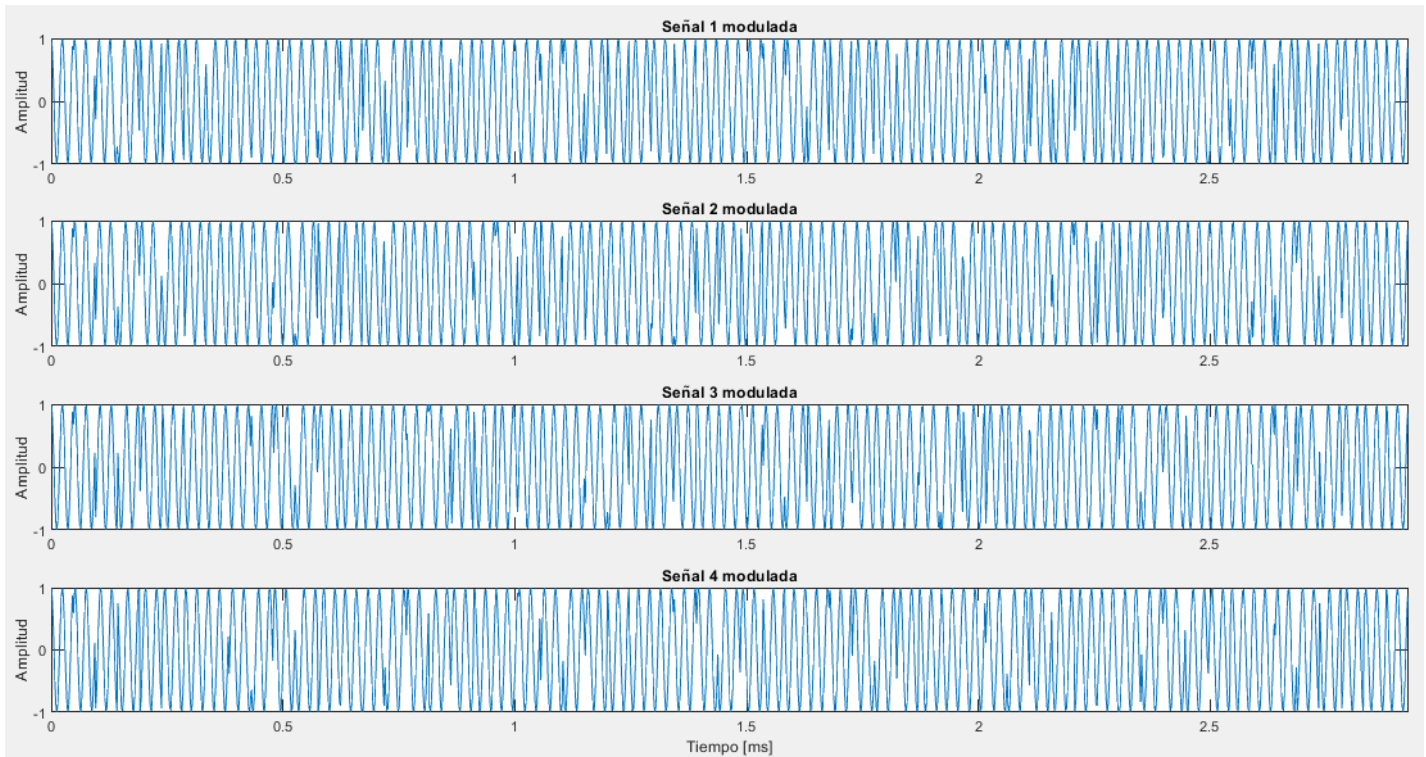


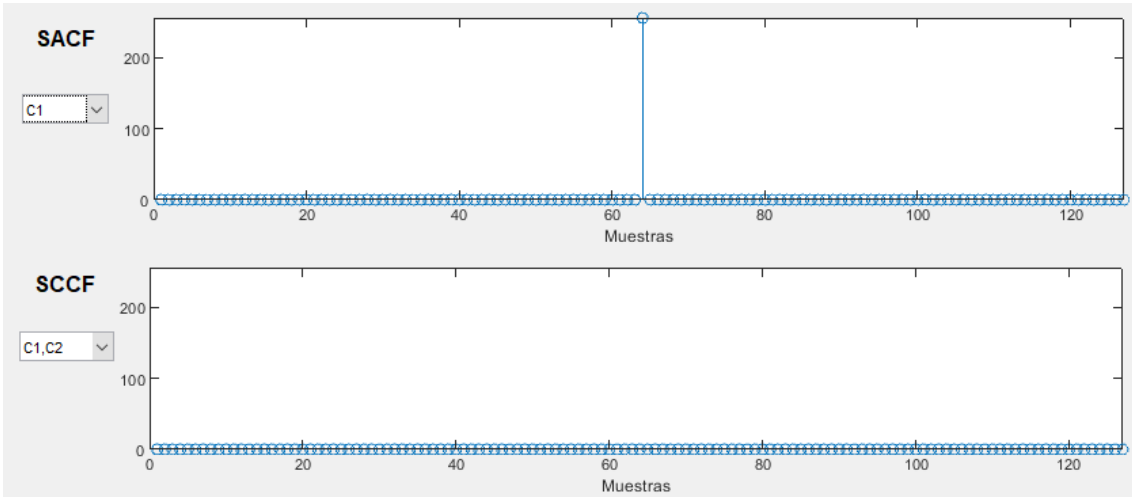
Fig. 27. Señal Zadoff.Chu modulada resultante

2.4. Conclusiones

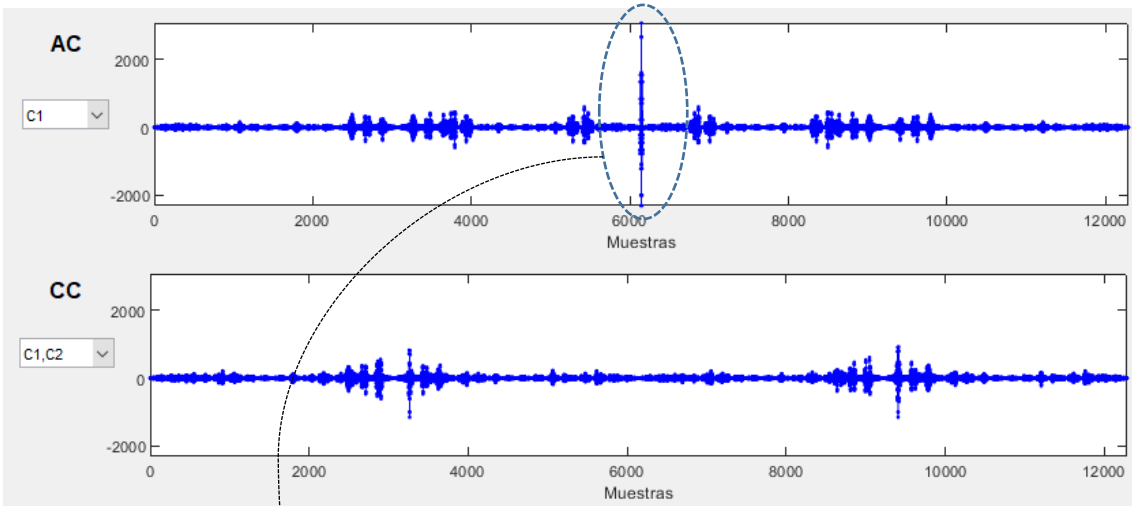
La modulación implica una degradación de las propiedades de AC y CC [GPÁ⁺15]. Los lóbulos laterales que aparecen implican un aumento de las cotas de θ_{AC} y θ_{CC} . Esta pérdida de propiedades depende del sobremuestreo y del número de períodos de portadora por bit transmitido.

La siguiente figura resalta la comparativa entre las propiedades de correlación cuando se emite el código en banda base y cuando se emite el modulado. Para este ejemplo, se han escogido las CSS, con $M=4$, $L=64$, $Nemo=2$, $fc=47,667$ kHz, $fse=500$ kHz y $Nc=2$.

Fig. 28. Correlaciones banda base VS modulada con $M=4$, $L=64$ CSS modulados con $Nc=2$ ciclos de portadora y factor de sobremuestreo $Of=12$



(a) Correlación en banda base



(b) Correlación de la señal modulada

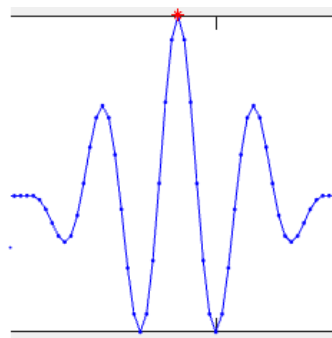


Fig. 29. Pico de correlación ampliado

3. MODELADO DE EFECTOS TÍPICOS SOBRE LA SEÑAL ULTRASÓNICA

3.1. Respuesta del transductor

El transductor utilizado es el modelo 328STR160. Su BW (*BandWidth*) es de aproximadamente 10 kHz centrado en 40 kHz. La Fig. 30 muestra su respuesta en frecuencia. La herramienta puede adaptarse fácilmente para la inclusión de otros modelos de transductor.

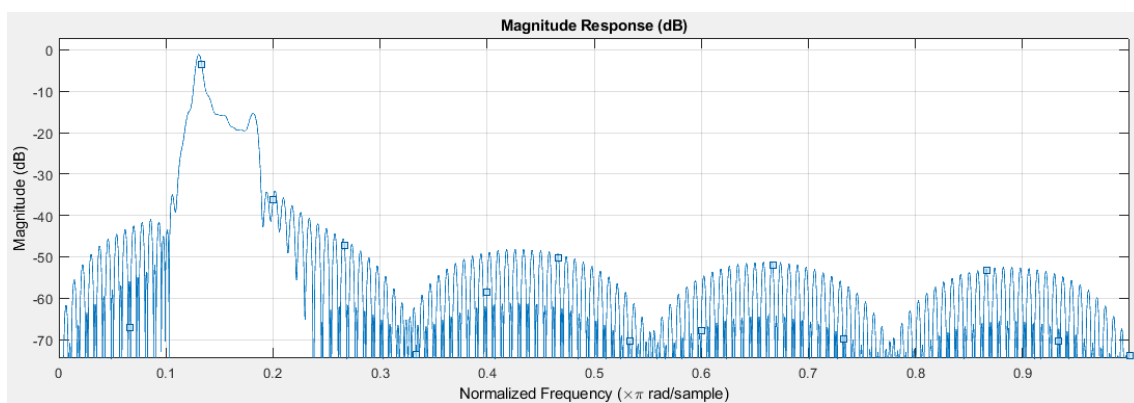


Fig. 30. Respuesta del transductor

En la Fig. 31 aparece la señal emitida tras aplicarle el efecto del transductor a unas CSS de $M=4$, $L=64$, $Nemo=2$, $f_c=47,667\text{kHz}$, $f_{se}=500\text{kHz}$ y $N_c=2$.

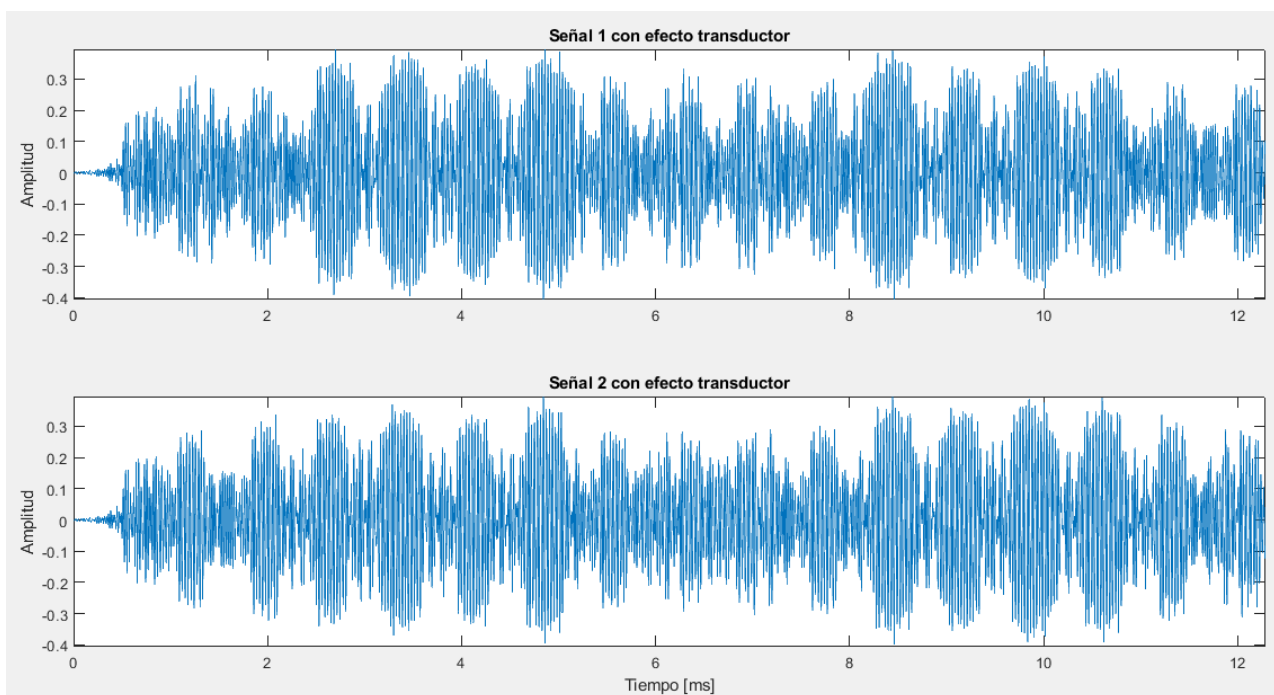


Fig. 31. Señal CSS con efecto del transductor

3.2. Transmisión de la señal

Para la transmisión de la señal, se deben que tener en cuenta una serie de parámetros. La Fig. 32 representa un esquema orientativo que aclara dichos conceptos donde T_e es el tiempo de emisión de un solo código (tiempo equivalente a la duración de su L), T_{gap} es el espaciado entre emisiones consecutivas, T_{shift} el retardo entre emisiones (aplicable sólo cuando N_{emo} es mayor que uno) y N_{rep} el número de repeticiones de cada código.

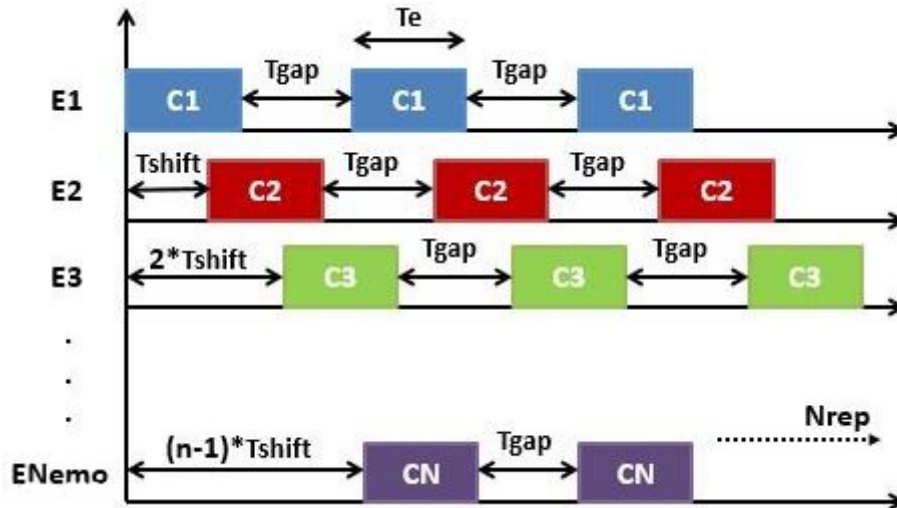


Fig. 32. Imagen orientativa de las configuraciones de emisión

3.3. Efectos del canal

Durante el recorrido que toma la señal configurada desde su emisión hasta su recepción, puede verse modificada debido a varios factores existentes en el ambiente. A continuación, se citan los efectos más comunes en las señales ultrasónicas, los cuales han sido añadidos a la aplicación diseñada para su posible interpretación.

3.3.1. Ruido blanco Gaussiano

El ruido ambiental está siempre presente, puede provenir de actividades humanas o mecánicas y puede afectar las estimaciones de los resultados en función de su energía y frecuencia [GPÁ⁺15]. El ruido ultrasónico puede aparecer en ambientes interiores debido al ruido de alguien que escribe en una computadora o el tintineo del llavero.

El Ruido Blanco Gaussiano es un modelo básico con potencia uniforme en toda la banda de frecuencias del sistema y con una distribución normal en el dominio del tiempo.

3.3.2. Ruido impulsivo

El ruido impulsivo es referido a sonidos agudos casi instantáneos. Son muchos los fenómenos mecánicos que contienen ruido ultrasónico que puede causar errores en la localización del pico de correlación [MPU⁺18].

3.3.3. Efecto cerca-lejos

Las señales más cercanas al receptor se reciben con una atenuación menor que las señales más alejadas. Así, las transmisiones más remotas pueden ser difíciles de recuperar [GPÁ⁺15].

3.3.4. Multicamino

En un ambiente interior, un receptor detecta múltiples réplicas retardadas y atenuadas de una señal transmitida debido a reflejos especulares de paredes y objetos [MPU⁺18].

3.3.5. Efecto Doppler

El movimiento del receptor se modela asumiendo un fse' virtual para la señal emitida dada como:

$$fse' = fse \left[1 - \frac{\vec{v}_r \cdot \frac{\vec{r}_e - \vec{r}_r}{|\vec{r}_e - \vec{r}_r|}}{c} \right] \quad (22)$$

Donde $c=343$ m/s (velocidad del sonido en el aire a una temperatura de 20°C y una presión de 1 atm), r_r es el vector de posición del receptor, r_e es el vector de posición del emisor, y v_r es el vector de velocidad del receptor. A partir de esta frecuencia, la señal adquirida por el receptor a la frecuencia de muestreo real se obtiene mediante un proceso de interpolación y diezmado [MPU⁺18].

3.4. Conclusiones

Cualquier tipo de efecto sobre la señal emitida va a provocar que el cálculo de las cotas de correlación se vea afectado. En las siguientes figuras, se compara la señal Kasami emitida sin efectos añadidos de $L=255$, $Nemo=5$, $fc=47,667kHz$, $fse=500kHz$, $Nc=2$, $Tgap=2ms$, $Tshift=3ms$, $Nrep=4$ y con transductor; con la señal resultante al aplicar cada uno de los efectos citados. A su vez, se visualiza la CC obtenida con el primer código generado.

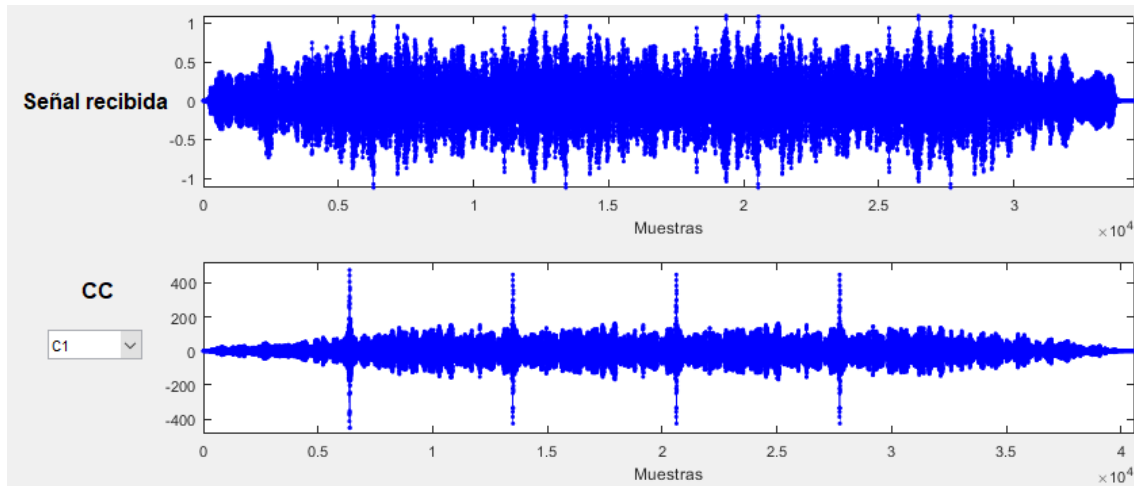


Fig. 33. Señal Kasami modulada en BPSK y con efecto del transductor

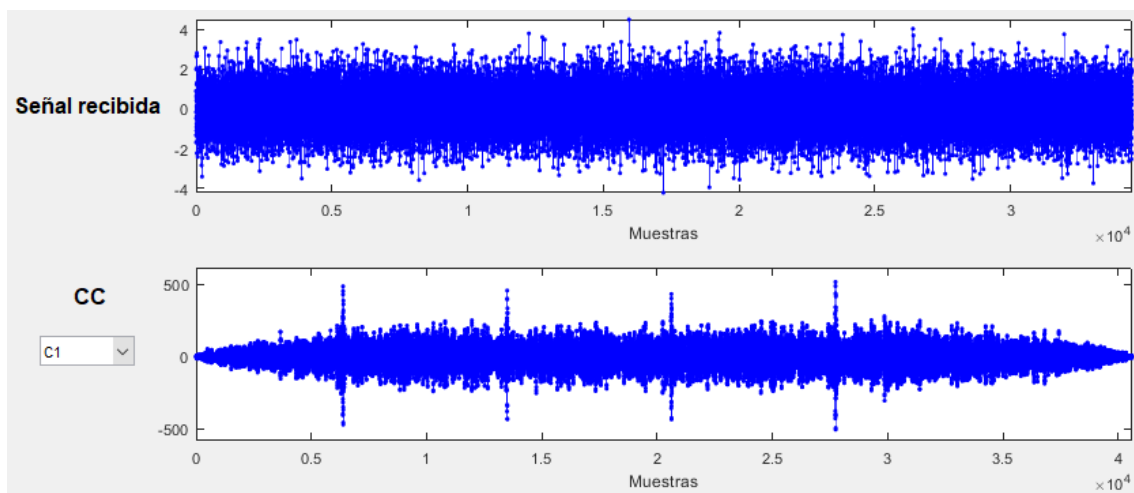


Fig. 34. Señal Kasami modulada en BPSK, con efecto del transductor y con Ruido Gaussiano de -10dB

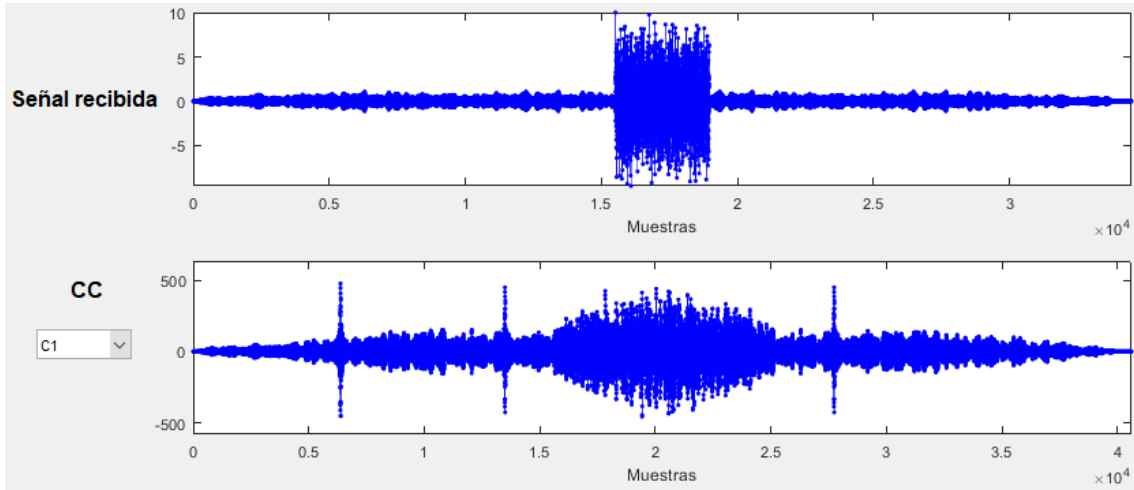


Fig. 35. Señal Kasami modulada en BPSK, con efecto del transductor y con Ruido Impulsivo de -20dB en la mitad del código con 10% de duración

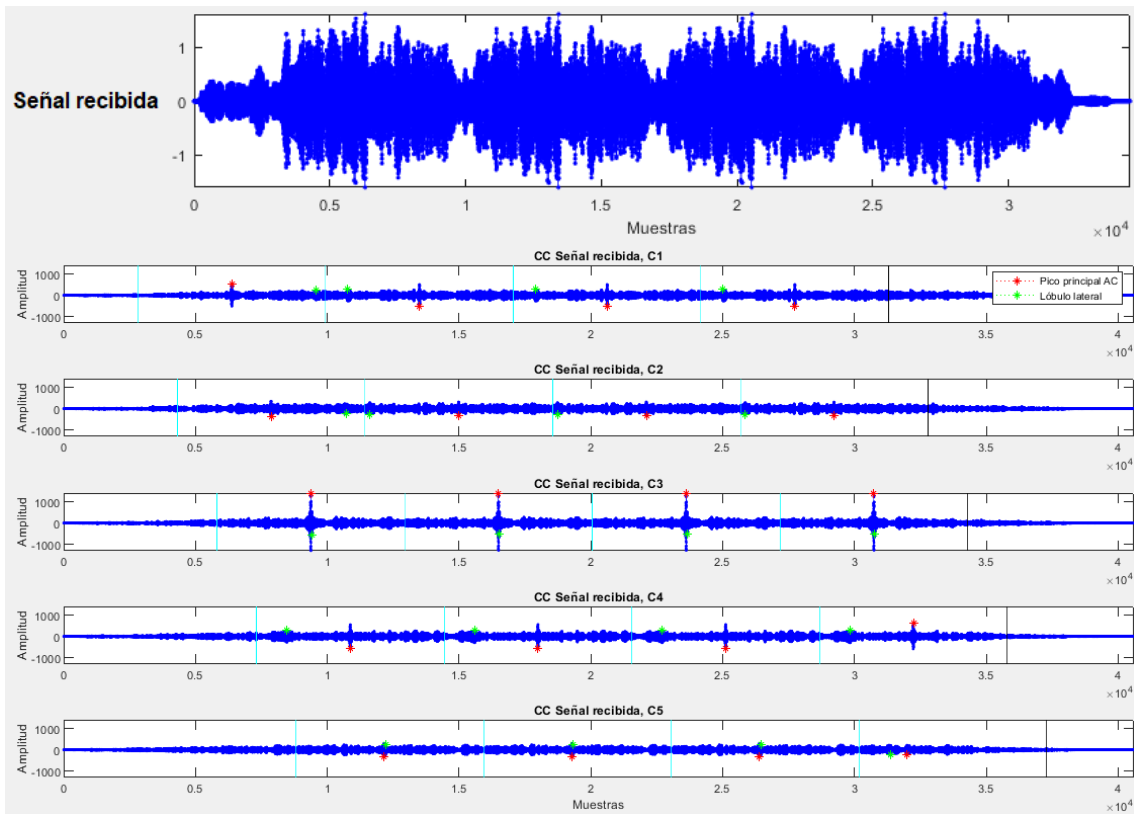


Fig. 36. Señal Kasami modulada en BPSK, con efecto del transductor y cerca-lejos de ganancia 1,0.6,3,1.3 y 0.2 para cada código respectivamente

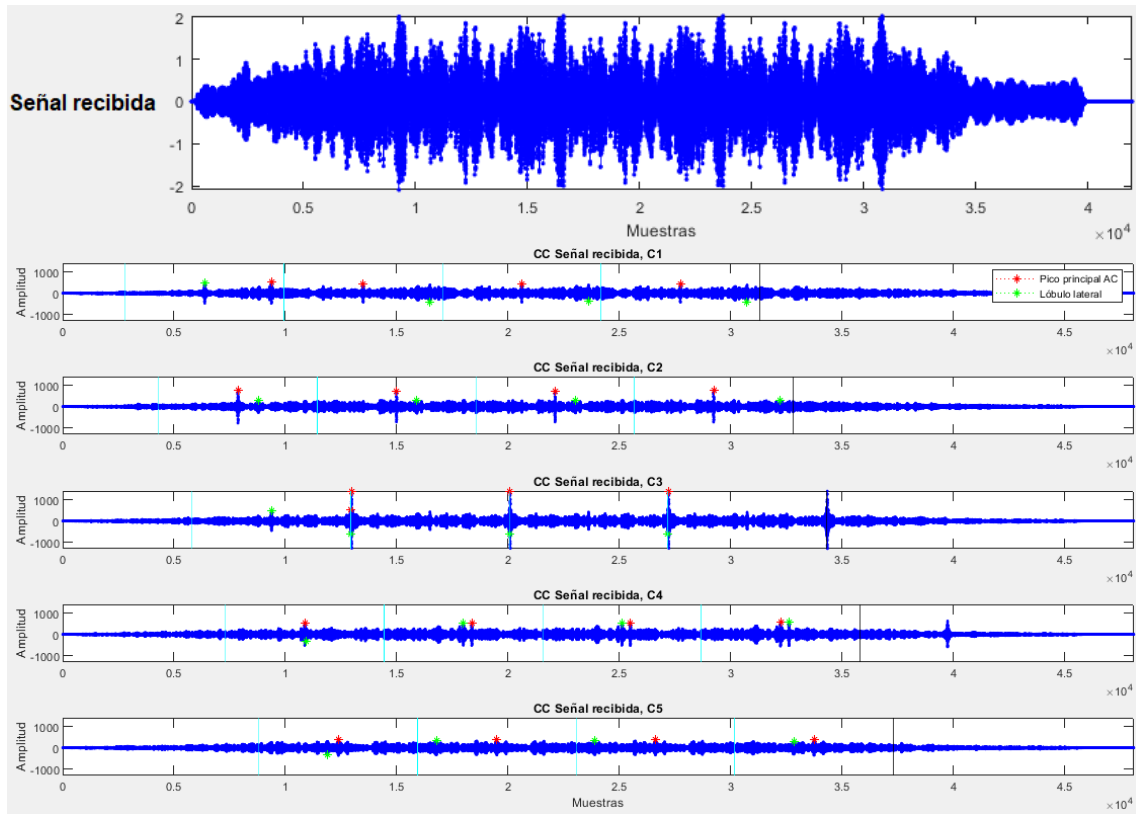


Fig. 37. Señal Kasami modulada en BPSK, con efecto del transductor y con Multicamino con retardo 6, 0, 7.2, 15 y 3 ms, y ganancia 1, 0.6, 3, 1.3 y 0.2, para cada código respectivamente

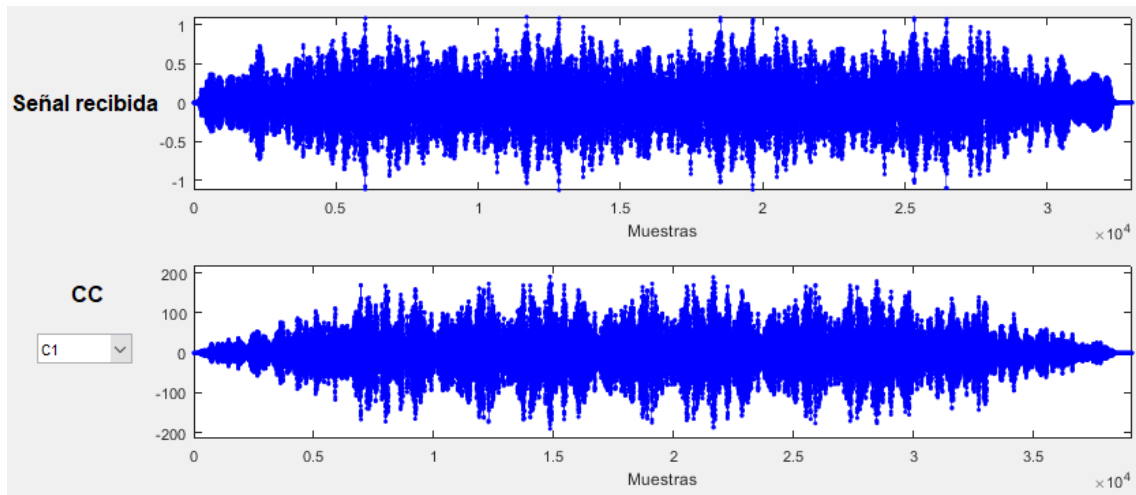


Fig. 38. Señal Kasami modulada en BPSK, con efecto del transductor y Doppler en el aire con $v_r = -15\text{m/s}$

Una forma de calcular el error provocado por el ruido en el cálculo de los picos de correlación es a través de la RMSE (Raíz del error cuadrático medio). Este concepto evalúa la diferencia de muestras entre el pico de correlación ideal de la señal y el pico resultante más elevado al aplicar ruido. Para su interpretación, se aplica Ruido Blanco Gaussiano.

En la siguiente figura, se ha aplicado ruido desde los -20dB hasta los 10dB a la señal Kasami anteriormente configurada y se ha calculado la RMSE con la señal emitida ideal. Se puede observar que, a medida que el ruido aumenta, la similitud entre señales aumenta y por lo tanto el valor del RMSE disminuye.

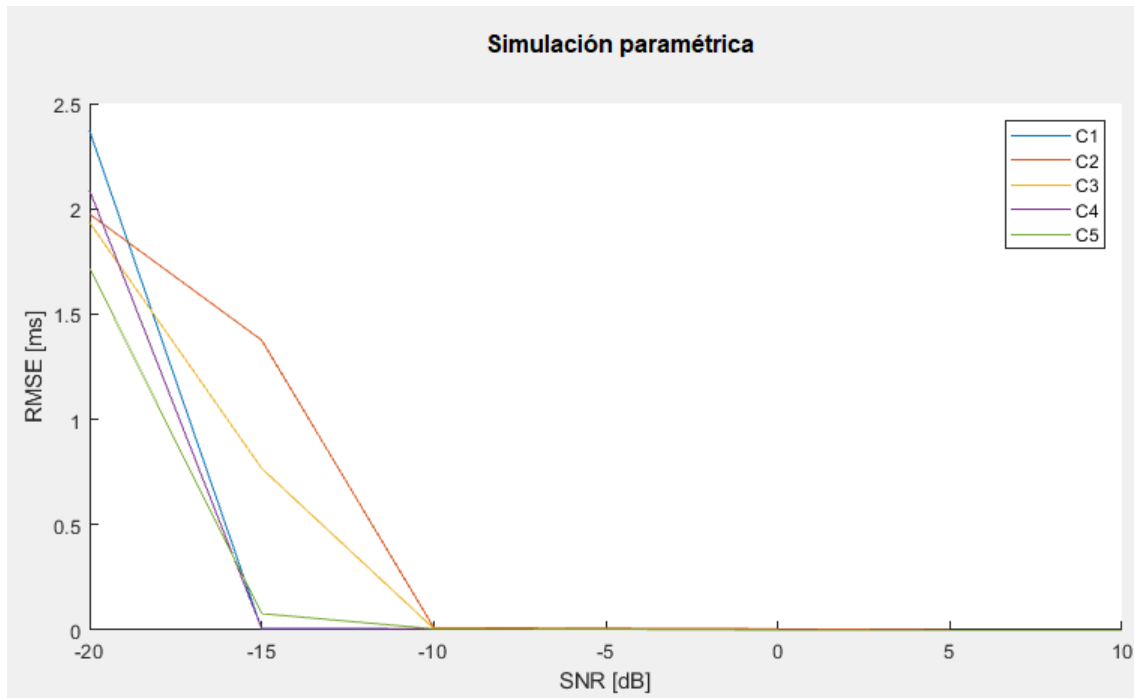


Fig. 39. Cálculo de RMSE entre -20dB y 10 dB

4. DISEÑO DE LA PLATAFORMA INTERACTIVA PARA LA PRUEBA DE ESQUEMAS DE CODIFICACIÓN EN SISTEMAS ULTRASÓNICOS

4.1. INTRODUCCIÓN

En este apartado se detalla el diseño de la aplicación desarrollada desde Matlab-GUIDE.

Mediante el editor de diseño de GUIDE es posible diseñar gráficamente la interfaz de usuario ya que incluye controles tales como menús, barras de herramientas, botones y controles deslizantes. GUIDE genera entonces de manera automática el código de MATLAB para construir la interfaz, el cual se puede modificar para programar el comportamiento de la app [MAT19].

Las funciones más relevantes de esta interfaz se encuentran en los Anexos de este documento.

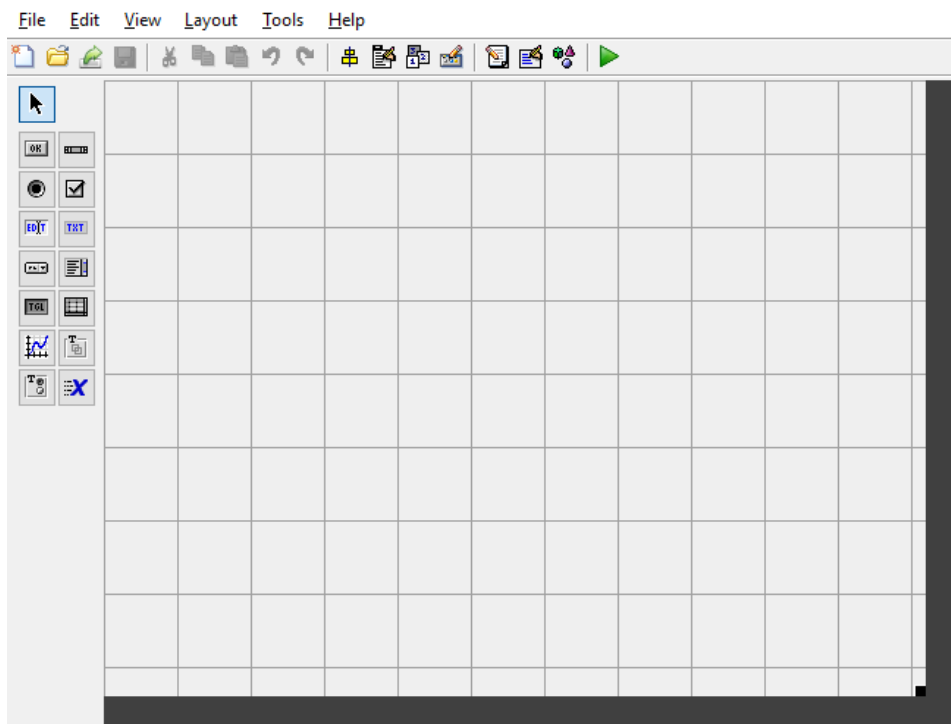


Fig. 40. Ventana de diseño GUIDE

4.2. INTERFAZ GRÁFICA: PANTALLA INICIO, ELECCIÓN CODIFICACIÓN

La primera ventana que aparece en la aplicación recoge un esquema genérico de los distintos pasos que va a seguir el usuario. Además, está acompañado de un texto que indica brevemente las funciones de cada ventana.

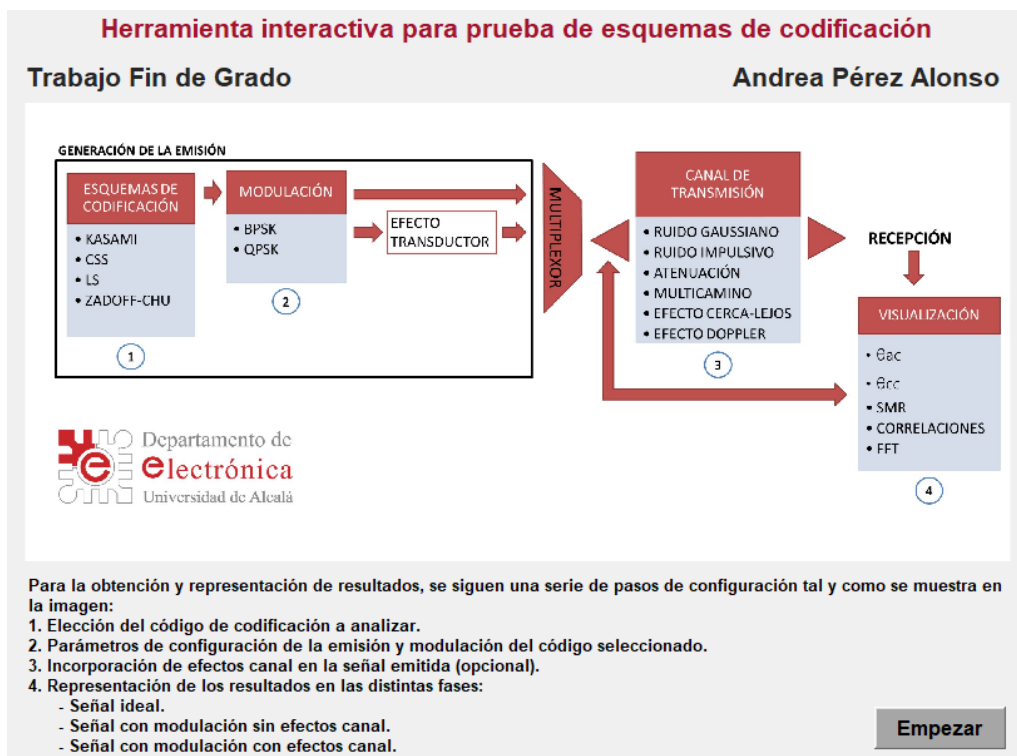


Fig. 41. TFG: Introducción

La siguiente ventana permite seleccionar el lenguaje de codificación en el que se quieren realizar las pruebas.

CODIFICACIÓN

Códigos

Kasami

LS

CSS

Zadoff-Chu

Volver

Fig. 42. Ventana 1 TFG: Elección de codificación

4.3. CONFIGURACIÓN EMISIÓN

Para este apartado, se va a analizar la configuración de las secuencias Kasami. A partir de ella, se particularizará para cada uno de los códigos restantes.

4.3.1. KASAMI

La Fig. 43 muestra la pantalla de configuración de emisión de las secuencias Kasami.

El botón ‘Valores por defecto’ introduce unos datos determinados para que el usuario pueda orientarse, ‘Borrar’ elimina cualquier valor que haya en los recuadros y ‘Ayuda’ abre el archivo correspondiente a dicha ventana en el Manual de Usuario adjuntado en esta memoria. ‘Volver’ redirige al programa a la selección del código de codificación y ‘Siguiente’ continua a la configuración de los efectos en la señal a emitir.

Para la configuración de propiedades de los códigos, se divide esta ventana en varias secciones.

- Entradas:

Configuración de los elementos necesarios para la generación del código.

La longitud $L=2^N-1$ viene restringida por N que debe ser un número par. De lo contrario, se manda un mensaje de aviso al usuario para que se modifique el parámetro. El tamaño de familia $M=2^{N/2}$ se calcula automáticamente una vez introducida L de donde se extrae N .

El usuario puede escoger el número de emisores simultáneos $Nemo$ que desea para la realización de pruebas dentro de los M códigos de longitud L generados. Este parámetro también está restringido y se mandará otro mensaje de aviso si $Nemo > M$.

La eficiencia energética η evalúa, según la fórmula (5), los códigos Kasami generados a partir de (11).

- Modulación:

Estos códigos utilizan la modulación BPSK, ver expresión (20), ya que son de naturaleza binaria.

Se selecciona la forma de onda de la señal portadora (sinusoidal o cuadrada) y sus parámetros característicos: frecuencia de la portadora f_c , frecuencia de muestro f_{se} y número de ciclos de la portadora N_c . El factor de sobremuestreo Of se obtiene a través de $Of=f_c/f_{se}$ y siempre va a ser un número entero. Para que esto se cumpla, cada vez que se

introduce un valor tanto en f_c como en f_{se} , se calcula Of y se redondea por su entero más próximo. Con este valor modificado, se recalcula el parámetro que se había introducido y se modificara en pantalla.

La duración de emisión de un código Te es el resultado de $Te=L \cdot Nc/f_c$.

- Emisión:

Generación de la señal a emitir configurando su espaciado entre emisiones consecutivas T_{gap} , su retardo entre emisores T_{shift} y su número de repeticiones N_{rep} . La Fig. 32 muestra una imagen orientativa de estos conceptos para facilitar el entendimiento al usuario.

Además, proporciona la opción de añadir el transductor o no. Éste está ajustado al ancho de banda del modulador mediante f_{se} .

- Salidas:

Permite almacenar en la carpeta que esté situado el programa las señales en banda base y/o moduladas que se han generado para su posible aplicación en pruebas reales.

También, ofrece la visualización de las señales a emitir durante todo el proceso de generación, de la señal portadora y del modelo del transductor.

Configuración de emisión: Kasami

Entradas

Longitud (L) Tamaño familia (M)

Nº emisores simultáneos (Nemo) Eficiencia energética (η)

Modulación BPSK:

Tipo portadora

Frecuencia portadora (fc) kHz Factor sobremuestreo (Of)

Frecuencia de muestreo (fse) kHz Duración emisión un código (Te) ms

Nº ciclos portadora (Nc)

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms **Imagen orientativa**

Número de repeticiones (Nrep)

Modelo transductor

Salidas

Fig. 43. Ventana 2 TFG: Configuración emisión Kasami

A continuación, se presenta un ejemplo de la aplicación para los siguientes valores: $L=255$, $Nemo=5$, portadora sinusoidal, $fc=42\text{kHz}$, $fse=500\text{kHz}$, $Nc=2$, $Tgap=2\text{ms}$, $Tshift=5\text{ms}$, $Nrep=3$ y con modelo transductor.

En la Fig. 44 se muestran los datos introducidos y los resultados obtenidos en M , η , Of y Te . Como se puede observar, fc pasa de valer 42kHz a ser $41,6667\text{kHz}$ para que Of sea un número entero.

Configuración de emisión: Kasami

Entradas

Longitud (L) Tamaño familia (M)

Nº emisores simultáneos (Nemo) Eficiencia energética (η)

Modulación BPSK:

Tipo portadora ▼

Frecuencia portadora (f_c) kHz

Frecuencia de muestreo (f_{se}) kHz Factor sobremuestreo (Of)

Nº ciclos portadora (N_c) Duración emisión un código (Te) ms

Emisión:

Espaciado entre emisiones consecutivas (T_{gap}) ms

Retardo entre emisores (T_{shift}) ms

Número de repeticiones (N_{rep})

Modelo transductor ▼

Salidas

Fig. 44. Ejemplo Ventana 2 Kasami: Configuración

Los archivos de almacenamiento serán .mat de 5×255 para el caso de banda base y 5×6120 para el caso modulado ($Te \cdot fse = 6120$ muestras).

La Fig. 45 muestra la visualización de los códigos generados en banda base.

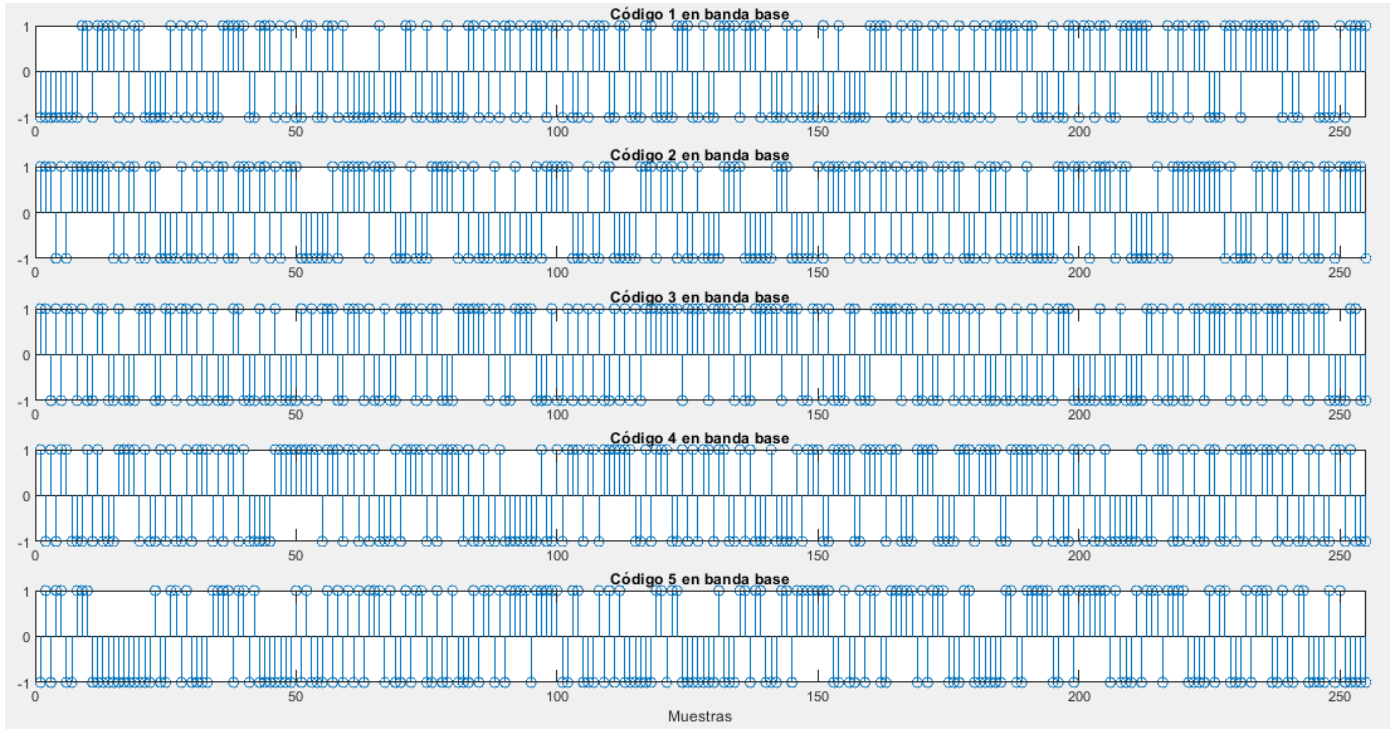


Fig. 45. Ejemplo ventana 2 Kasami: banda base

Las Fig. 46 y 47 son el resultado de la portadora y de la señal modulada.

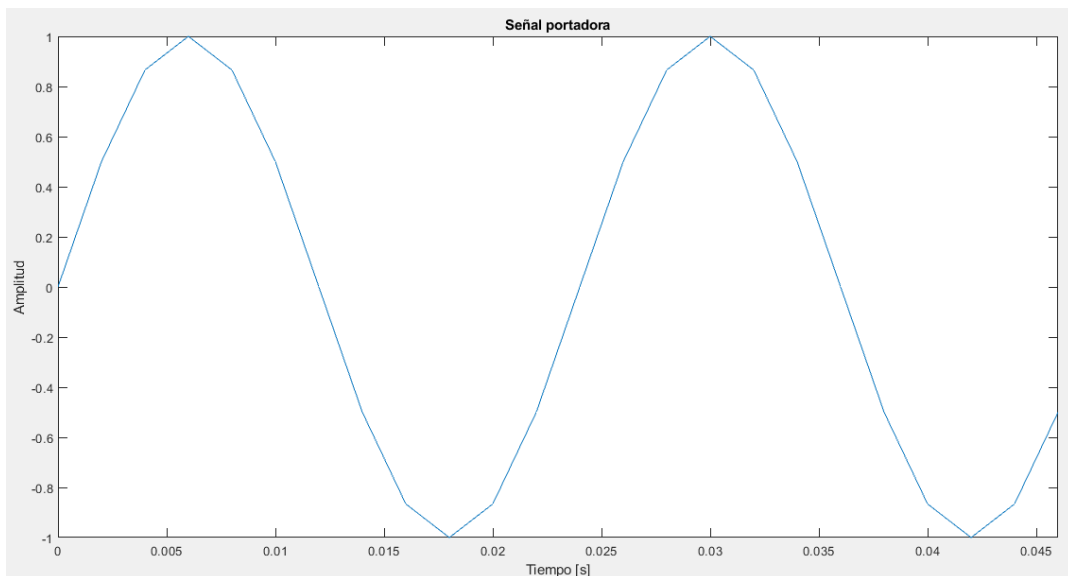


Fig. 46. Ejemplo ventana 2 Kasami: portadora

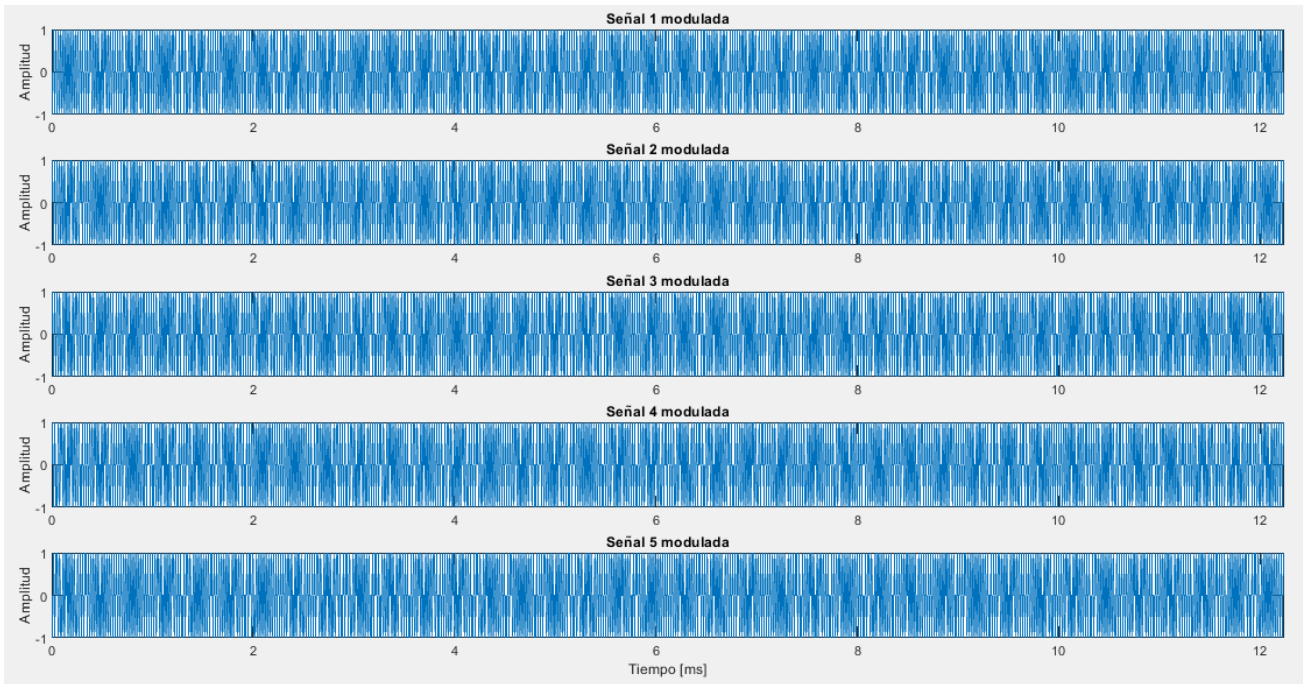


Fig. 47. Ejemplo ventana 2 Kasami: modulada

En la Fig. 48 aparece la señal modulada con el efecto del transductor.

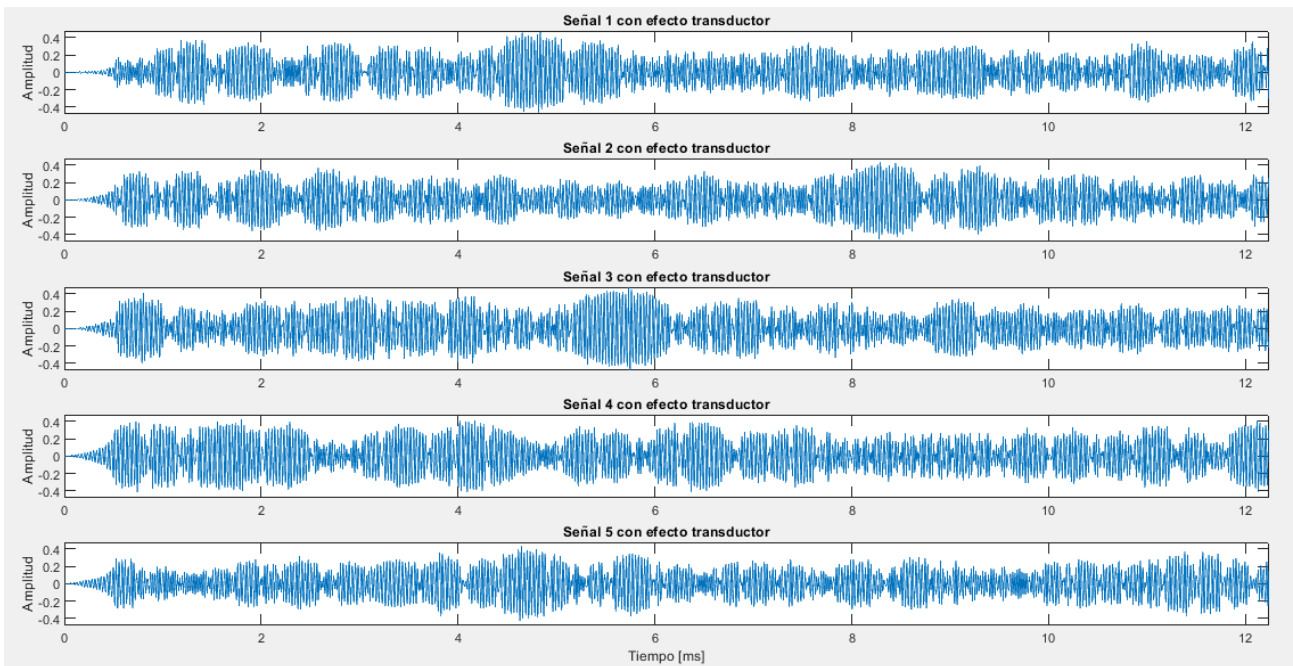


Fig. 48. Ejemplo ventana 2 Kasami: transductor

Por último, la Fig. 49 representa la señal a emitir por cada emisor.

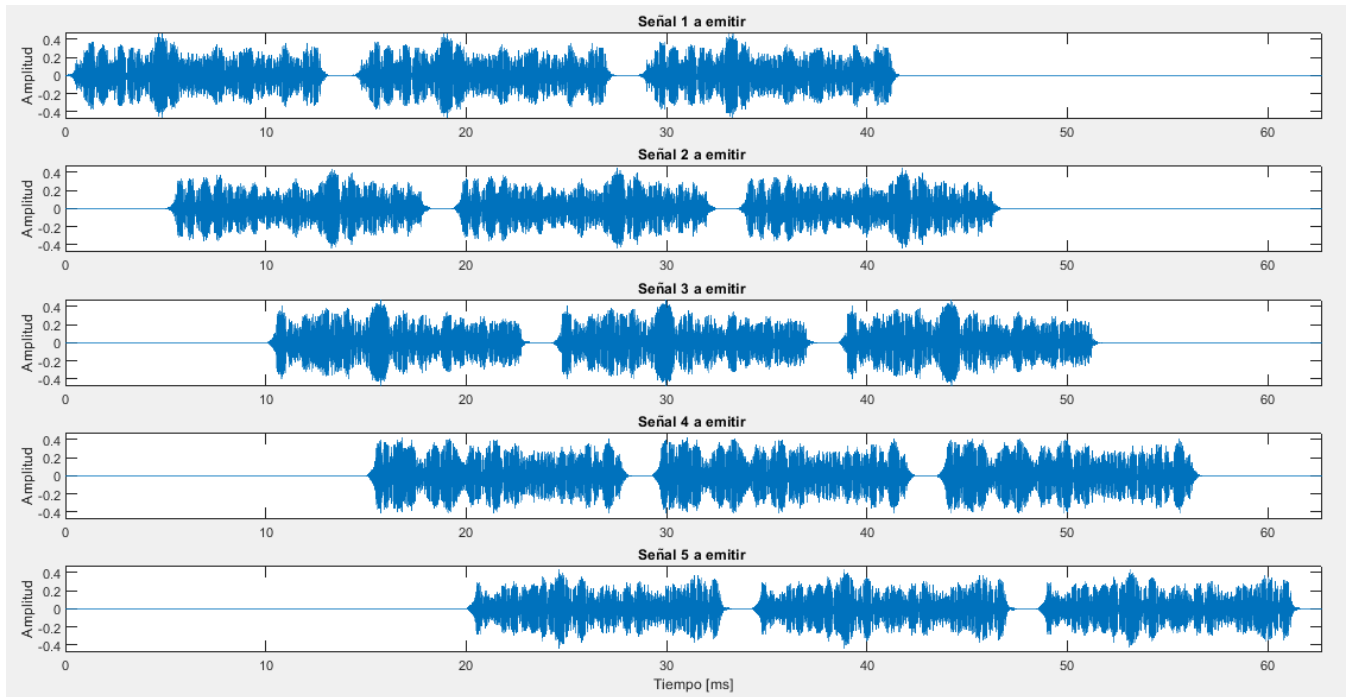


Fig. 49. Ejemplo ventana 2 Kasami: emisión

4.3.2. CSS

En la Fig. 50 aparece la pantalla de configuración de las CSS.

Como cada código tiene M secuencias, los parámetros de configuración incluyen M que debe ser potencia de 2 y restringe el valor de la longitud $L=M^N$ siendo N un número natural.

Para que los códigos resultantes sean incorrelados, la semilla de generación tiene que tener una separación entre ellos de L/M muestras. Dicha semilla inicialmente vale cero y se va incrementando L/M muestras hasta conseguir el *Nemo* seleccionado.

A la hora de realizar la modulación, las secuencias deben reordenarse y formar un sólo código de longitud $M \cdot L$. Existen dos mecanismos de reordenación: concatenación (15) y entrelazado (16). Por lo tanto, la duración de emisión de un código viene dada como $Te=M \cdot L \cdot Nc/fc$.

La visualización de los códigos en banda base se realiza de forma individual. Según el código que se seleccione en el menú de su derecha, saldrán representadas sus M secuencias junto con el resultado de la reordenación con el método escogido.

Para que la aplicación ofrezca buenos resultados en las correlaciones, hay que tener en cuenta que los códigos deben tener un T_{gap} de guardia para evitar picos muy elevados y que la emisión no debe ser simultánea por lo que T_{shift} debe ser, por lo menos, el mínimo valor que reconoce el programa ($0,1$ ms).

Fig. 50. Ventana 2 TFG: Configuración de la emisión CSS

A continuación, se presenta un ejemplo de la aplicación para los siguientes valores: $M=4$, $L=64$, $Nemo=2$, ordenación mediante entrelazado, portadora sinusoidal, $f_c=42\text{kHz}$, $f_{se}=500\text{kHz}$, $N_c=2$, $T_{gap}=10\text{ms}$, $T_{shift}=1\text{ms}$, $N_{rep}=4$ y con modelo transductor.

En la Fig. 51 se muestran los datos introducidos y los resultados obtenidos en η , O_f y T_e .

Configuración de emisión: CSS

Entradas

Tamaño familia (M) Concatenado Entrelazado

Longitud (L) Eficiencia energética (η)

Nº emisoros simultáneos (Nemo)

Modulación BPSK:

Tipo portadora

Frecuencia portadora (f_c) kHz

Frecuencia de muestreo (f_{se}) kHz Factor sobremuestreo (O_f)

Nº ciclos portadora (N_c) Duración emisión un código (T_e) ms

Emisión:

Espaciado entre emisiones consecutivas (T_{gap}) ms

Retardo entre emisoros (T_{shift}) ms

Número de repeticiones (N_{rep})

Modelo transductor

Salidas

Fig. 51. Ejemplo Ventana 2 CSS: Configuración

Los archivos de almacenamiento serán .mat de $4 \times 255 \times 2$ para el caso de banda base y 5×6144 para el caso modulado.

La Fig. 52 muestra la visualización de las secuencias generadas del código 2 en banda base.

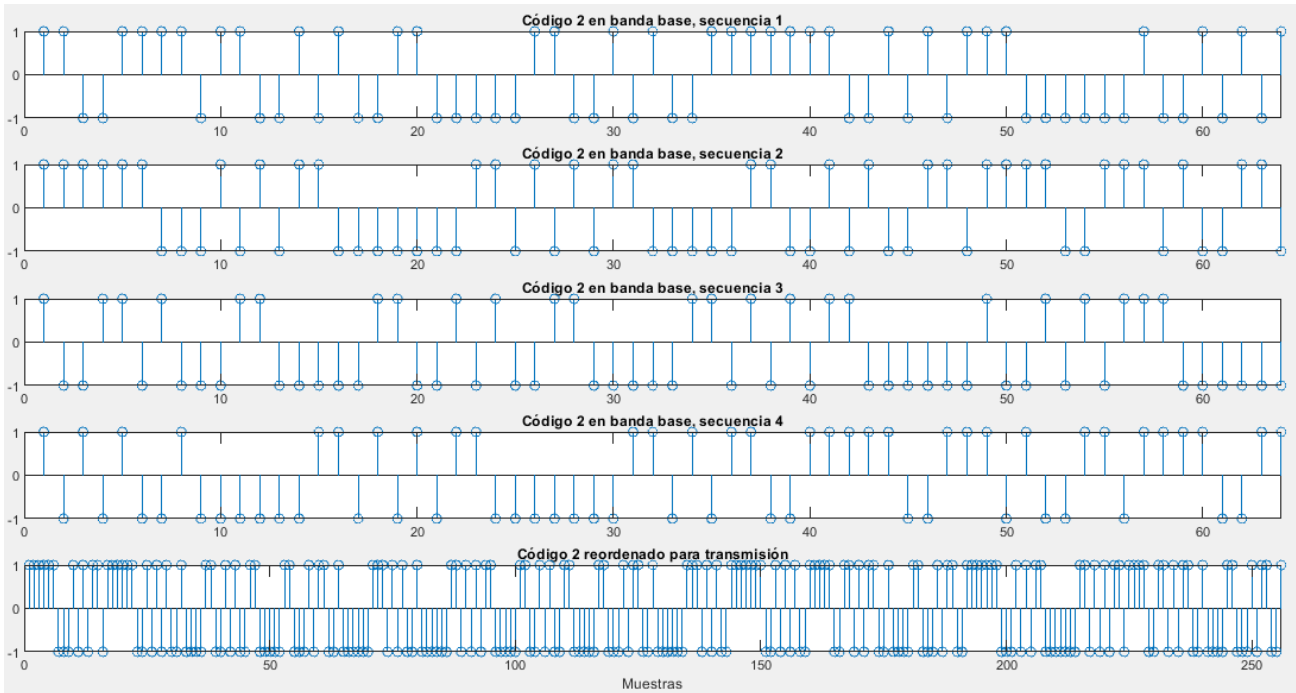


Fig. 52. Ejemplo ventana 2 CSS: banda base

Las Fig. 53 y 54 son el resultado de la portadora y de la señal modulada.

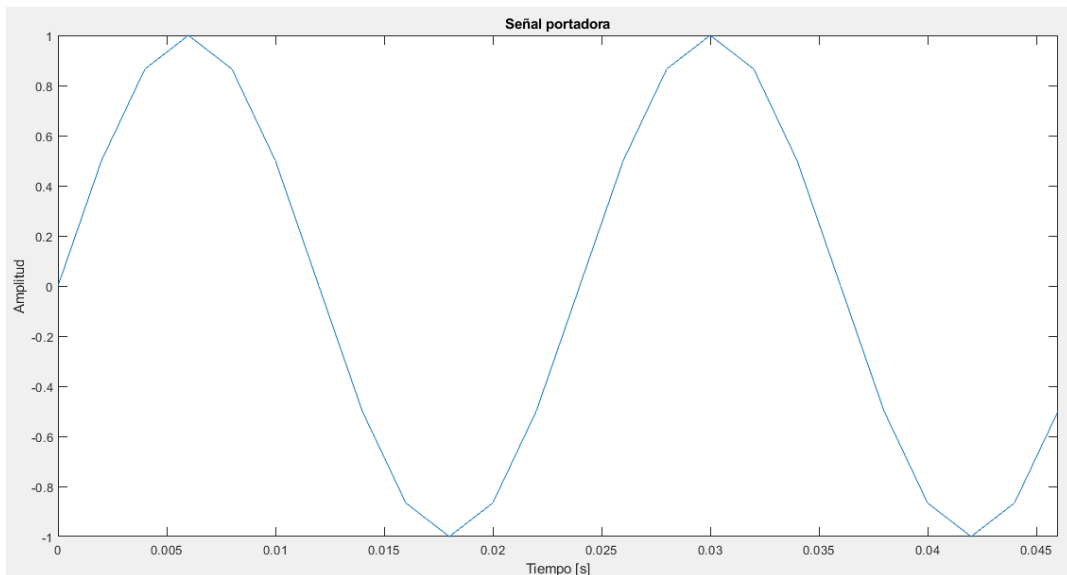


Fig. 53. Ejemplo ventana 2 CSS: portadora

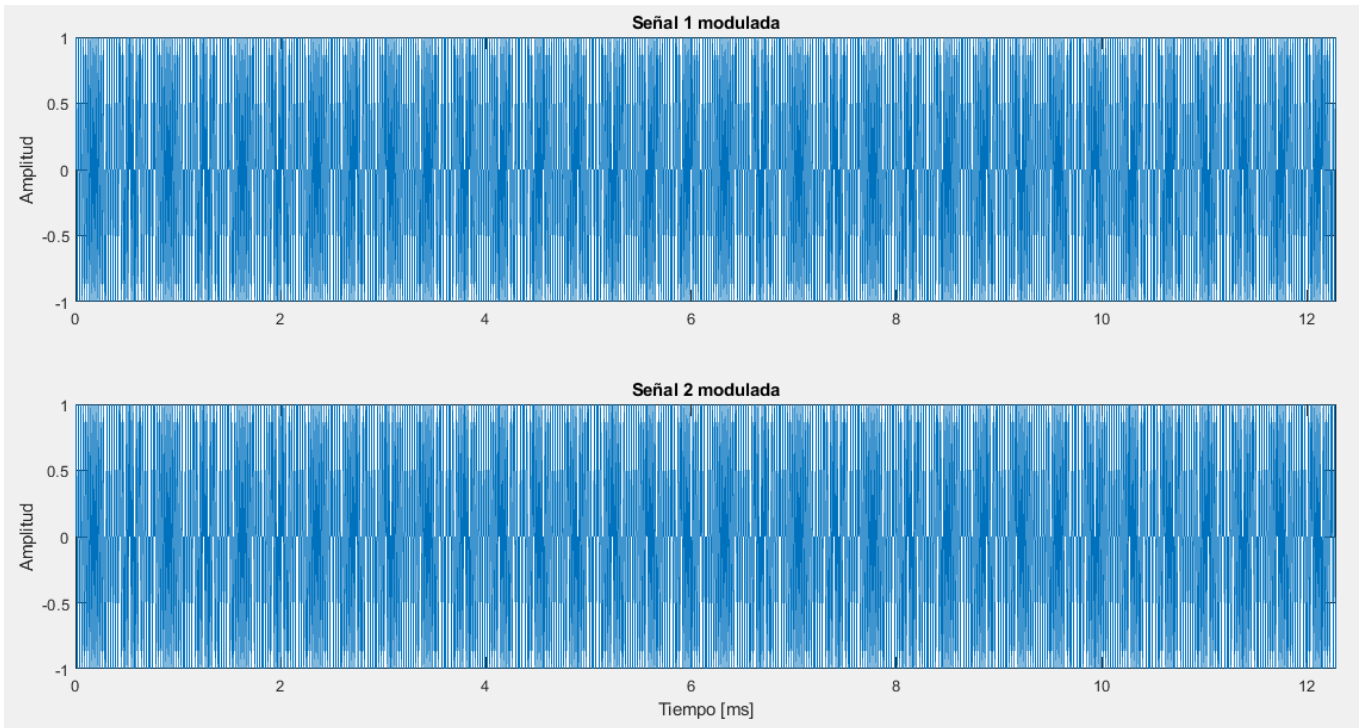


Fig. 54. Ejemplo ventana 2 CSS: modulada

En la Fig. 55 aparece la señal modulada con el efecto del transductor.

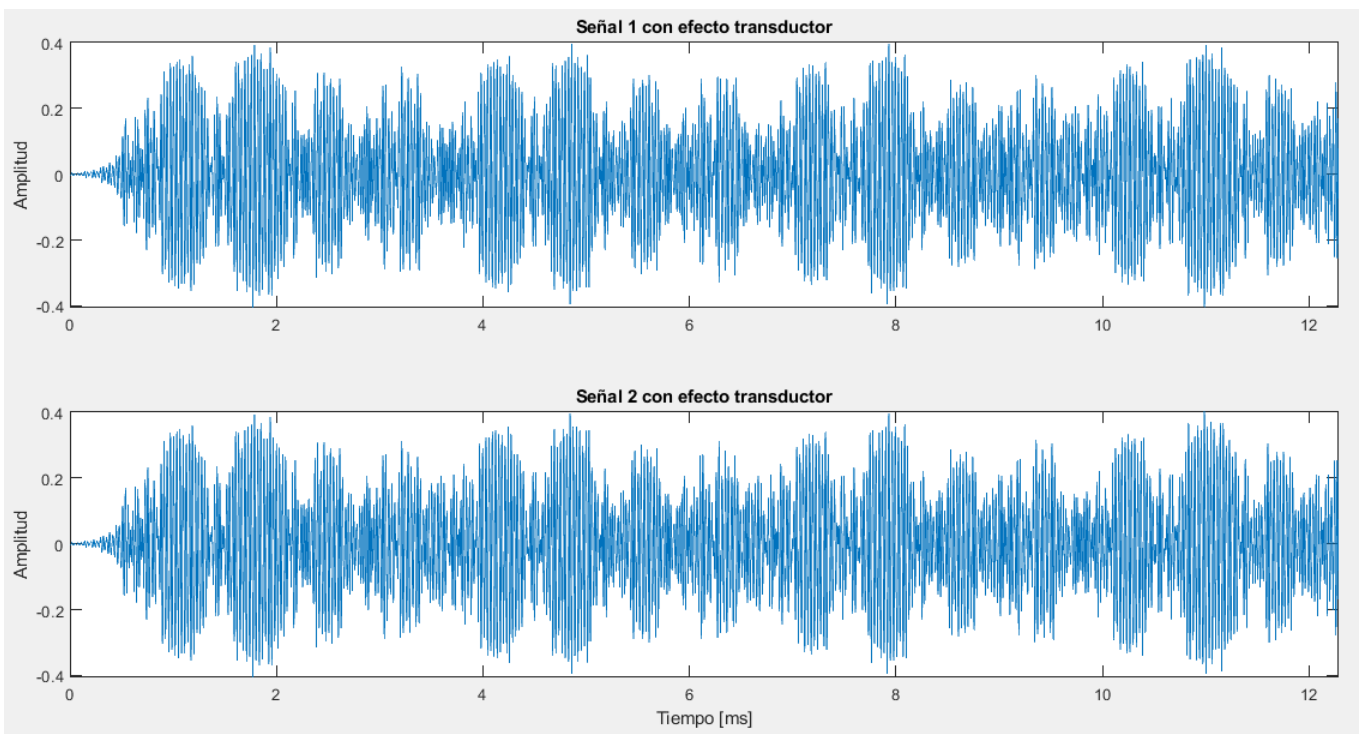


Fig. 55. Ejemplo ventana 2 CSS: transductor

Por último, la Fig. 56 representa la señal a emitir por cada emisor.

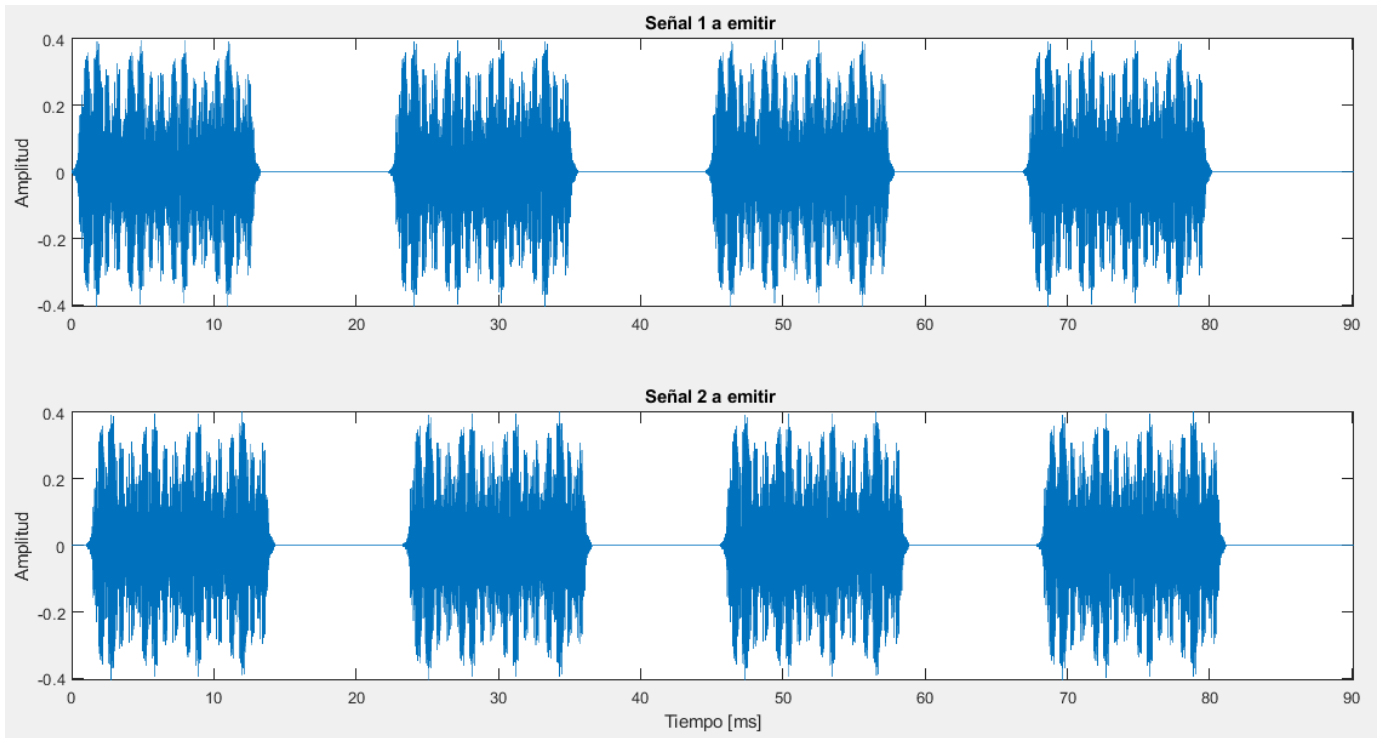


Fig. 56. Ejemplo ventana 2 CSS: emisión

4.3.3. LS

La Fig. 57 representa la ventana de configuración de estos códigos.

En ella, aparecen nuevos parámetros como la longitud de Parejas Golay Base L_o y la Zona de Correlación Cero ZCZ .

ZCZ se obtiene como $ZCZ=2 \cdot W+1$ siendo W la semiventana de correlación cero la cual está fijada por $W=L_o-1$. Así mismo, L equivale a $L=L_o \cdot M+W$.

Al igual que ocurre en las CSS, se necesita tener un mínimo de valor en T_{gap} y en T_{shift} para obtener buenos resultados de correlación. En este caso, T_{gap} debe tener mínimo $ms = W \cdot f_c / (f_{se} \cdot N_c)$ ms.

Fig. 57. Ventana 2 TFG: Configuración emisión LS

A continuación, se presenta un ejemplo de la aplicación para los siguientes valores: $M=8$, $L_o=16$, $Nemo=4$, portadora cuadrada, $f_c=42\text{kHz}$, $f_{se}=500\text{kHz}$, $N_c=2$, $T_{gap}=10\text{ms}$, $T_{shift}=1\text{ms}$, $N_{rep}=4$ y con modelo transductor.

En la Fig. 58 se muestran los datos introducidos y los resultados obtenidos en ZCZ , L , η , Of y Te . Como se puede observar, la eficiencia energética η ya no es de valor 1 como en los anteriores lenguajes de codificación ya que contiene 0s además de 1s y -1s en su código.

Configuración de emisión: LS

Entradas

Tamaño familia (M)	<input type="text" value="8"/>	Zona correlación cero (ZCZ)	<input type="text" value="31"/>
Longitud parejas Golay Base (Lo)	<input type="text" value="16"/>	Longitud código (L)	<input type="text" value="143"/>
Nº emisores simultáneos (Nemo)	<input type="text" value="4"/>	Eficiencia energética (η)	<input type="text" value="0.895105"/>

Modulación BPSK:

Tipo portadora	<input type="text" value="Sinusoidal"/>		
Frecuencia portadora (fc)	<input type="text" value="41.6667"/>	kHz	
Frecuencia de muestreo (fse)	<input type="text" value="500"/>	kHz	
Nº ciclos portadora (Nc)	<input type="text" value="2"/>		
		Factor sobremuestreo (Of)	<input type="text" value="12"/>
		Duración emisión un código (Te)	<input type="text" value="6.86399"/>

Emisión:

Espaciado entre emisiones consecutivas (Tgap)	<input type="text" value="10"/>	ms	
Retardo entre emisores (Tshift)	<input type="text" value="1"/>	ms	<input type="button" value="Imagen orientativa"/>
Número de repeticiones (Nrep)	<input type="text" value="4"/>		
Modelo transductor	<input type="text" value="Sí"/>		

Salidas

Fig. 58. Ejemplo Ventana 2 LS: Configuración

Los archivos de almacenamiento serán .mat de 4x143 para el caso de banda base y 5x3492 para el caso modulado.

La Fig. 59 muestra la visualización de los códigos generados en banda base.

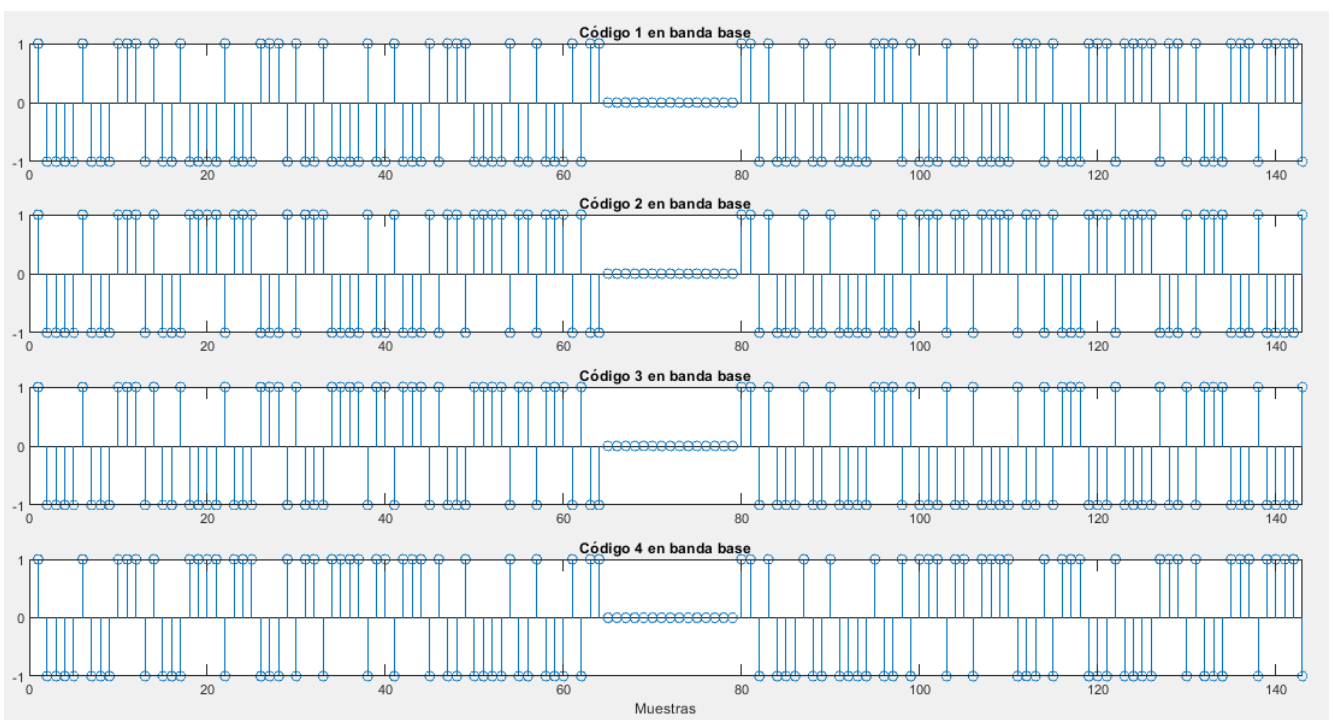


Fig. 59. Ejemplo ventana 2 LS: banda base

Las Fig. 60 y 61 son el resultado de la portadora y de la señal modulada.

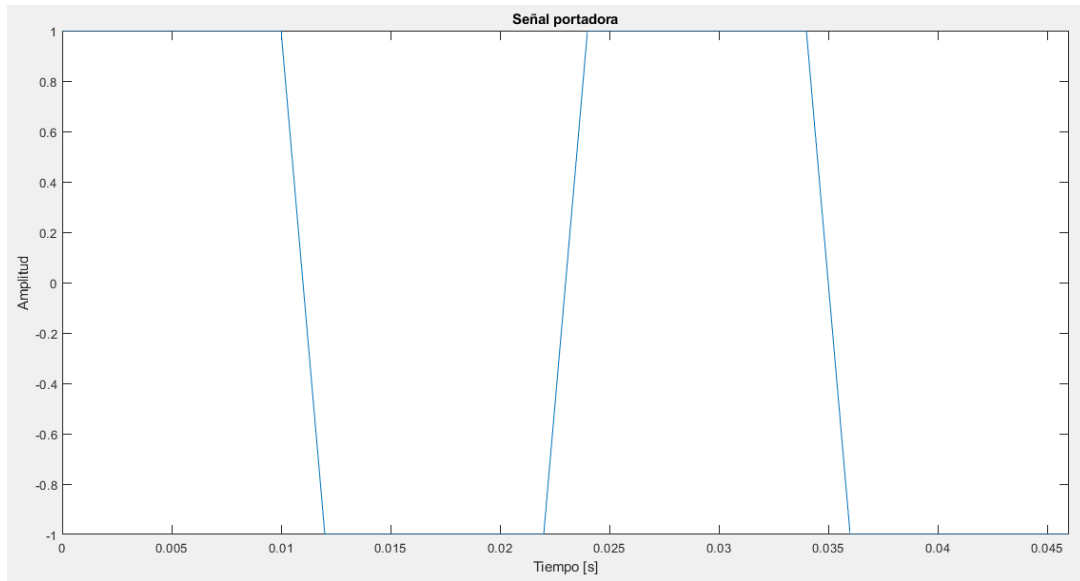


Fig. 60. Ejemplo ventana 2 LS: portadora

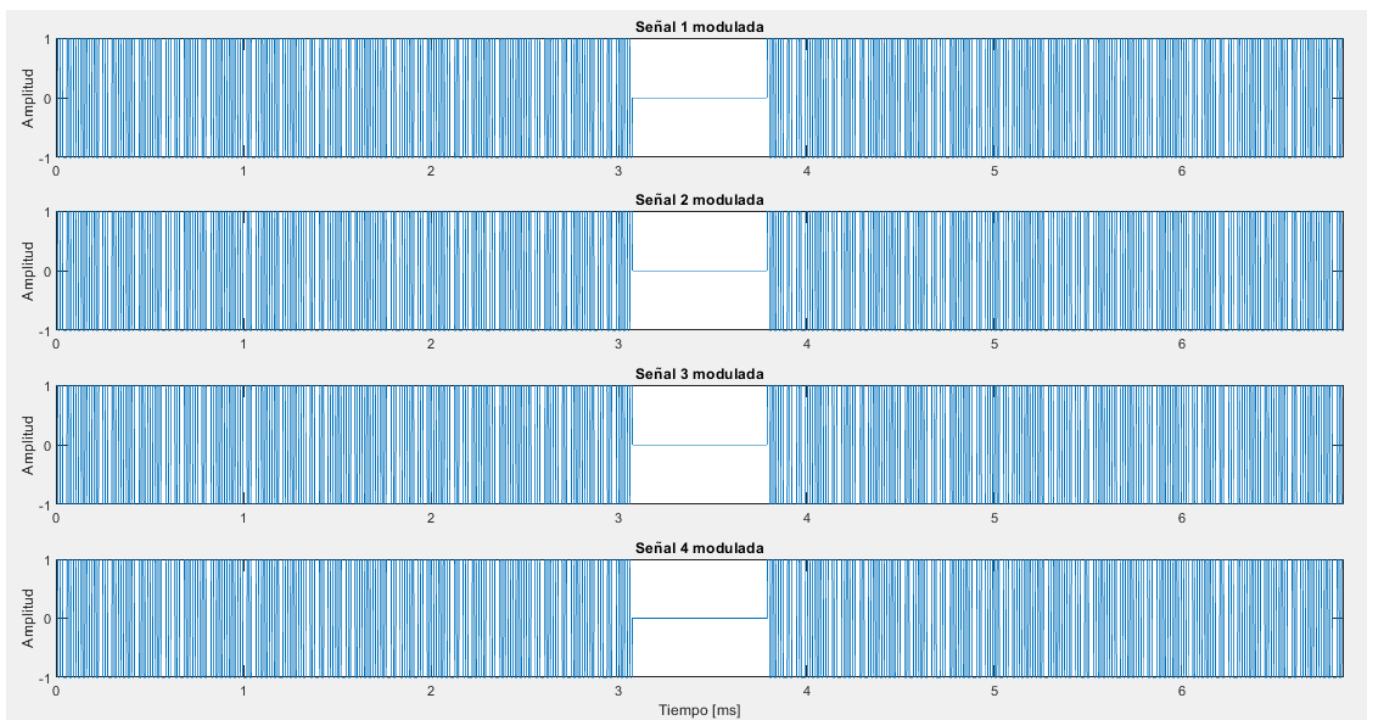


Fig. 61. Ejemplo ventana 2 LS: modulada

En la Fig. 62 aparece la señal modulada con el efecto del transductor.

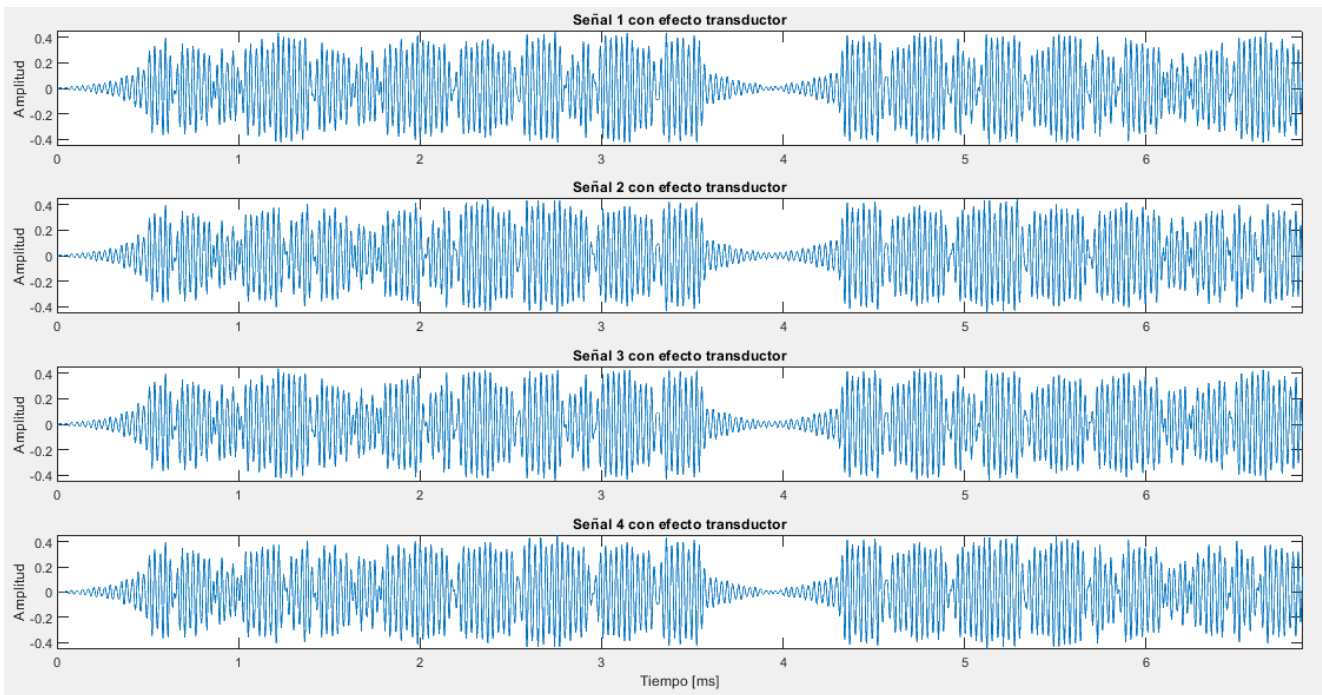


Fig. 62. Ejemplo ventana 2 LS: transductor

Por último, la Fig. 63 representa la señal a emitir por cada emisor.

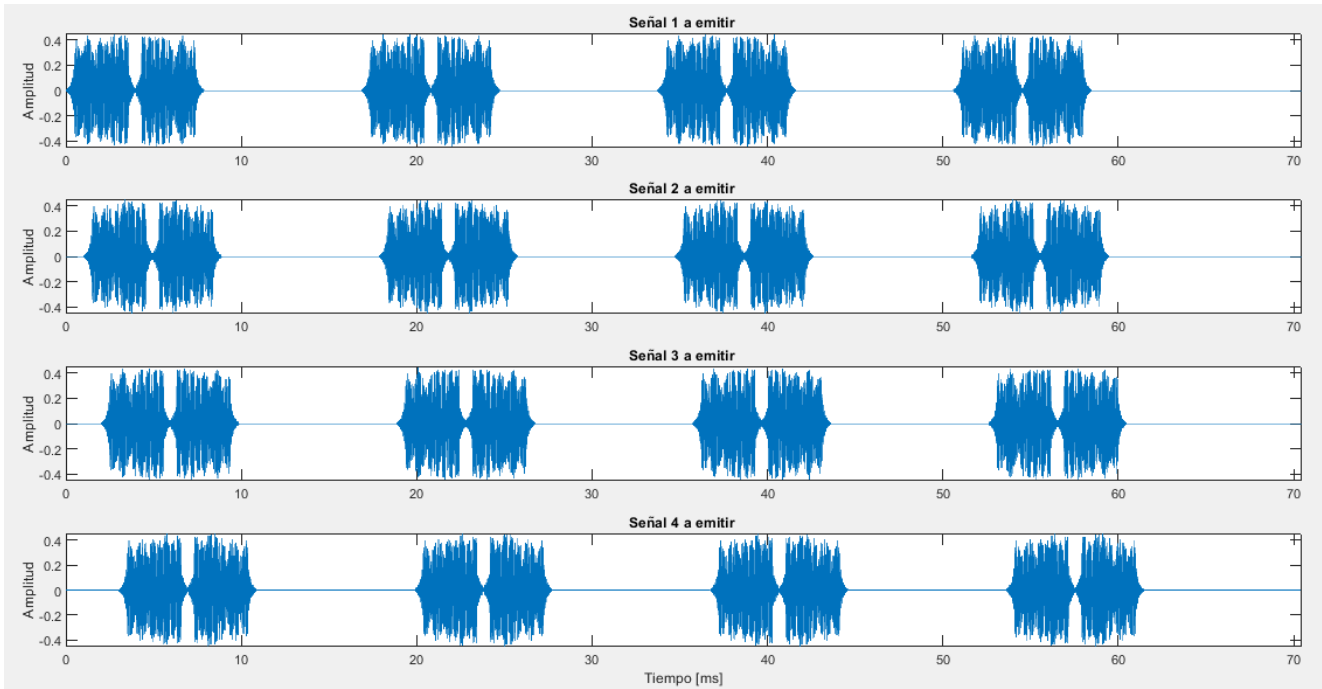


Fig. 63. Ejemplo ventana 2 LS: emisión

4.3.4. Zadoff-Chu

En la Fig. 64 se muestra la pantalla correspondiente a la configuración en los códigos Zadoff-Chu. Es similar al de las Kasami excepto por algunos matices que se exponen a continuación.

La longitud L ya no está restringida, pero se recomienda que sea un número primo ya que ofrece mejores resultados. M está definido como $M=L-1$.

Cuando se selecciona el *Nemo* se escogen, dentro de las M posibilidades, aquellos códigos que den mejores propiedades de correlación. Esto es, que su pico principal de correlación sea menor.

Al ser códigos compuestos por números complejos, a la hora de modularlos o aplicar correlación, se trabaja con ellos en valor absoluto. Además, se utiliza la modulación QPSK (21).

Fig. 64. Ventana 2 TFG: Configuración emisión Zadoff-Chu

A continuación, se presenta un ejemplo de la aplicación para los siguientes valores: $L=61$, $Nemo=5$, portadora sinusoidal $fc=42\text{kHz}$, $fse=500\text{kHz}$, $Nc=2$, $Tgap=0\text{ms}$, $Tshift=0\text{ms}$, $Nrep=4$ y con modelo transductor.

En la Fig. 65 se muestran los datos introducidos y los resultados obtenidos en M , η , Of y Te .

Configuración de emisión: Zadoff_Chú

Entradas

Longitud (L) Tamaño familia (M)

Nº emisores simultáneos (Nemo) Eficiencia energética (η)

Modulación QPSK:

Tipo portadora

Frecuencia portadora (fc) kHz Factor sobremuestreo (Of)

Frecuencia de muestreo (fse) kHz Duración emisión un código (Te) ms

Nº ciclos portadora (Nc)

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms **Imagen orientativa**

Número de repeticiones (Nrep)

Modelo transductor

Salidas

Fig. 65. Ventana 2 Zadoff-Chú: Configuración

Los archivos de almacenamiento serán .mat de 5x61 complejos para el caso de banda base y 5x1464 para el caso modulado.

La Fig. 66 muestra la visualización de los códigos generados en banda base. Para poder representarlos, tienen que estar en valor absoluto.

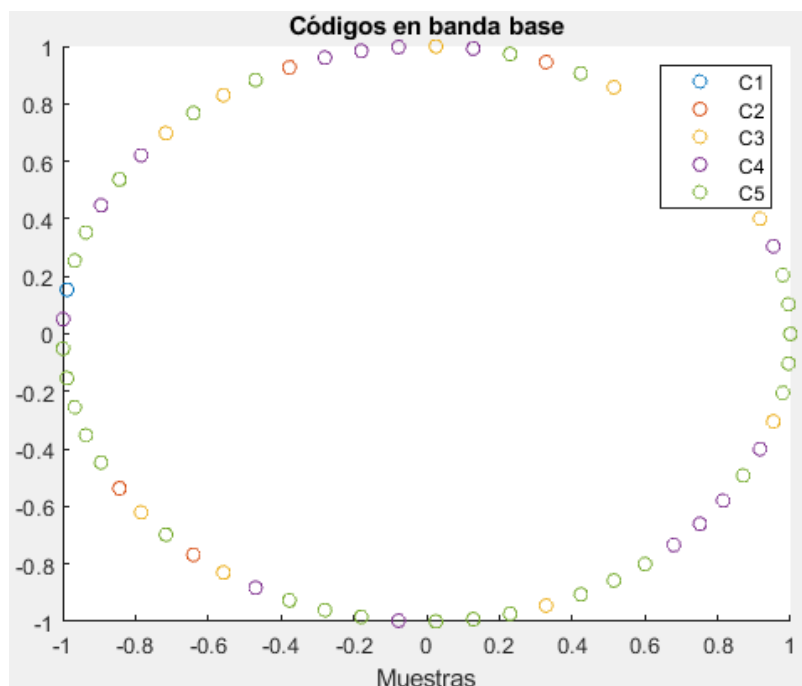


Fig. 66. Ejemplo ventana 2 Zadoff-Chú: banda base

Las Fig. 67 y 68 son el resultado de la portadora y de la señal modulada.

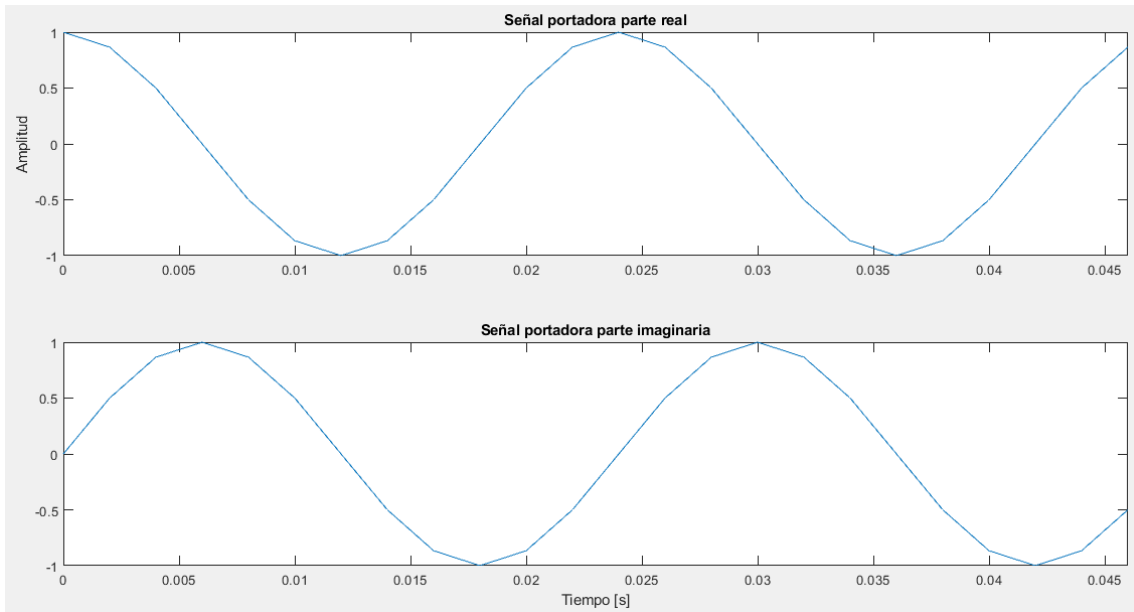


Fig. 67. Ejemplo ventana 2 Zadoff-Chu: portadora

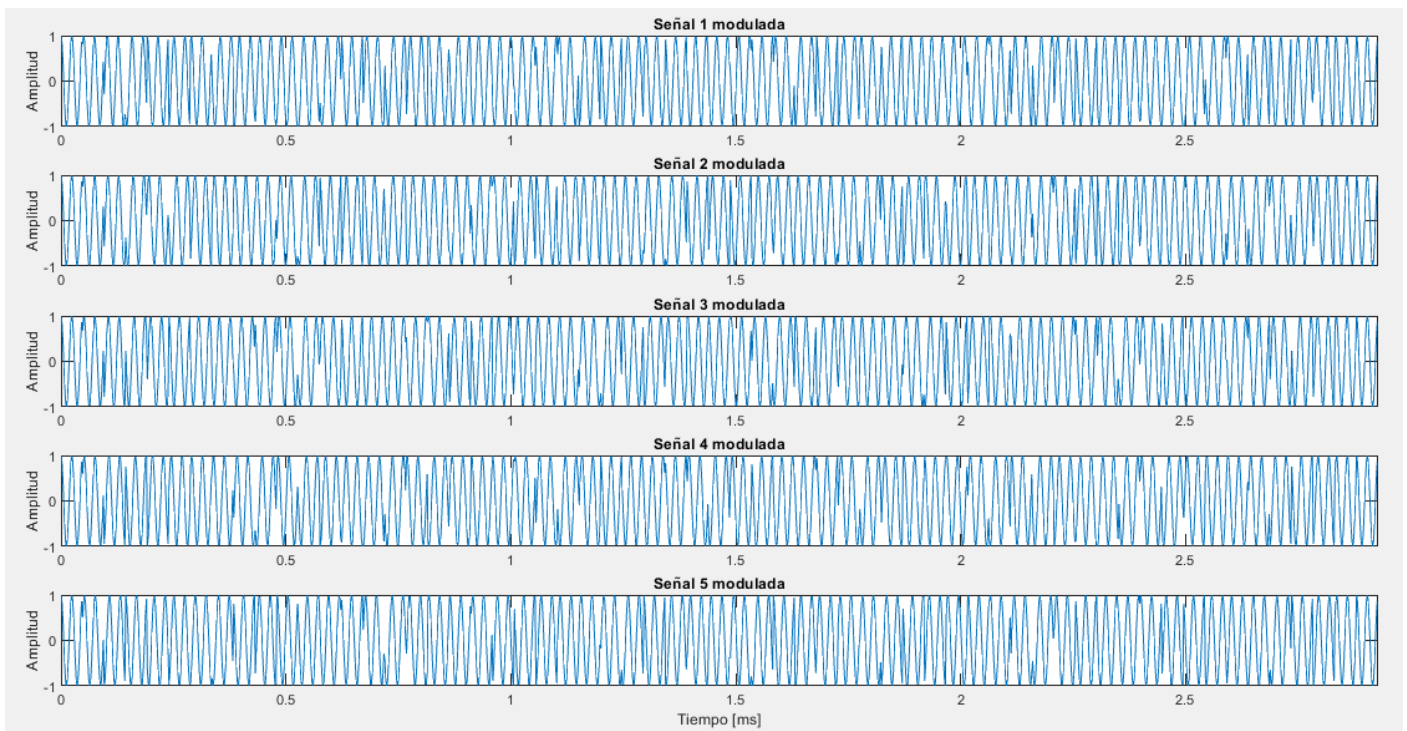


Fig. 68. Ejemplo ventana 2 Zadoff-Chu: modulada

En la Fig. 69 aparece la señal modulada con el efecto del transductor.

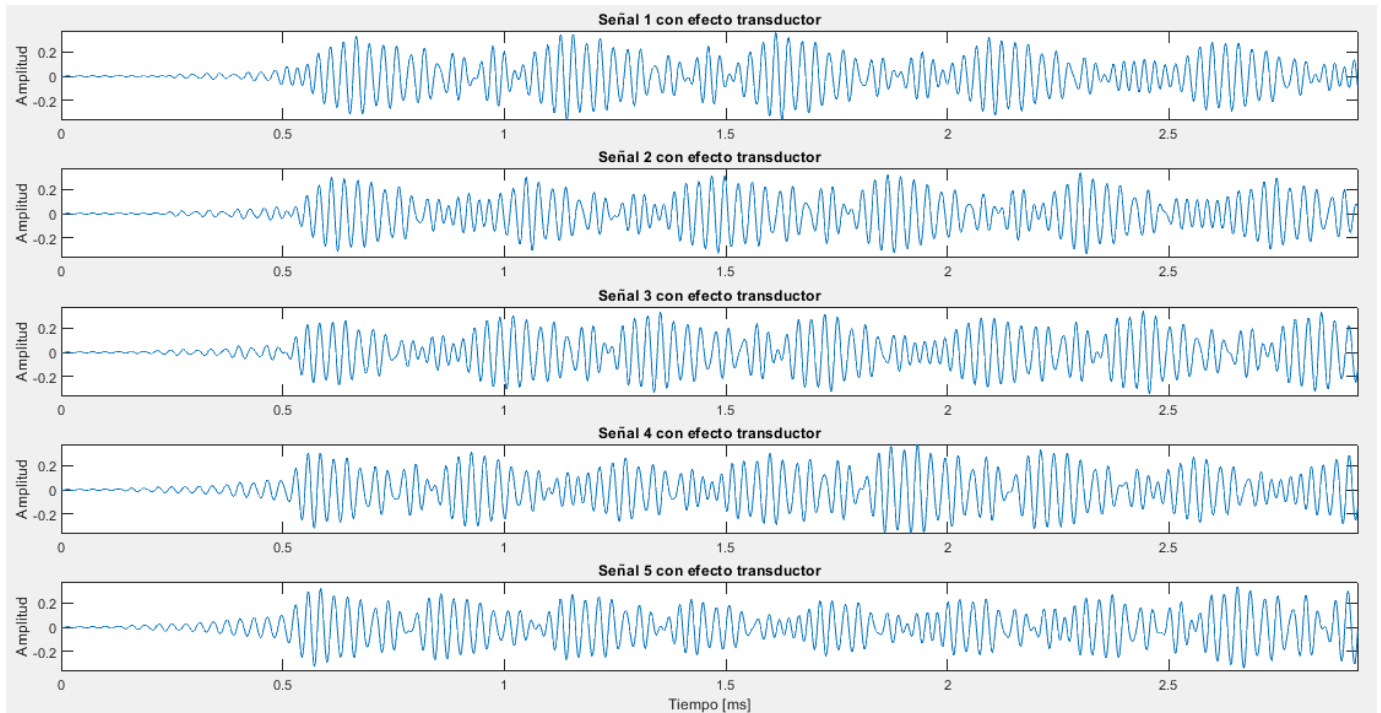


Fig. 69. Ejemplo ventana 2 Zadoff-Chu: transductor

Por último, la Fig. 70 representa la señal a emitir por cada emisor.

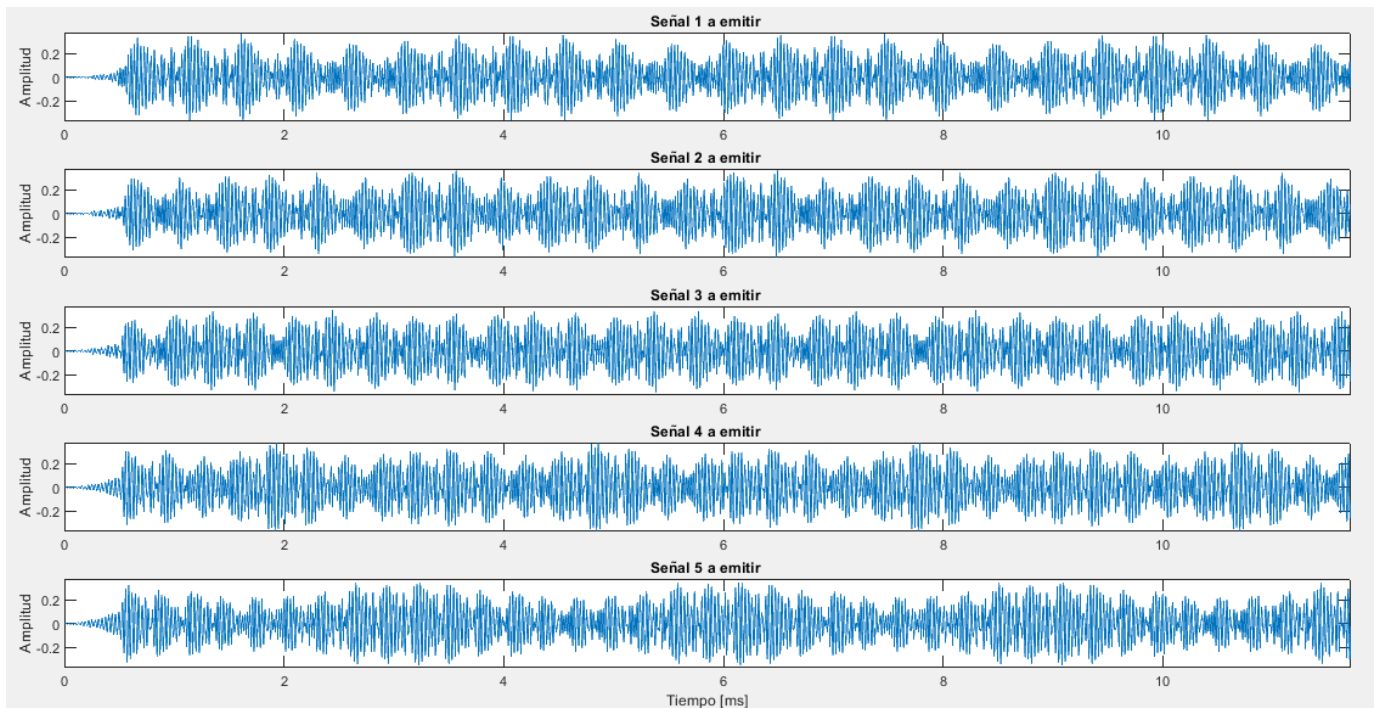


Fig. 70. Ejemplo ventana 2 Zadoff-Chu: emisión

4.4. CONFIGURACIÓN EFECTOS

La ventana de configuración de efectos es la misma para todos los códigos. Dicha ventana es opcional y se puede avanzar el programa sin tener que hacer uso de ella. Dentro de esta pantalla, se pueden diferenciar dos tipos de configuración: aplicación de efectos a la señal a emitir o simulación paramétrica de ruido en la señal a emitir. La primera opción aparece en la Fig. 71 en la cual salen señalados todos los efectos disponibles con sus respectivos parámetros de configuración. La segunda opción es la que representa la Fig. 72. Ambas opciones no pueden evaluarse a la vez, es decir, si por ejemplo se seleccionan los efectos cerca-lejos y multicamino, si después se selecciona la simulación paramétrica automáticamente se anulan y viceversa.

El botón ‘Valores por defecto’ introduce unos datos determinados para que el usuario pueda orientarse, ‘Borrar’ elimina cualquier valor que haya en los recuadros y ‘Ayuda’ abre el archivo correspondiente a dicha ventana en el Manual de Usuario adjuntado en esta memoria. ‘Volver’ redirige al programa a la configuración de las propiedades del código seleccionado, en la cual aparecerán los últimos datos que se introdujeron para su configuración, y ‘Siguiente’ continua a la visualización de resultados.

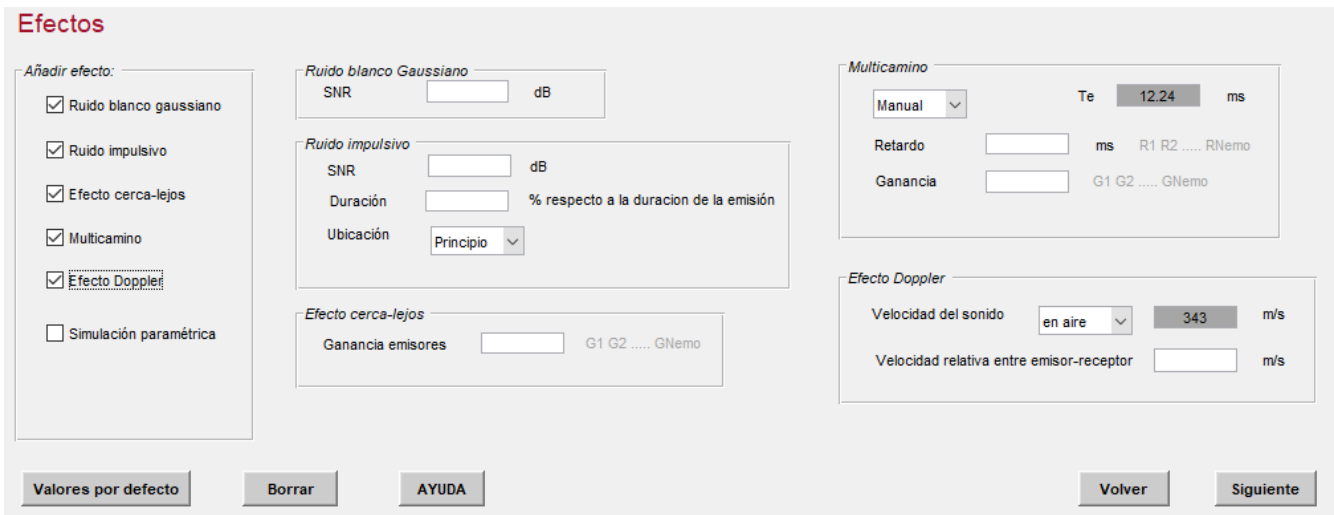


Fig. 71. Ventana 3 TFG: Configuración de efectos (1)

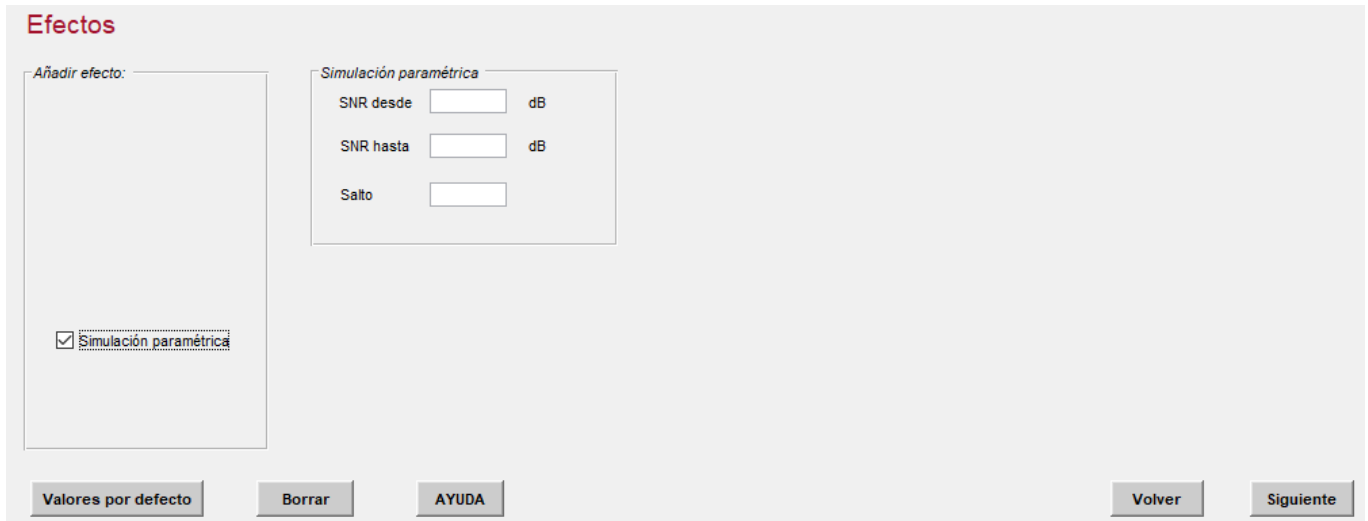


Fig. 72. Ventana 3 TFG: Configuración de efectos (2)

La primera opción permite señalar los efectos que se desee aplicar sin la necesidad de que sean todos. Algo a tener en cuenta es que, los efectos que no vayan a ser aplicados, deben quedar deseleccionados para que la aplicación evite extraer datos inexistentes. Una vez configurados, el programa aplica los efectos sobre la señal en este orden: cerca-lejos, multicamino, efecto Doppler, ruido blanco Gaussiano y finalmente ruido impulsivo. Esto hace que la señal que se utiliza para superponer el siguiente efecto no sea la ideal, si no la última resultante. Los efectos cerca-lejos y multicamino se aplican a cada uno de los *Nemo* códigos de la señal a emitir y el resto de efectos sobre la señal emitida la cual está compuesta por la suma de los códigos anteriores.

A continuación, se evalúan cada uno de los efectos:

El **Ruido Blanco Gaussiano** se configura a través de la relación señal-ruido SNR (*Signal-to-Noise Ratio*). Este parámetro se aplicará en toda la señal y causará mayor efecto cuanto más negativo sea el SMR (menor que cero).



Fig. 73. Ventana 3: Ruido Blanco Gaussiano

El **Ruido Impulsivo** aplica Ruido Blanco Gaussiano sólo en una zona de la señal. La duración que tendrá ese ruido se configurará en tanto por ciento respecto a la duración de la señal a emitir. Este efecto puede estar ubicado en varias zonas: principio, medio, final o aleatorio. Si es al principio comenzará en la primera muestra de la señal hasta la duración establecida, si es en el medio se repartirá de igual manera a ambos lados de la muestra situada a la mitad la duración establecida y si es al final se aplicará desde una muestra que resulte que la diferencia entre la última y ella sea la duración establecida. El modo aleatorio establece dos parámetros: la duración y la ubicación de la primera muestra. Esta duración se modifica cuando la muestra aleatoria está próxima al inicio o al final de la señal y no cabe.

Ruido impulsivo

SNR dB

Duración % respecto a la duracion de la emisión

Ubicación

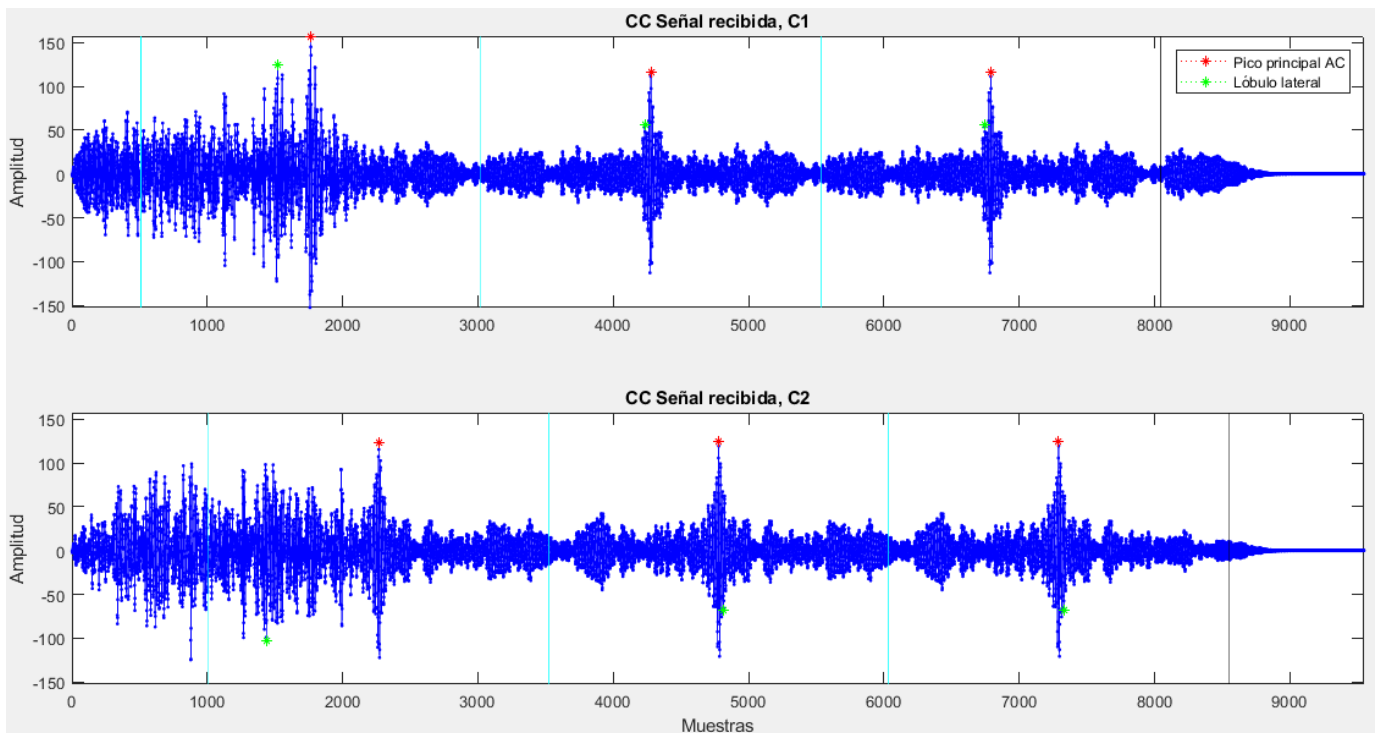


Fig. 74. Ventana 3: Ruido Impulsivo: Ejemplo

Para aplicar el efecto **cerca-lejos** el programa recoge los valores mediante un vector de datos separados por un espacio y de longitud $Nemo$. Si la cantidad de valores introducidos son distintos a $Nemo$, saltará un aviso y se deberá volver a ajustar. El primer dato está referido a la ganancia del código C_1 y así sucesivamente hasta llegar a la ganancia del código C_{Nemo} .

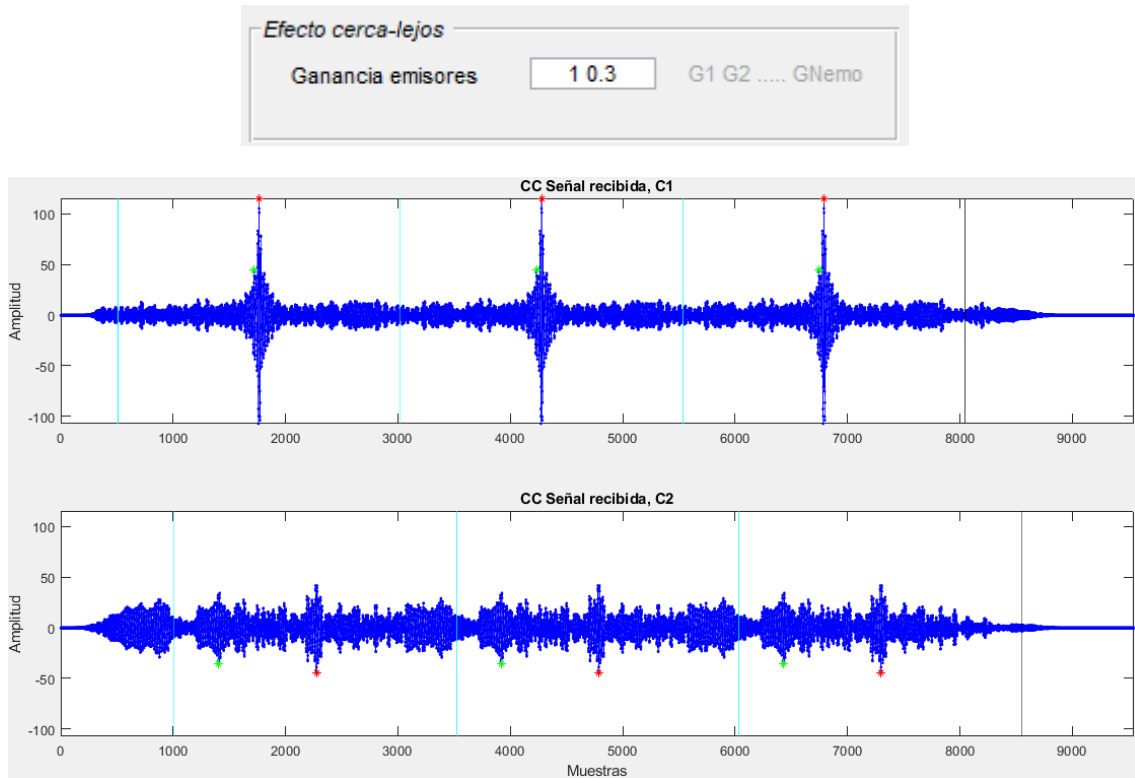


Fig. 75. Ventana 3: Efecto cerca-lejos: Ejemplo

El **multicamino** tiene dos opciones de configuración: manual o aleatoria. Como orientación, se muestra por pantalla el tiempo que dura un código Te . En el modo manual, al igual que el anterior, el programa recoge los datos mediante dos vectores de longitud $Nemo$, uno para los retardos y otro para las ganancias, que contienen la información para cada código separados por un espacio. Para simplificar, en este modo la señal solo va a rebotar una vez, es decir, solo va a tener un multicamino. El modo aleatorio está configurado para que cada código tenga dos multicaminos y que los retardos tengan un valor comprendido entre 40 y 1500. En este caso, cada valor de retardo tiene asignada una ganancia determinada a través de una función exponencial de la forma $x = 0.8e^{-t}$ representada en la Fig. 76 donde t es un vector que toma valores entre 0 y 1500 con un paso de 1 muestra.

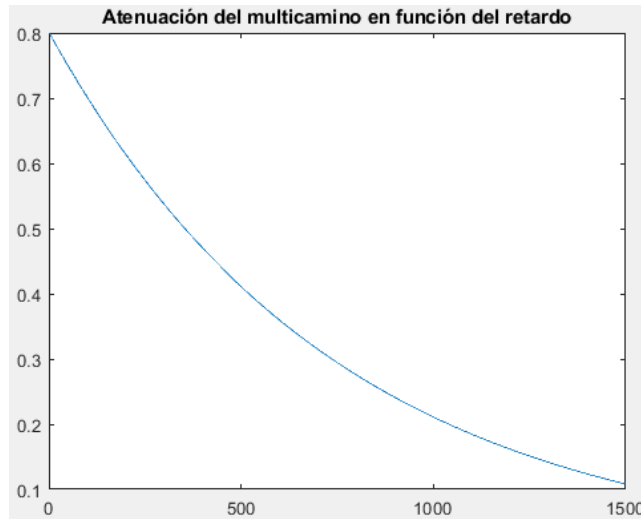


Fig. 76. Función cálculo multicamino modo aleatorio

Multicamino

Manual Te ms

Retardo ms R1 R2 RNemo

Ganancia G1 G2 GNemo

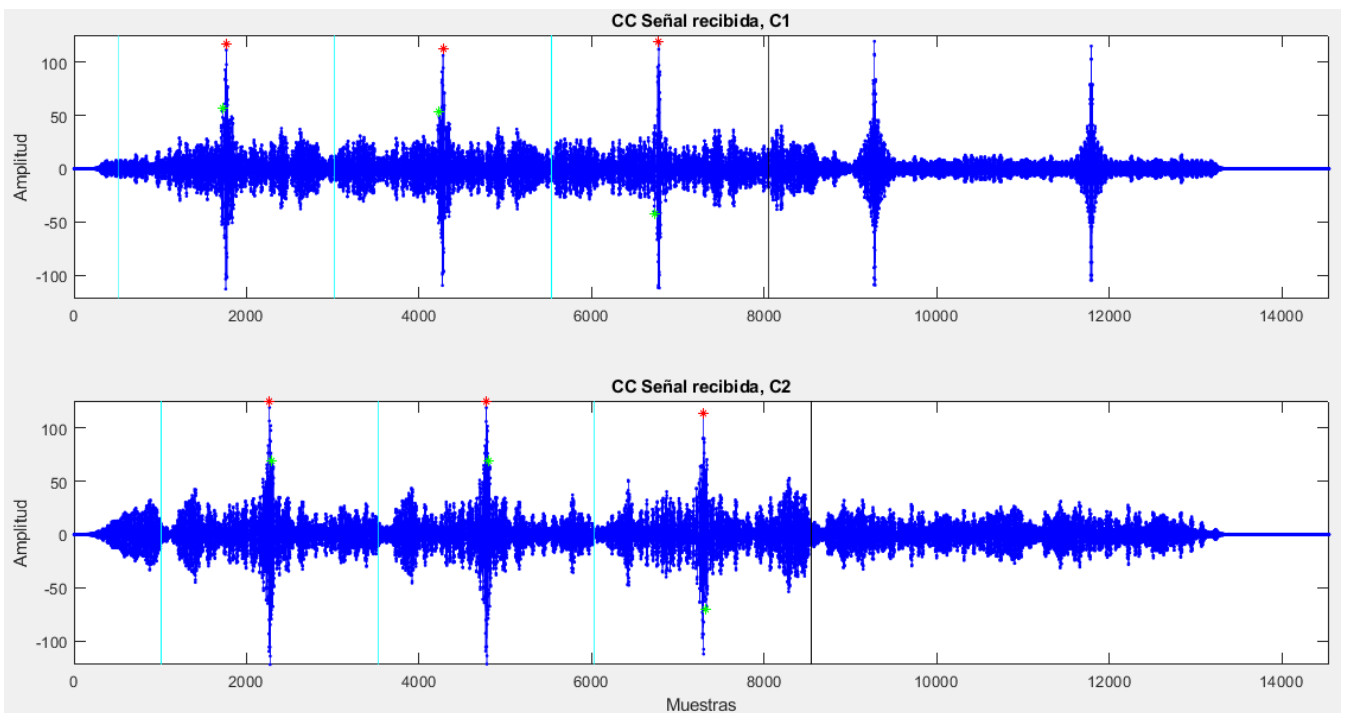


Fig. 77. Ventana 3: Multicamino: Ejemplo

El **efecto Doppler** tiene dos parámetros de configuración: la velocidad del sonido y la velocidad relativa entre emisor-receptor. En la primera, se puede elegir entre velocidad del sonido en el aire o en el agua y se mostrará por pantalla el valor que se va a aplicar según el caso. En el segundo parámetro, si se introduce un valor positivo quiere decir que emisor y receptor se están alejando y si es un valor negativo es que se están acercando. El objetivo de este efecto es resamplear la señal con una nueva frecuencia de muestreo obtenida como $f_{se}' = f_{se} \cdot (1 + v_{relativa}/v_{sonido})$.

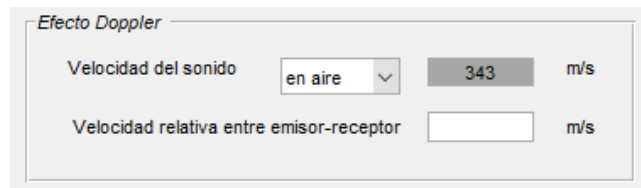


Fig. 78. Efecto Doppler

La segunda opción hace referencia a la **simulación paramétrica del ruido blanco gaussiano** en la señal emitida. En la siguiente pantalla de la aplicación (Visualización de resultados) se podrá observar el RMSE de cada código que existe entre el pico principal de la señal emitida idealmente y la señal emitida real, en función del rango de ruido establecido en esta ventana (SNR_desde:salto:SNR_hasta). El RMSE debe tender a cero a medida que el ruido aumenta en valor.

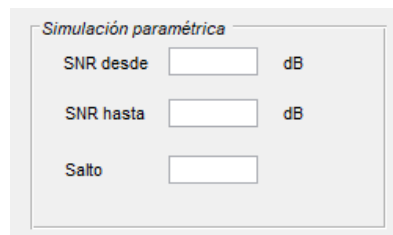


Fig. 79. Simulación paramétrica

4.5. VISUALIZACIÓN DE RESULTADOS

4.5.1. Introducción

En esta pantalla se analizan los resultados obtenidos en los distintos procesos de configuración de la señal. El botón ‘Ayuda’ abre el archivo correspondiente a dicha ventana en el Manual de Usuario adjuntado en esta memoria, ‘Volver’ redirige al programa a la configuración de efectos en la señal, ‘Inicio’ vuelve a la ventana de presentación de la app y ‘Salir’ lleva a una última pantalla informativa que enlaza este programa con el proyecto “*Locate-US*” que ofrece la posibilidad de realizar pruebas experimentales reales con los códigos configurados.

En función de lo que se indique en la pestaña ‘Elección de resultados’, se van a ofrecer distintas ventanas con una serie de resultados determinados. Estas ventanas se pueden agrupar en tres casos generales: señal antes de emitir (tres primeras opciones, Fig. 80), señal tras ser emitida (cuarta y quinta opción, Fig. 86) y simulación paramétrica del ruido (última opción, Fig. 90). Por el momento, se definen las particularidades de cada caso general sin entrar en detalles del código que se está visualizando y más adelante se explican cada una de las opciones de resultados que se pueden analizar. La Tabla 1 recoge los distintos casos y subcasos que se van a explicar a lo largo de este apartado.

Casos generales	Descripción	Subcasos
1º	Señal antes de emitir	Banda base
		Modulada
		Con transductor
2º	Señal recibida	Con modulación y efecto del transductor
		Con efectos añadidos
3º	Simulación paramétrica	

Tabla 1. Resumen de las distintas ventanas de la pantalla de resultados

Caso 1. Señales antes de emitir

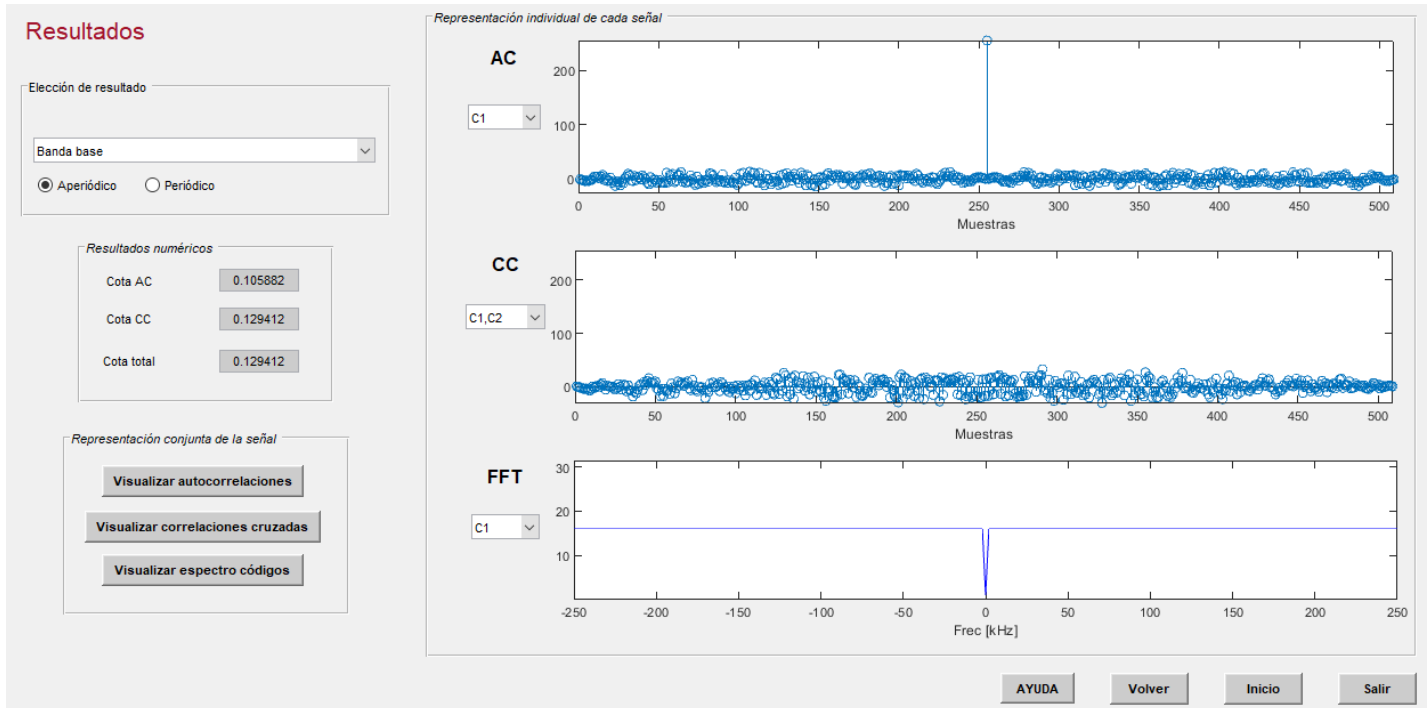


Fig. 80. Ventana 4 TFG: Resultados caso 1

En el **primer caso**, se distinguen varias secciones:

- **Selección de emisión:** periódica o aperiódica. Si es aperiódica, se realizan las correlaciones con las señales directamente diseñadas. Si es periódica, se crea una nueva señal para cada código en la que aparece su contenido repetido tres veces seguidas sin espacio entre ellos. Estas nuevas señales son las que se correlan con las señales iniciales (sin repeticiones). Como resultado, aparecen tres picos máximos de correlación, dos a los laterales y uno en el medio. Se recorta la señal a los lados de tal forma que se solamente se quede la correlación del medio que es cuando existe el caso periódico. En función de la opción que se seleccione, las gráficas tendrán diferentes resultados de correlación siendo más favorables en el caso periódico, como se comentó anteriormente.

Las Fig. 81 y 82 muestran la comparación de resultados ofrecidos para un mismo código por las dos opciones de visualización: aperiódica y periódica.

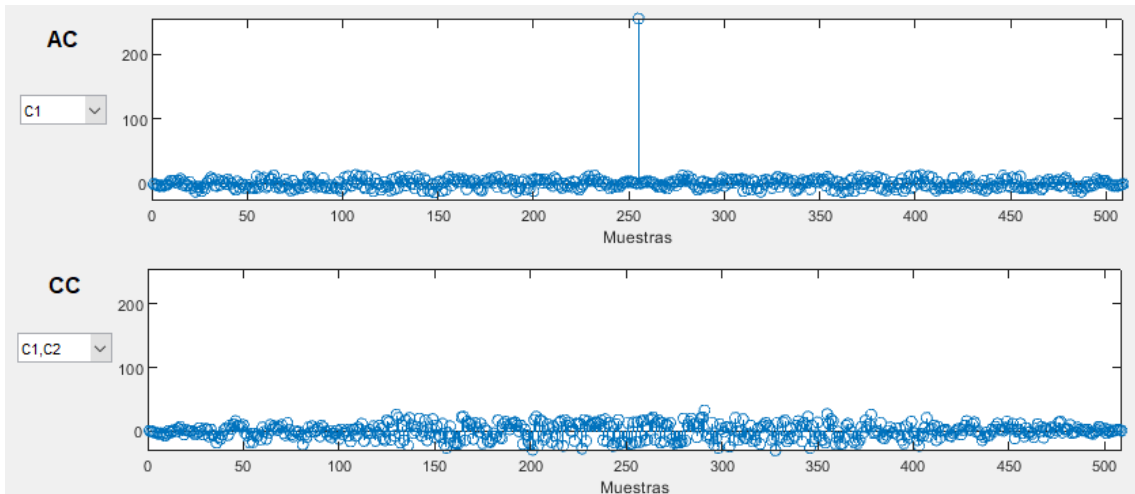


Fig. 81. Ventana 4: Resultados caso 1, Representación aperiódica

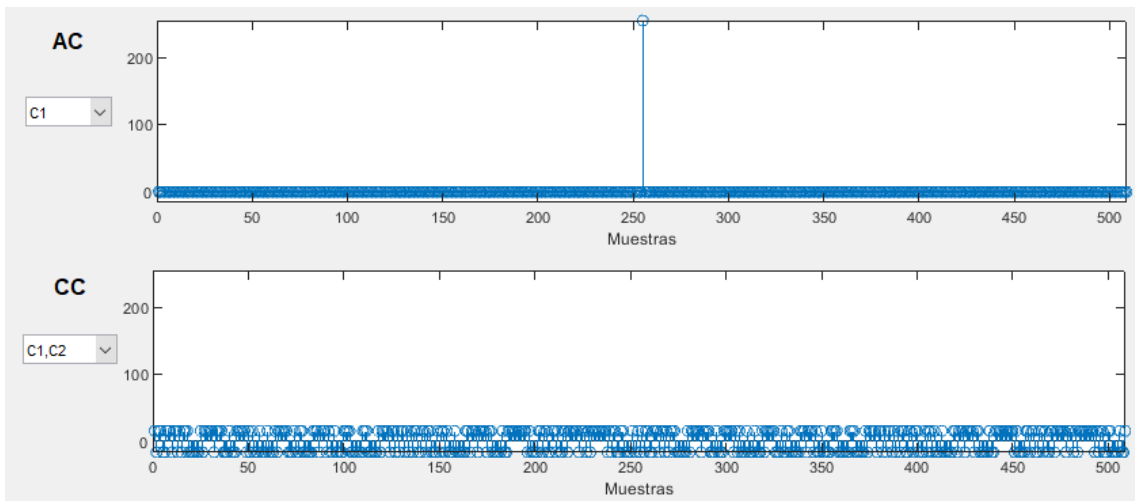


Fig. 82. Resultados caso 1, Representación periódica

- Resultados numéricos: Cota AC θ_{AC} , Cota CC θ_{CC} y cota total θ . Estas cotas equivalen a la relación entre el lóbulo principal y el lóbulo lateral en las correlaciones. Para las AC, el programa localiza el máximo valor de la señal, el cual debe estar a L muestras, y lo anula. Cuando la señal está modulada, se opta además por anular una zona $Of \cdot Nc$ alrededor del pico principal para evitar los elevados valores que se pueden dar en esa zona por efecto de la modulación y que llevarían a un mal cálculo de las cotas. Seguidamente, se vuelve a buscar el máximo de la señal restante y se asigna como lóbulo lateral. Para las CC, se utiliza el pico principal obtenido en las AC y se obtiene el lóbulo lateral de CC como el máximo valor de la señal. Se calculan para cada código y se muestra por pantalla el máximo de todos ellos.
- Representación individual de la AC, CC y FFT de cada código. Mediante las pestañas que aparecen junto con las gráficas, se selecciona aquella AC o FFT de código C_i , o CC de códigos C_i, C_j que se quiera visualizar siendo $1 \leq i \leq Nemo-1$; $2 \leq j \leq Nemo$.

- Representación conjunta de la AC, CC o FFT. En esta representación aparece para las AC, marcado con un asterisco rojo el pico principal de correlación y para las AC y CC, con un asterisco verde el lóbulo lateral obtenidos para el cálculo de θ_{AC} y de θ_{CC} .

Las Fig. 83, 84 y 85 corresponden a las representaciones de la última sección.

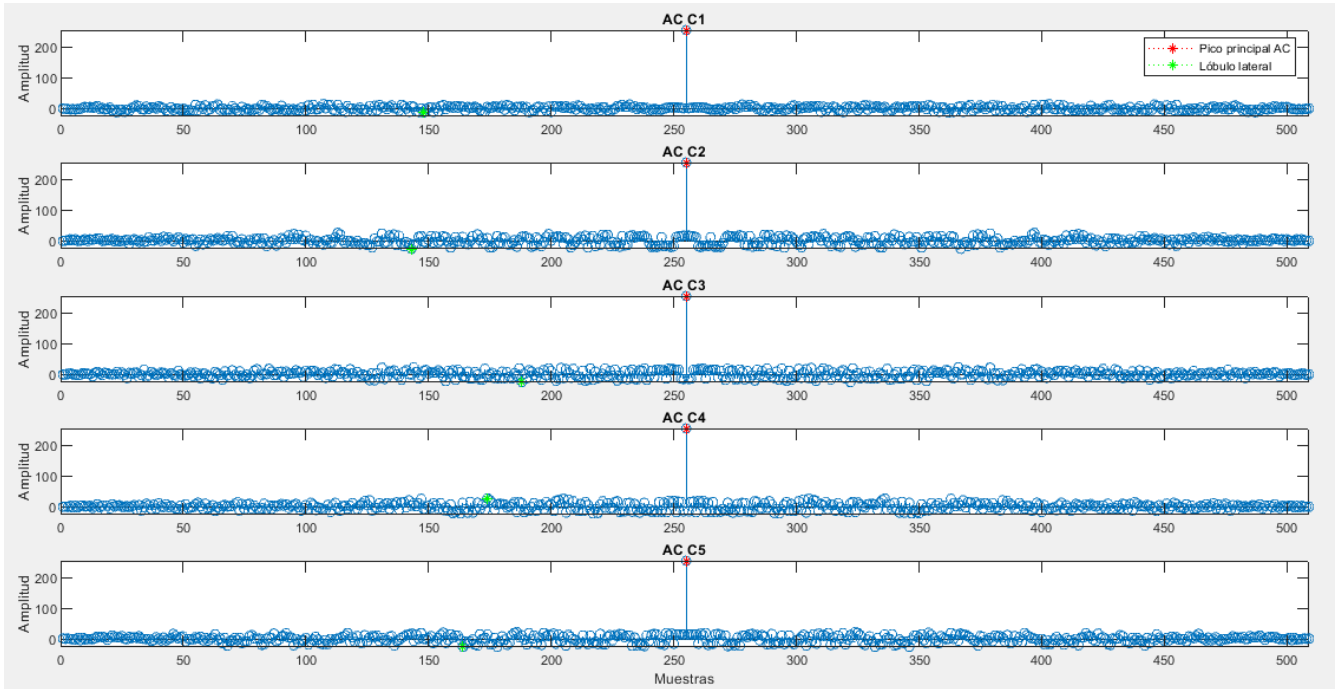


Fig. 83. Ventana 4: Resultados caso 1, Representación AC

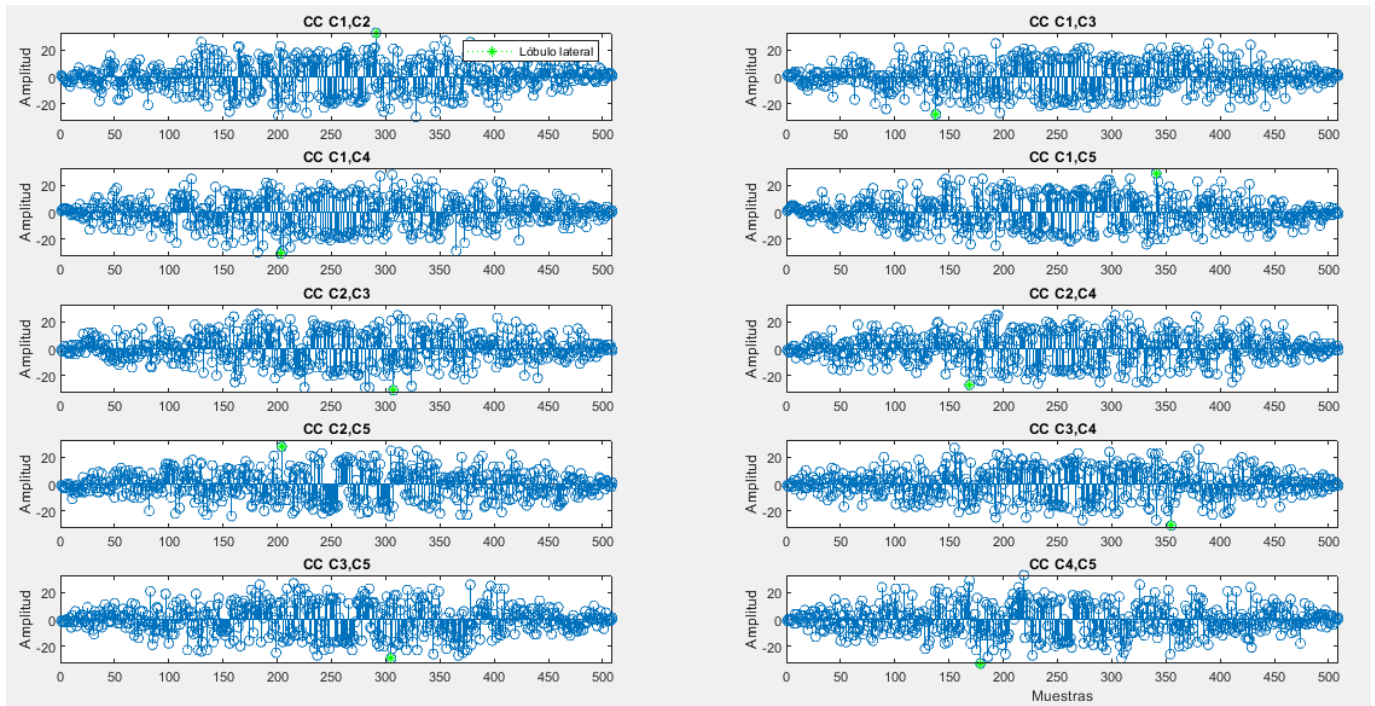


Fig. 84. Ventana 4: Resultados caso 1, Representación CC

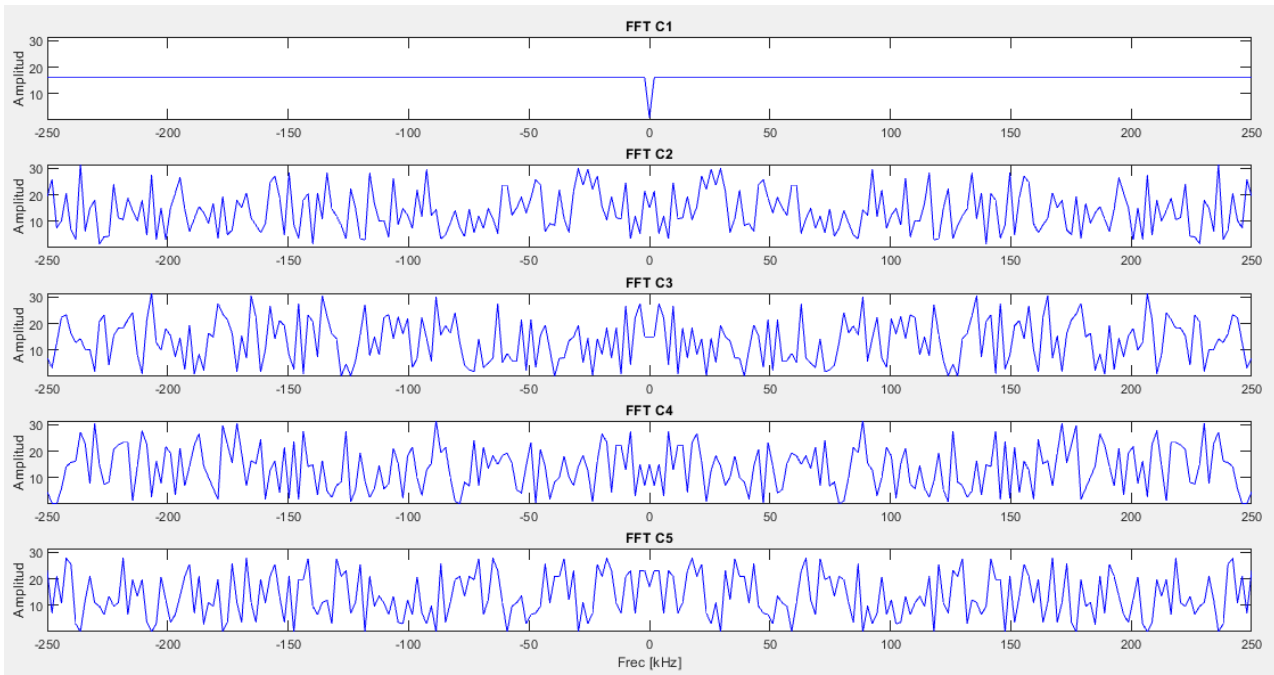


Fig. 85. Ventana 4: Resultados caso 1, Representación FFT

Caso 2. Señal recibida

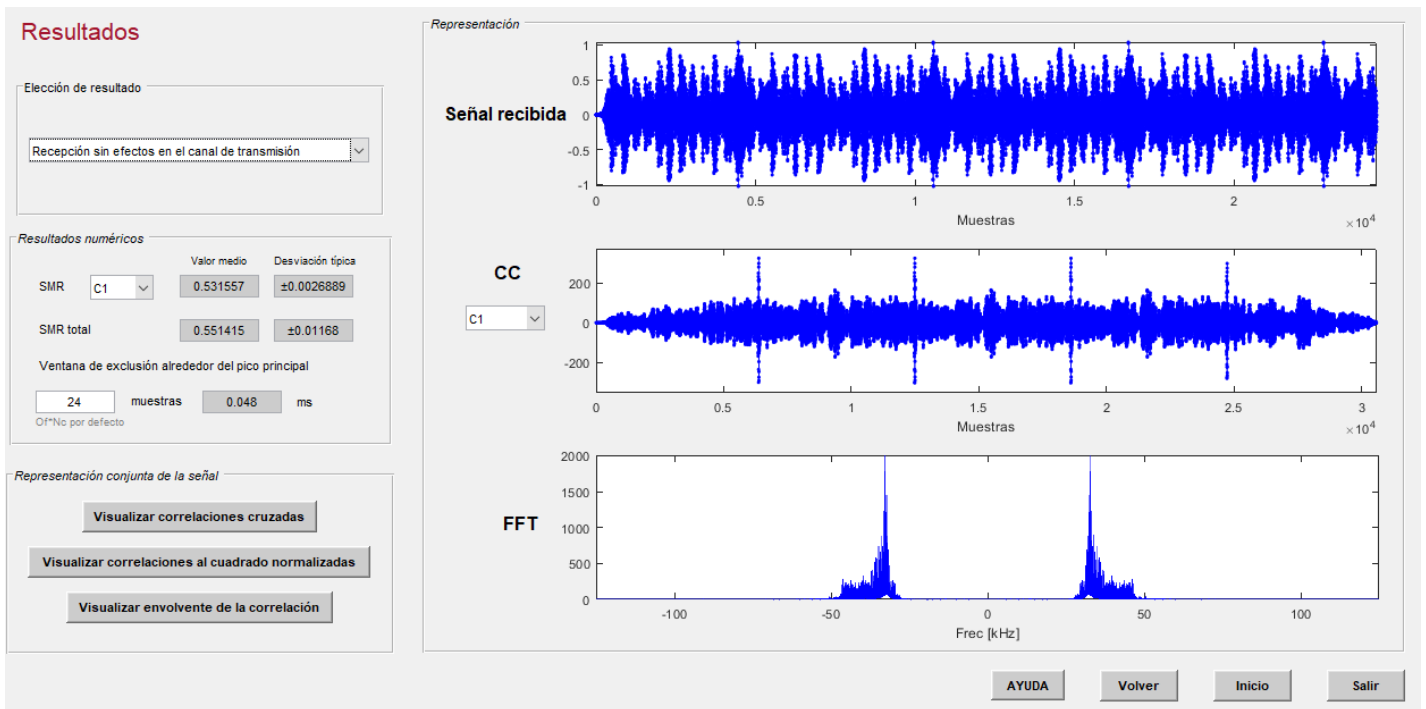


Fig. 86. Ventana 4 TFG: Resultados caso 2

En el **segundo caso**, se distinguen también varias secciones:

- **Resultados numéricos:** SMR de cada código y SMR total. La señal emitida es el resultado de la suma de todos los códigos los cuales se han repetido N_{rep} veces. Esto provoca que cuando se efectúa la correlación de la señal emitida con la señal modulada del código C_i ($1 \leq i \leq N_{emo}$) aparezcan N_{rep} picos principales de correlación. Por lo tanto, se forman N_{rep} ventanas en cada correlación y se calculan la θ_{AC} de cada una. El SMR de cada código se obtiene mediante la media de dichas θ_{AC} y la desviación típica. Se visualiza por pantalla el valor del código seleccionado a través del menú. El SMR total será el máximo de entre todos los códigos. Ya se comentó que, una vez modulada la señal, se pierden las propiedades de correlación y aparecen valores muy elevados alrededor del pico principal que pueden dar resultados erróneos de SMR. Para resolver este problema se opta por anular, a parte del pico principal, una zona alrededor suya que por defecto es $N_c \cdot Of$ pero que se puede modificar en esta pantalla.
- **Representación de la señal emitida** y su FFT, y de las CC entre dicha señal y el código modulado C_i seleccionado.
- **Representación conjunta** de las CC, de las CC al cuadrado normalizadas y de la envolvente de las CC (transformada de Hilbert). Junto con las dos primeras gráficas aparecen marcadas las ventanas que se han utilizado para el cálculo parcial de cotas, los lóbulos principales marcados con un asterisco rojo y los

lóbulos laterales con un asterisco verde obtenidos para el cálculo del SMR. Las Fig. 87, 88 y 89 corresponden a las representaciones de la última sección.

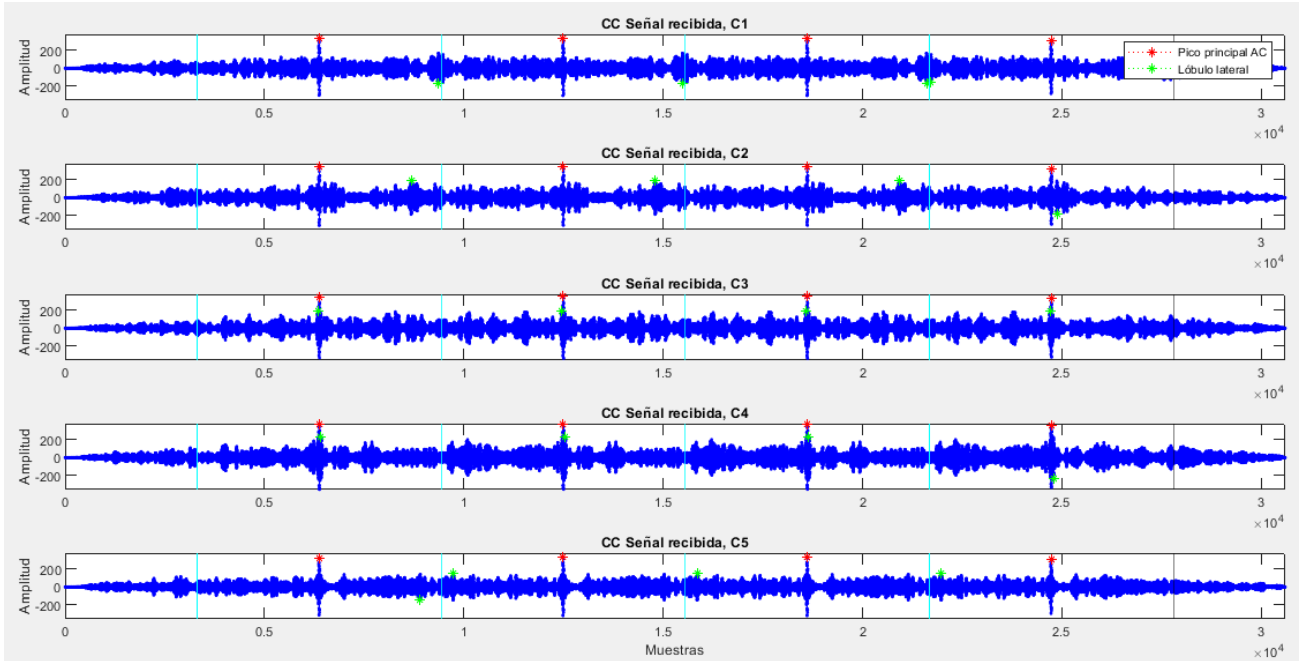


Fig. 87. Ventana 4: Resultados caso 2, Visualización CC con señal emitida

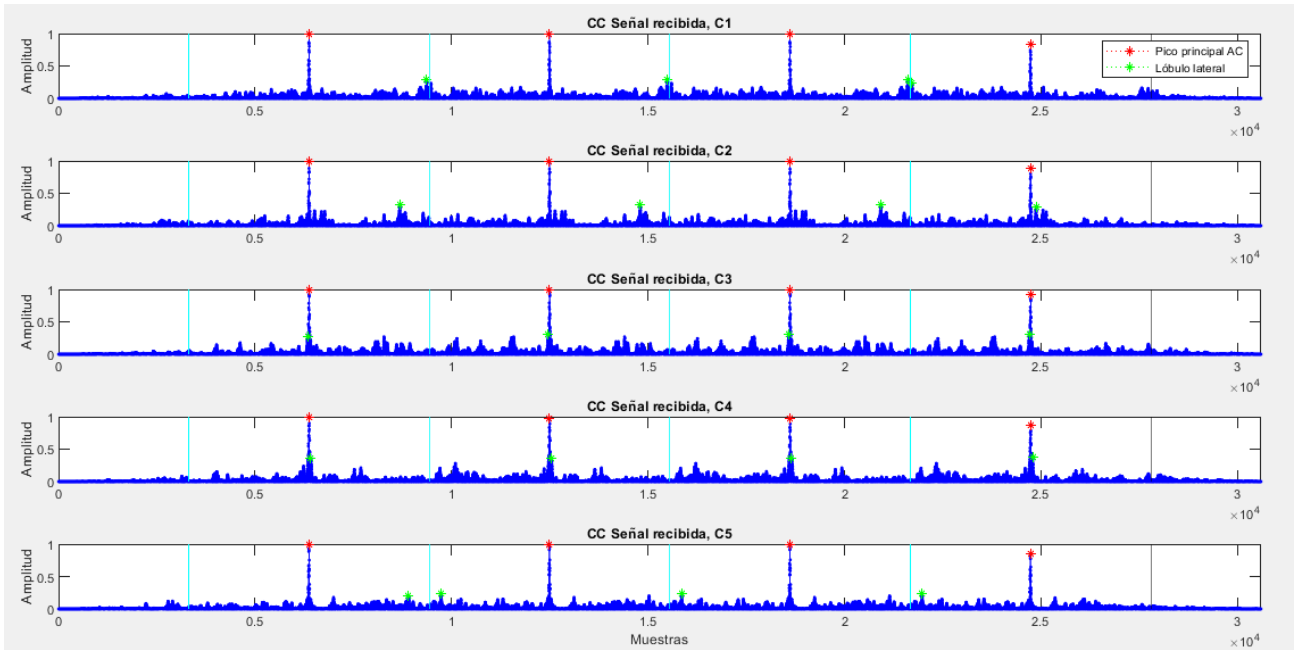


Fig. 88. Ventana 4: Resultados caso 2, Visualización correlaciones al cuadrado normalizadas

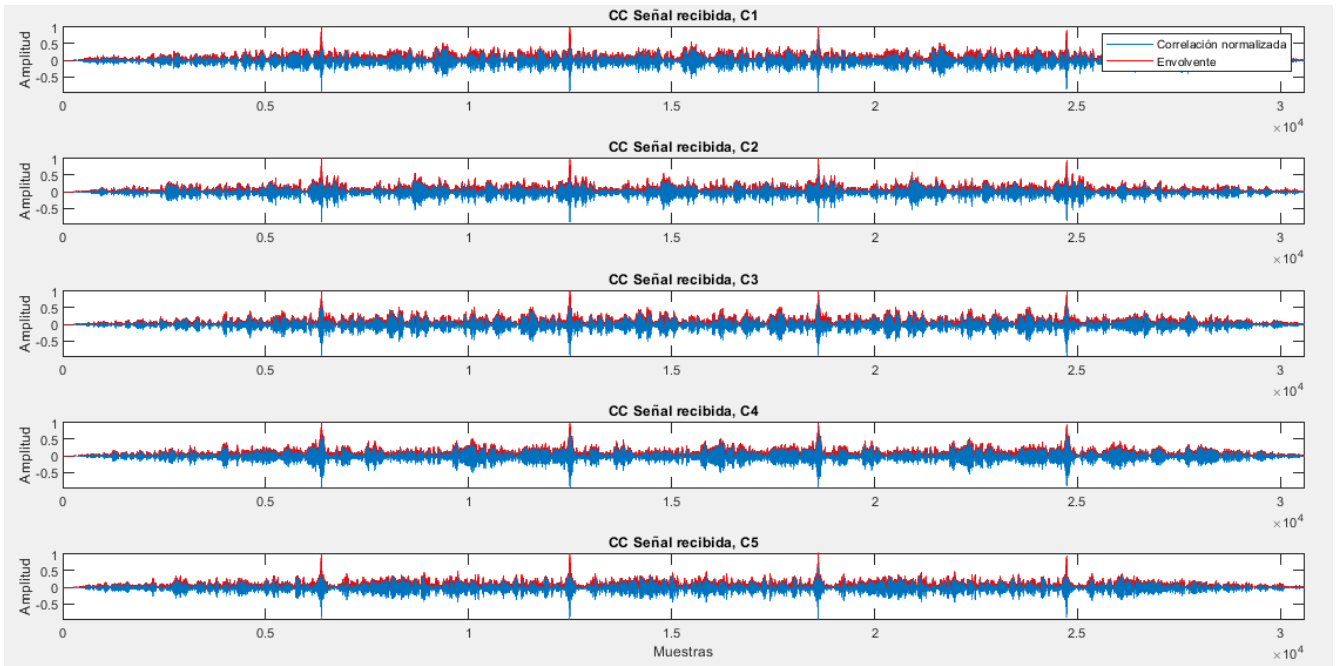


Fig. 89. Ventana 4: Resultados caso 2, Visualización envolvente de correlación

Caso 3. Simulación paramétrica

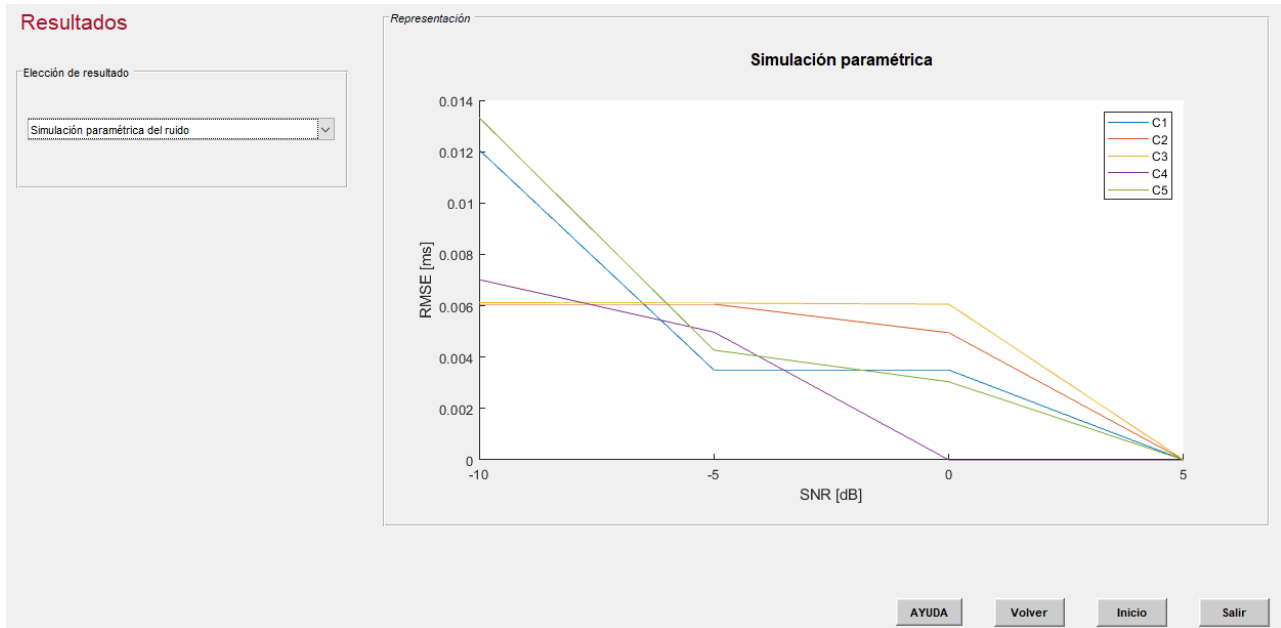


Fig. 90. Ventana 4 TFG: Resultados caso 3

El **tercer caso** representa el RMSE para cada código C_i en función de la simulación paramétrica de ruido configurada en la pantalla de efectos. Este parámetro tiende a cero cuanto menor es el ruido.

4.5.2. Caso práctico: Kasami

A continuación, se evalúa un caso práctico de las secuencias Kasami recorriendo todas las opciones que se pueden seleccionar: banda base, señal modulada, señal con efecto del transductor, señal recibida, señal recibida con efectos añadidos y simulación paramétrica del ruido. Además, en las tres primeras opciones se analiza la emisión en modo periódico y aperiódico. Las características de configuración son las siguientes: $L=255$, $Nemo=5$, portadora sinusoidal, $f_c=41,667\text{kHz}$, $f_{se}=500\text{kHz}$, $N_c=2$, $T_{gap}=2\text{ms}$, $T_{shift}=4\text{ms}$, $N_{rep}=4$ y con transductor. A la señal se le añade el efecto cerca-lejos con ganancias 0,5; 3; 4; 0,9; 0.2 y ruido blanco gaussiano con $SNR=-5\text{ dB}$.

En **banda base** y en modo **aperiódico** la ventana se muestra en la Fig. 91. El resultado de las cotas es de $\theta_{AC}=0,1058$, $\theta_{CC}=0,1294$ y $\theta=0,1294$. Las autocorrelaciones AC, correlaciones cruzadas CC y las FFT se dan en las figuras 92, 93 y 94 respectivamente. Esta última va a ser similar para el caso periódico y aperiódico por lo que sólo se representarán una vez.

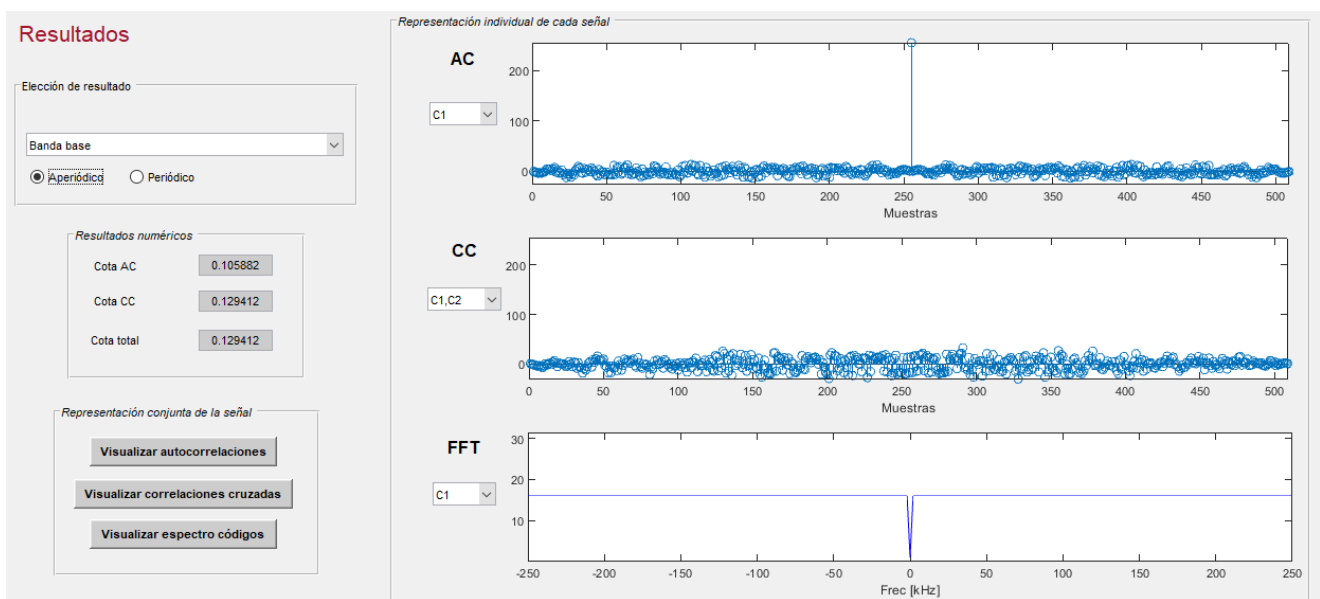


Fig. 91. Ventana 4 Kasami: banda base aperiódica

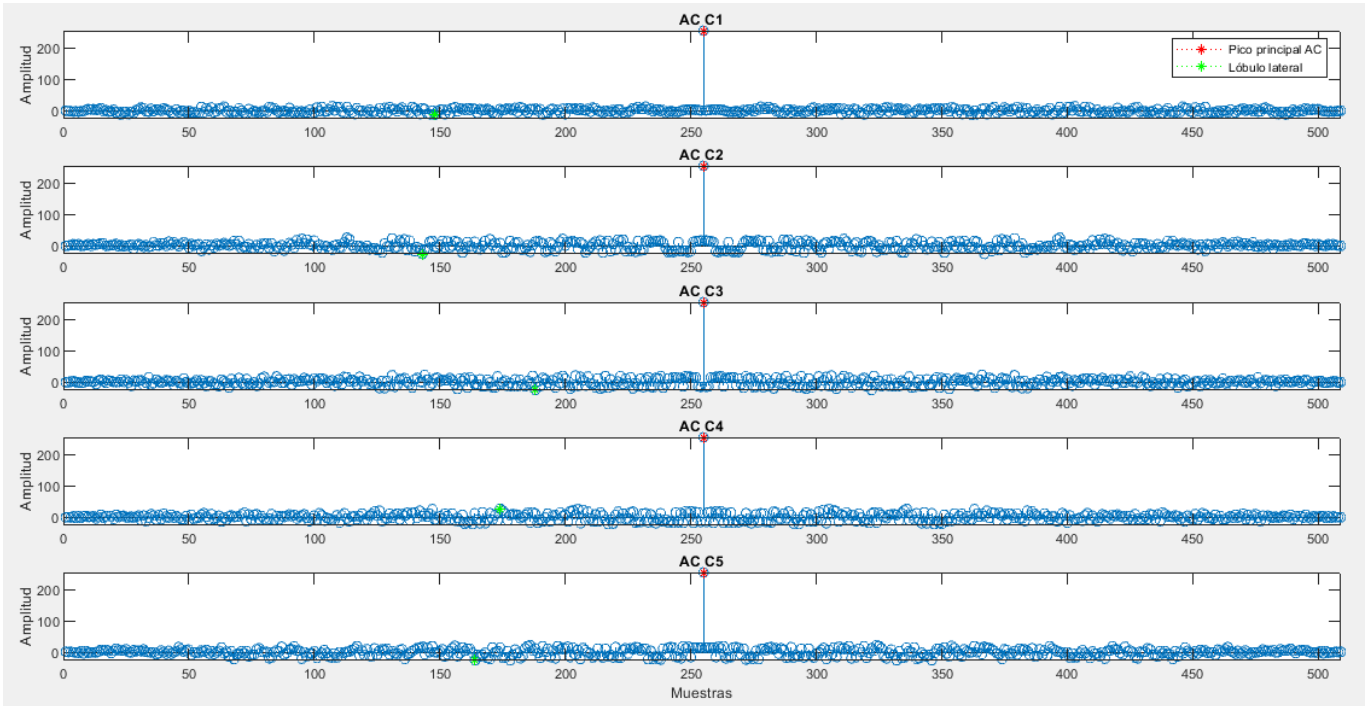


Fig. 92. Ventana 4 Kasami: AC banda base aperiódica

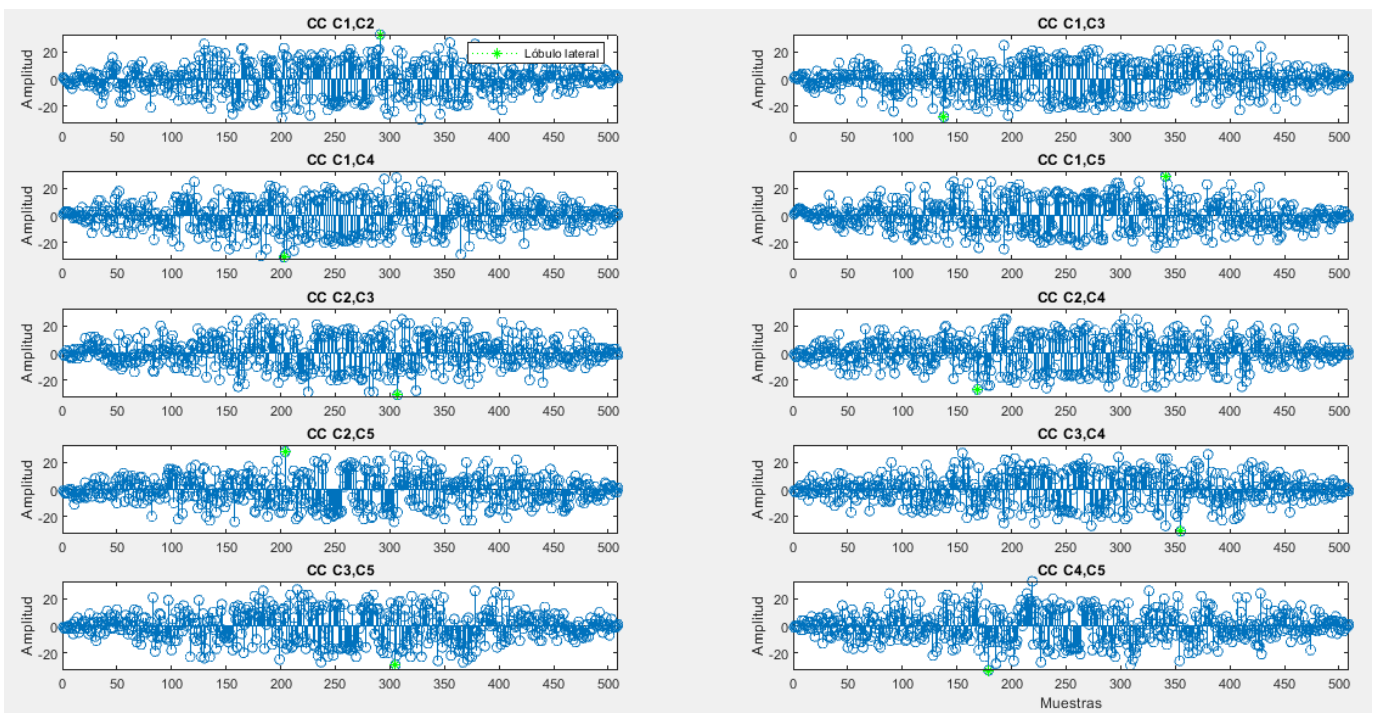


Fig. 93. Ventana 4 Kasami: CC banda base aperiódica

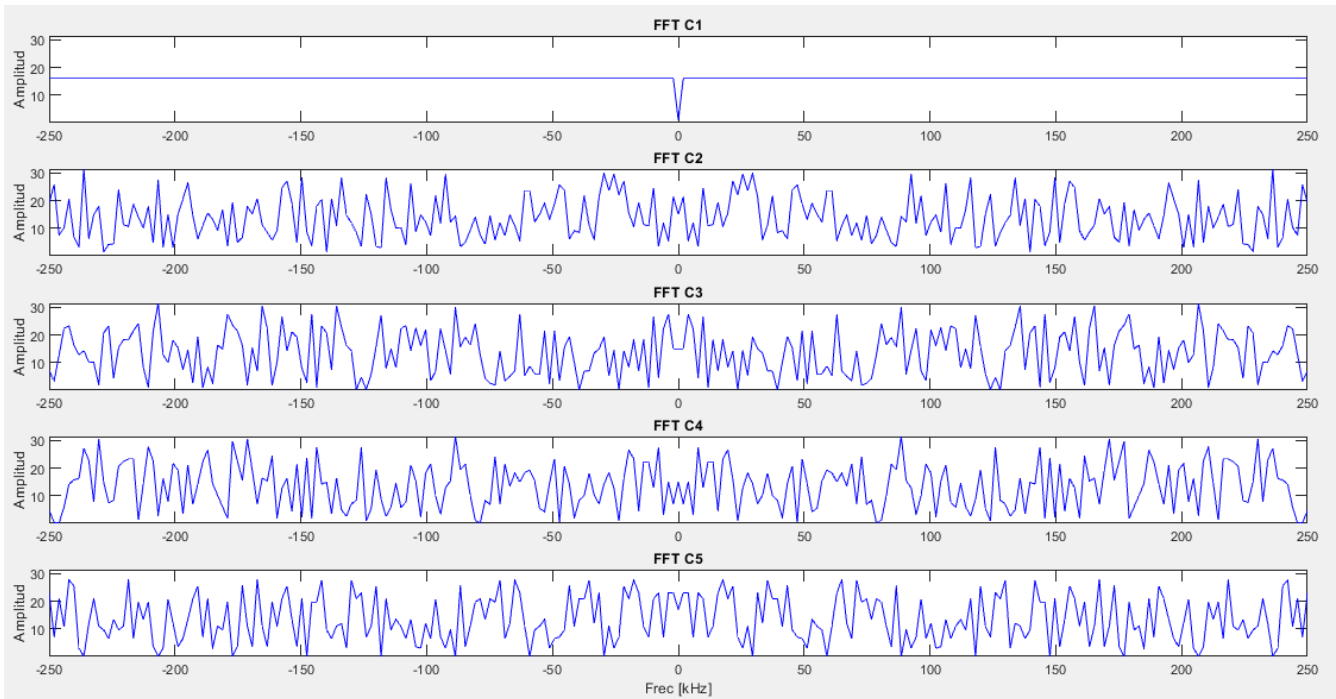


Fig. 94. Ventana 4 Kasami: FFT banda base

La figura 95 muestra la ventana **en banda base** y en modo **periódico**. El resultado de las cotas es de $\theta_{AC}=0,0667$, $\theta_{CC}=0,0667$ y $\theta=0,0667$. Como se puede comprobar, dan mejores resultados que en el modo aperiódico. Las autocorrelaciones AC, correlaciones cruzadas CC se muestran en las figuras 96 y 97 respectivamente.

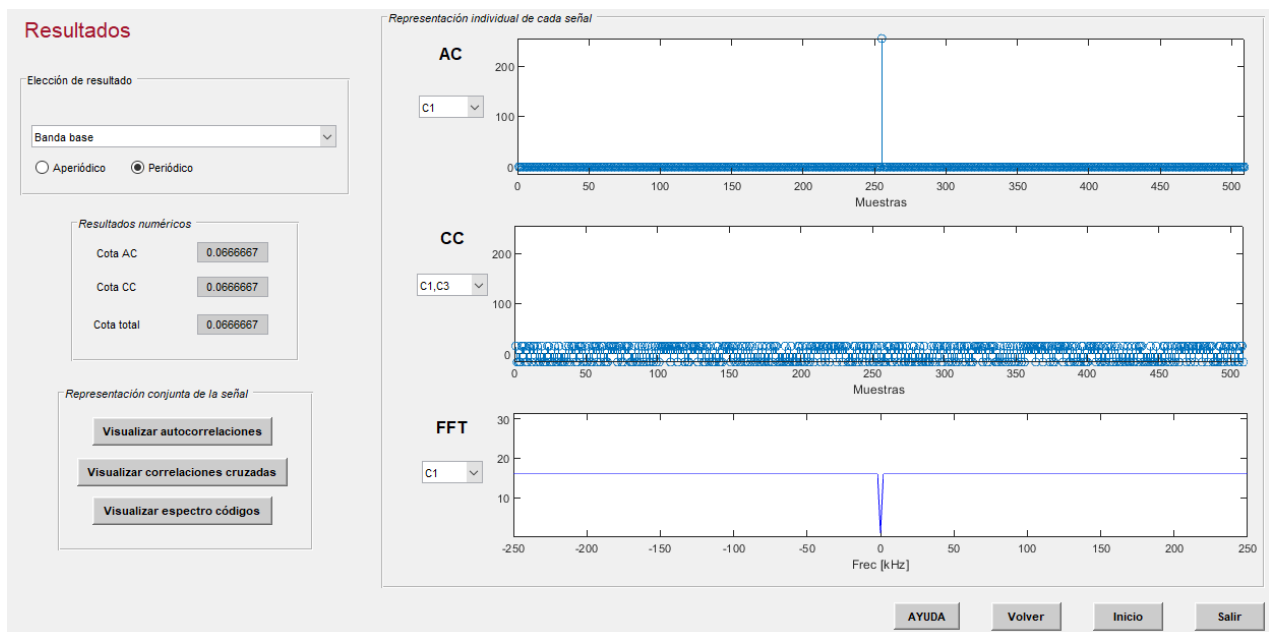


Fig. 95. Ventana 4 Kasami: banda base periódica

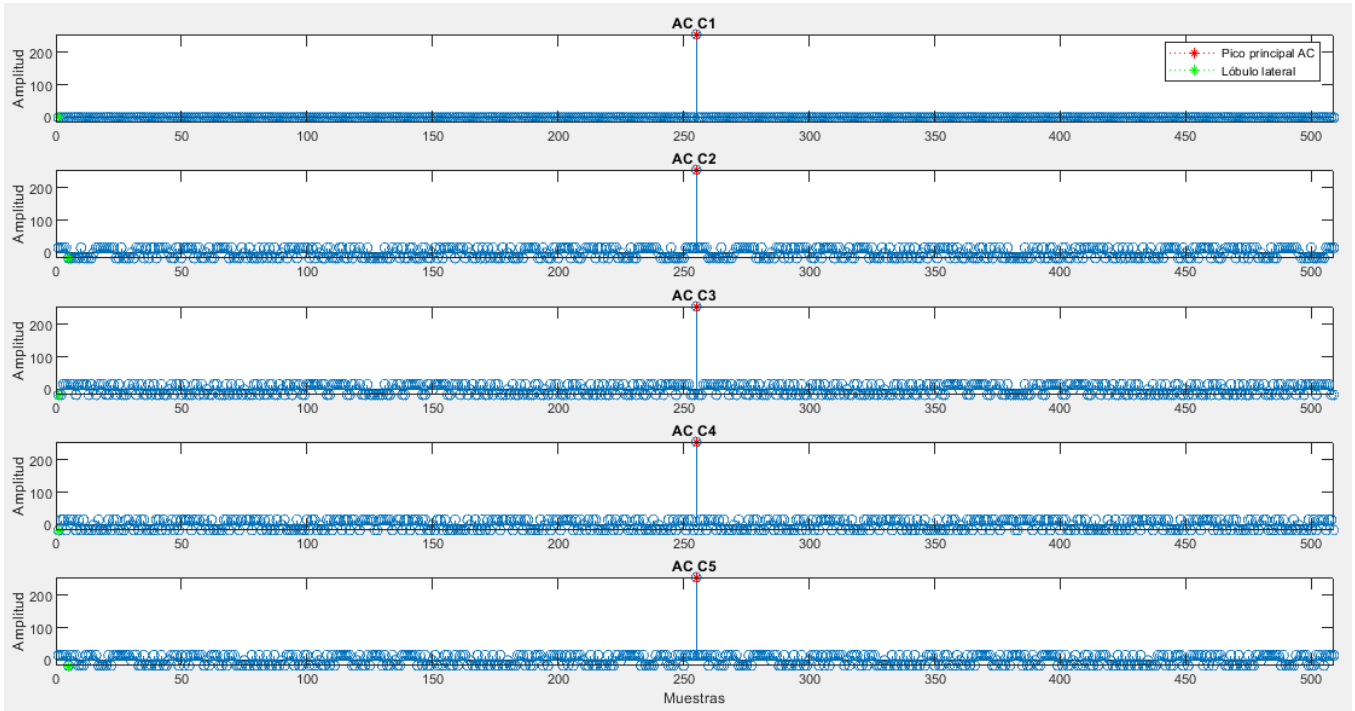


Fig. 96. Ventana 4 Kasami: AC banda base periódica



Fig. 97. Ventana 4 Kasami: CC banda base periódica

Una vez aplicada la **modulación**, en modo **aperiódico** se muestra en la Fig. 98. El resultado de las cotas es de $\theta_{AC}=0,1058$, $\theta_{CC}=0,1294$ y $\theta=0,1294$. Las autocorrelaciones AC, correlaciones cruzadas CC y las FFT se dan en las figuras 99, 100 y 101 respectivamente.

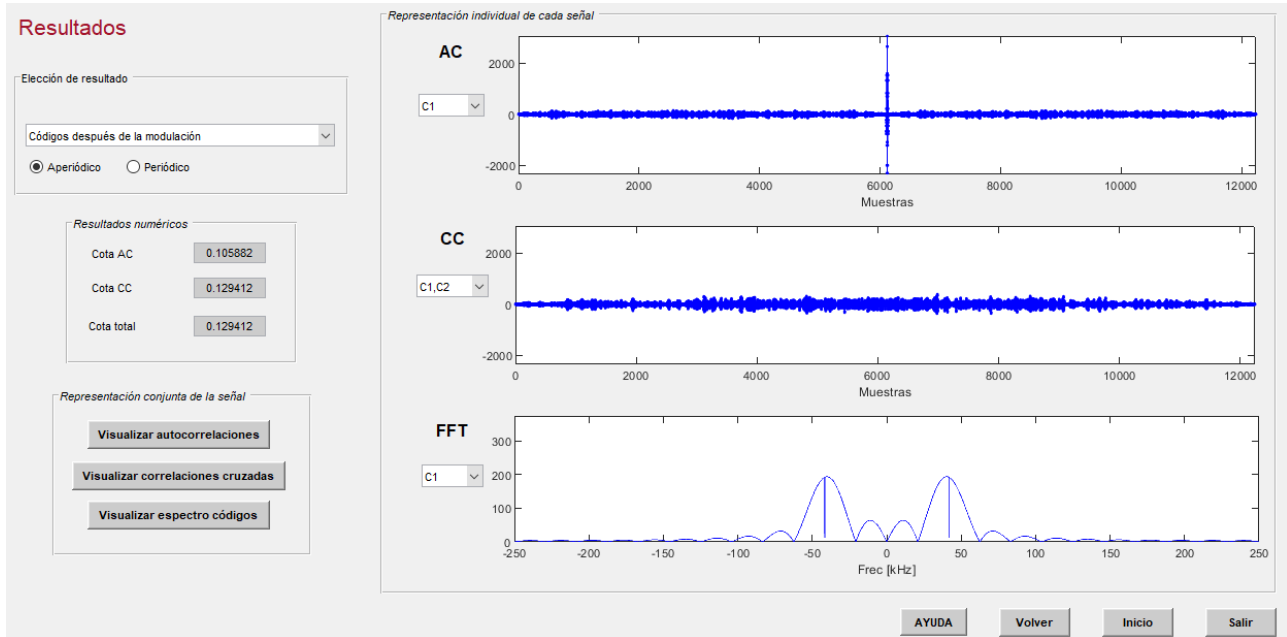


Fig. 98. Ventana 4 Kasami: modulación aperiódica

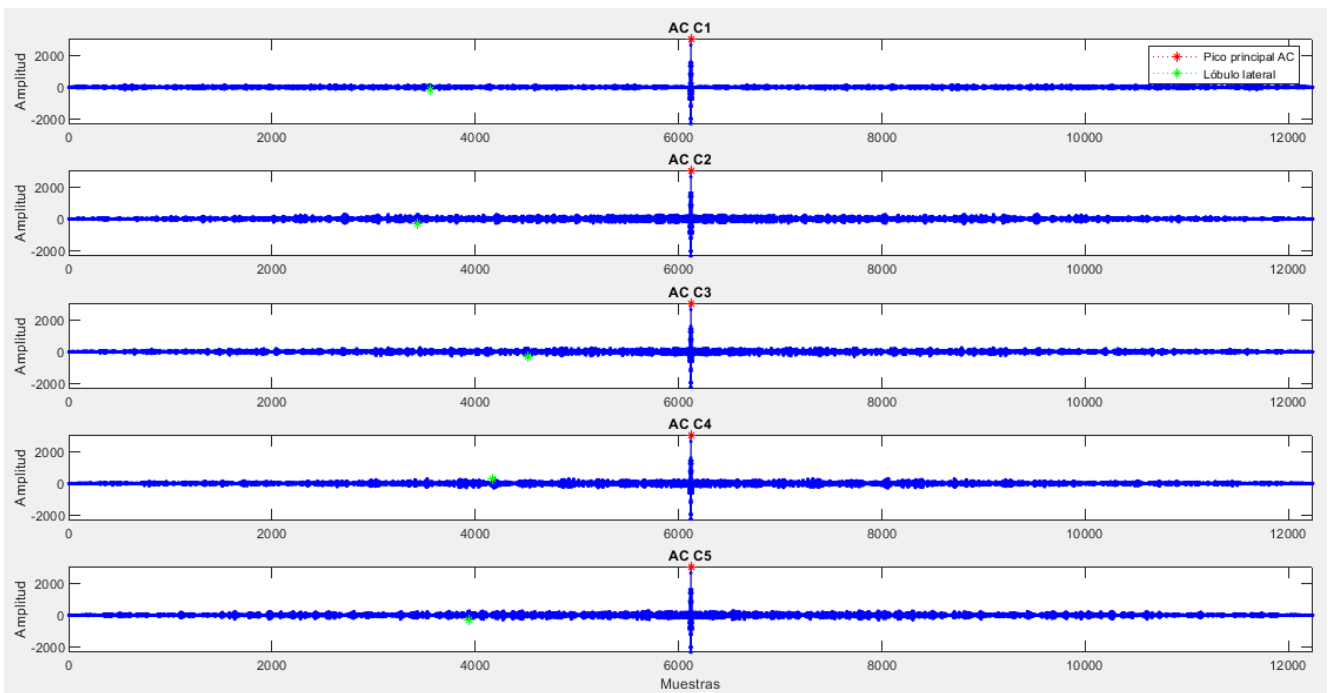


Fig. 99. Ventana 4 Kasami: AC modulada aperiódica

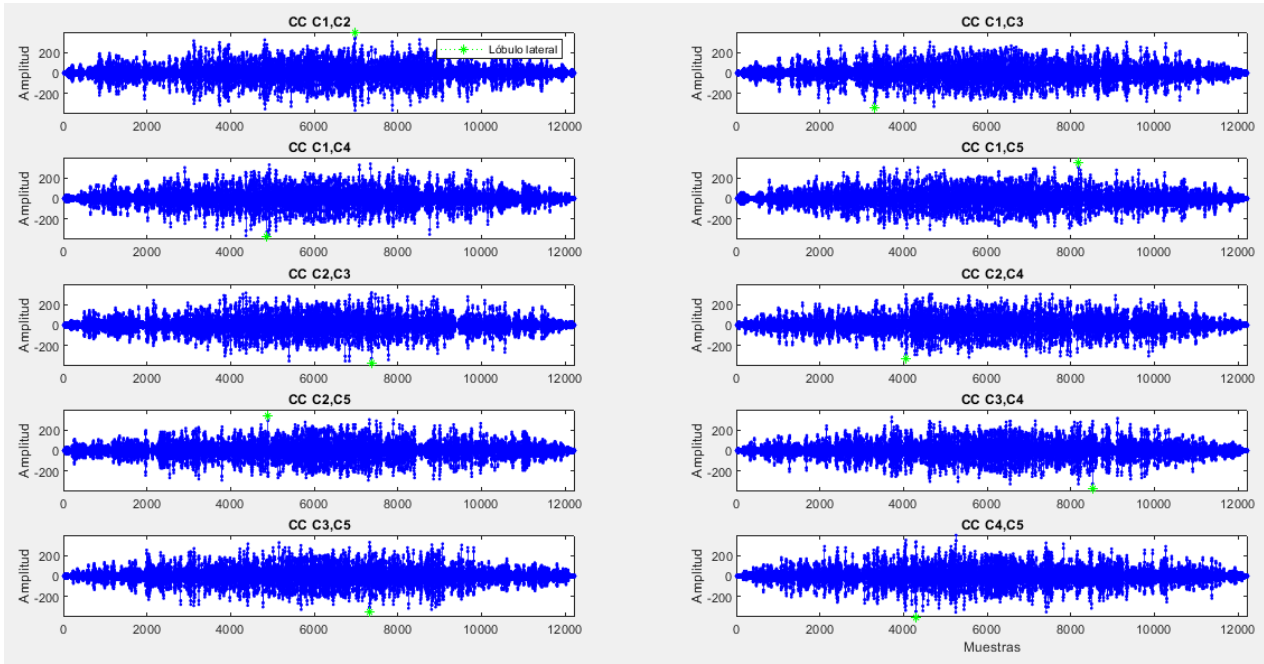


Fig. 100. Ventana 4 Kasami: CC modulada aperiódica

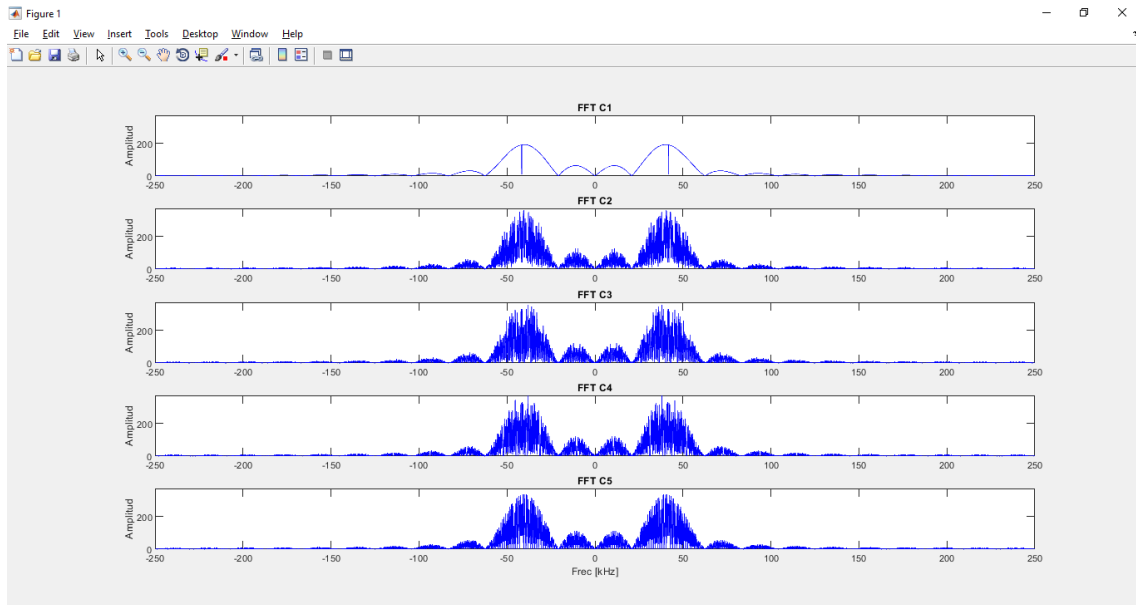


Fig. 101. Ventana 4 Kasami: FFT modulada

La figura 102 muestra la ventana **modulada** y en modo **periódico**. El resultado de las cotas es de $\theta_{AC}=0,0667$, $\theta_{CC}=0,0667$ y $\theta=0,0667$. Las autocorrelaciones AC, correlaciones cruzadas CC se muestran en las figuras 103 y 104 respectivamente.

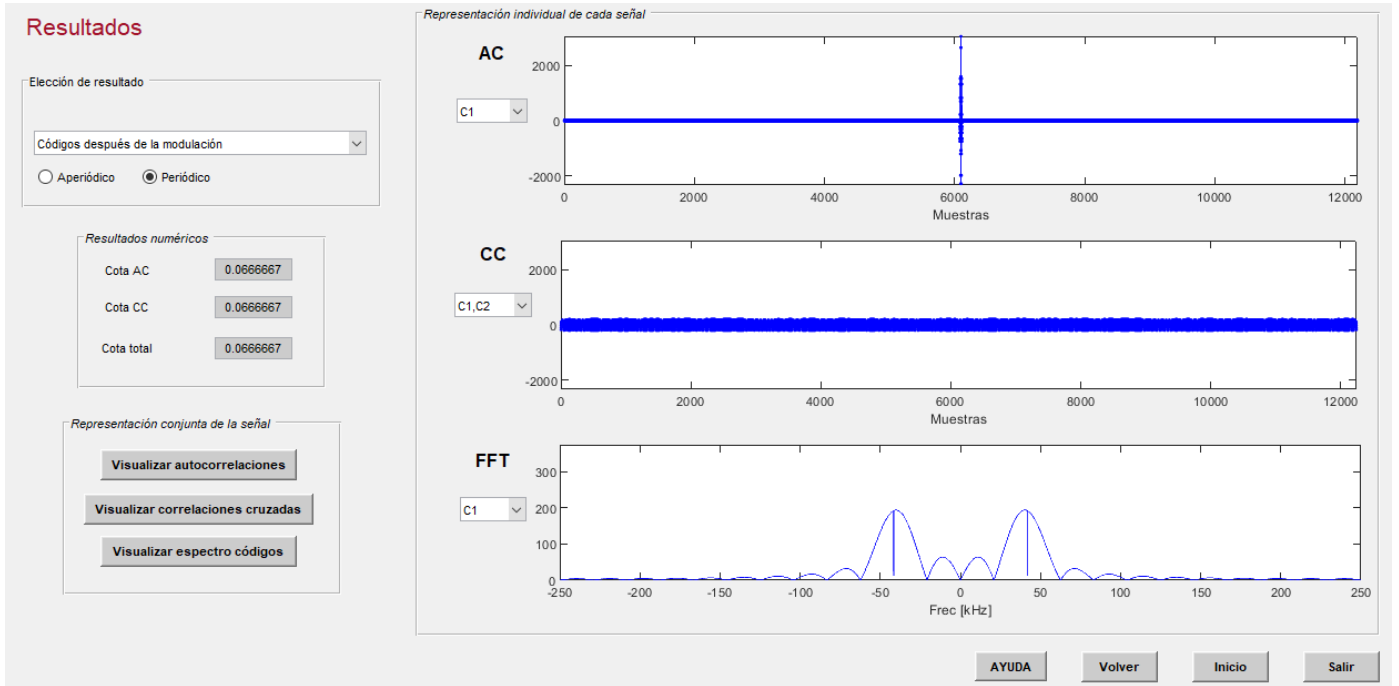


Fig. 102. Ventana 4 Kasami: modulada periódica

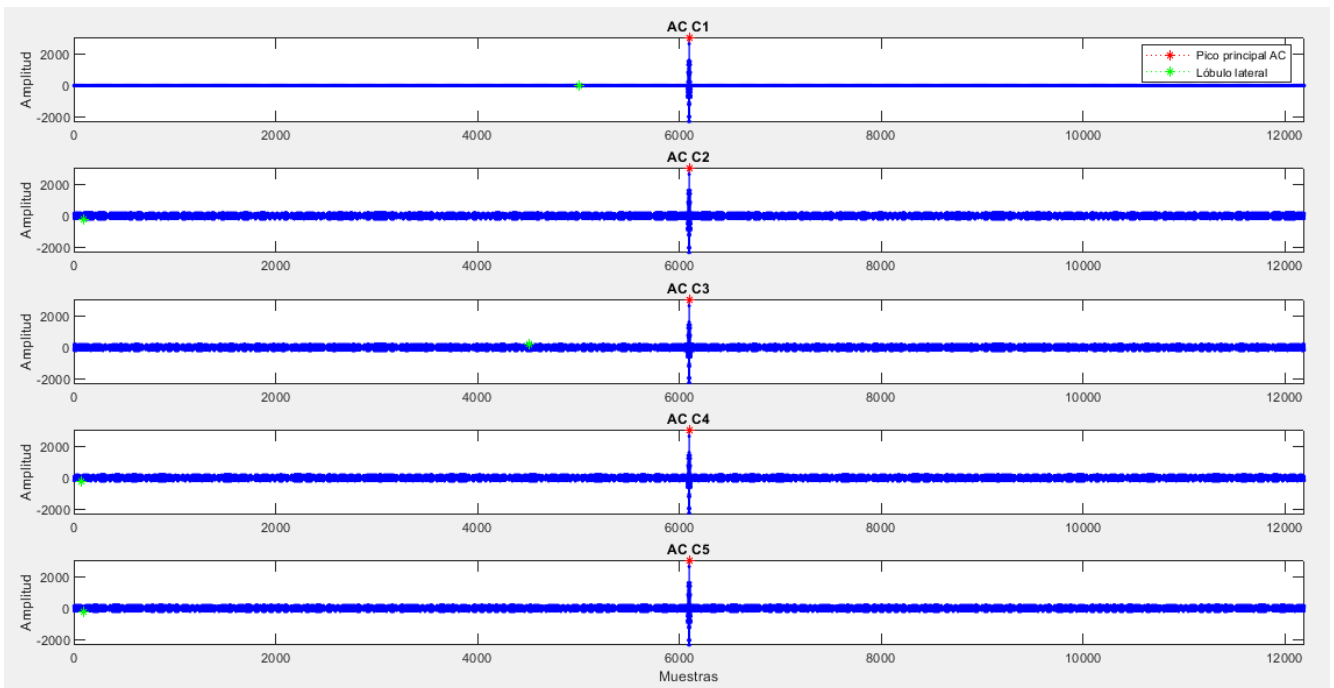


Fig. 103. Ventana 4 Kasami: AC modulada periódica

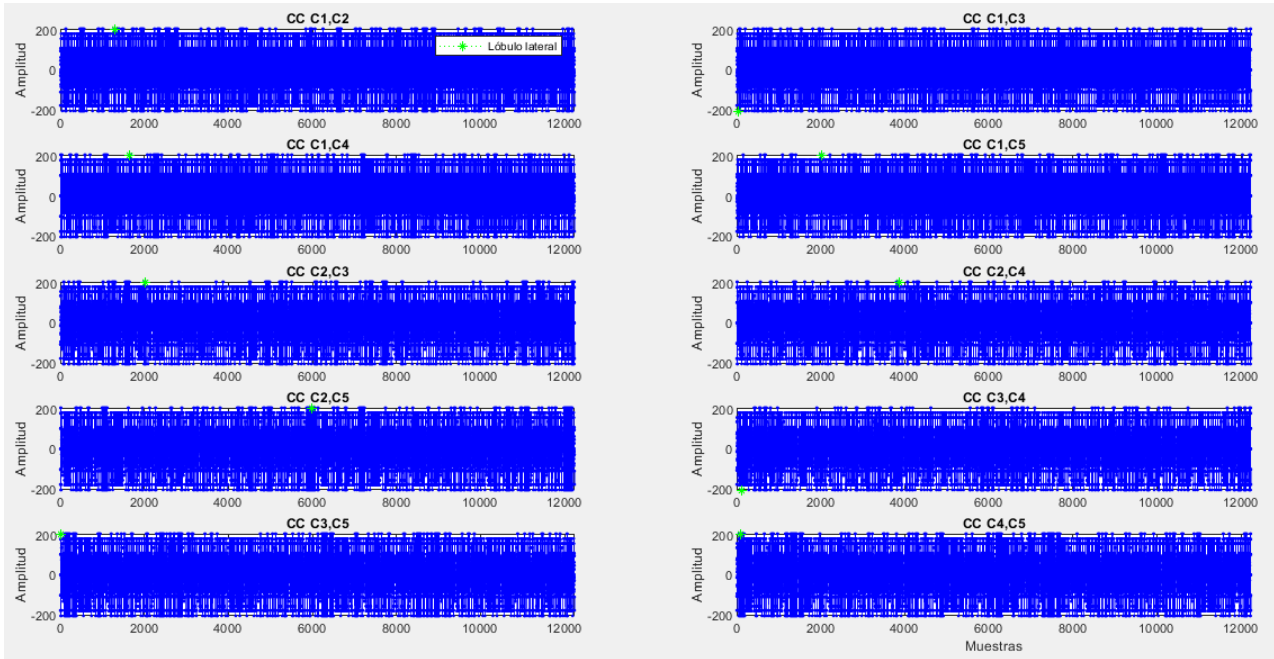


Fig. 104. Ventana 4 Kasami: CC modulada periódica

Si se selecciona la opción de **transductor**, aparece la pantalla de la Fig. 105 (modo **aperiódico**). El resultado de las cotas es de $\theta_{AC}=0,4255$, $\theta_{CC}=0,0989$ y $\theta=0,4255$. Las autocorrelaciones AC, correlaciones cruzadas CC y las FFT se dan en las figuras 106, 107 y 108 respectivamente.

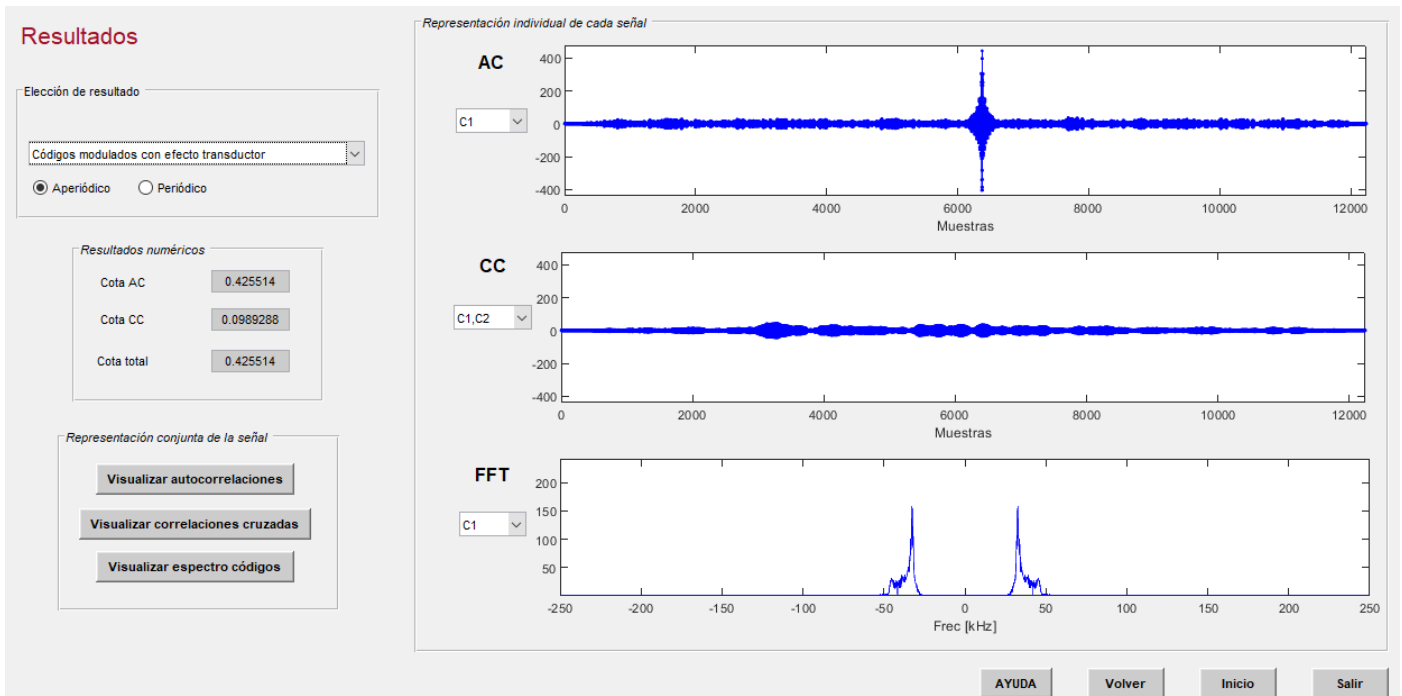


Fig. 105. Ventana 4 Kasami: transductor aperiódica

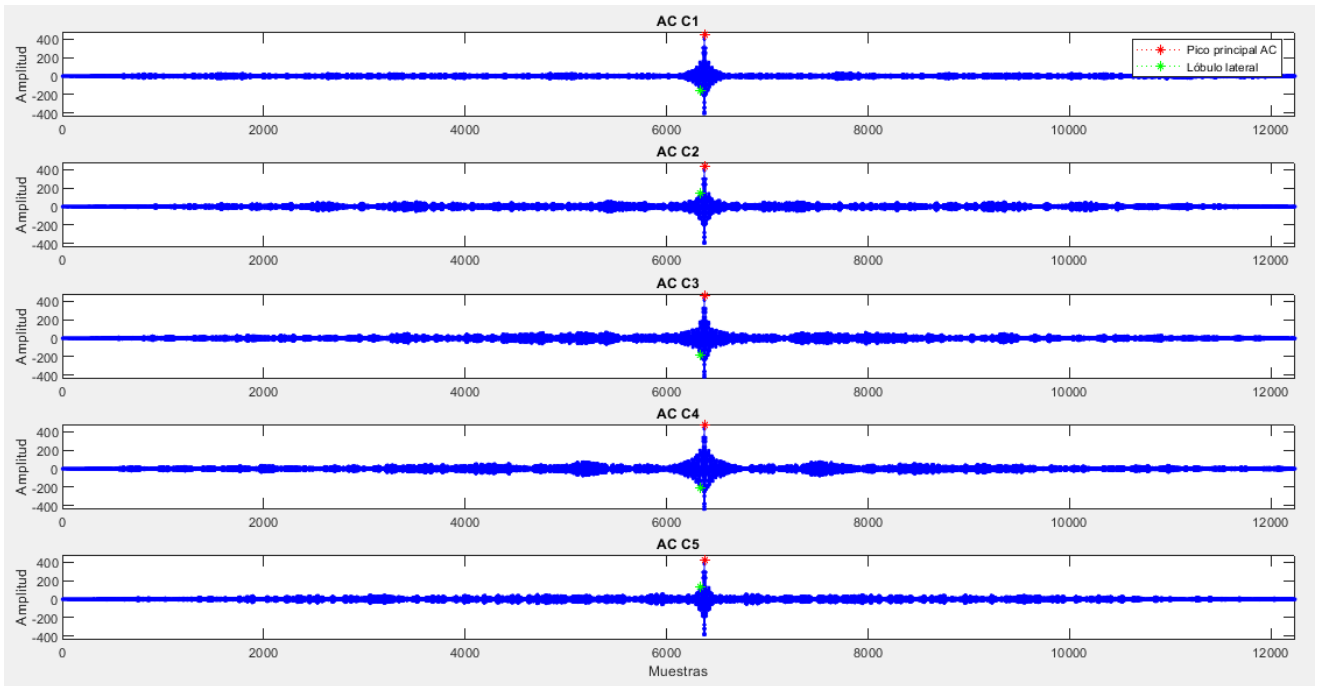


Fig. 106. Ventana 4 Kasami: AC transductor aperiódica

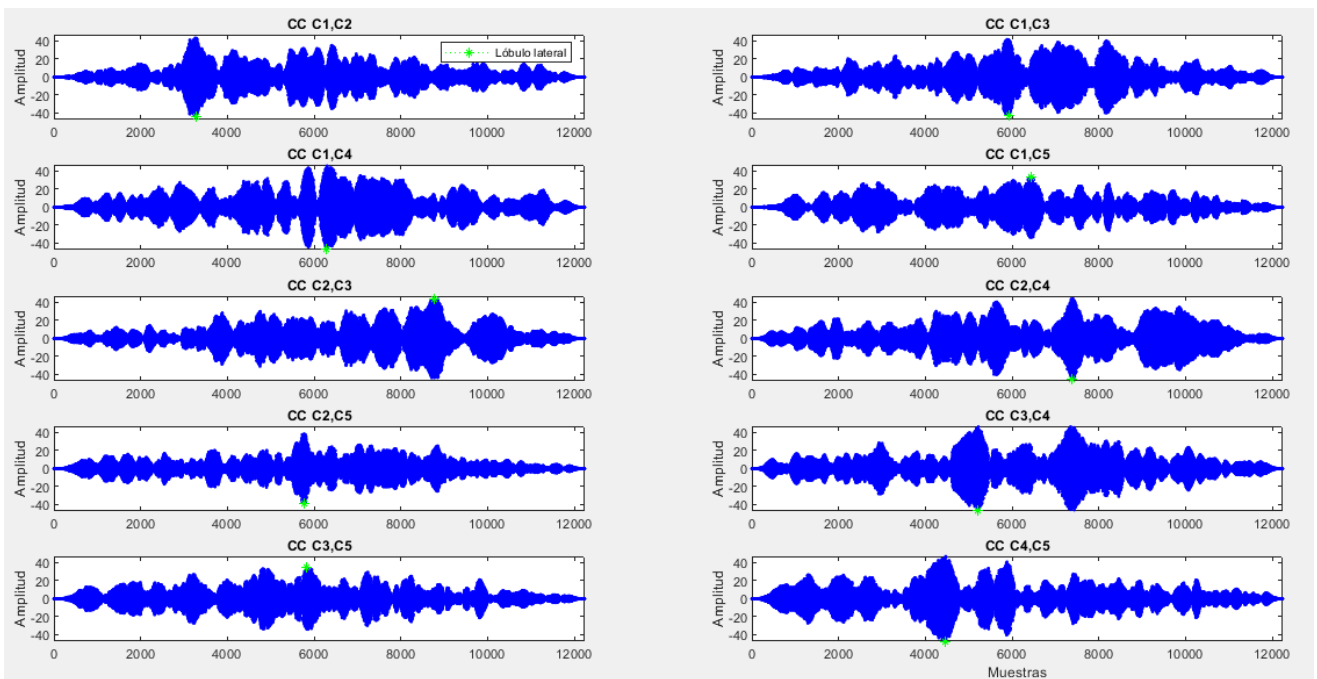


Fig. 107. Ventana 4 Kasami: CC transductor aperiódica

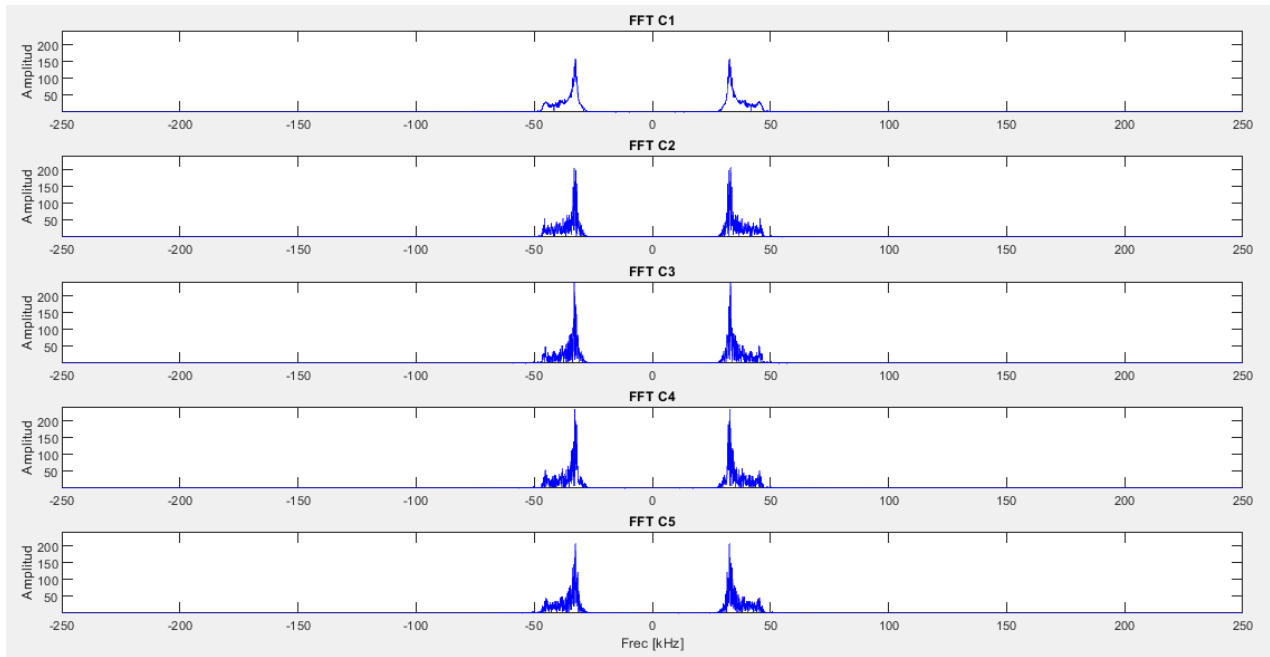


Fig. 108. Ventana 4 Kasami: FFT transductor

La figura 109 muestra la ventana con **transductor** y en modo **periódico**. El resultado de las cotas es de $\theta_{AC}=0,4255$, $\theta_{CC}=0,1169$ y $\theta=0,4255$. Las autocorrelaciones AC, correlaciones cruzadas CC se dan en las figuras 110 y 111 respectivamente. En este caso también se anula la búsqueda un espacio $N_c \cdot Of$ a los laterales sin embargo los efectos del transductor provocan más variación.

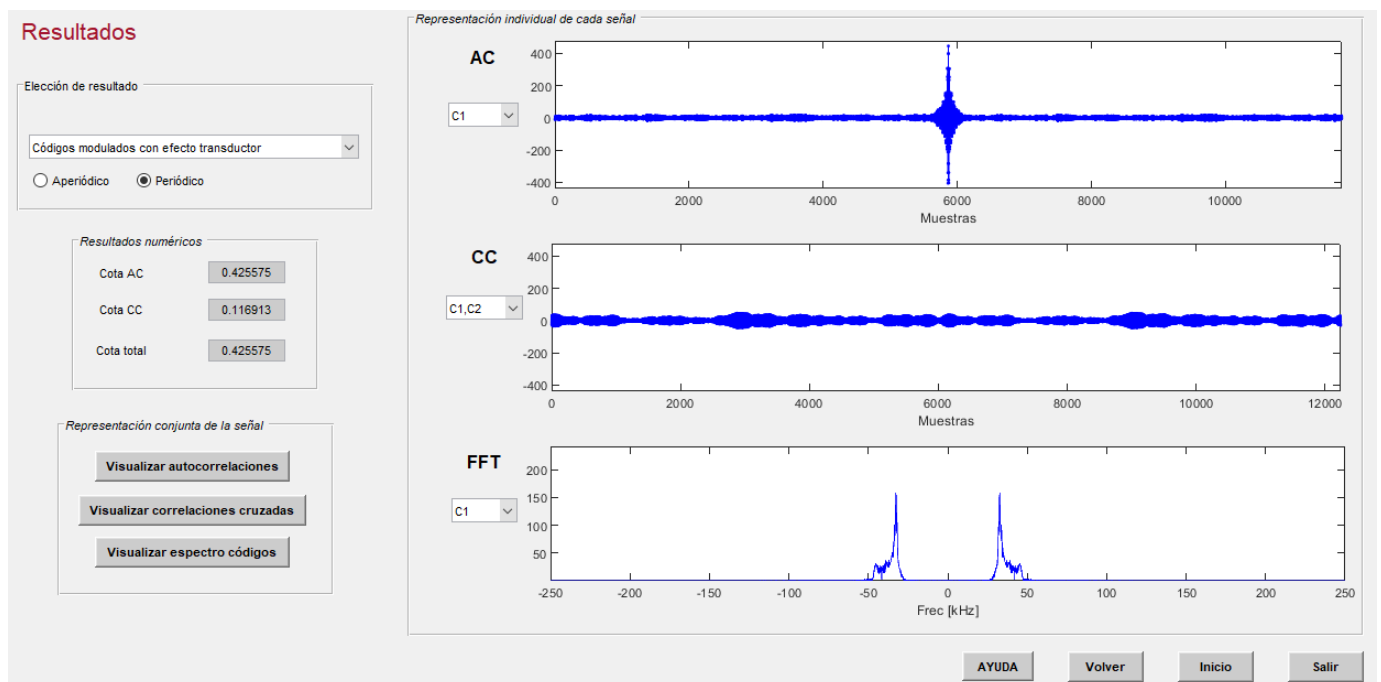


Fig. 109. Ventana 4 Kasami: transductor periódica

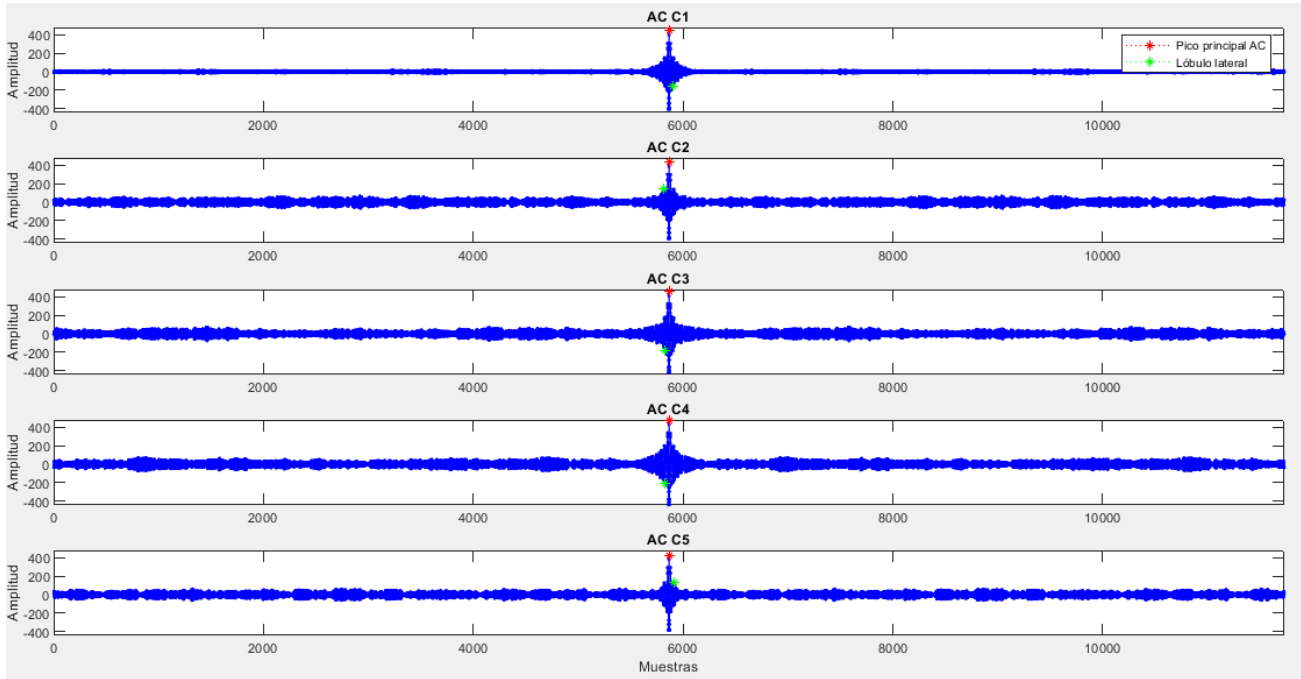


Fig. 110. Ventana 4 Kasami: AC transductor periódica

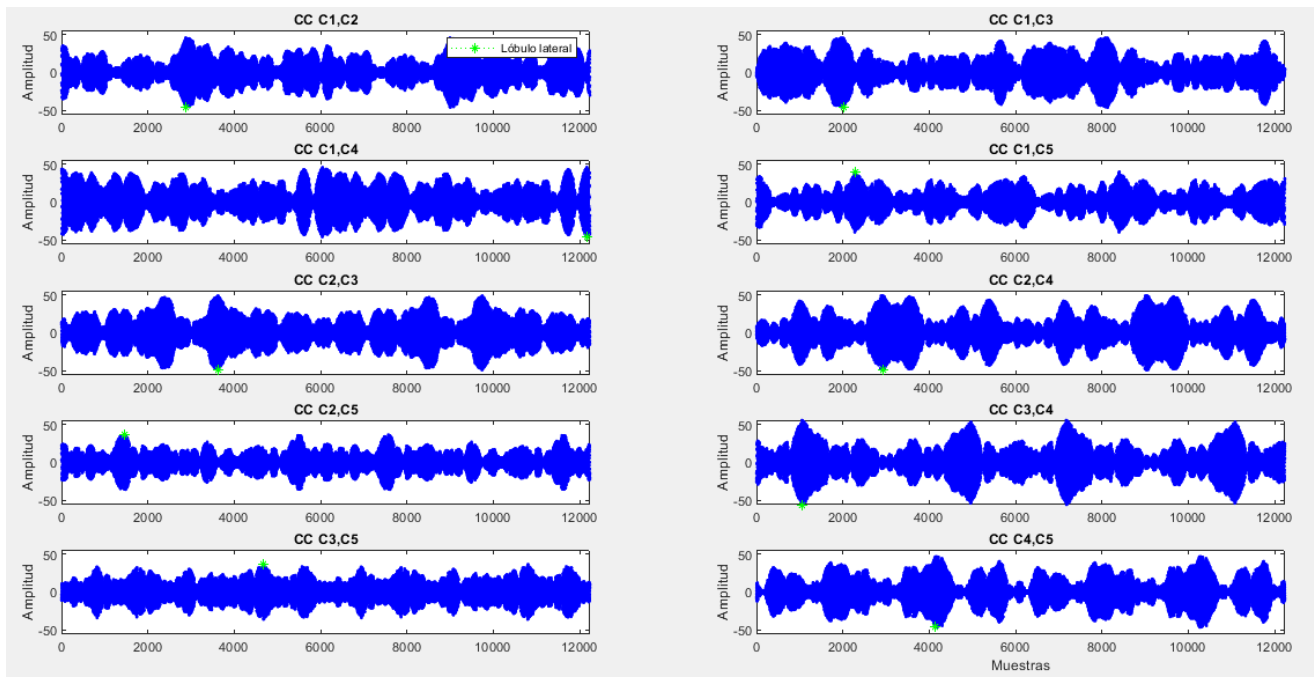


Fig. 111. Ventana 4 Kasami: CC transductor periódica

La ventana de la **señal recibida real (sin aplicar efectos añadidos)** se muestra en la Fig. 112. El SMR para cada código es $SMR_{C1}=0,3839\pm 0,0389$, $SMR_{C2}=0,3905\pm 0,0271$, $SMR_{C3}=0,4606\pm 0,0088$, $SMR_{C4}=0,4845\pm 0,0128$, $SMR_{C5}=0,3508\pm 0,0320$ siendo $SMR_{total}=0,4141\pm 0,0127$. Las figuras 113, 114 y 115 representan las gráficas de correlaciones cruzadas, correlaciones al cuadrado normalizadas y envolvente de correlación respectivamente. Además, en las dos primeras se visualizan las ventanas de

código que se han tenido en cuenta para el cálculo del SMR parcial, así como el pico principal (asterisco rojo) y el lóbulo lateral (asterisco verde).

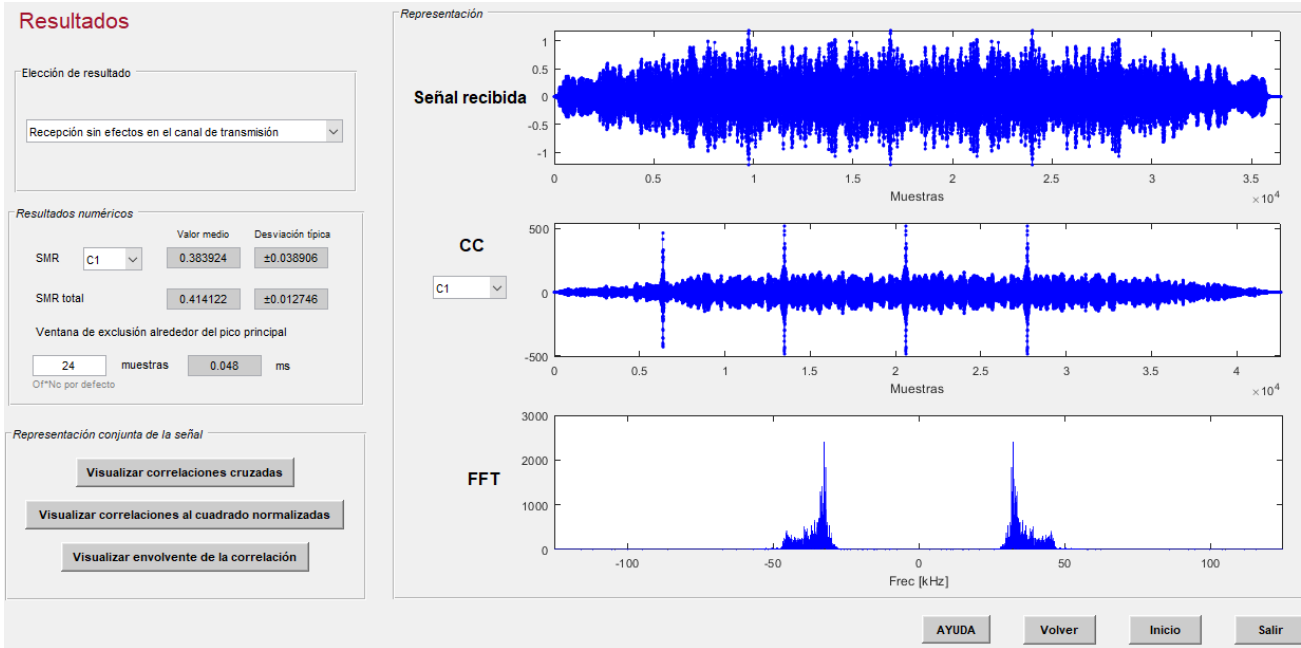


Fig. 112. Ventana 4 Kasami: señal recibida ideal

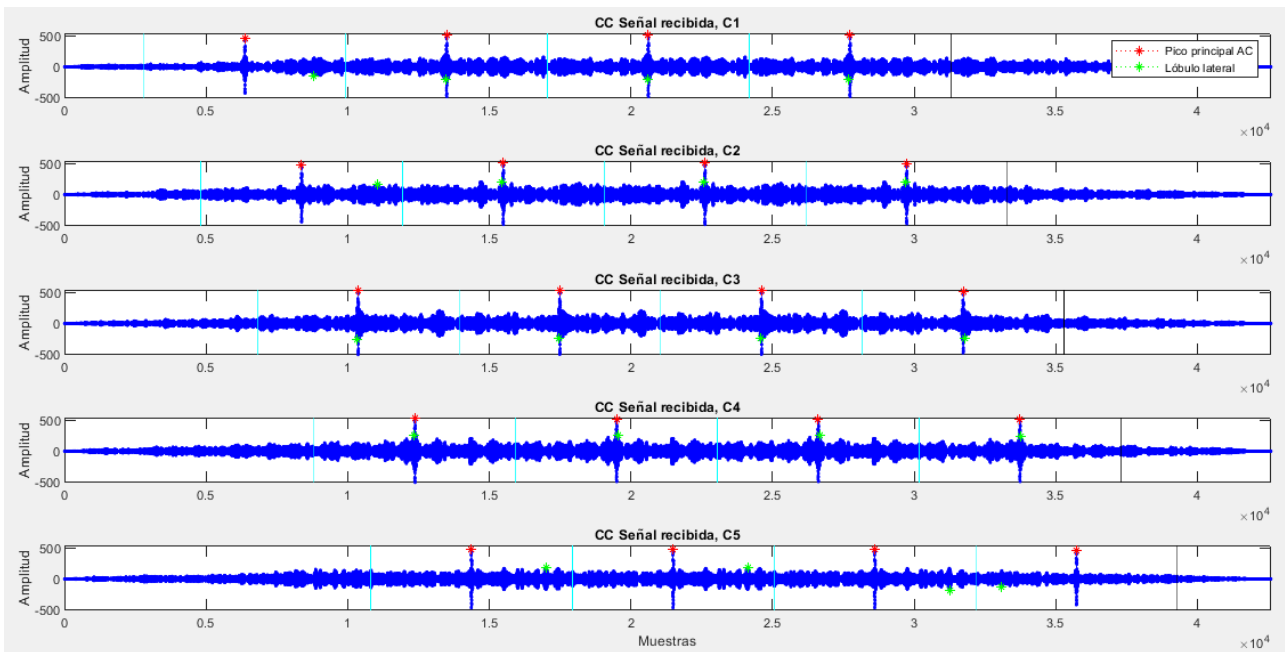


Fig. 113. Ventana 4 Kasami: CC Señal recibida ideal

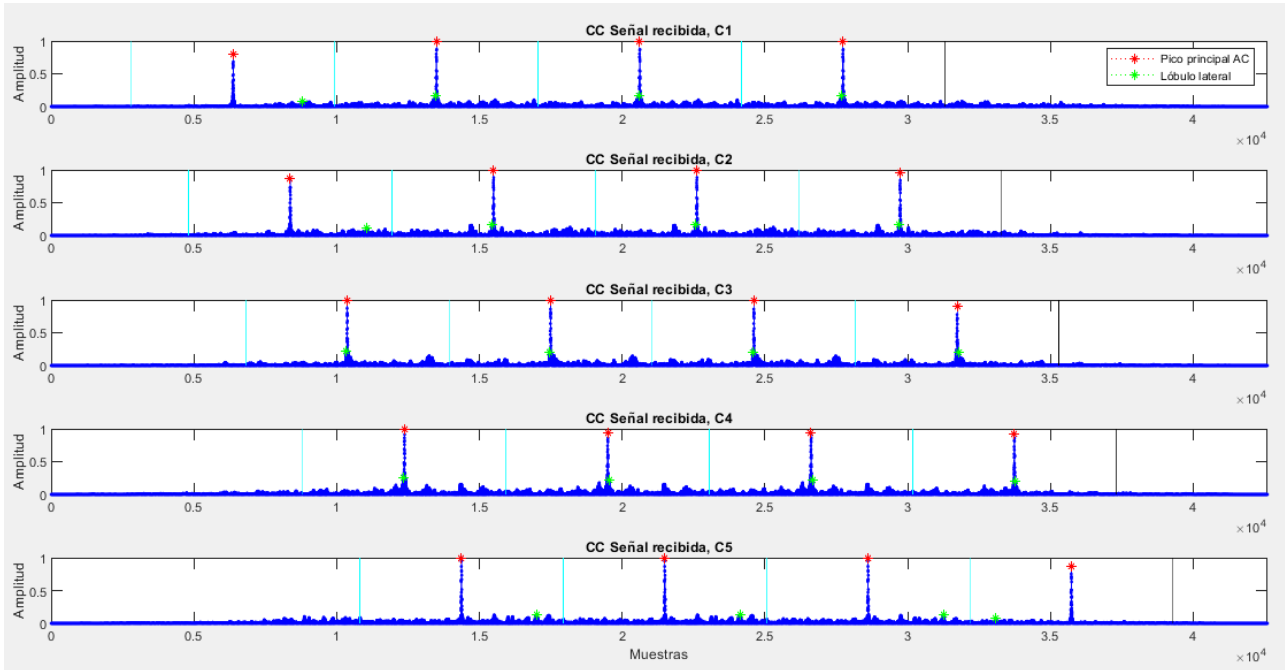


Fig. 114. Ventana 4 Kasami: correlaciones normalizadas al cuadrado señal recibida ideal

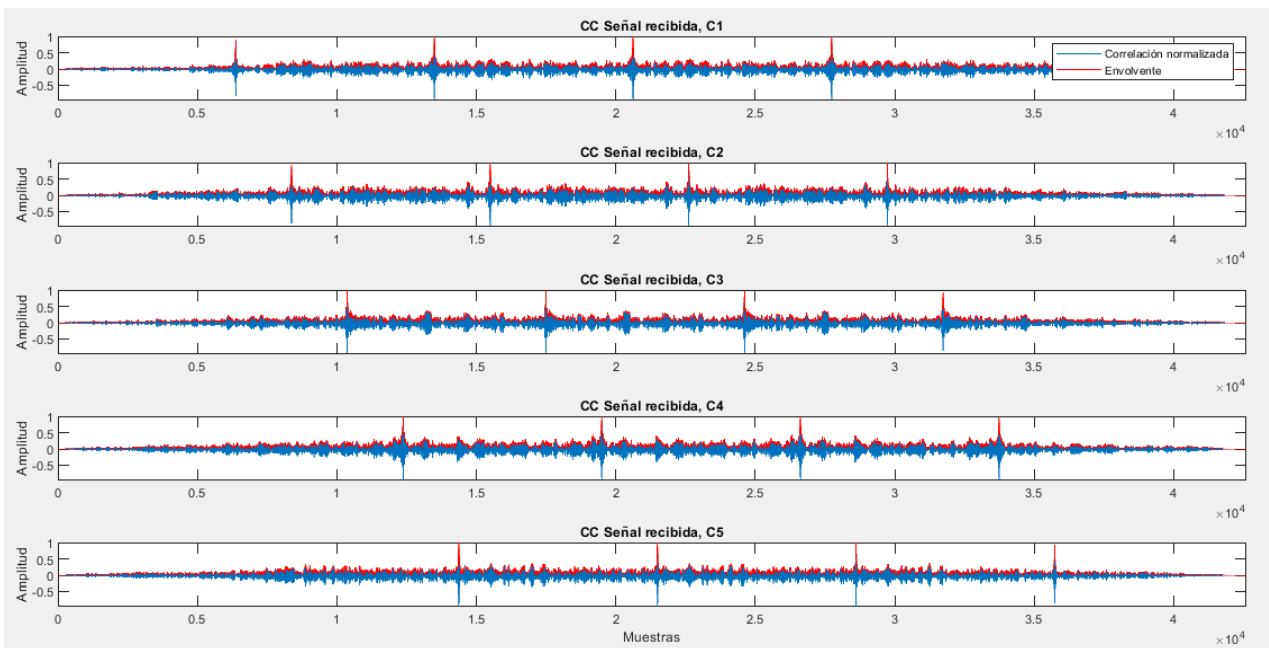


Fig. 115. Ventana 4 Kasami: envolvente correlación señal recibida ideal

En la Fig. 116, se ha modificado el valor de la ventana de exclusión alrededor del pico principal a 140 muestras. Esto provoca que se obtengan nuevos valores de SMR: $SMR_{C1}=0,3128\pm 0,0209$, $SMR_{C2}=0,3881\pm 0,0255$, $SMR_{C3}=0,3775\pm 0,0153$, $SMR_{C4}=0,4118\pm 0,0308$, $SMR_{C5}=0,3508\pm 0,0320$ siendo $SMR_{total}=0,3682\pm 0,0069$. En la Fig. 117 se muestra la gráfica de las correlaciones normalizadas al cuadrado para este valor de exclusión, el cual da mejores resultados ya que evalúa zonas de baja correlación.

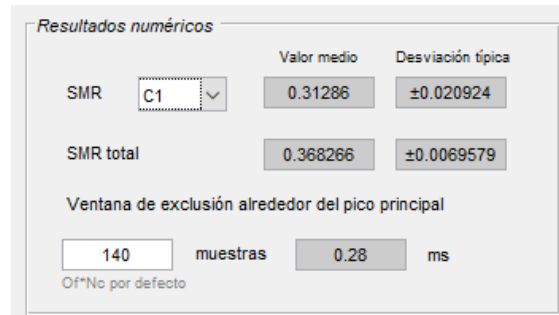


Fig. 116. Ventana 4 Kasami: SMR para diferente ventana de exclusión

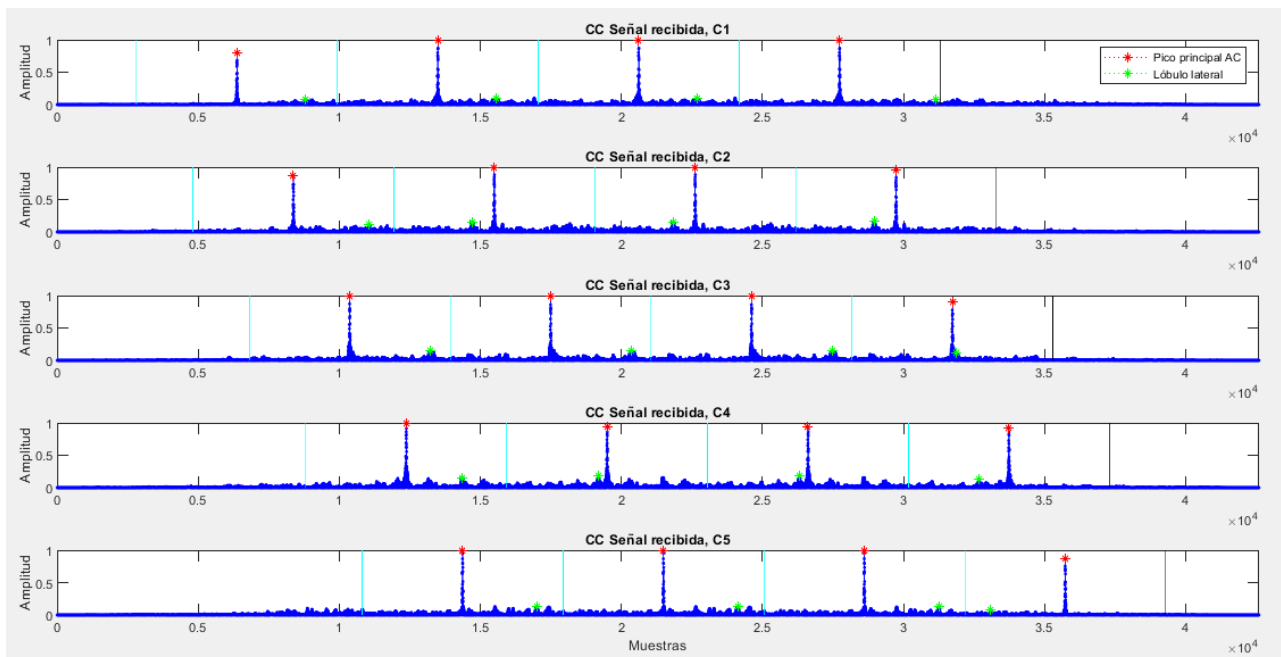


Fig. 117. Ventana 4 Kasami: CC normalizadas al cuadrado Señal recibida ideal con distinta exclusión

Al añadirse los efectos configurados a la señal recibida, aparece la ventana de la Fig. 118. El SMR para cada código es $SMR_{C1}=0,9743\pm 0,0276$, $SMR_{C2}=0,4299\pm 0,0162$, $SMR_{C3}=0,4290\pm 0,0311$, $SMR_{C4}=0,9261\pm 0,353$ $SMR_{C5}=0,9551\pm 0,0287$ siendo $SMR_{total}=0,7429\pm 0,0071$. Las figuras 119, 120 y 121 representan las gráficas de correlaciones cruzadas, correlaciones al cuadrado normalizadas y envolvente de correlación respectivamente. Además, en las dos primeras se visualizan las ventanas de código que se han tenido en cuenta para el cálculo del SMR parcial, así como el pico principal (asterisco rojo) y el lóbulo lateral (asterisco verde).

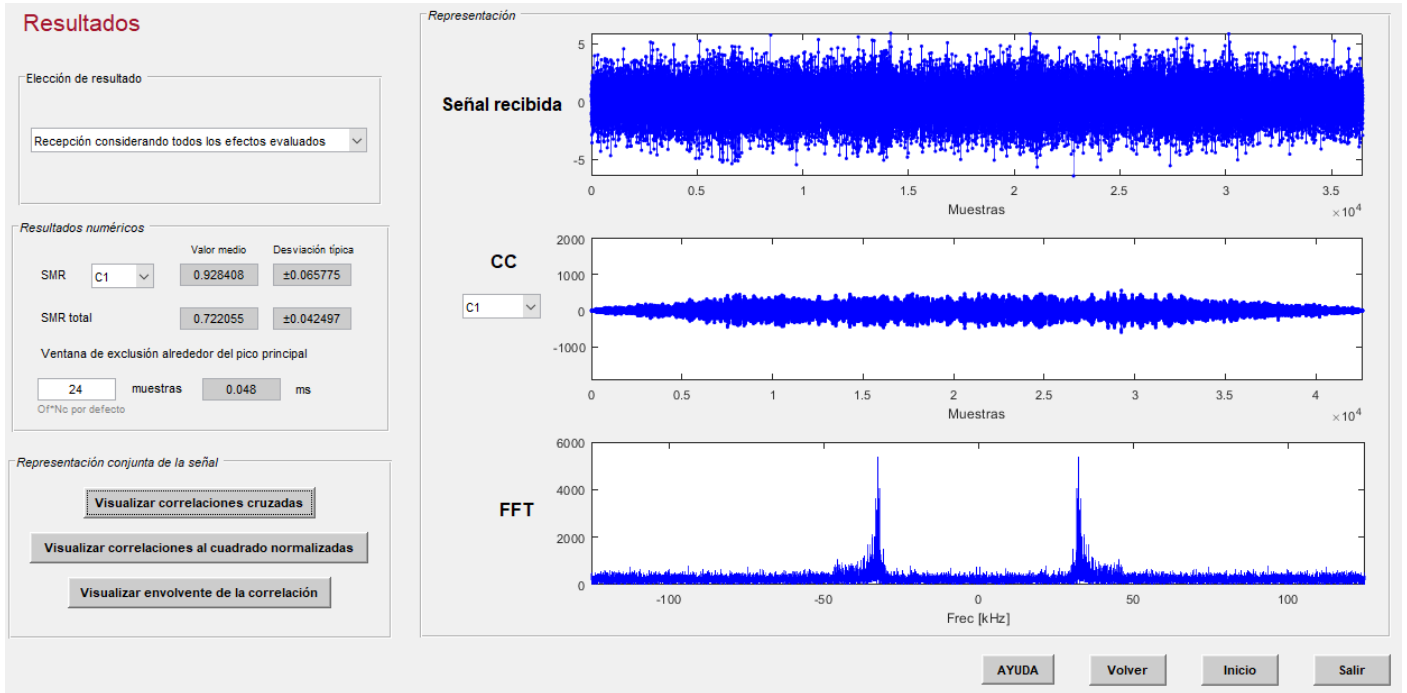


Fig. 118. Ventana 4 Kasami: Señal recibida con efectos

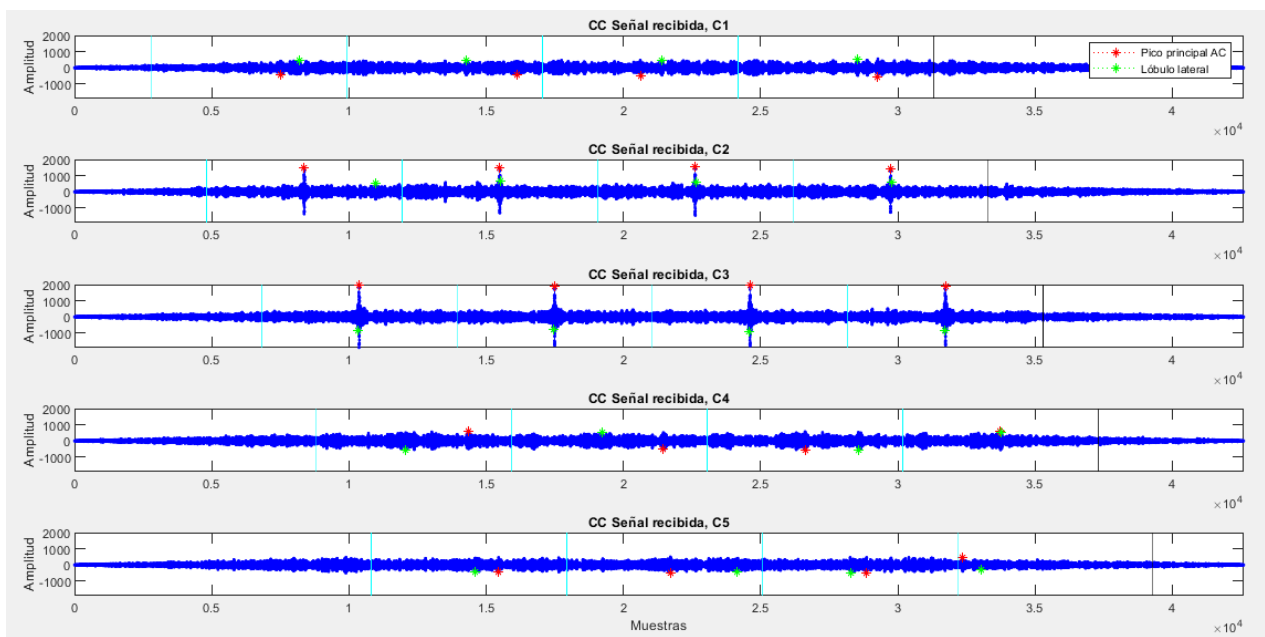


Fig. 119. Ventana 4 Kasami: CC Señal recibida con efectos

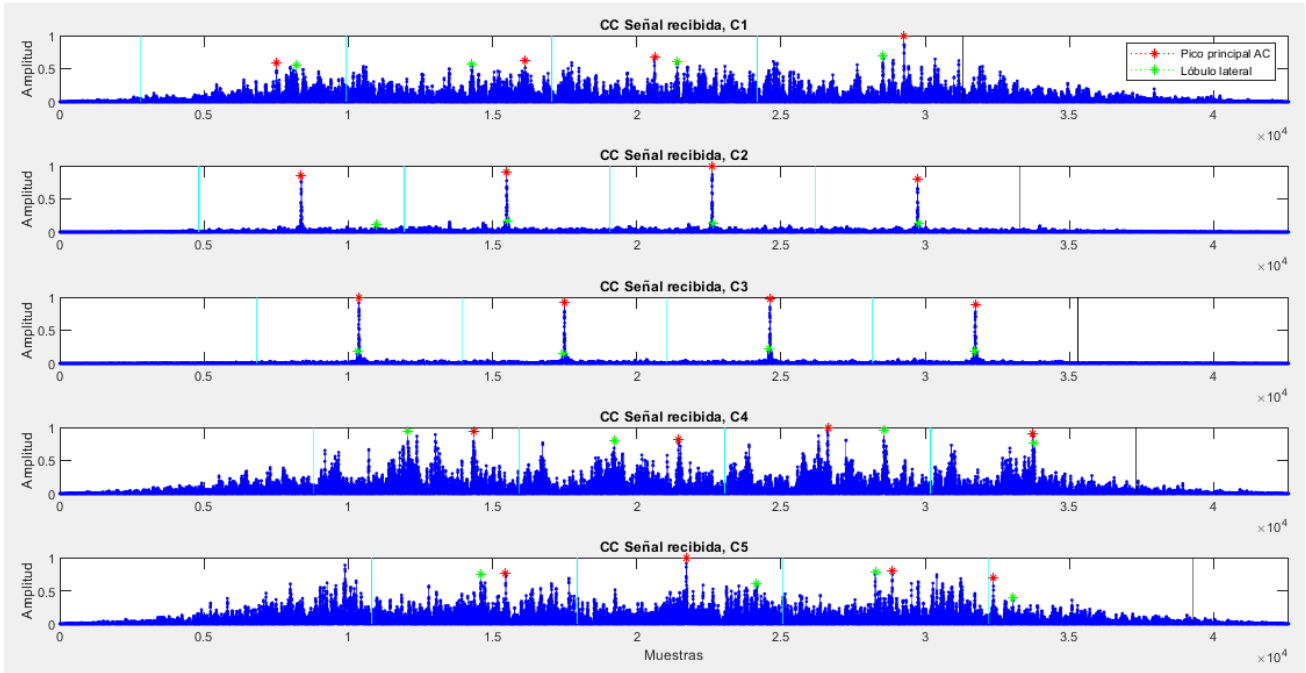


Fig. 120. Ventana 4 Kasami: CC normalizadas al cuadrado señal recibida con efectos

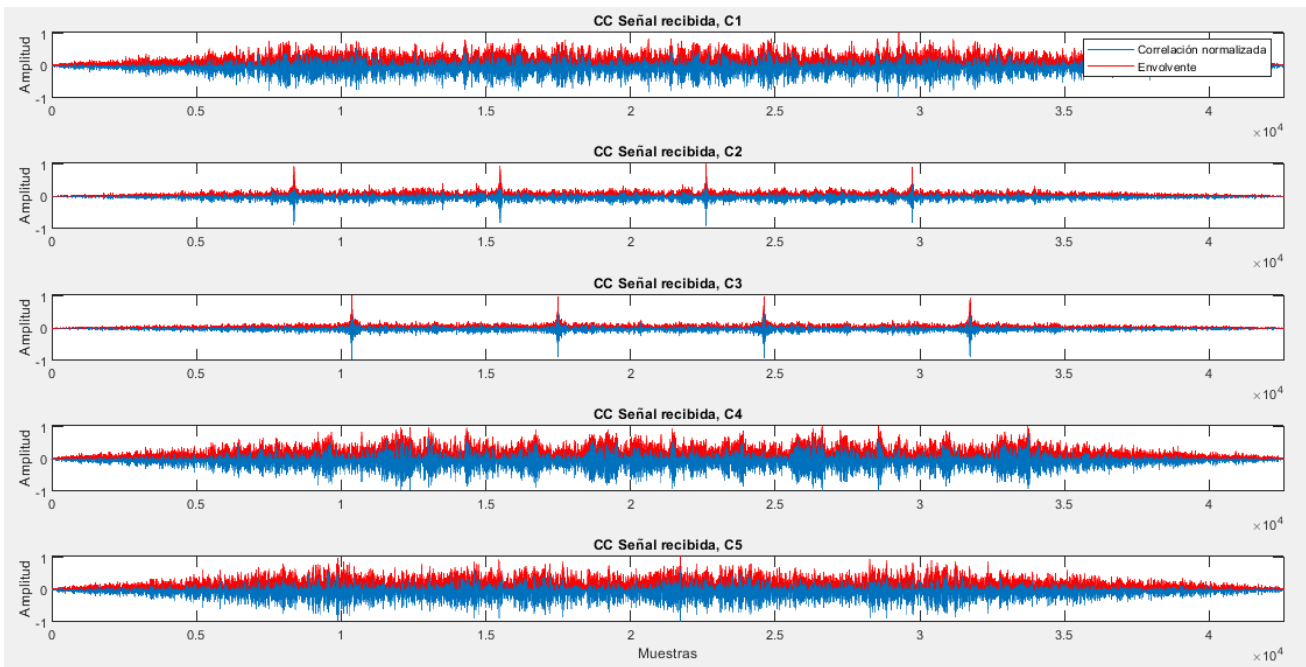


Fig. 121. Ventana 4 Kasami: envoltura de correlación señal recibida con efectos

Si ahora se retrocede a la ventana de configuración de efectos y se cambia por la **simulación paramétrica** para los valores $SNR_{desde}=-20$, $SNR_{hasta}=15$ y $Salto=5$, se obtiene el RMSE para cada código representado en la figura 122.

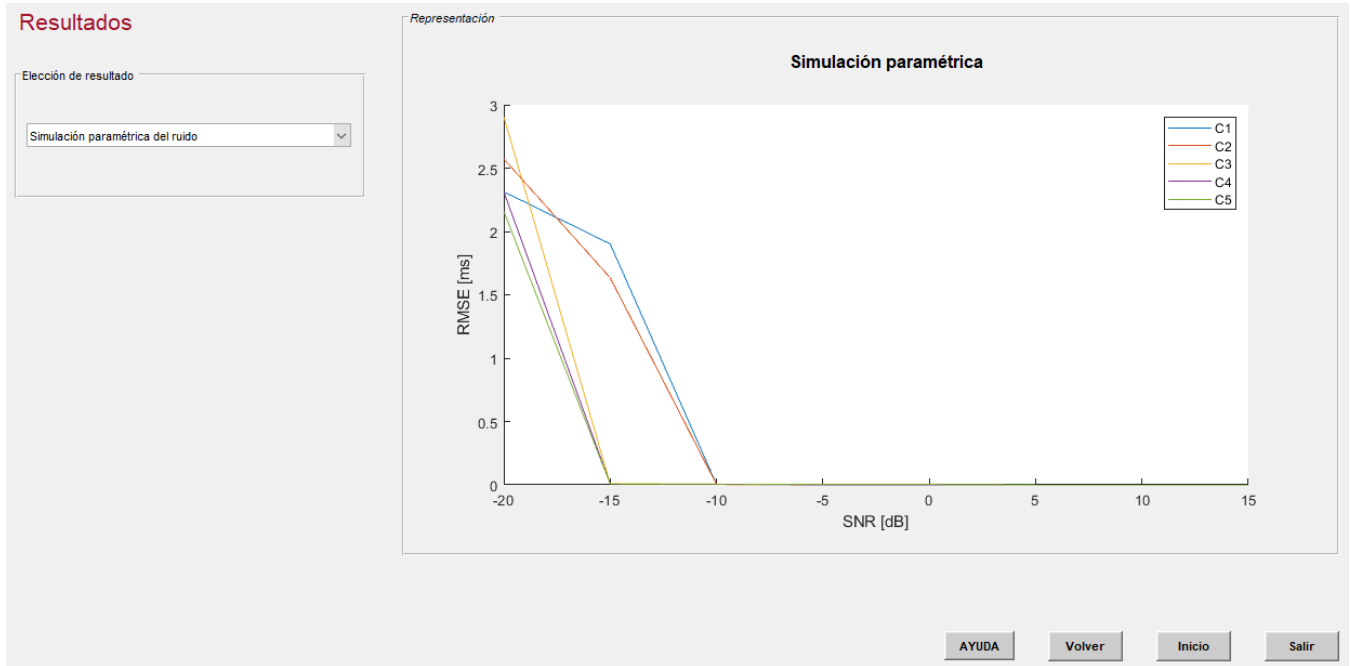


Fig. 122. Ventana 4 Kasami: Simulación paramétrica

4.5.3. Particularidades: CSS

Estos códigos tienen una particularidad frente a los demás y es que se puede seleccionar una opción más llamada **banda base ideal** (Fig. 123). Esta opción permite evaluar los resultados de las secuencias de cada código antes de ser reordenadas y agrupadas para su emisión. Se distinguen varias secciones:

- Resultados numéricos: Cota AC θ_{AC} , Cota CC θ_{CC} y cota total θ . Estas cotas equivalen a la relación entre el lóbulo principal y el lóbulo lateral en las correlaciones. En este caso, aunque sea visualicen también las correlaciones de las secuencias de cada código, el cálculo de cotas se realiza solamente en la SACF y en la SCCF. Para las SACF, el programa localiza el máximo valor de la señal, el cual debe estar a L muestras, y lo anula. Seguidamente, se vuelve a buscar el máximo de la señal restante y se asigna como lóbulo lateral. Para las SCCF, se utiliza el pico principal obtenido en las SACF y se obtiene el lóbulo lateral de SCCF como el máximo valor de la señal. Se calculan para cada código y se muestra por pantalla el máximo de todos ellos.
- Representación individual de la SACF y SCCF de cada código, y de la FFT de cada secuencia de cada código. Mediante las pestañas que aparecen junto con las gráficas, se selecciona aquella SACF de código C_i , SCCF de códigos C_i, C_j o FFT de secuencia S_k del código C_i , siendo $1 \leq i \leq Nemo-1$; $2 \leq j \leq Nemo$; $1 \leq k \leq M$.
- Representación conjunta de las AC de las secuencias junto con su SACF, de las CC de las secuencias junto con su SCCF, o de la FFT de las secuencias. Mediante las pestañas que aparecen junto a los botones de visualización, se selecciona el código de M secuencias a visualizar. En esta representación aparece en la SACF, marcado con un asterisco rojo el pico principal de correlación y para la SACF y SCCF, con un asterisco verde el lóbulo lateral obtenido para el cálculo de θ_{AC} y de θ_{CC} .

Las Fig. 124, 125 y 126 corresponden a las representaciones de la última sección.

Las propiedades de los códigos representados son: $M=4$, $L=64$, $Nemo=2$, ordenación mediante entrelazado, portadora sinusoidal, $f_c=41,667\text{kHz}$, $f_{se}=500\text{kHz}$, $N_c=2$, $T_{gap}=10\text{ms}$, $T_{shift}=1\text{ms}$, $N_{rep}=4$, con efecto del transductor y sin efectos añadidos.

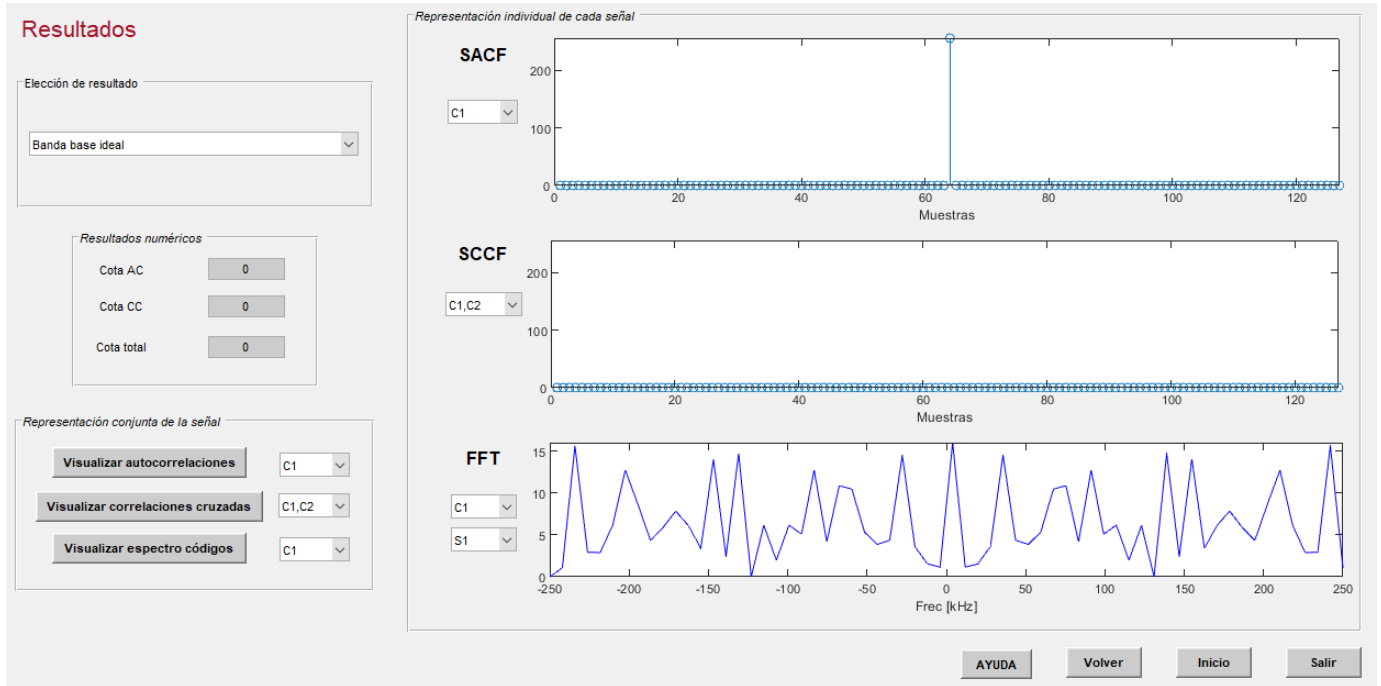


Fig. 123. Ventana 4 CSS: Banda base ideal

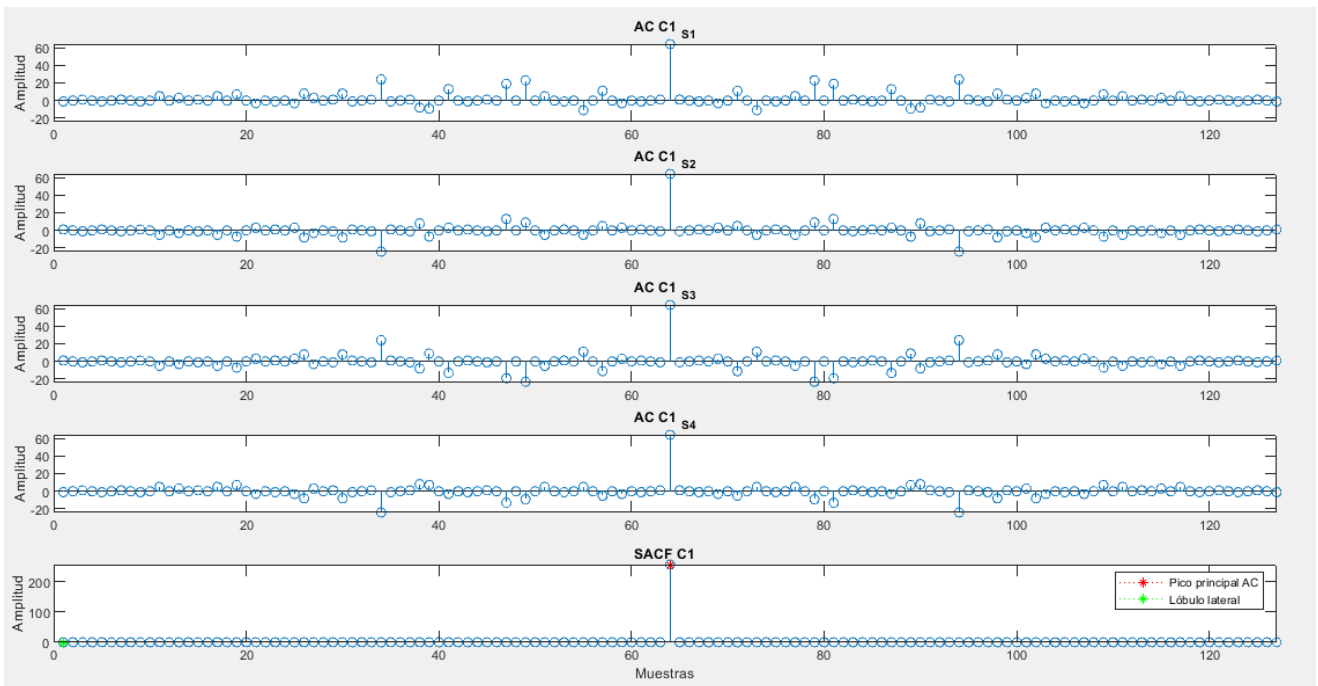


Fig. 124. Ventana 4 CSS: AC de las M secuencias y SACF del código 1 banda base ideal

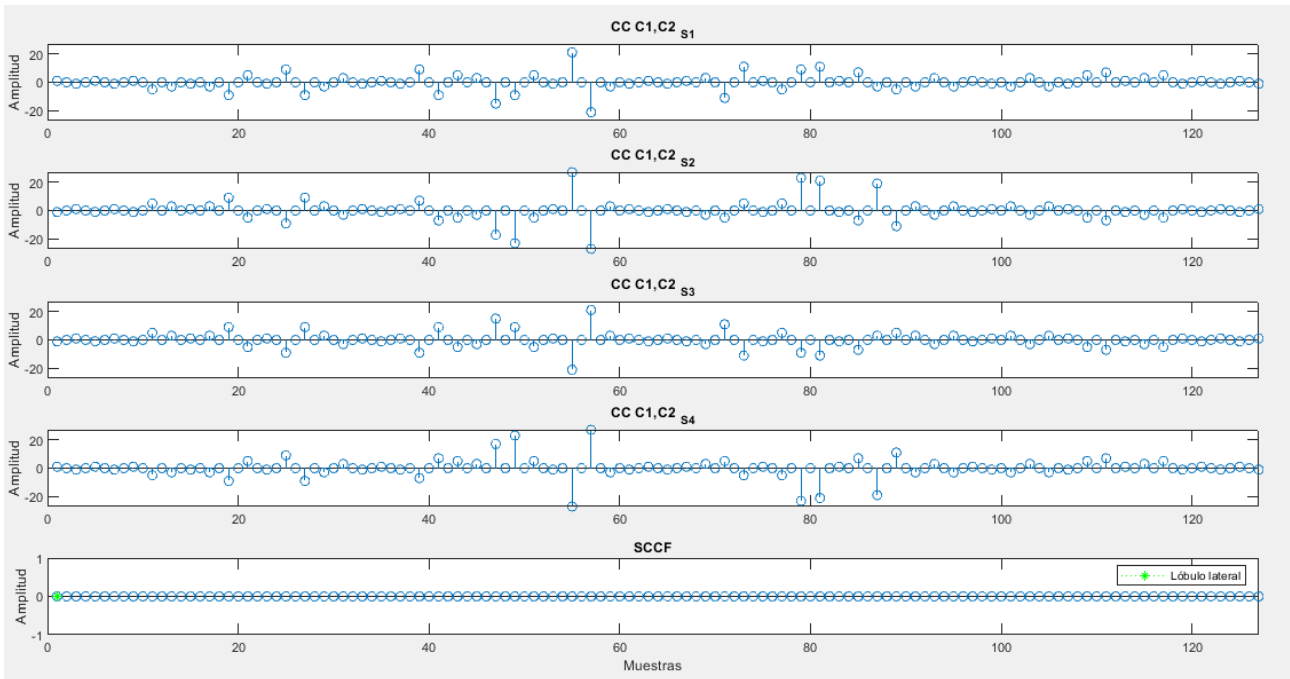


Fig. 125. Ventana 4 CSS: CC de las M secuencias y SCCF del código 1 banda base ideal

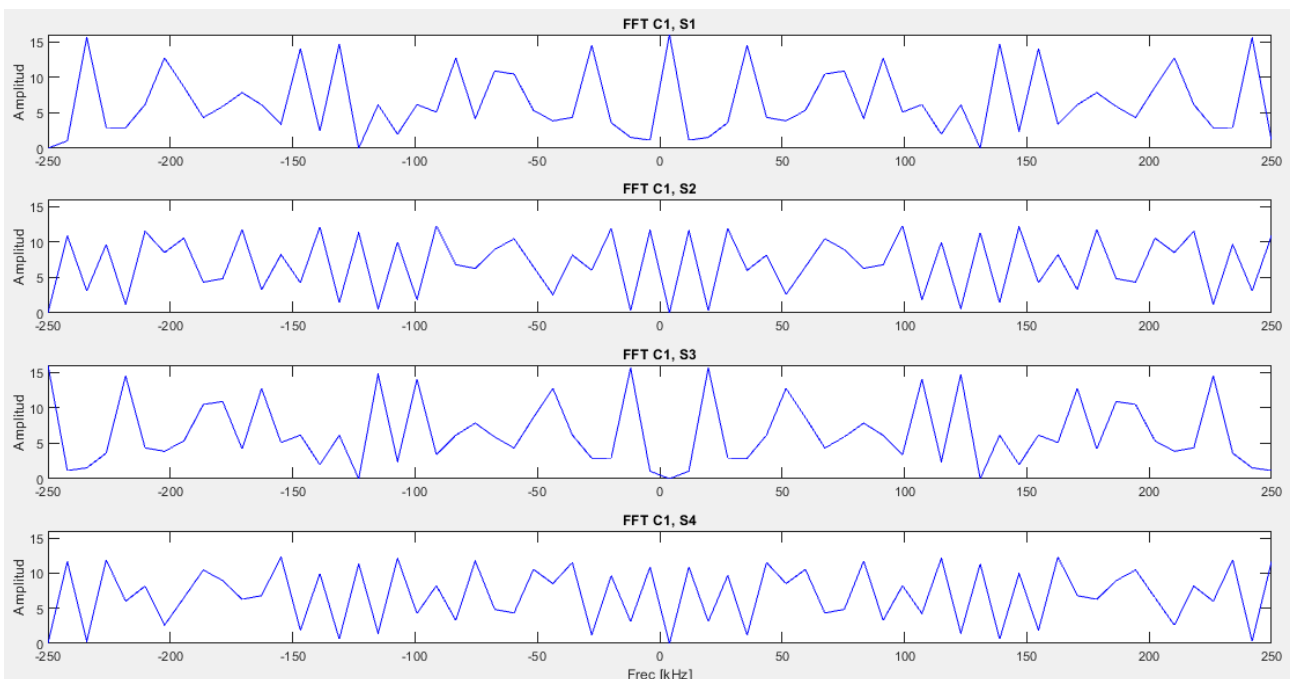


Fig. 126. Ventana 4 CSS: FFT de las M secuencias del código 1 banda base ideal

Una vez aplicada la reordenación de secuencias de cada código (**banda base**), el resto de la aplicación funciona igual que para el caso de las Kasami. Hay que tener en cuenta el detalle de que, al haber aumentado la longitud del código $M \cdot L$, los picos de correlación se encontrarán en ese punto en lugar de en L .

Como se explicó anteriormente, estos códigos necesitan un T_{gap} y un T_{shift} mínimo para no perder las propiedades de correlación. En la Fig. 127 se muestra la ventana de la **señal recibida sin efectos añadidos** con los valores de T_{gap} y T_{shift} citados más arriba. En ella se puede observar que, debido a la naturaleza que tienen los códigos, hay zonas de la señal que al sumarse se hacen nulas.

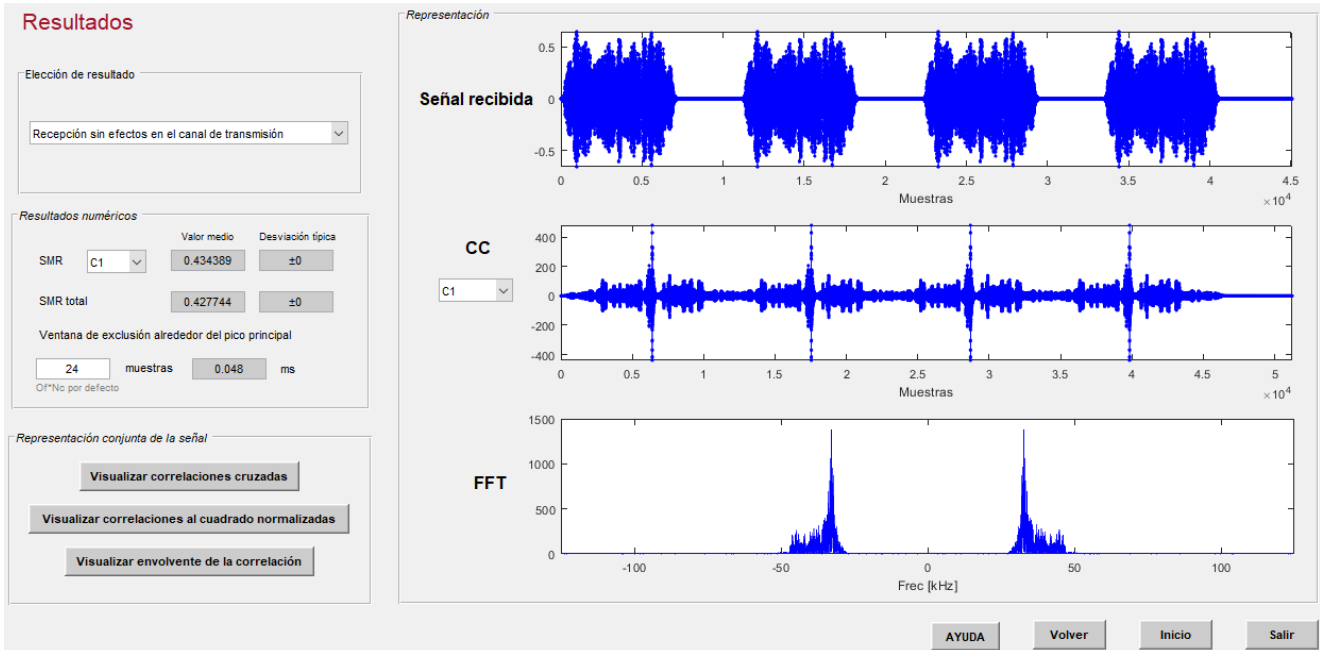


Fig. 127. Ventana 4 CSS: señal recibida sin efectos añadidos

4.5.4. Particularidades: LS

Estos códigos permiten realizar el cálculo de cotas considerando sólo la ZCZ o toda la señal. Esta opción se da en las ventanas de banda base, modulación y transductor. La figura 128 muestra la pantalla de **banda base** de los codigos LS. La figura 129 recoge la comparación de evaluar las cotas de dicha pantalla dentro y fuera de la ZCZ siendo la primera la más favorable.

Las propiedades de los códigos representados son: $M=8$, $Lo=16$, $Nemo=4$, portadora sinusoidal, $f_c=41,667\text{kHz}$, $f_{se}=500\text{kHz}$, $N_c=2$, $T_{gap}=10\text{ms}$, $T_{shift}=1\text{ms}$, $N_{rep}=4$, con efecto del transductor y sin efectos añadidos.

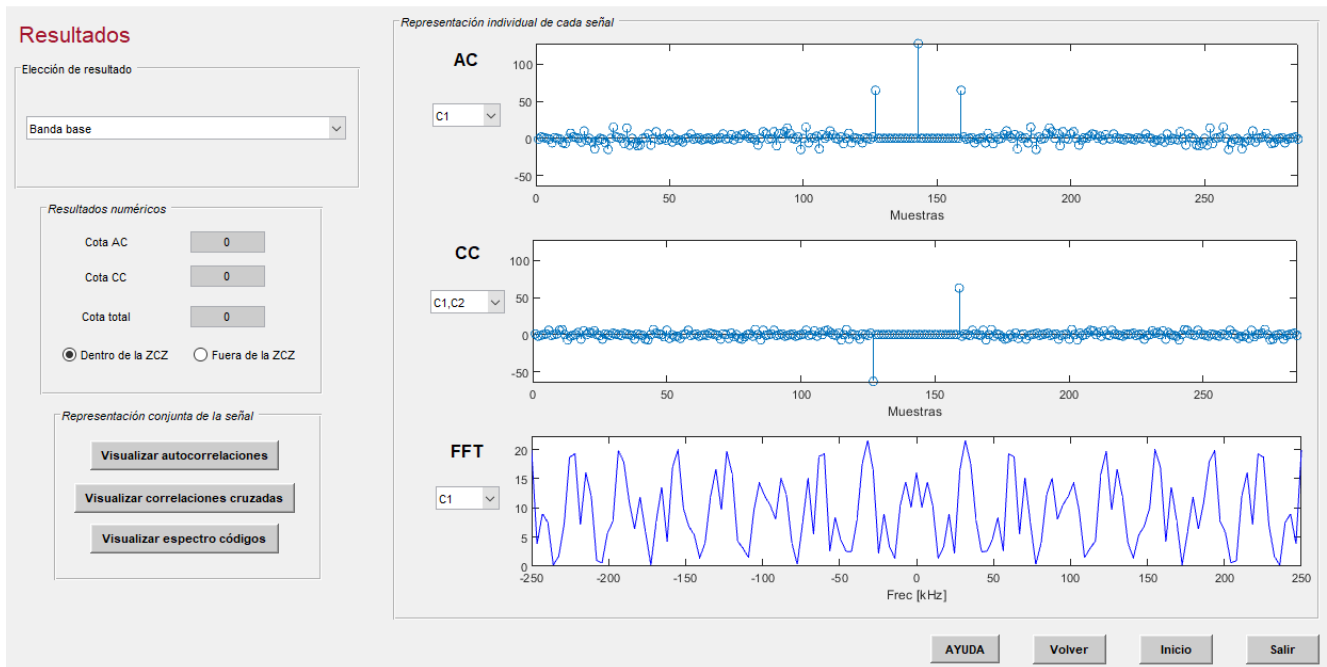


Fig. 128. Ventana 4 LS: banda base



Fig. 129. Comparación del resultado de cotas dentro y fuera de la ZCZ

Como pasa en las CSS, T_{gap} y T_{shift} deben ser mínimos para que no aparezcan partes nulas en la señal recibida. La Fig. 130 representa la pantalla de **señal recibida sin efectos** añadidos para T_{gap} y T_{shift} configurados más arriba.

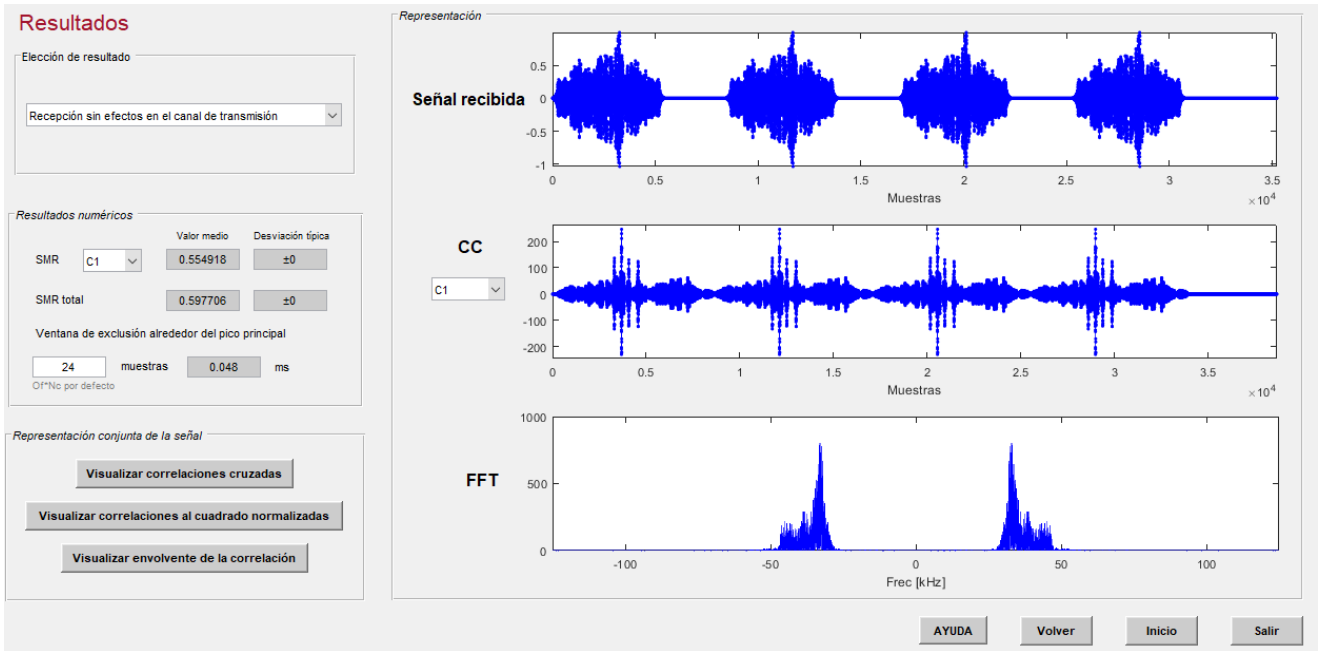


Fig. 130. Ventana 4 LS: señal recibida sin efectos añadidos

4.5.5. Particularidades: Zadoff-Chu

La visualización de resultados de los códigos Zadoff-Chu es similar a la de las Kasami. La única peculiaridad que tienen es que el resultado de las FFT (Fig. 131) en **banda base** son otras señales complejas de módulo distinto a uno.

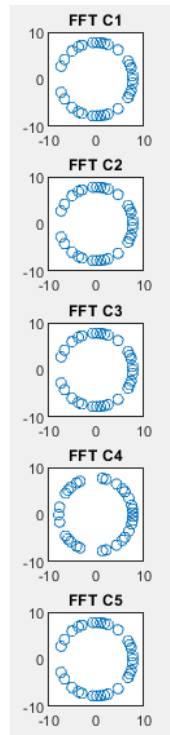


Fig. 131. Ventana 4 Zadoff-Chu: FFT banda base

4.6. VENTANA DE SALIDA

Cabe la posibilidad de ampliar este proyecto realizando pruebas experimentales a través de la herramienta Locate-Us cuyo proyecto trata del Desarrollo de un demostrador online para Sistemas de Posicionamiento Local Ultrasónicos [CHG⁺19a] [CHG⁺19b].

Las restricciones que existen para poder realizar estas pruebas aparecen en la figura 132 la cual representa la última pantalla que aparece en este TFG.



Fig. 132. Ventana 5 TFG: Locate-US

5. CONCLUSIONES Y LÍNEAS FUTURAS

5.1. Recopilación de logros del trabajo

Este trabajo de fin de grado ha desarrollado una herramienta experimental y didáctica para la prueba de distintos esquemas de codificación en sistemas basados en medida de tiempos de vuelo de señales ultrasónicas. La herramienta es útil tanto para investigadores del área, pudiendo analizar diseños de emisión, como para alumnos iniciados en el tema, ofreciendo una comprensión más fácil del funcionamiento de las señales ultrasónicas en sistemas de localización. Se puede comprobar de manera experimental los conocimientos adquiridos teóricamente y evaluar el efecto de los diferentes parámetros sobre los diferentes códigos: propiedades inherentes a los códigos emitidos, a la modulación utilizada (número de ciclos de portadora o sobremuestreo), ancho de banda del transductor o efectos típicos sobre la señal ultrasónica (cerca-lejos, ruido, Doppler, multicamino, entre otros).

El programa permite generar y almacenar códigos en formato .mat para su uso en sistemas prácticos, así la herramienta se ha enlazado con una aplicación desarrollada previamente en el grupo GEINTRA para la realización de pruebas reales con los códigos generados. La herramienta es intuitiva, fácil de utilizar con unos conocimientos básicos de sistemas de localización y siguiendo el Manual de Usuario, permitiendo afianzar los conceptos.

A nivel personal, este proyecto ha supuesto una oportunidad para poder conocer la teoría de los sistemas de localización ultrasónicos, esquemas de codificación y modulación y comprobar sus diversas propiedades mientras se diseñaba. Además, ha servido para los conocimientos de Matlab y de la programación.

5.2. Trabajos futuros

De cara a trabajos futuros, se proponen las siguientes mejoras y ampliaciones que no pudieron ser abarcadas por el presente TFG:

- Incorporación de otros esquemas de codificación y modulación: Chirp, códigos GPC, ...
- Añadir una ampliación que contenga la posibilidad de posicionar según la ubicación de las balizas.
- Relacionar el diseño de los códigos generados con pruebas experimentales.

6. PRESUPUESTO

6.1. Presupuesto de material y licencias

Material / licencia	Cantidad	Precio unitario (€)	Período de amortización (años)	Uso (meses)	Coste de amortización (€)
Ordenador	1	800	5	9	150
Licencia Matlab 2018a	1	800	1	9	600
Licencia Windows 10	1	145	1	9	108,75
Licencia Microsoft Office 2016	1	150	1	9	112,5
Total		3.130			971,25

Tabla 2. Presupuesto de materiales y licencias.

6.2. Presupuesto de las horas de desarrollo

Elemento desarrollado	Horas	Coste
Estudio de los esquemas de codificación	80	4.800 €
Interfaz gráfica Matlab-GUIDE	100	6.000 €
Pruebas y correcciones de errores	50	3.000 €
Escritura de la memoria	40	2.400 €
Escritura del manual de usuario	8	480 €
Grabación del videotutorial	5	200 €
Total	283	16.880 €

Tabla 3. Presupuesto de las horas de desarrollo

6.3. Presupuesto global

Bloque presupuestado	Coste
Material es y licencias	971,25 €
Horas de desarrollo	16.880 €
Total	17.851,25 €

Tabla 4. Presupuesto global

7. PLIEGO DE CONDICIONES

Para la programación y utilización de la aplicación se requieren los siguientes elementos:

- Ordenador
- Licencia Matlab para versión 2017a o superiores
- Conexión a internet para realizar pruebas experimentales

8. MANUAL DE USUARIO

8.1. Introducción

Dado que el Manual de Usuario será extraído para su uso por parte de los usuarios y debe ser autocontenido, algunas de las figuras, tablas o descripciones vistas en la memoria serán replicadas para ofrecer explicaciones del funcionamiento del sistema.

Para el uso de la aplicación desarrollada, se debe descargar la carpeta de nombre 'TFG' en la que se encuentran todos los archivos necesarios para el funcionamiento del programa.

Una vez instalado el programa 'Matlab', situaremos como directorio de trabajo la carpeta descargada llamada 'TFG'. Para el inicio de la interfaz (Fig. 133), se escribe en la ventana de comandos '*TFG_AndreaPerez*' (opción azul) o se abre el archivo '*TFG_AndreaPerez.m*' y se pulsa 'Run' desde el editor (opción roja).

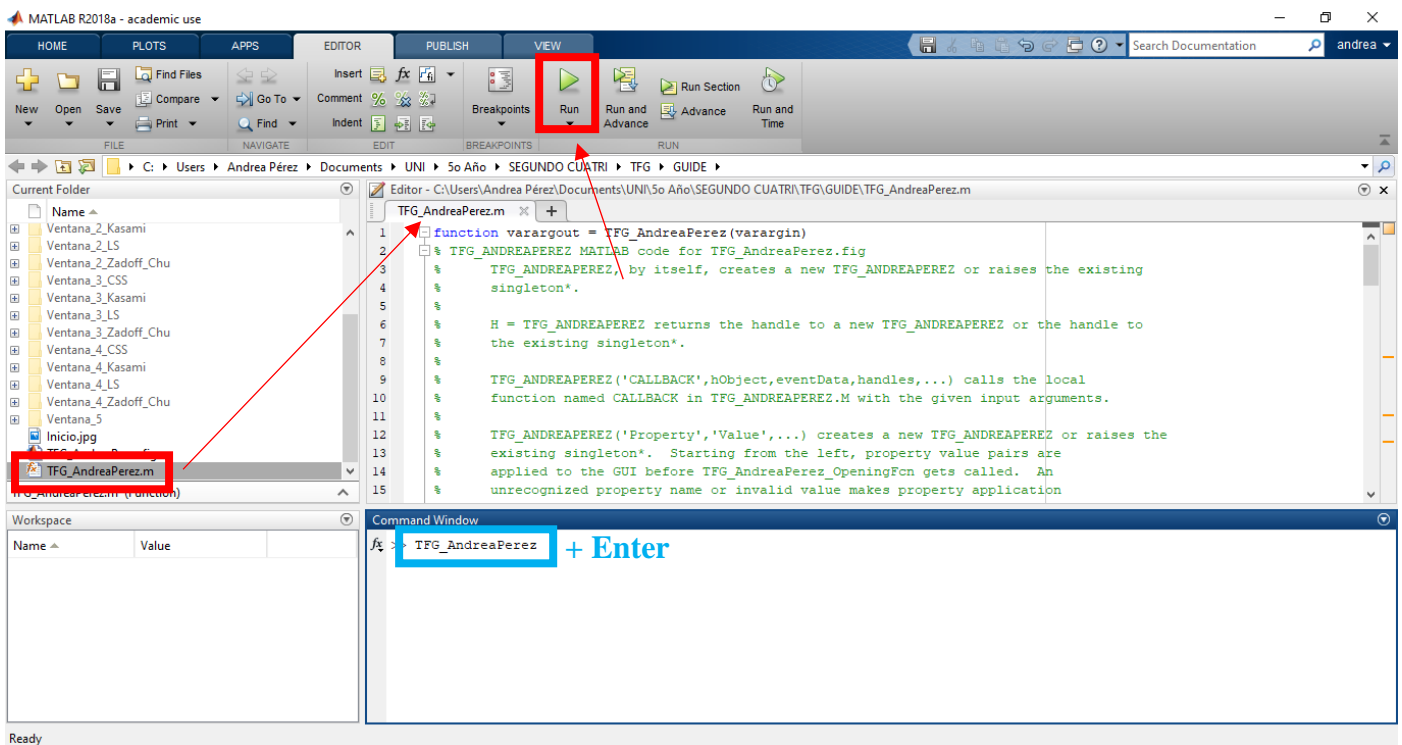


Fig. 133. Manual de Usuario: Inicio de la interfaz

8.2. Ventana Inicio

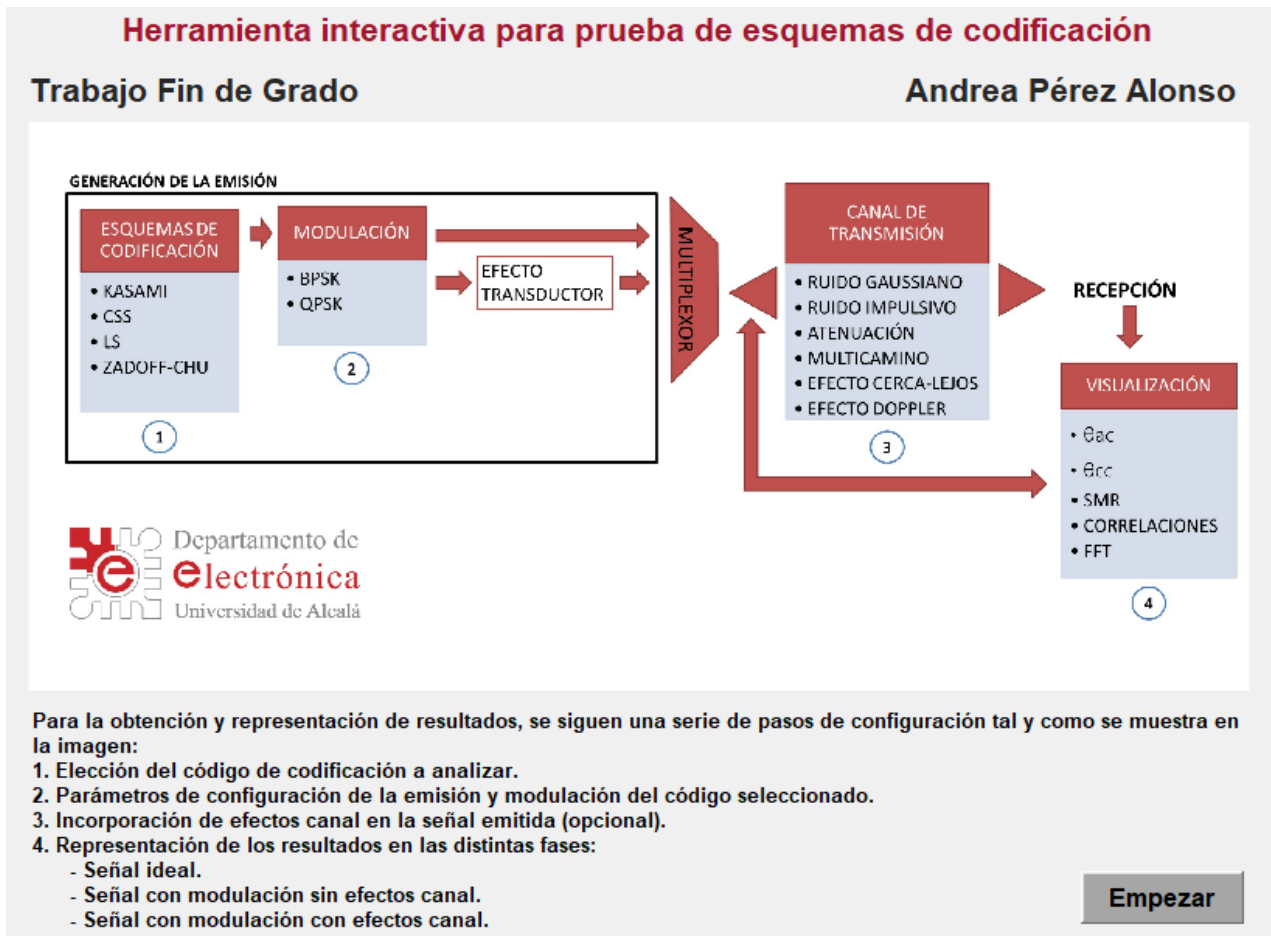


Fig. 134. Manual de Usuario: Ventana de Inicio

Esta ventana expone de forma general los distintos apartados que sigue la aplicación.



Continúa hacia la ventana de Selección del lenguaje de codificación.

8.3. Ventana de selección del lenguaje de codificación

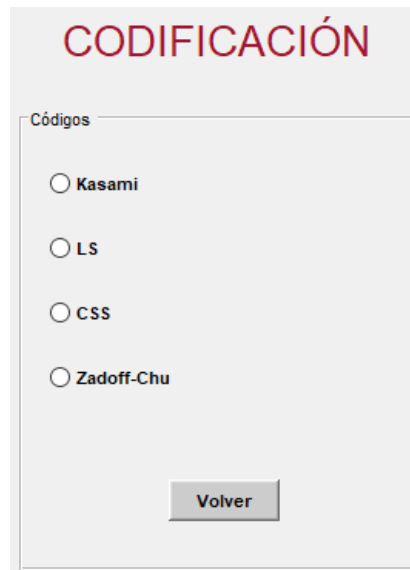


Fig. 135. Manual de Usuario: Selección del lenguaje de codificación

- Kasami** Continúa hacia la ventana de Configuración de emisión de secuencias Kasami (Capítulo 8.4.1)
- LS** Continúa hacia la ventana de Configuración de emisión de secuencias LS (Capítulo 8.4.2)
- CSS** Continúa hacia la ventana de Configuración de emisión de secuencias CSS (Capítulo 8.4.3)
- Zadoff-Chu** Continúa hacia la ventana de Configuración de emisión de secuencias Zadoff-Chu (Capítulo 8.4.4)
- Retrocede a la ventana de Inicio.

8.4. Ventana de Configuración de la emisión

8.4.1. Configuración de la emisión de secuencias Kasami

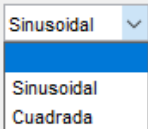
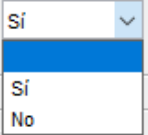
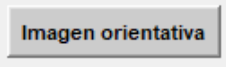
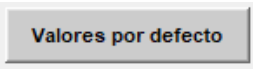
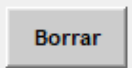
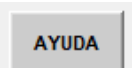

Fig. 136. Manual de Usuario: Ventana de Configuración de la emisión de secuencias Kasami

Las secuencias Kasami son de naturaleza binaria y se generan mediante la fórmula (23) [Pér19] [Kas68]. Debido a esta estructura binaria, utilizan la modulación BPSK, ver expresión (24).

$$C = \{m_1, m_1 \oplus m_2, m_1 \oplus Dm_2, m_1 \oplus D^2m_2, \dots, m_1 \oplus D^{L-1}m_2\} \quad (23)$$

$$x(t) = \begin{cases} A \sin(2\pi fct), & \text{Binario "1"} \\ A \sin(2\pi fct + \pi), & \text{Binario "0"} \end{cases} \quad (24)$$

Nomenclatura	Definición	Observaciones
L	Longitud del código	$L=2^N-1$. Restringida por N que debe ser un número par
M	Tamaño de familia	$M=2^{N/2}$. N se extrae de la fórmula de L
$Nemo$	Número de emisores simultáneos	$Nemo \leq M$
η	Eficiencia energética	$\eta = \frac{1}{L} \sum_{l=0}^{L-1} C_m[l] \leq 1$ siendo $1 \leq m \leq Nemo$
f_c	Frecuencia portadora	
f_{se}	Frecuencia de muestreo	
N_c	Número de ciclos de la portadora	
Of	Factor de sobremuestreo	$Of=f_{se}/f_c$. Se aproxima este valor al número entero más cercano y se modifica la f_c
T_e	Duración de la emisión de un código	$T_e=L \cdot N_c/f_c$
T_{gap}	Espaciado entre emisiones consecutivas	
T_{shift}	Retardo entre emisores	
N_{rep}	Número de repeticiones del código	

Opciones	Definición
	Forma de la señal portadora
	Incorporar efecto de transductor o no
	Gráfica en la que aparecen T_{gap} , T_{shift} y N_{rep} para un mejor entendimiento. Véase Fig. 137
	Introduce en los campos de configuración unos valores determinados para proporcionar una mejor orientación al usuario
	Elimina todo valor introducido en los campos de configuración
	Abre el capítulo correspondiente del Manual de Usuario para esta pantalla
	Retrocede a la ventana de Selección del lenguaje de codificación

Siguiente	Continúa a la ventana de Configuración de efectos añadidos
Guardar archivo .mat código sin modular	Almacena en el directorio de trabajo el código sin modular de dimensiones $Nemo \times L$
Visualizar código sin modular	Visualiza los <i>Nemo</i> códigos sin modular generados. Véase Fig. 139
Guardar archivo .mat código modulado	Almacena en el directorio de trabajo el código sin modular de dimensiones $Nemo \times T_{e[muestras]}$
Visualizar señal portadora	Visualiza la señal portadora que se va a utilizar para modular la señal. Véase Fig. 140
Visualizar señal modulada	Visualiza los <i>Nemo</i> códigos modulados. Véase Fig. 141
Visualizar señal con efecto transductor	Visualiza los <i>Nemo</i> códigos modulados con efecto transductor. Véase Fig. 142
Ver modelo transductor	Indica la respuesta en frecuencia del transductor utilizado, así como sus características principales
Visualizar señales a emitir	Visualiza los <i>Nemo</i> códigos a emitir. Véase Fig. 143

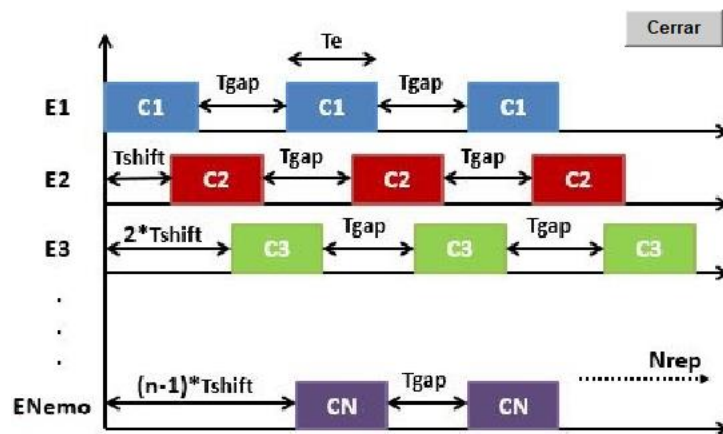


Fig. 137. Imágen representativa

Ejemplo.

Configuración de emisión: Kasami

Entradas

Longitud (L) Tamaño familia (M)

Nº emisores simultáneos (Nemo) Eficiencia energética (η)

Modulación BPSK:

Tipo portadora

Frecuencia portadora (fc) kHz Factor sobremuestreo (Of)

Frecuencia de muestreo (fse) kHz Duración emisión un código (Te) ms

Nº ciclos portadora (Nc)

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms

Número de repeticiones (Nrep)

Modelo transductor

Salidas

Fig. 138. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Datos

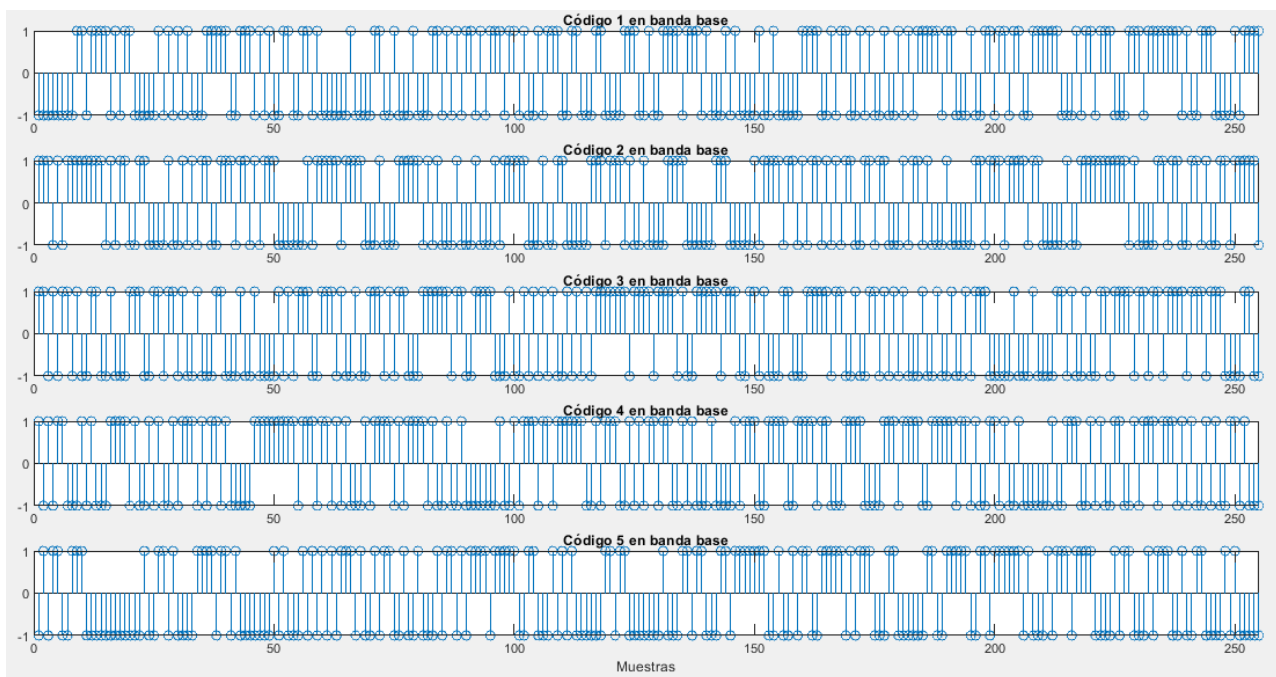


Fig. 139. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización banda base

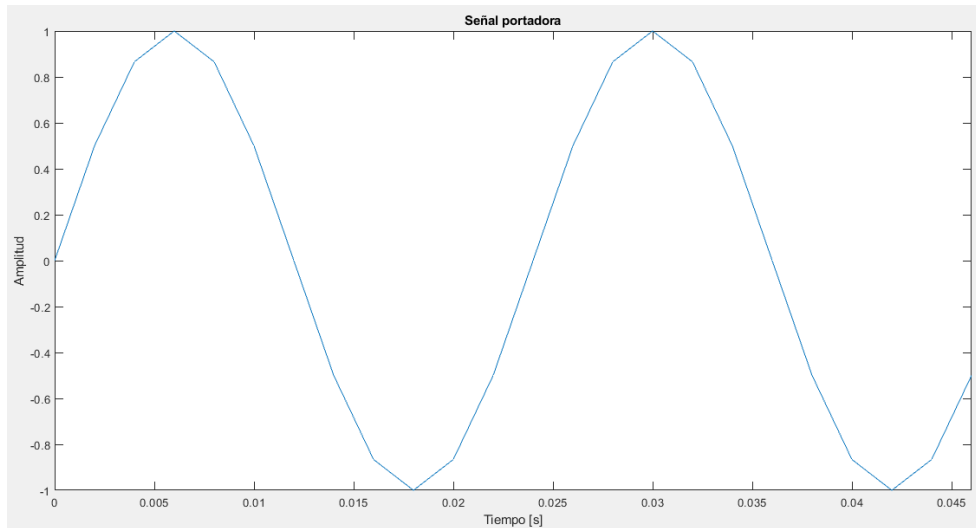


Fig. 140. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la portadora

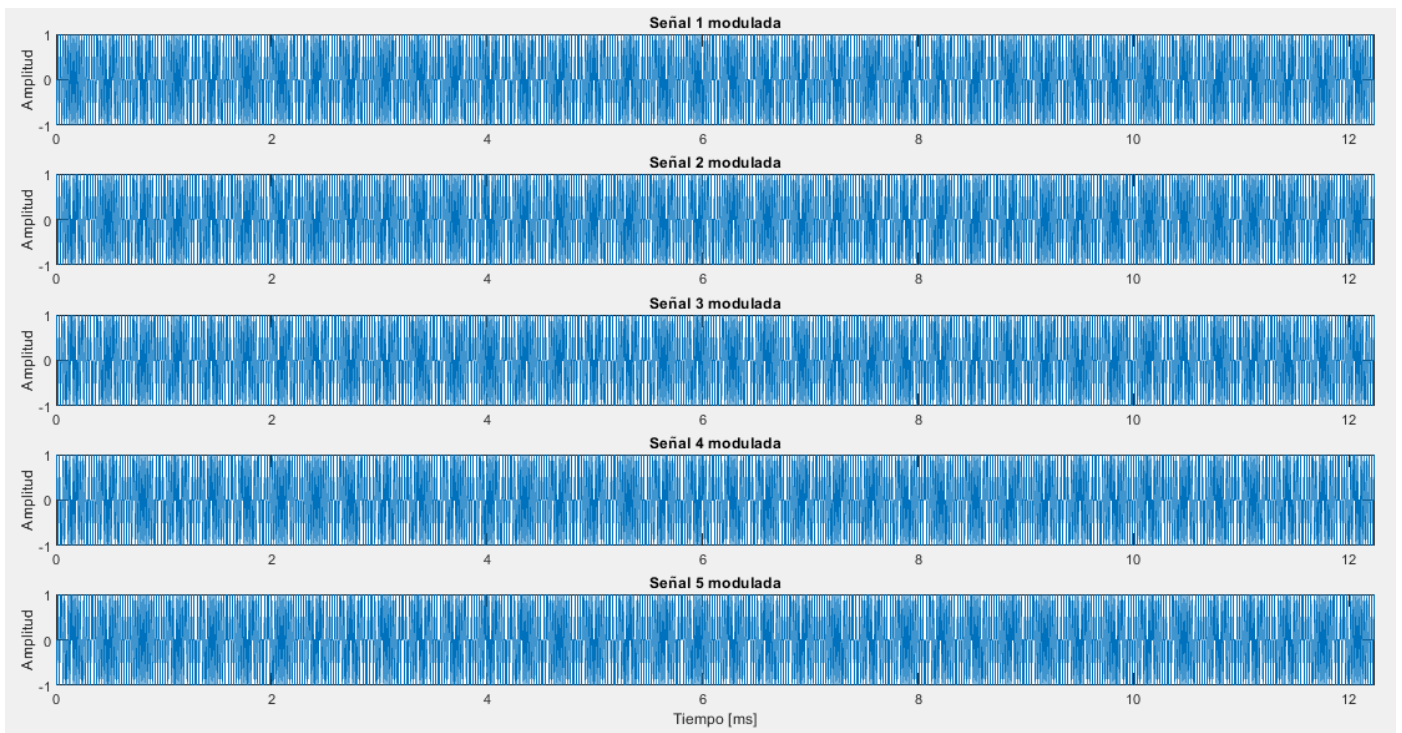


Fig. 141. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la señal modulada

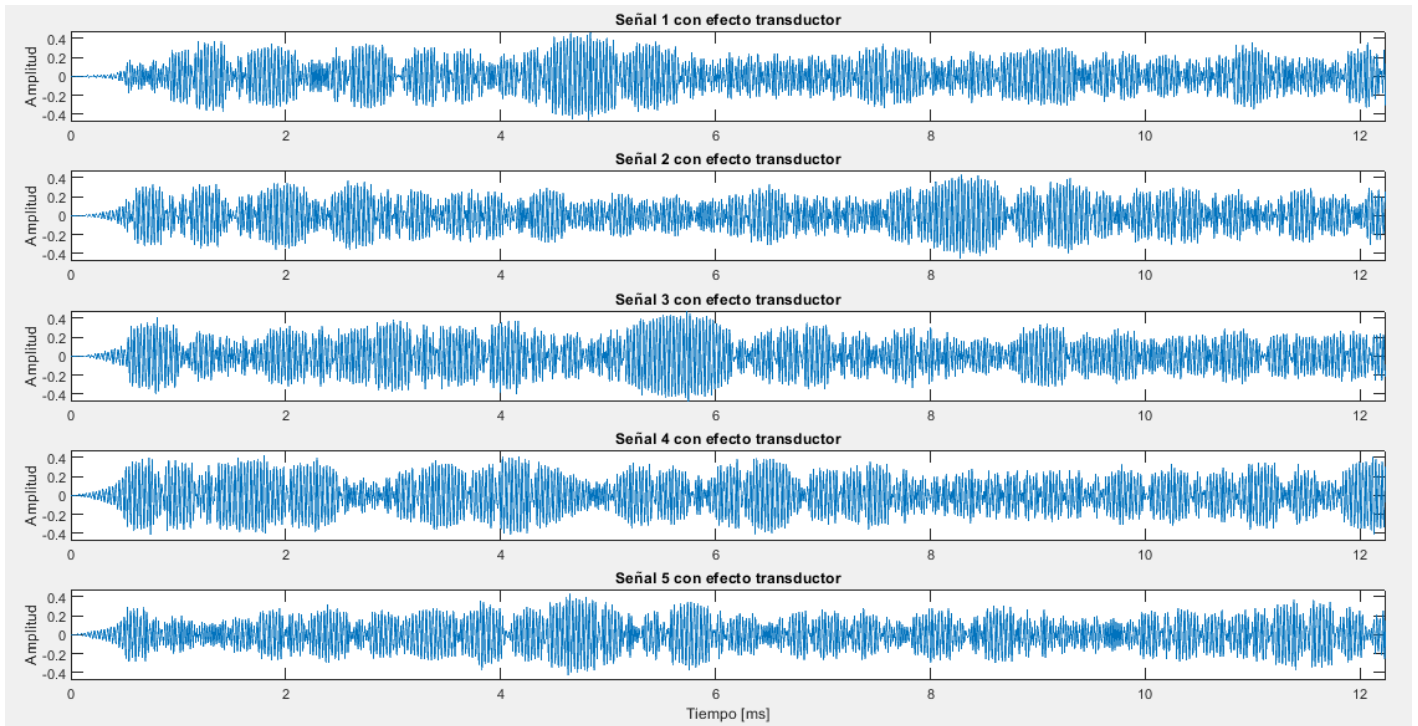


Fig. 142. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la señal con efecto del transductor

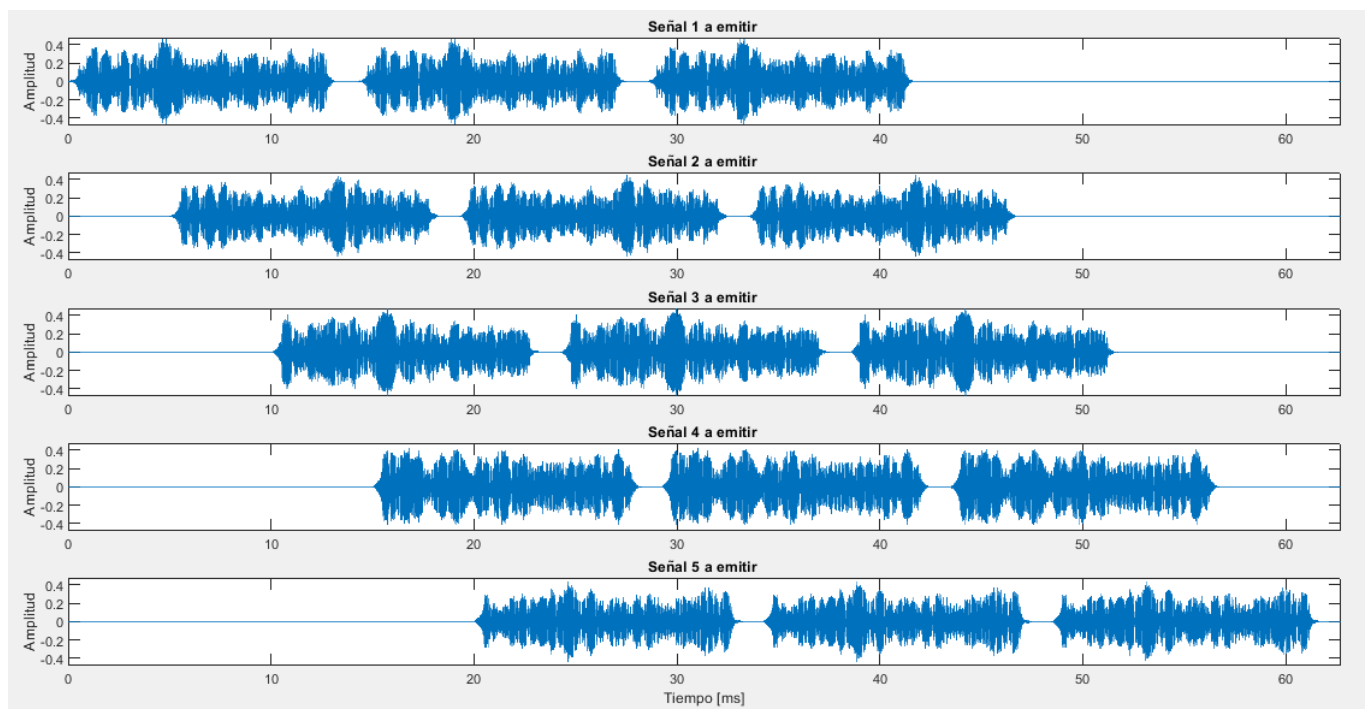


Fig. 143. Manual de Usuario: Configuración de emisión de secuencias Kasami: Ejemplo: Visualización de la señal a emitir

8.4.2. Configuración de la emisión de secuencias CSS

Fig. 144. Manual de Usuario: Ventana de Configuración de la emisión de secuencias CSS

Los conjuntos M -CSS (*Complementary Set of Sequences*) constituyen una generalización de las parejas Golay, pudiendo componerse con un número de secuencias M mayor de dos. Un sistema CSS está constituido como $C_{i_s_j}[l]$ (M secuencias S_j para cada $Nemo$ código C_i de longitud L) donde $1 \leq i \leq Nemo$, $1 \leq j \leq M$, $1 \leq l \leq L$ [TL72]. Son de naturaleza binaria y por ello utilizan la modulación BPSK, ver expresión (25) [GPA⁺15].

$$x(t) = \begin{cases} A \sin(2\pi fct), & \text{Binario "1"} \\ A \sin(2\pi fct + \pi), & \text{Binario "0"} \end{cases} \quad (25)$$

Para su correspondiente emisión, es necesario agrupar las secuencias de cada código en una sola señal de longitud $M \cdot L$. Para ello, se utilizan mecanismos de reordenación de códigos:

Concatenación: (26)

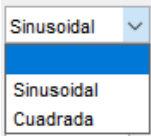
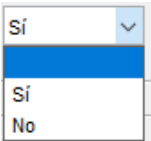
$$\{C_{i,s_1}[0] \dots C_{i,s_1}[L-1] \ C_{i,s_2}[0] \dots C_{i,s_2}[L-1] \ \dots \ C_{i,s_M}[0] \dots C_{i,s_M}[L-1]\}$$

Entrelazado:

(27)

$$\{C_{i,s1}[0] C_{i,s2}[0] \dots C_{i,sM}[0] \dots C_{i,s1}[L-1] C_{i,s2}[L-1] \dots C_{i,sM}[L-1]\}$$

Nomenclatura	Definición	Observaciones
M	Tamaño de familia	Debe ser potencia de 2
L	Longitud del código	$L=M^N$ siendo N un número natural. Debe ser potencia de M
$Nemo$	Número de emisores simultáneos	$Nemo \leq M$
η	Eficiencia energética	$\eta = \frac{1}{L} \sum_{l=0}^{L-1} C_m[l] \leq 1$ siendo $1 \leq m \leq Nemo$
fc	Frecuencia portadora	
fse	Frecuencia de muestreo	
Nc	Número de ciclos de la portadora	
Of	Factor de sobremuestreo	$Of=fse/fc$. Se aproxima este valor al número entero más cercano y se modifica la fc
Te	Duración de la emisión de un código	$Te=L \cdot Nc/fc$
$Tgap$	Espaciado entre emisiones consecutivas	Se recomienda un valor de por lo menos Te para obtener mejores resultados
$Tshift$	Retardo entre emisores	Se recomienda introducir un valor mínimo, 1 ms p.e.
$Nrep$	Número de repeticiones del código	

Opciones	Definición
<input checked="" type="radio"/> Concatenado	Método de la concatenación para la reordenación de las M secuencias de cada código. Véase (26)
<input type="radio"/> Entrelazado	Método del entrelazado para la reordenación de las M secuencias de cada código. Véase Fig. (27)
	Forma de la señal portadora
	Incorporar efecto de transductor o no

<p>Imagen orientativa</p>	<p>Gráfica en la que aparecen T_{gap}, T_{shift} y N_{rep} para un mejor entendimiento. Véase Fig. 145</p>
<p>Valores por defecto</p>	<p>Introduce en los campos de configuración unos valores determinados para proporcionar una mejor orientación al usuario</p>
<p>Borrar</p>	<p>Elimina todo valor introducido en los campos de configuración</p>
<p>AYUDA</p>	<p>Abre el capítulo correspondiente del Manual de Usuario para esta pantalla</p>
<p>Volver</p>	<p>Retrocede a la ventana de Selección del lenguaje de codificación</p>
<p>Siguiente</p>	<p>Continúa a la ventana de Configuración de efectos añadidos</p>
<p>Guardar archivo .mat código sin modular</p>	<p>Almacena en el directorio de trabajo el código sin modular de dimensiones $M \times L \times Nemo$</p>
<p>Visualizar código sin modular <input type="text" value="C1"/></p>	<p>Visualiza las M secuencias de código C_i que se seleccione en la pestaña siendo $1 \leq i \leq Nemo$, y la señal reordenada a través del mecanismo configurado. Véase Fig. 147</p>
<p>Guardar archivo .mat código modulado</p>	<p>Almacena en el directorio de trabajo el código sin modular de dimensiones $Nemo \times T_{e[muestras]}$</p>
<p>Visualizar señal portadora</p>	<p>Visualiza la señal portadora que se va a utilizar para modular la señal. Véase Fig. 148</p>
<p>Visualizar señal modulada</p>	<p>Visualiza los $Nemo$ códigos modulados. Véase Fig. 149</p>
<p>Visualizar señal con efecto transductor</p>	<p>Visualiza los $Nemo$ códigos modulados con efecto transductor. Véase Fig. 150</p>
<p>Ver modelo transductor</p>	<p>Indica la respuesta en frecuencia del transductor utilizado, así como sus características principales</p>
<p>Visualizar señales a emitir</p>	<p>Visualiza los $Nemo$ códigos a emitir. Véase Fig. 151</p>

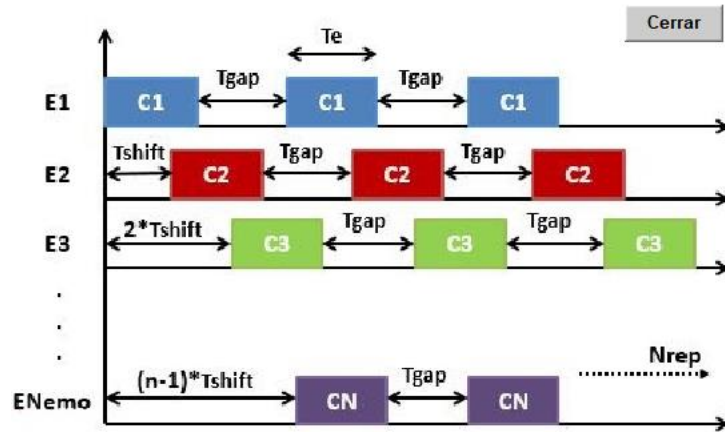


Fig. 145. Imágen representativa

Ejemplo.

Configuración de emisión: CSS

Entradas

Tamaño familia (M) Concatenado Entrelazado

Longitud (L) Eficiencia energética (η)

Nº emisores simultáneos (Nemo)

Modulación BPSK:

Tipo portadora kHz

Frecuencia portadora (fc) kHz

Frecuencia de muestreo (fse) ms

Nº ciclos portadora (Nc) Factor sobremuestreo (Of)

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms

Número de repeticiones (Nrep)

Modelo transductor

Salidas

C2

Fig. 146. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Datos

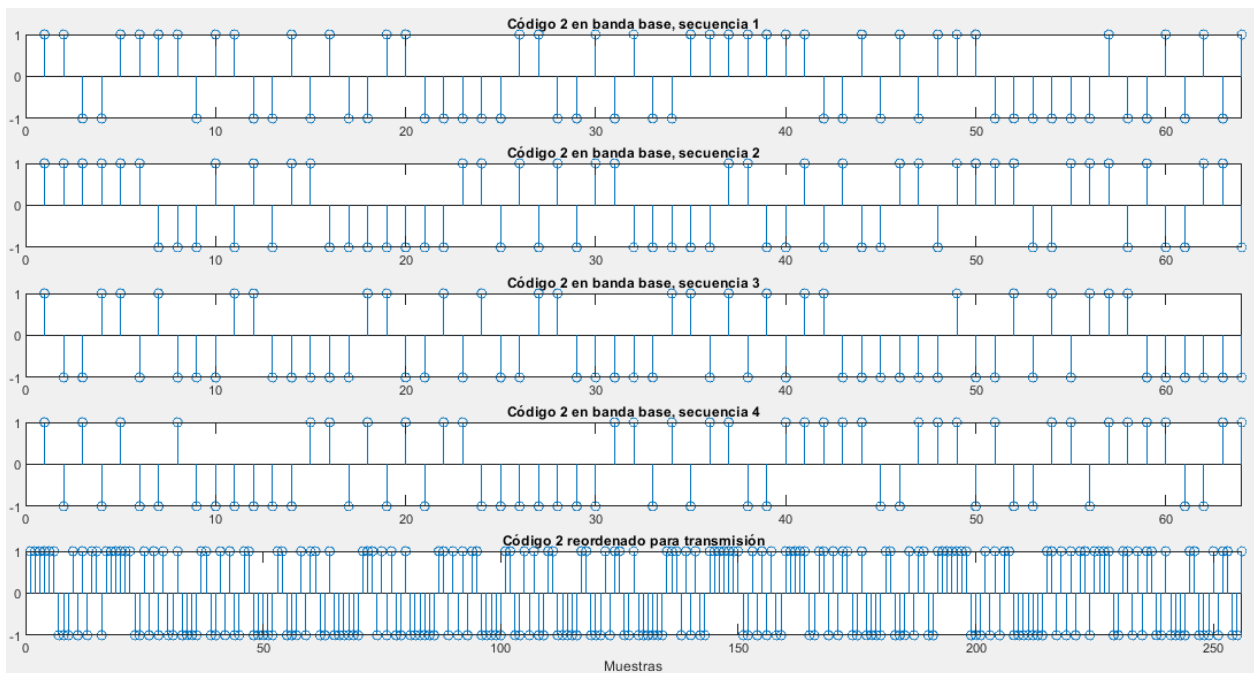


Fig. 147. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización banda base

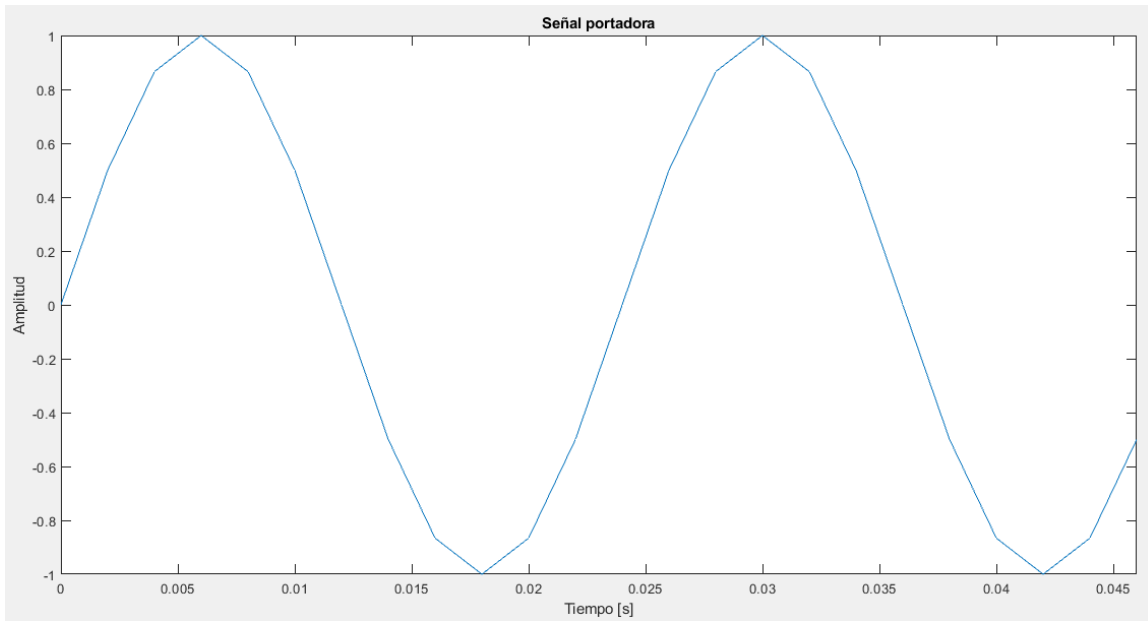


Fig. 148. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la portadora

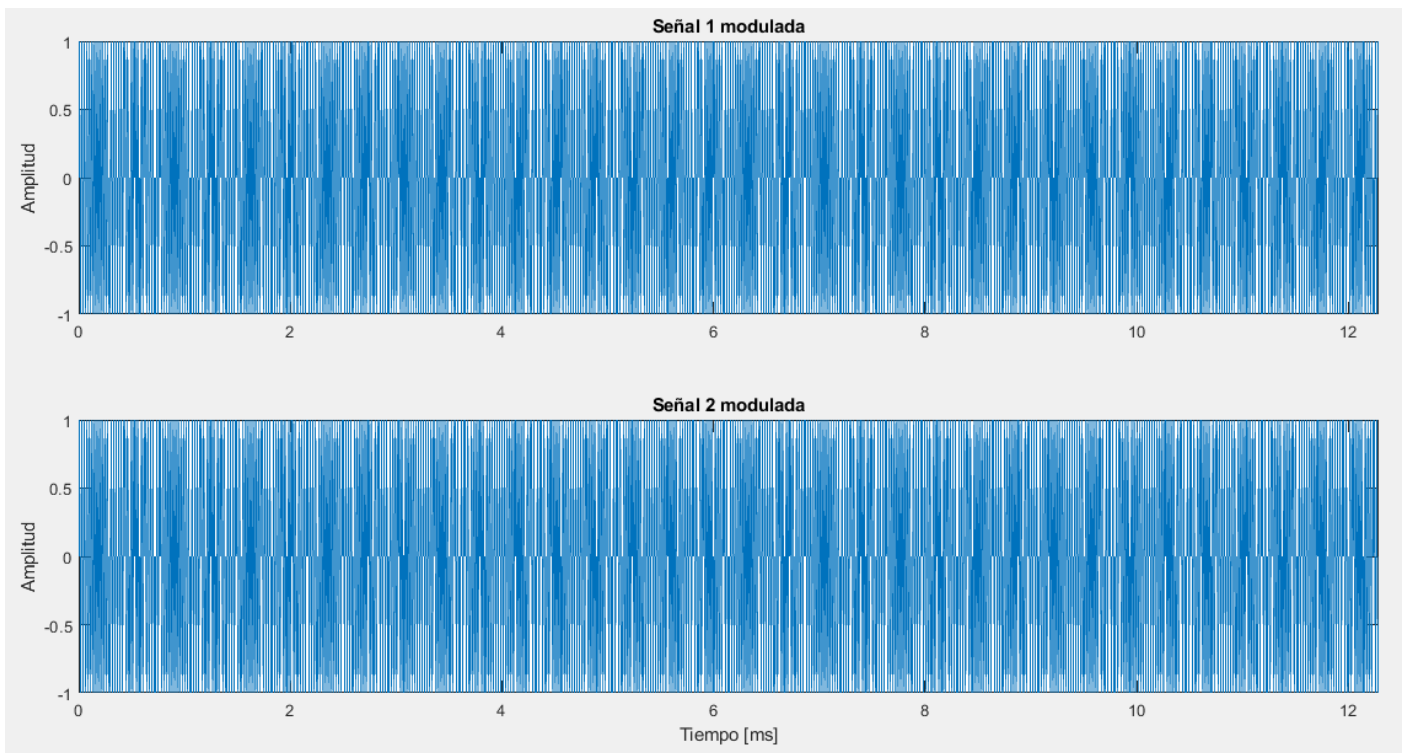


Fig. 149. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la señal modulada

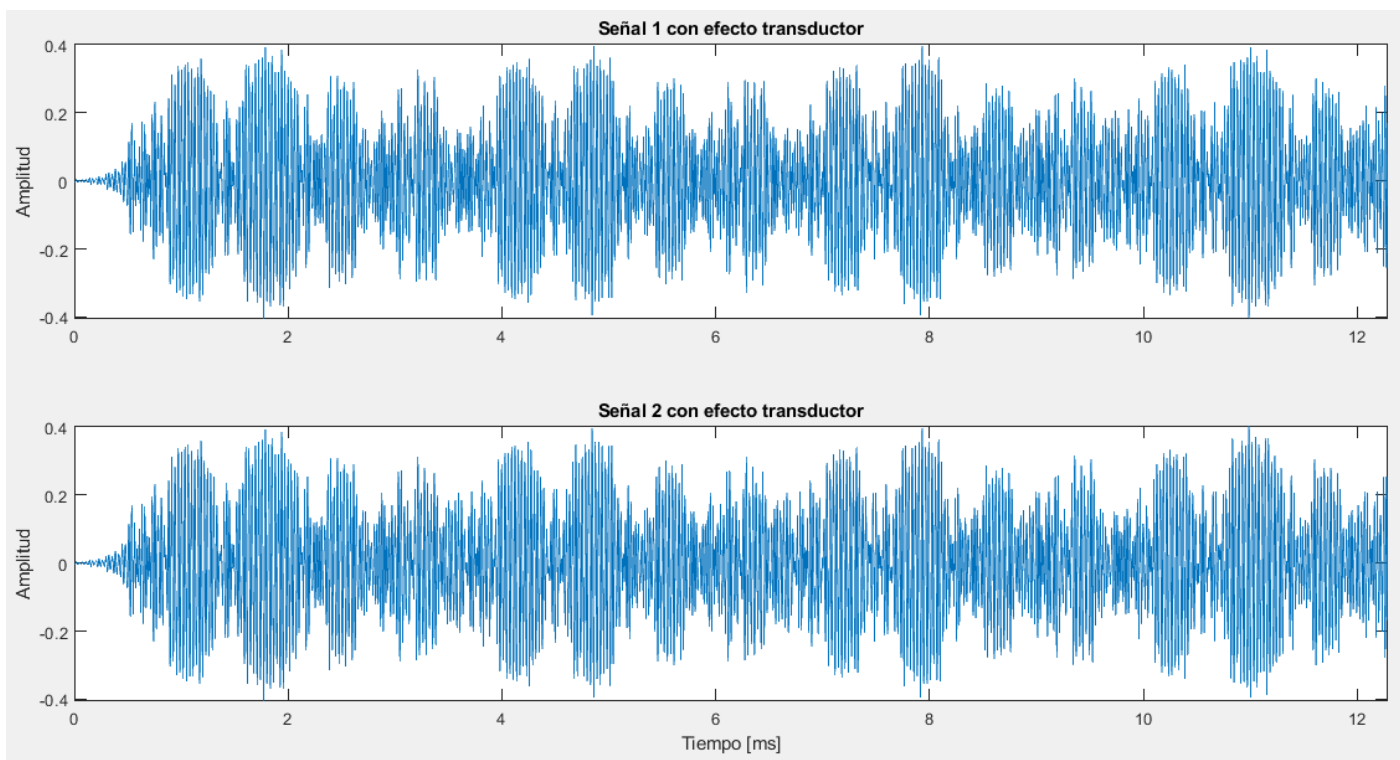


Fig. 150. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la señal con efecto del transductor

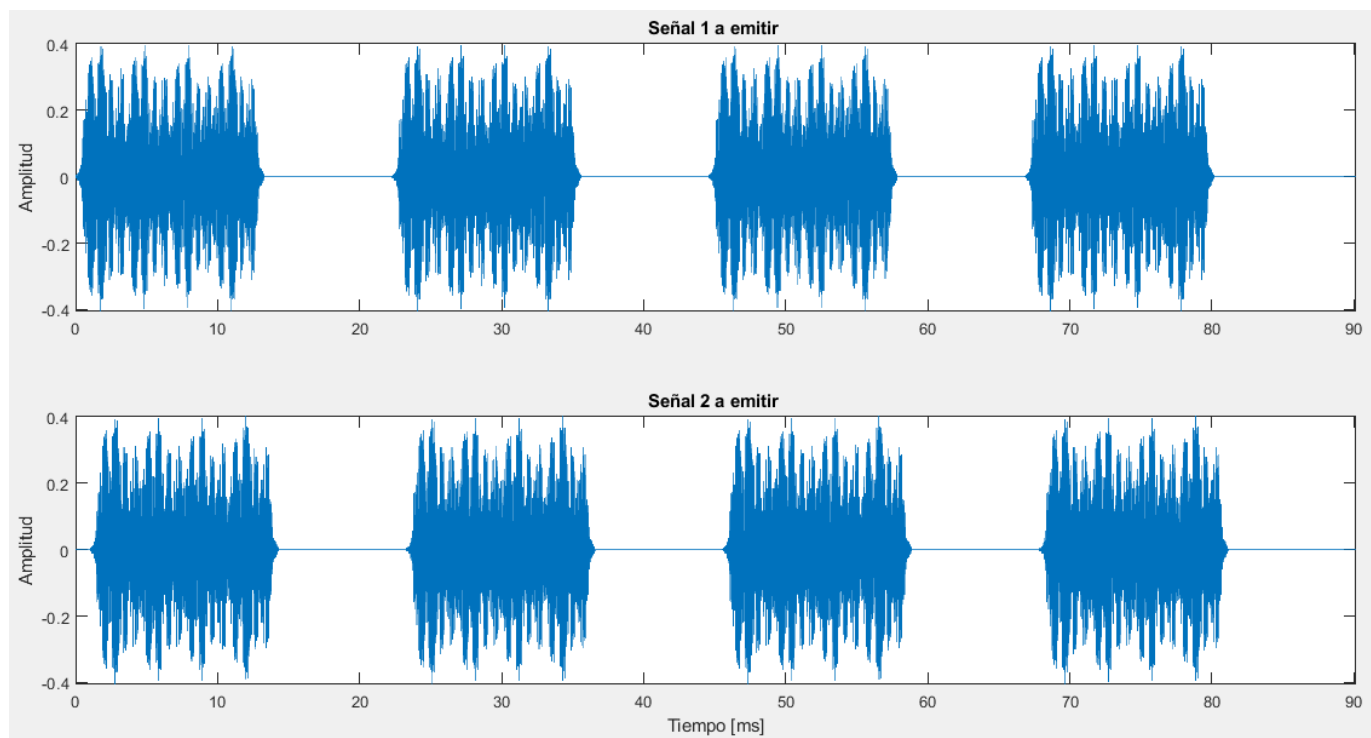


Fig. 151. Manual de Usuario: Configuración de emisión de secuencias CSS: Ejemplo: Visualización de la señal a emitir

8.4.3. Configuración de la emisión de secuencias LS

Fig. 152. Manual de Usuario: Ventana de Configuración de la emisión de secuencias CSS

Un conjunto de $M = 2^n$, $n \in \mathbb{N}$, códigos LS (*Loosely Synchronous codes*) de longitud L $\{C = c_m[l]; 0 \leq m \leq M-1; 0 \leq l \leq L-1\}$ compuesto por elementos $\{-1,0,1\}$ puede generarse a partir de dos parejas Golay $G_1 = (G_{1,s_1}, G_{1,s_2})$ y $G_2 = (G_{2,s_1}, G_{2,s_2})$ de longitud L_0 e incorreladas entre sí, según la siguiente ecuación [Pér19] [SBH01]:

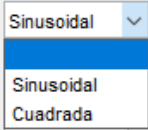
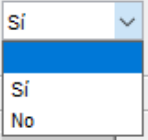
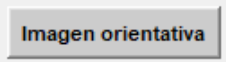
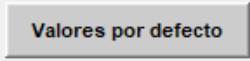
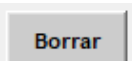
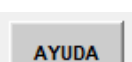
$$C[z] = \sum_{i=1}^{M/2} h_{m,i} \cdot z^{-(i-1)L_0} \cdot [s_{\pi_{i,0}}(z) + z^{-\left(\frac{M}{2}L_0+W\right)} s_{\pi_{i,1}}(z)] \quad (28)$$

Estos conjuntos tienen una zona de correlación cero ZCZ alrededor del origen al ser códigos ortogonales generalizados.

Por su naturaleza binaria, utilizan la modulación BPSK, ver expresión (29).

$$x(t) = \begin{cases} A \sin(2\pi fct), & \text{Binario "1"} \\ A \sin(2\pi fct + \pi), & \text{Binario "0"} \end{cases} \quad (29)$$

Nomenclatura	Definición	Observaciones
M	Tamaño de familia	Debe ser potencia de 2
L_0	Longitud de parejas Golay base	Debe ser potencia de 2
ZCZ	Zona de correlación cero alrededor del origen	$ZCZ=2 \cdot W+1$
W	Semiventana libre de interferencias	$W=L_0-1$
L	Longitud del código	$L=L_0 \cdot M+W$
$Nemo$	Número de emisores simultáneos	$Nemo \leq M$
η	Eficiencia energética	$\eta = \frac{1}{L} \sum_{l=0}^{L-1} C_m[l] \leq 1$ siendo $1 \leq m \leq Nemo$
f_c	Frecuencia portadora	
f_{se}	Frecuencia de muestreo	
N_c	Número de ciclos de la portadora	
Of	Factor de sobremuestreo	$Of=f_{se}/f_c$. Se aproxima este valor al número entero más cercano y se modifica la f_c
T_e	Duración de la emisión de un código	$T_e=L \cdot N_c/f_c$
T_{gap}	Espaciado entre emisiones consecutivas	Se recomienda un valor de por lo menos T_e para obtener mejores resultados
T_{shift}	Retardo entre emisores	Se recomienda introducir un valor mínimo, 1 ms p.e.
N_{rep}	Número de repeticiones del código	

Opciones	Definición
	Forma de la señal portadora
	Incorporar efecto de transductor o no
	Gráfica en la que aparecen T_{gap} , T_{shift} y N_{rep} para un mejor entendimiento. Véase Fig. 153
	Introduce en los campos de configuración unos valores determinados para proporcionar una mejor orientación al usuario
	Elimina todo valor introducido en los campos de configuración
	Abre el capítulo correspondiente del Manual de Usuario para esta pantalla

Volver	Retrocede a la ventana de Selección del lenguaje de codificación
Siguiente	Continúa a la ventana de Configuración de efectos añadidos
Guardar archivo .mat código sin modular	Almacena en el directorio de trabajo el código sin modular de dimensiones $Nemo \times L$
Visualizar código sin modular	Visualiza los <i>Nemo</i> códigos generados sin modular. Véase Fig. 155
Guardar archivo .mat código modulado	Almacena en el directorio de trabajo el código sin modular de dimensiones $Nemo \times T_{e[muestras]}$
Visualizar señal portadora	Visualiza la señal portadora que se va a utilizar para modular la señal. Véase Fig. 156
Visualizar señal modulada	Visualiza los <i>Nemo</i> códigos modulados. Véase Fig. 157
Visualizar señal con efecto transductor	Visualiza los <i>Nemo</i> códigos modulados con efecto transductor. Véase Fig. 158
Ver modelo transductor	Indica la respuesta en frecuencia del transductor utilizado, así como sus características principales
Visualizar señales a emitir	Visualiza los <i>Nemo</i> códigos a emitir. Véase Fig. 159

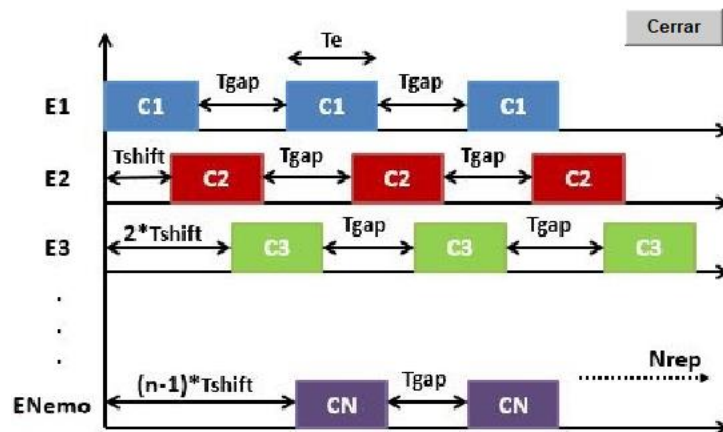


Fig. 153. Imágen representativa

Ejemplo.

Configuración de emisión: LS

Entradas

Tamaño familia (M)	<input type="text" value="8"/>	Zona correlación cero (ZCZ)	<input type="text" value="31"/>
Longitud parejas Golay Base (Lo)	<input type="text" value="16"/>	Longitud código (L)	<input type="text" value="143"/>
Nº emisores simultáneos (Nemo)	<input type="text" value="4"/>	Eficiencia energética (η)	<input type="text" value="0.895105"/>

Modulación BPSK:

Tipo portadora	<input type="text" value="Cuadrada"/>		
Frecuencia portadora (fc)	<input type="text" value="41.6667"/>	kHz	
Frecuencia de muestreo (fse)	<input type="text" value="500"/>	kHz	Factor sobremuestreo (Of)
Nº ciclos portadora (Nc)	<input type="text" value="2"/>		<input type="text" value="12"/>
		Duración emisión un código (Te)	<input type="text" value="6.86399"/>
		ms	

Emisión:

Espaciado entre emisiones consecutivas (Tgap)	<input type="text" value="10"/>	ms	Imagen orientativa
Retardo entre emisores (Tshift)	<input type="text" value="1"/>	ms	
Número de repeticiones (Nrep)	<input type="text" value="4"/>		
Modelo transductor	<input type="text" value="Sí"/>		

Salidas

Fig. 154. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Datos

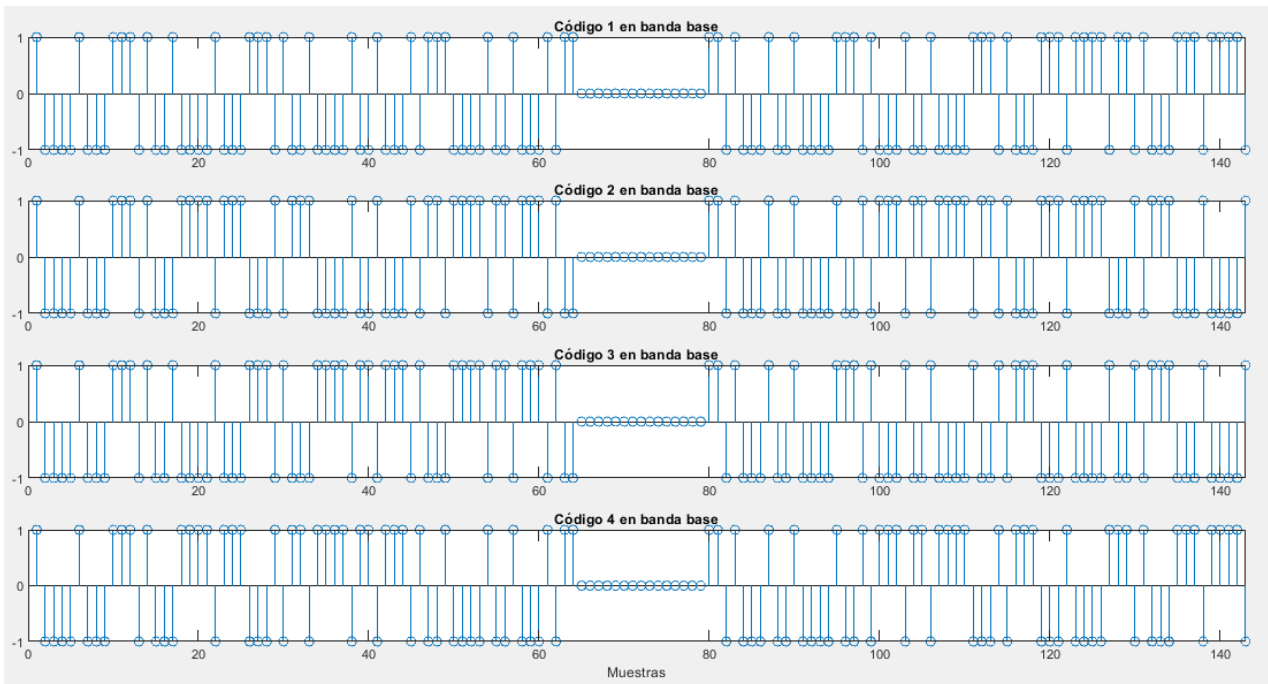


Fig. 155. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización banda base

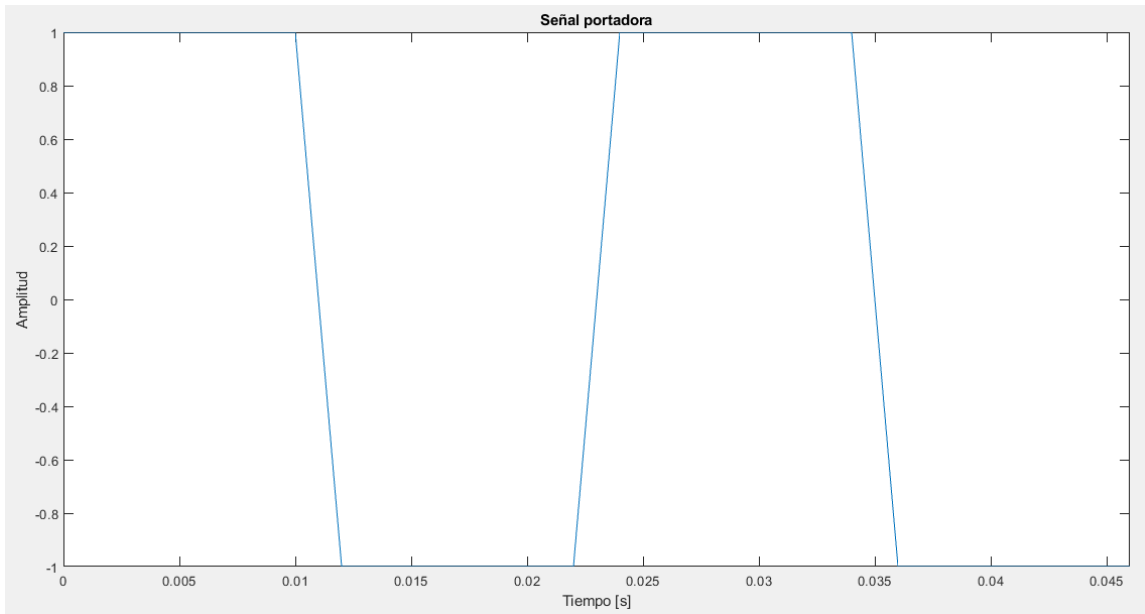


Fig. 156. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la portadora

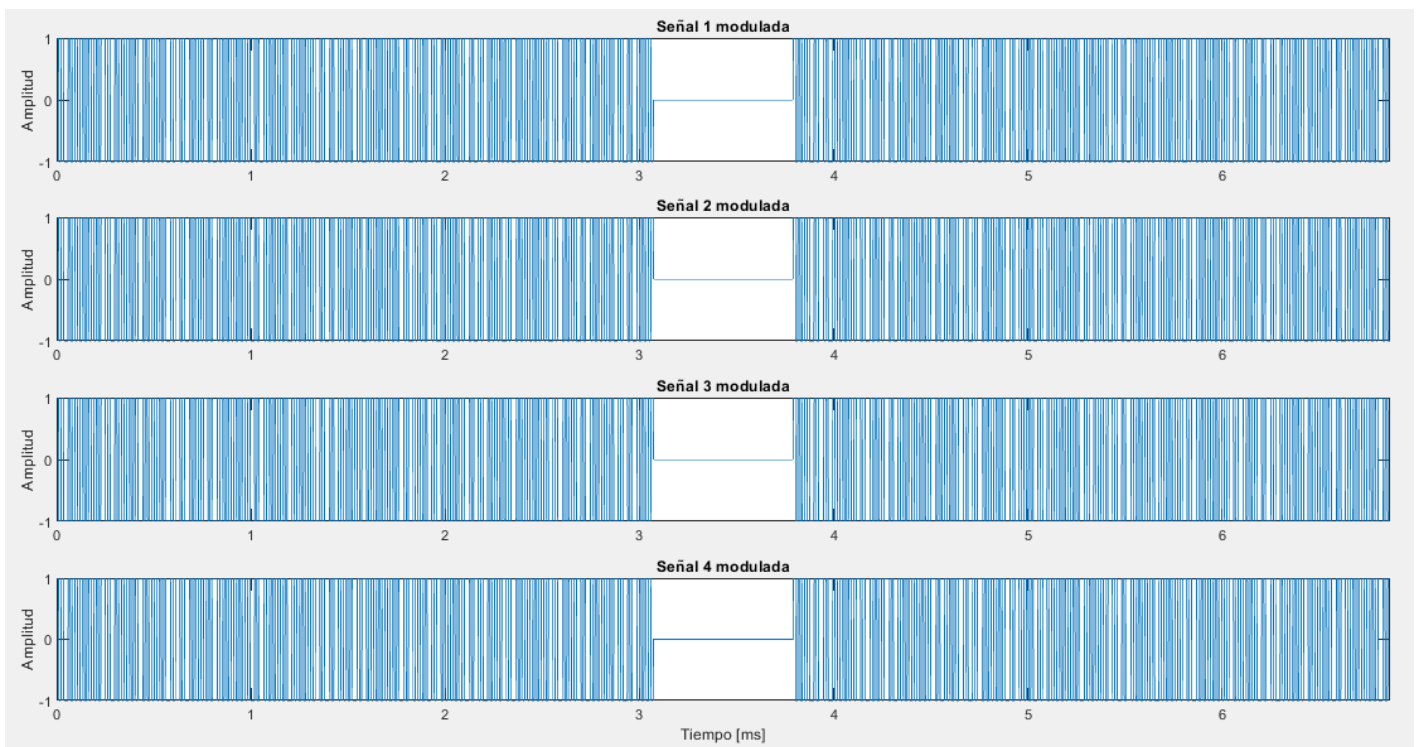


Fig. 157. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la señal modulada

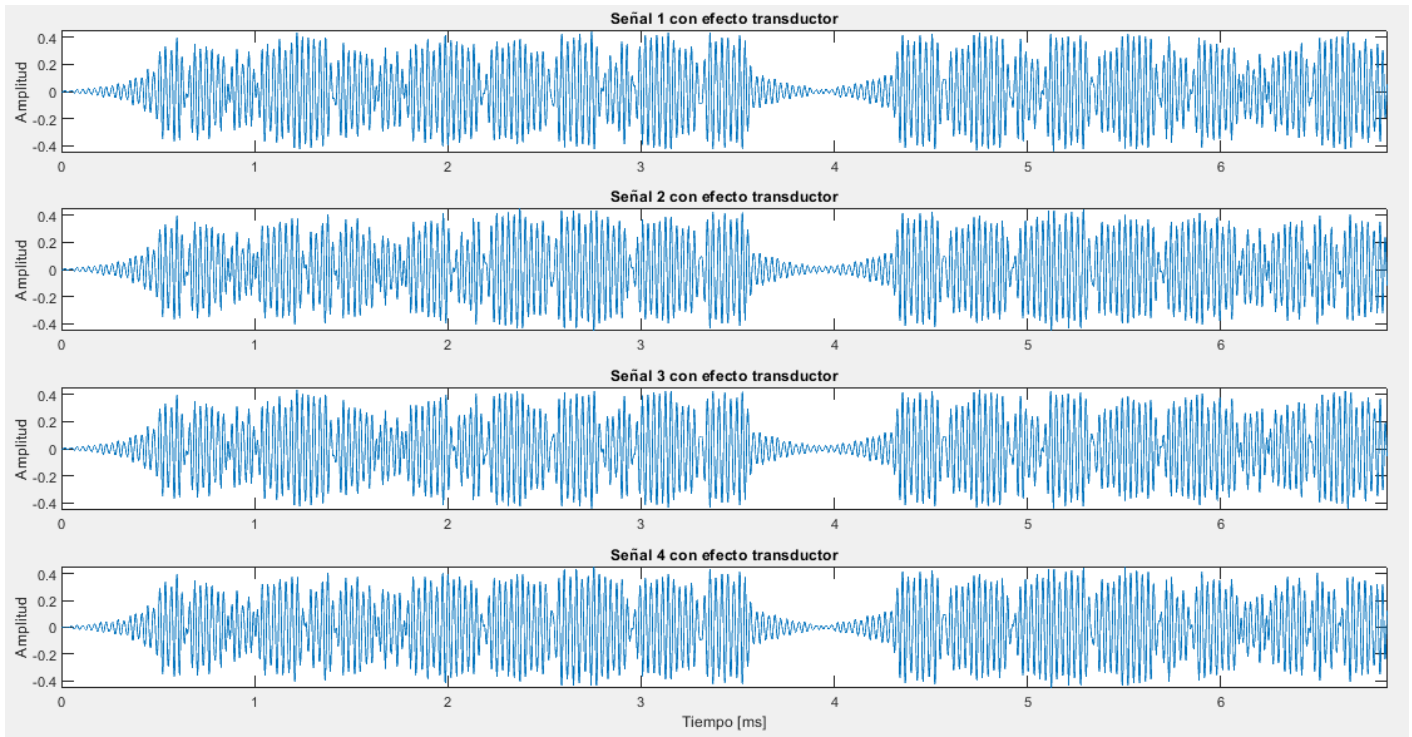


Fig. 158. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la señal con efecto del transductor

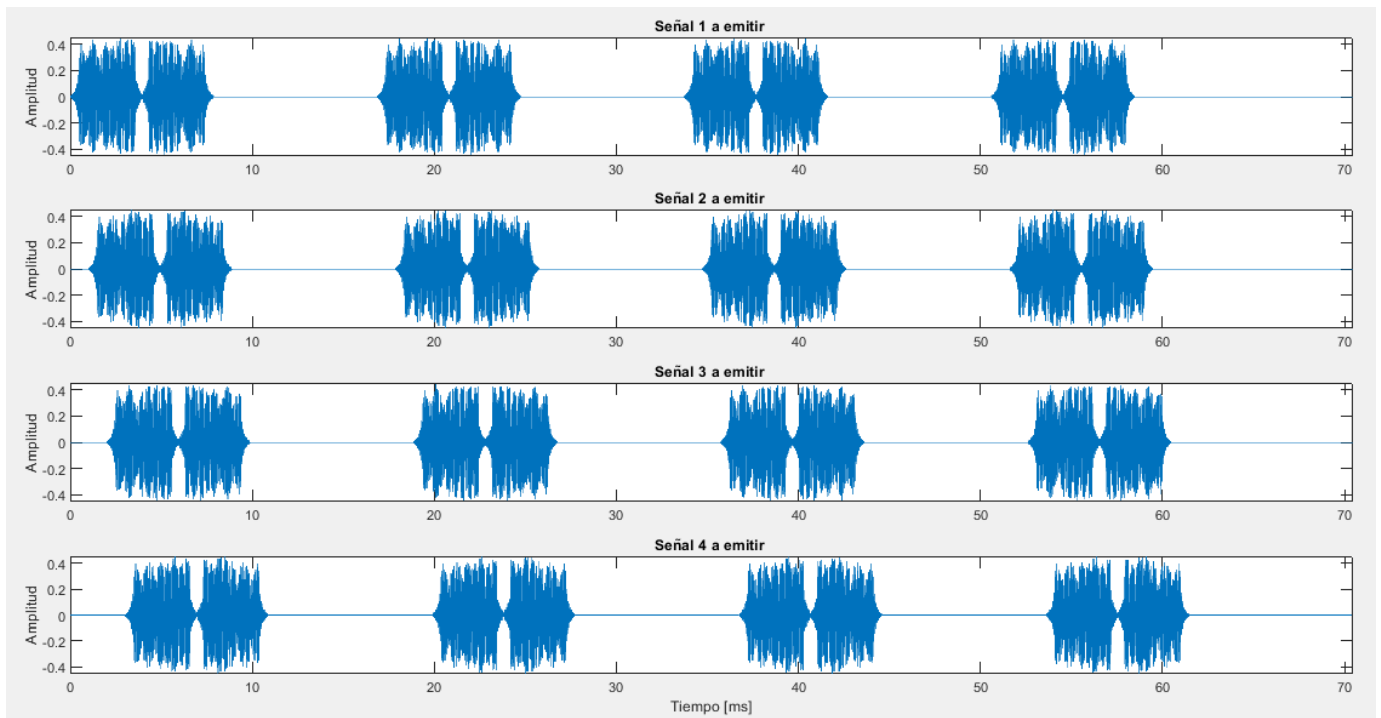


Fig. 159. Manual de Usuario: Configuración de emisión de secuencias LS: Ejemplo: Visualización de la señal a emitir

8.4.4. Configuración de la emisión de secuencias Zadoff-Chu

Fig. 160. Manual de Usuario: Ventana de Configuración de la emisión de secuencias Zadoff-Chu

Las secuencias Zadoff-Chu, son códigos polifásicos y complejos de módulo unitario. Dichas secuencias C_r de longitud L se puede definir de la siguiente manera [MPU⁺18] [Chu72]:

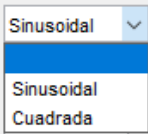
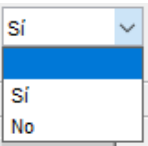
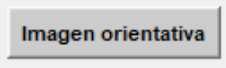
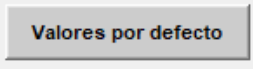
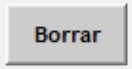
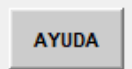
$$C_r[l] = e^{-i\frac{2\pi r l(l+1)}{2L}}, \quad 0 \leq l \leq L-1 \quad (30)$$

La longitud del código L es un número impar y r es un número entero conocido como índice raíz o semilla. La secuencia va a tener un total de $L-1$ semillas. En función del $Nemo$ que se escojan a través de la interfaz, se seleccionarán por orden ascendente aquellas semillas que tengan el pico de correlación principal menor.

Al ser de naturaleza compleja, se modula mediante QPSK. Véase (31).

$$x(t) = \begin{cases} A \sin(2\pi fct + \frac{\pi}{4}), & \text{Binario "11"} \\ A \sin(2\pi fct + \frac{3\pi}{4}), & \text{Binario "01"} \\ A \sin(2\pi fct - \frac{3\pi}{4}), & \text{Binario "00"} \\ A \sin(2\pi fct - \frac{\pi}{4}), & \text{Binario "10"} \end{cases} \quad (31)$$

Nomenclatura	Definición	Observaciones
L	Longitud del código	Debe ser impar. Se recomienda que sea un número primo
M	Tamaño de familia	$M=L-1$
$Nemo$	Número de emisores simultáneos	$Nemo \leq M$
η	Eficiencia energética	$\eta = \frac{1}{L} \sum_{l=0}^{L-1} C_m[l] \leq 1$ siendo $1 \leq m \leq Nemo$
fc	Frecuencia portadora	
fse	Frecuencia de muestreo	
Nc	Número de ciclos de la portadora	
Of	Factor de sobremuestreo	$Of=fse/fc$. Se aproxima este valor al número entero más cercano y se modifica la fc
Te	Duración de la emisión de un código	$Te=L \cdot Nc/fc$
$Tgap$	Espaciado entre emisiones consecutivas	
$Tshift$	Retardo entre emisores	
$Nrep$	Número de repeticiones del código	

Opciones	Definición
	Forma de la señal portadora
	Incorporar efecto de transductor o no
	Gráfica en la que aparecen $Tgap$, $Tshift$ y $Nrep$ para un mejor entendimiento. Véase Fig. 161
	Introduce en los campos de configuración unos valores determinados para proporcionar una mejor orientación al usuario
	Elimina todo valor introducido en los campos de configuración
	Abre el capítulo correspondiente del Manual de Usuario para esta pantalla

Volver	Retrocede a la ventana de Selección del lenguaje de codificación
Siguiente	Continúa a la ventana de Configuración de efectos añadidos
Guardar archivo .mat código sin modular	Almacena en el directorio de trabajo el código sin modular de dimensiones $NemoL$ de números complejos
Visualizar código sin modular	Visualiza los <i>Nemo</i> códigos complejos sin modular generados en valor absoluto. Véase Fig. 163
Guardar archivo .mat código modulado	Almacena en el directorio de trabajo el código sin modular de dimensiones $NemoTe_{[muestras]}$
Visualizar señal portadora	Visualiza la señal portadora que se va a utilizar para modular la señal. Véase Fig. 164
Visualizar señal modulada	Visualiza los <i>Nemo</i> códigos modulados. Véase Fig. 165
Visualizar señal con efecto transductor	Visualiza los <i>Nemo</i> códigos modulados con efecto transductor. Véase Fig. 166
Ver modelo transductor	Indica la respuesta en frecuencia del transductor utilizado, así como sus características principales
Visualizar señales a emitir	Visualiza los <i>Nemo</i> códigos a emitir. Véase Fig. 167

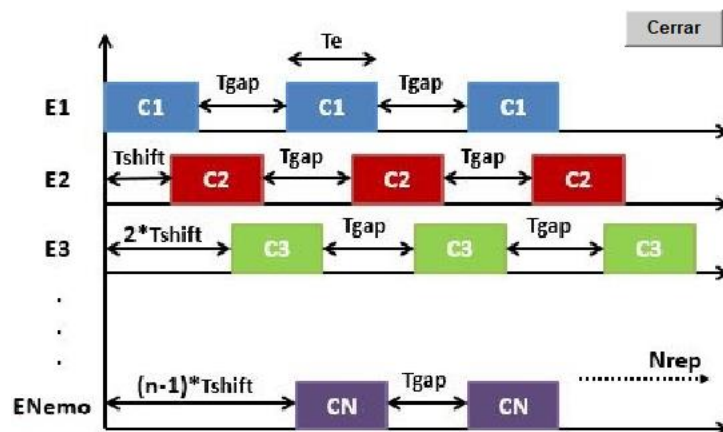


Fig. 161. Imágen representativa

Ejemplo.

Configuración de emisión: Zadoff_Chú

Entradas

Longitud (L) Tamaño familia (M)

Nº emisores simultáneos (Nemo) Eficiencia energética (η)

Modulación QPSK:

Tipo portadora

Frecuencia portadora (fc) kHz

Frecuencia de muestreo (fse) kHz Factor sobremuestreo (Of)

Nº ciclos portadora (Nc) Duración emisión un código (Te) ms

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms

Número de repeticiones (Nrep)

Modelo transductor

Salidas

Fig. 162. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Datos

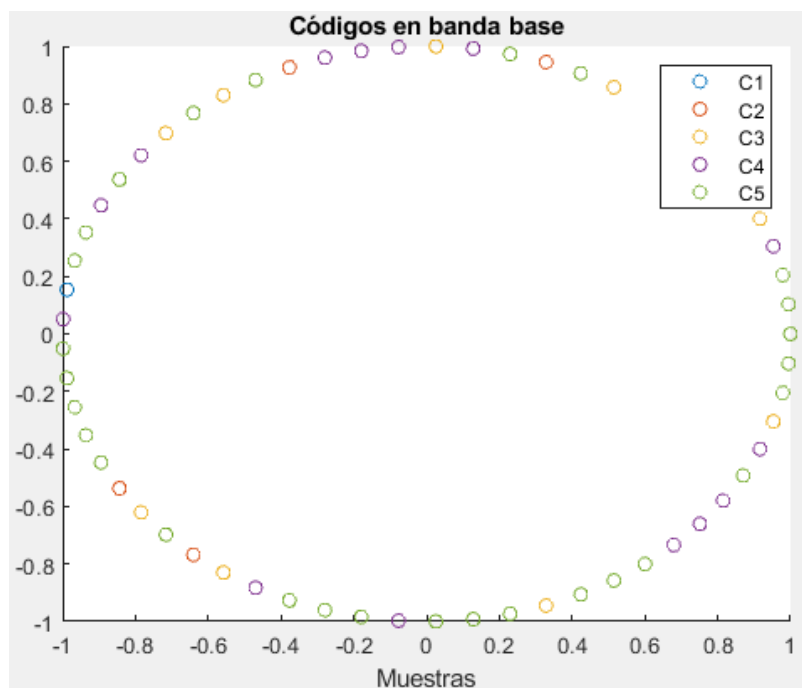


Fig. 163. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización banda base

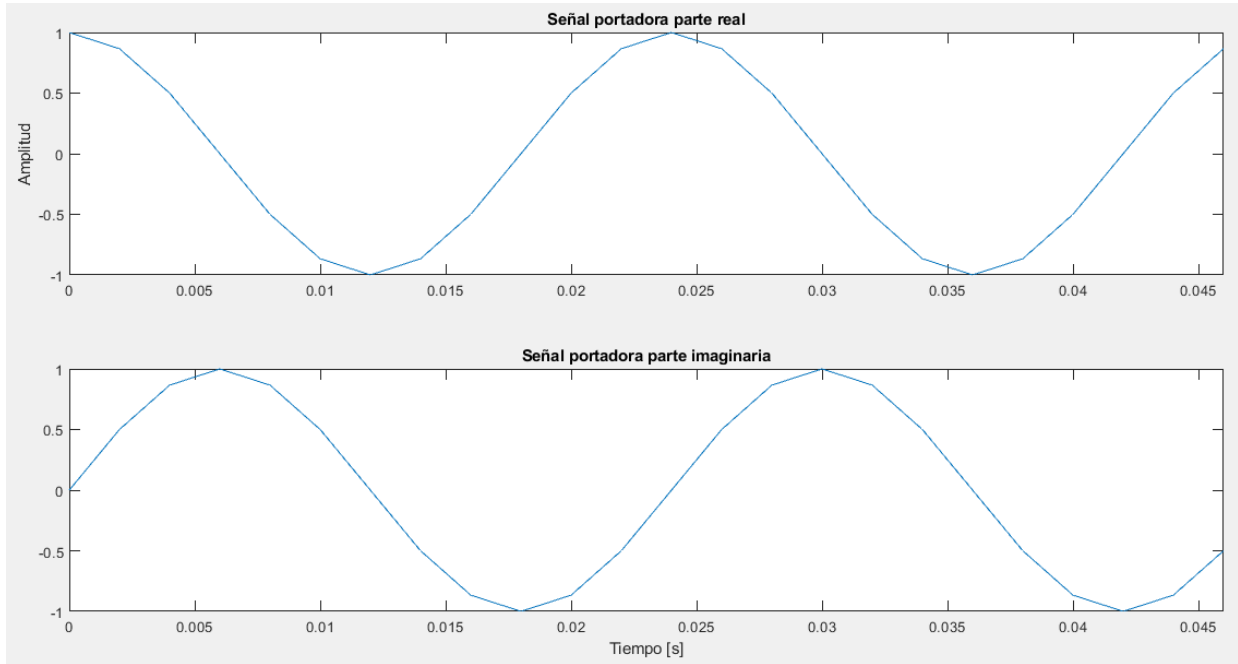


Fig. 164. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la portadora

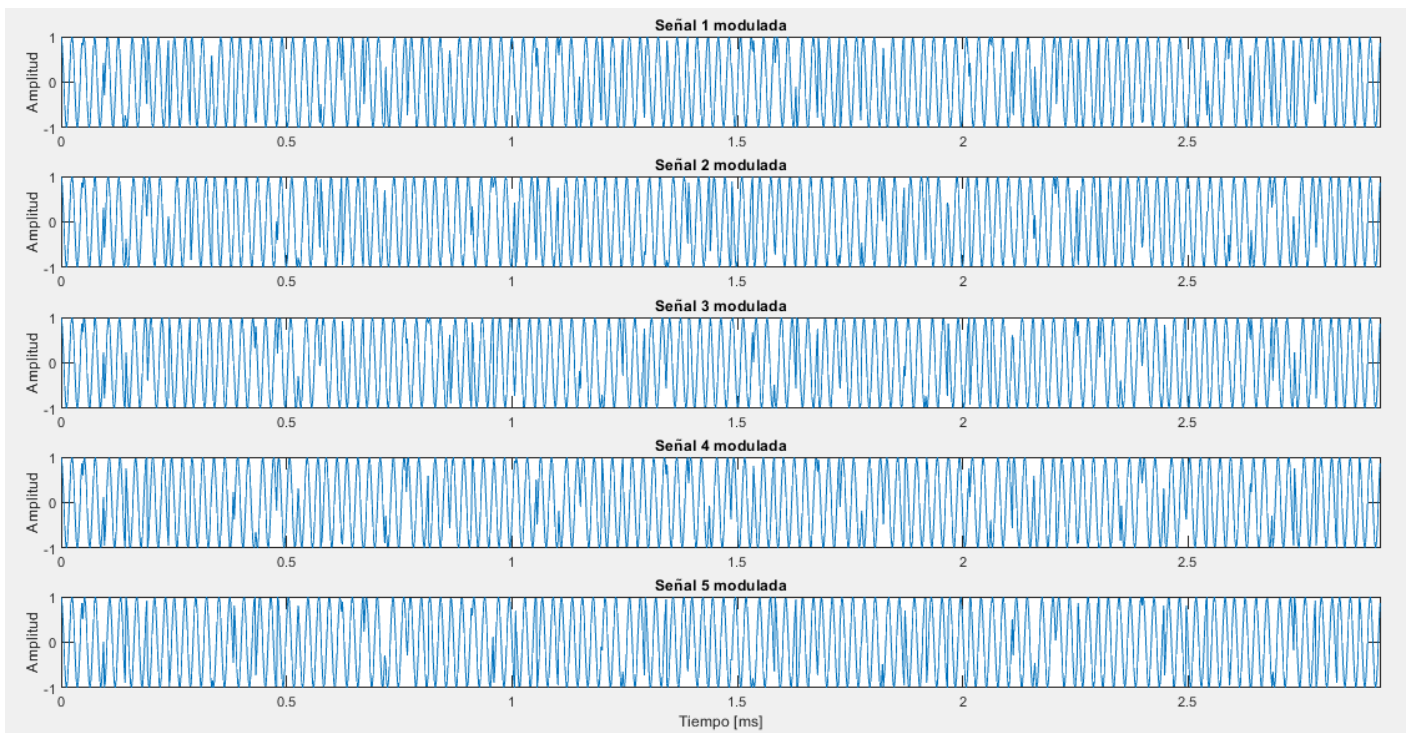


Fig. 165. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la señal modulada

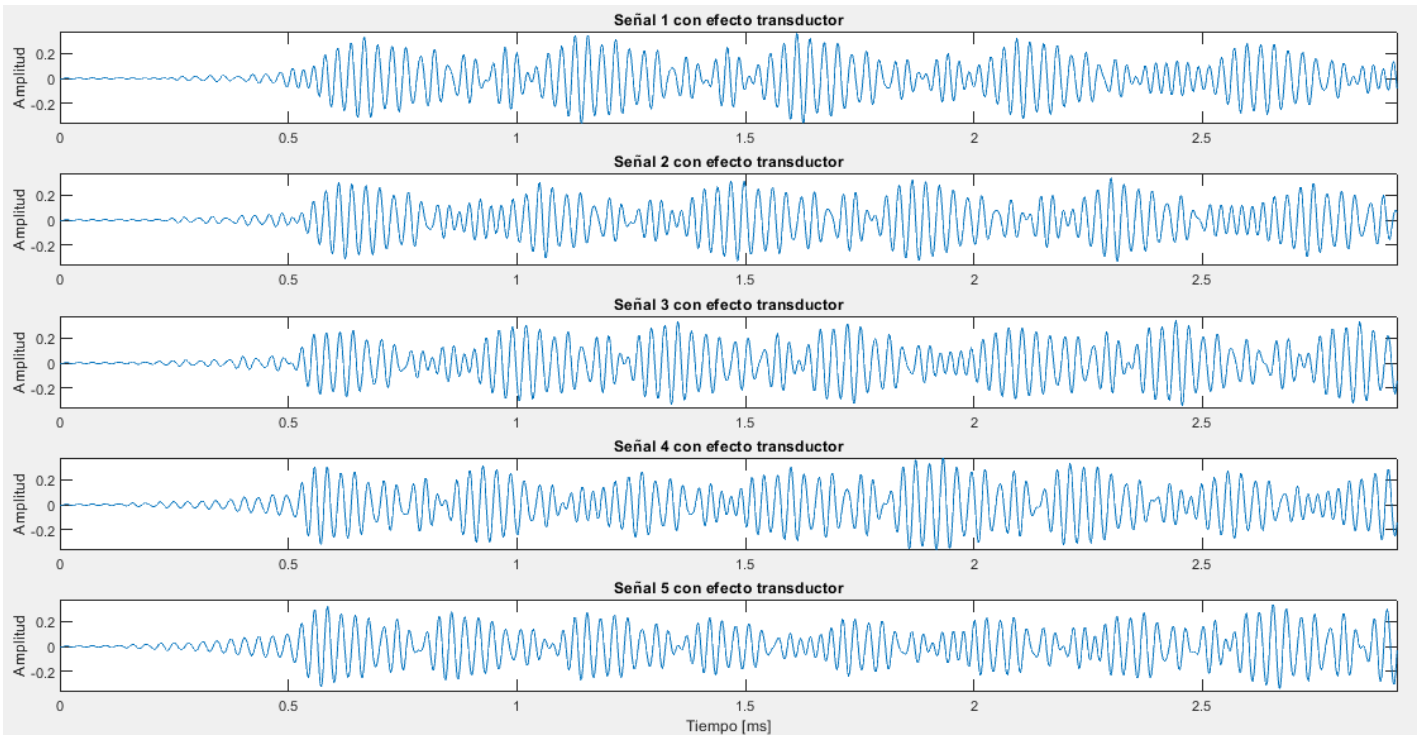


Fig. 166. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la señal con efecto del transductor

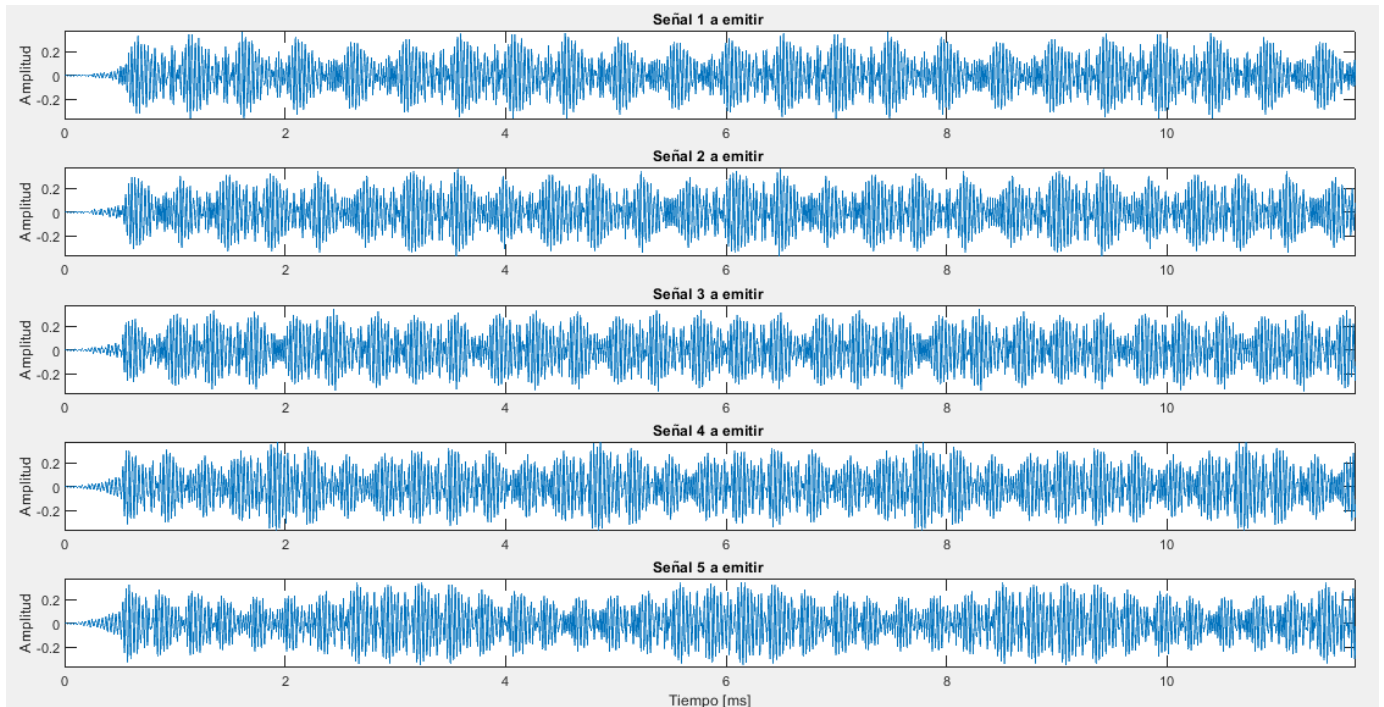
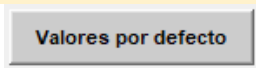
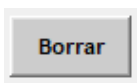





Fig. 167. Manual de Usuario: Configuración de emisión de secuencias Zadoff-Chu: Ejemplo: Visualización de la señal a emitir

8.5. Ventana de Configuración de efectos añadidos

Partiendo de que la señal está afectada por el efecto de la modulación y del transductor, se pueden añadir otros efectos mediante esta ventana. Existen dos opciones de configuración: aplicación de efectos a la señal a emitir (Capítulo 8.5.1) o simulación paramétrica de ruido en la señal a emitir (Capítulo 8.5.2).

Ambas opciones no pueden evaluarse a la vez, es decir, si por ejemplo se seleccionan los efectos cerca-lejos y multicamino, si después se selecciona la simulación paramétrica automáticamente se anulan y viceversa.

<i>Opciones generales</i>	Definición
	Introduce en los campos de configuración de los efectos seleccionados unos valores determinados para proporcionar una mejor orientación al usuario
	Elimina todo valor introducido en los campos de configuración de los efectos seleccionados
	Abre el capítulo correspondiente del Manual de Usuario para esta pantalla
	Retrocede a la ventana de Configuración de la emisión del código seleccionado
	Continúa a la ventana de Resultados

8.5.1. Efectos

Fig. 168. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables

La aplicación permite señalar los efectos que se desee aplicar sin la necesidad de que sean todos. Algo a tener en cuenta es que, los efectos que no vayan a ser aplicados deben quedar deseleccionados para que la aplicación evite extraer datos inexistentes. Una vez configurados, el programa aplica los efectos sobre la señal en este orden: cerca-lejos, multicamino, efecto Doppler, ruido blanco Gaussiano y finalmente ruido impulsivo. Esto hace que la señal que se utiliza para superponer el siguiente efecto no sea la ideal, si no la última resultante. Los efectos cerca-lejos y multicamino se aplican a cada uno de los *Nemo* códigos de la señal a emitir y el resto de efectos sobre la señal emitida la cual está compuesta por la suma de los códigos anteriores.

A continuación, se evalúan cada uno de los efectos:

El **Ruido Blanco Gaussiano** se configura a través de la relación señal-ruido SNR (*Signal-to-Noise Ratio*). Este parámetro se aplicará en toda la señal y causará mayor efecto cuanto más negativo sea el SMR (menor que cero).

Fig. 169. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Ruido Blanco Gaussiano

El **Ruido Impulsivo** aplica Ruido Blanco Gaussiano sólo en una zona de la señal. La duración que tendrá ese ruido se configurará en tanto por ciento respecto a la duración de la señal a emitir. Este efecto puede estar ubicado en varias zonas: principio, medio, final o aleatorio. Si es al principio comenzará en la primera muestra de la señal hasta la duración establecida, si es en el medio se repartirá de igual manera a ambos lados de la muestra situada a la mitad la duración establecida y si es al final se aplicará desde una muestra que resulte que la diferencia entre la última y ella sea la duración establecida. El modo aleatorio establece dos parámetros: la duración y la ubicación de la primera muestra. Esta duración se modifica cuando la muestra aleatoria está próxima al inicio o al final de la señal y no cabe.

Ruido impulsivo

SNR dB

Duración % respecto a la duracion de la emisión

Ubicación

Fig. 170. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Ruido Impulsivo: Ejemplo

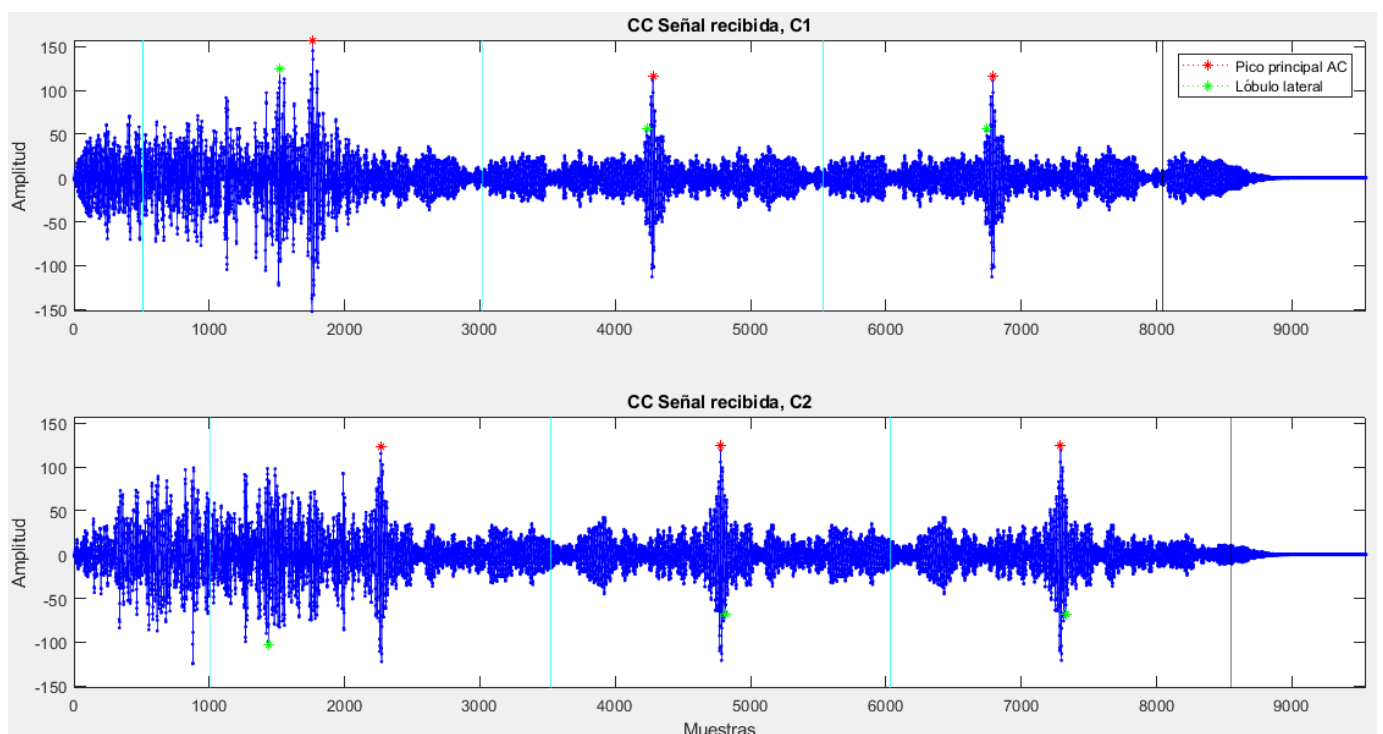


Fig. 171. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Ruido Impulsivo: Ejemplo (2)

Para aplicar el efecto **cerca-lejos** el programa recoge los valores mediante un vector de datos separados por un espacio y de longitud $Nemo$. Si la cantidad de valores introducidos son distintos a $Nemo$, saltará un aviso y se deberá volver a ajustar. El primer dato está referido a la ganancia del código C_1 y así sucesivamente hasta llegar a la ganancia del código C_{Nemo} .

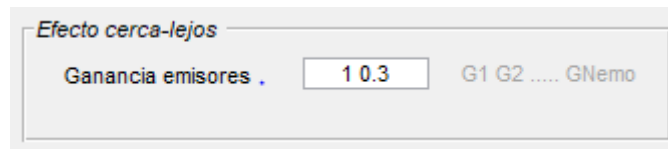


Fig. 172. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Efecto cerca-lejos: Ejemplo

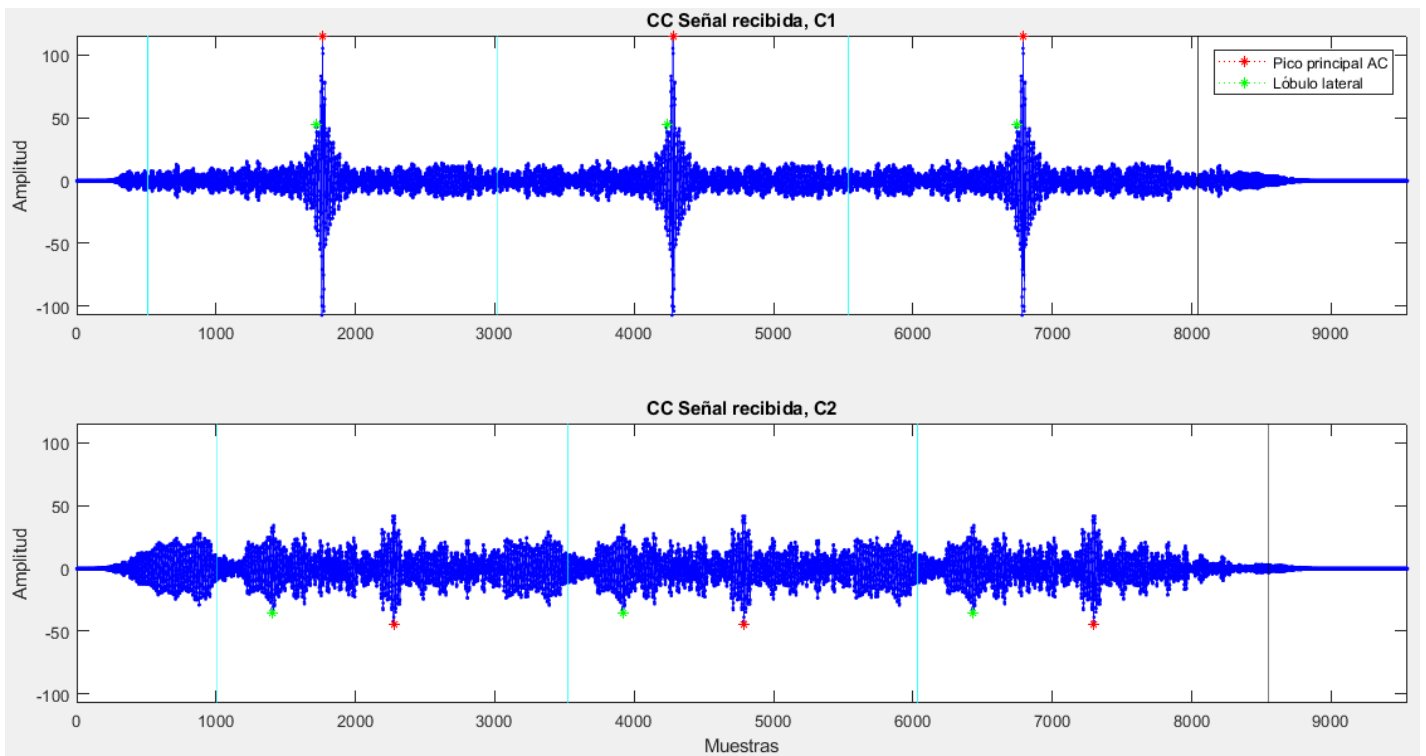


Fig. 173. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Efecto cerca-lejos: Ejemplo (2)

El **multicamino** tiene dos opciones de configuración: manual o aleatoria. Como orientación, se muestra por pantalla el tiempo que dura un código T_e . En el modo manual, al igual que el anterior, el programa recoge los datos mediante dos vectores de longitud $Nemo$, uno para los retardos y otro para las ganancias, que contienen la información para cada código separados por un espacio. Para simplificar, en este modo la señal solo va a rebotar una vez, es decir, solo va a tener un multicamino. El modo aleatorio está configurado para que cada código tenga dos multicaminos y que los retardos tengan un valor comprendido entre 40 y 1500. En este caso, cada valor de retardo tiene asignada una ganancia determinada a través de una función exponencial de la forma $x = 0.8e^{-t}$ representada en la Fig. 174 donde t es un vector que toma valores entre 0 y 1500 con un paso de 1 muestra.

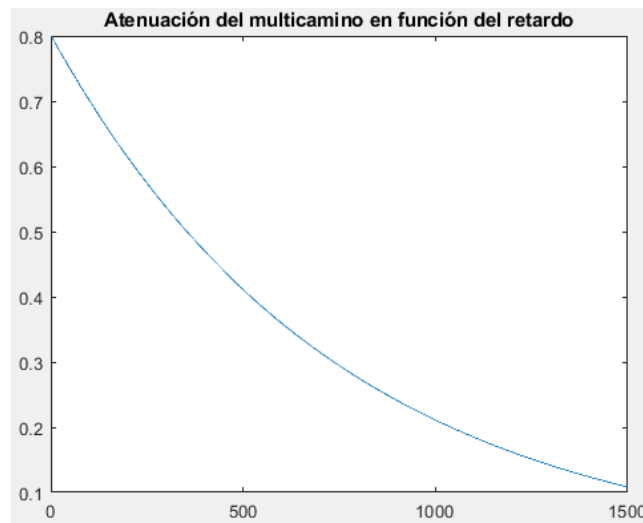
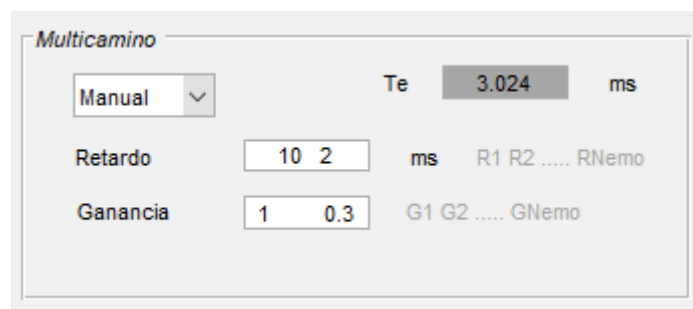


Fig. 174. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Función cálculo multicamino modo aleatorio



Multicamino
 Manual
 Te ms
 Retardo ms R1 R2 RNemo
 Ganancia G1 G2 GNemo

Fig. 175. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Multicamino: Ejemplo

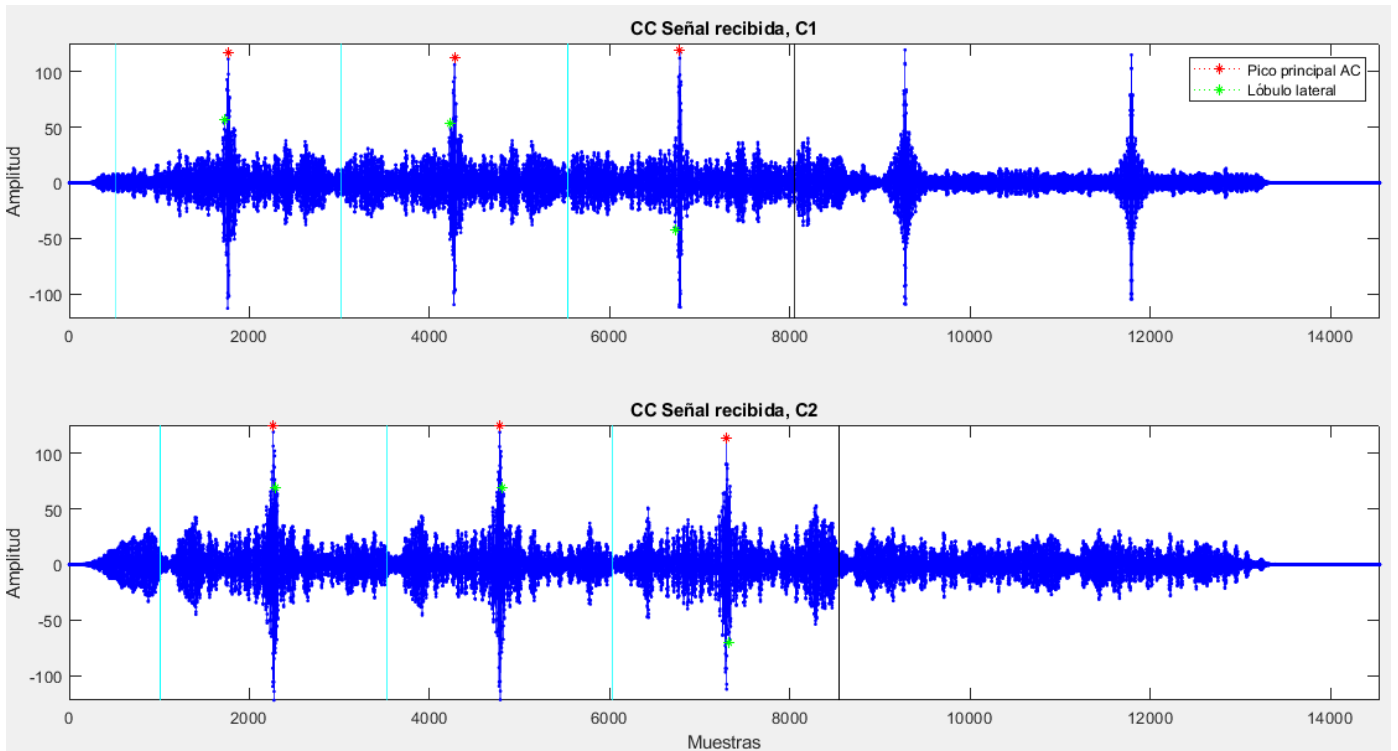


Fig. 176. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Multicamino: Ejemplo (2)

El **efecto Doppler** tiene dos parámetros de configuración: la velocidad del sonido y la velocidad relativa entre emisor-receptor. En la primera, se puede elegir entre velocidad del sonido en el aire o en el agua y se mostrará por pantalla el valor que se va a aplicar según el caso. En el segundo parámetro, si se introduce un valor positivo quiere decir que emisor y receptor se están alejando y si es un valor negativo es que se están acercando. El objetivo de este efecto es resamplear la señal con una nueva frecuencia de muestreo obtenida como $fse' = fse \cdot (1 + v_{relativa}/v_{sonido})$.

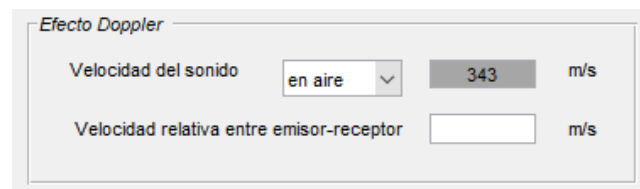


Fig. 177. Manual de Usuario – Ventana de Configuración de efectos añadidos: Efectos configurables: Efecto Doppler

8.5.2. Simulación paramétrica del ruido

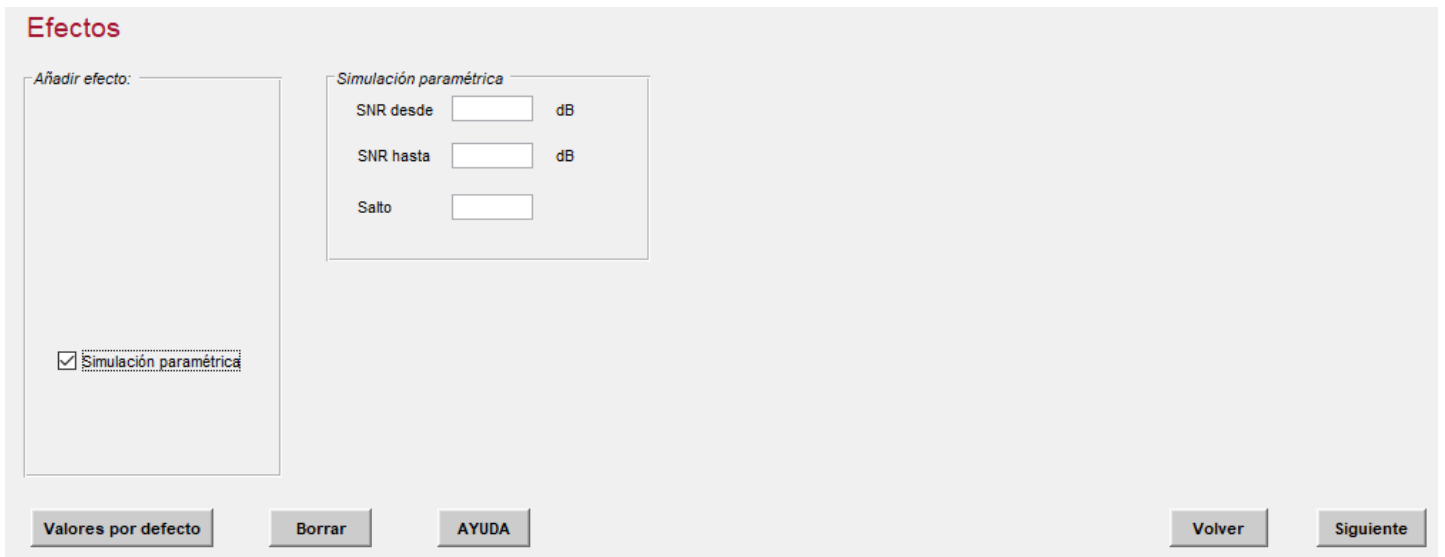


Fig. 178. Manual de Usuario: Ventana de Configuración de efectos añadidos: Simulación paramétrica

Esta opción hace referencia a la **simulación paramétrica del ruido blanco gaussiano** en la señal emitida. En la siguiente pantalla de la aplicación (Capítulo 8.6) se podrá observar el RMSE de cada código que existe entre el pico principal de la señal emitida idealmente y la señal emitida real, en función del rango de ruido establecido en esta ventana (SNR_desde:salto:SNR_hasta). El RMSE debe tender a cero a medida que el ruido aumenta en valor.

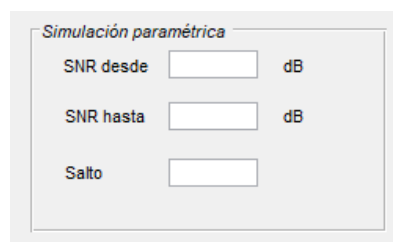


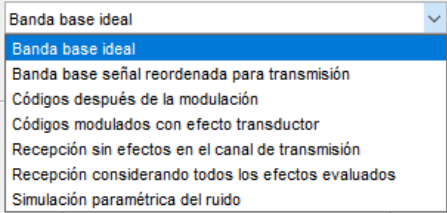
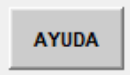
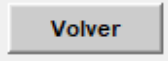
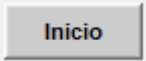
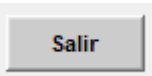
Fig. 179. Manual de Usuario – Ventana de Configuración de efectos añadidos: Simulación paramétrica de ruido

8.6. Ventana de Visualización de resultados

8.6.1. Introducción

En esta pantalla se analizan los resultados obtenidos en los distintos procesos de configuración de la señal.

Se explica detalladamente la ventana para el caso de las secuencias Kasami y se limita a indicar las peculiaridades del resto de códigos frente a esa ella.

Opciones generales	Definición
	<p>Selección de resultados para los distintos procesos que lleva a cabo la señal</p>
	<p>Abre el capítulo correspondiente del Manual de Usuario para esta pantalla</p>
	<p>Retrocede a la ventana de Configuración de la emisión del código seleccionado</p>
	<p>Retrocede a la ventana de Inicio del programa</p>
	<p>Finaliza el programa y continúa a una última ventana, <i>Locate-Us</i>, que ofrece la posibilidad de enviar códigos modulados, generados desde esta aplicación, para realizar pruebas experimentales</p>

8.6.2. Resultados secuencias Kasami

Dentro de las 6 posibles opciones que se pueden seleccionar para la obtención de resultados, se pueden agrupar en tres ventanas diferentes en función de los resultados que se ofrecen.

De esta forma, se dan los 3 siguientes casos:

Elección de resultado	Caso
Banda base (sin modular)	1
Códigos después de la modulación	
Códigos modulados con efecto transductor	
Recepción sin efectos en el canal de transmisión	2
Recepción considerando todos los efectos evaluados	
Simulación paramétrica del ruido	3

Caso 1. Banda base sin modular

Nomenclatura	Definición	Observaciones
AC	Función de autocorrelación	Véase (32) para el caso periódico y (33) para el caso aperiódico cuando $m=s$
CC	Función de correlación cruzada	Véase (32) para el caso periódico y (33) para el caso aperiódico cuando $m \neq s$
FFT	Función transformada de Fourier	
AC C_i	AC del código C_i	$1 \leq i \leq Nemo$
CC C_i, C_j	CC del código C_i con el código C_j	$1 \leq i \leq Nemo-1, 2 \leq j \leq Nemo$
FFT C_i	FFT del código C_i	$1 \leq i \leq Nemo$
Cota AC	Máximo valor de las cotas AC de los códigos. Pico máximo AC entre lóbulo lateral AC	El pico principal debe estar a L muestras. El lóbulo lateral se obtiene anulando el pico principal y volviendo a calcular el valor máximo de la señal. En modulación y en modulación con transductor, se anula también una zona $N_c \cdot Of$ alrededor del pico principal para evitar los efectos que se producen
Cota CC	Máximo valor de las cotas CC de los códigos. Pico máximo AC entre lóbulo lateral CC	El lóbulo lateral se obtiene calculando el valor máximo de la señal
Cota total	Valor máximo entre cota AC y cota CC	

Función de Autocorrelación y Correlación Cruzada

(32)

$$R_{C_m, C_s}[\tau] = \sum_{l=0}^{L-1} C_m[l] C_s[l + \tau]$$

$$X_{C_m, C_s}[\tau] = \begin{cases} \sum_{l=0}^{L-1-\tau} C_m[l]C_s[l+\tau], & 0 \leq \tau \leq L-1 \\ \sum_{l=0}^{L-1+\tau} C_m[l-\tau]C_s[l], & 1-L \leq \tau \leq 0 \\ 0, & |\tau| \geq L \end{cases} \quad (33)$$

Opciones	Definición
<input checked="" type="radio"/> Aperiódico	Cálculo de las correlaciones en modo aperiódico.
<input type="radio"/> Periódico	Cálculo de las correlaciones en modo periódico.
<input type="text" value="C1"/>	Selección de la AC o de la FFT código C_i que se desee visualizar en pantalla ($1 \leq i \leq Nemo-1$)
<input type="text" value="C1,C2"/>	Selección de la CC de los códigos C_i, C_j que se quiera visualizar en pantalla ($1 \leq i \leq Nemo-1, 2 \leq j \leq Nemo$)
<input type="button" value="Visualizar autocorrelaciones"/>	Representación conjunta de las AC de la señal. Aparece marcado con un asterisco rojo el pico principal AC y con un asterisco verde el lóbulo lateral AC
<input type="button" value="Visualizar correlaciones cruzadas"/>	Representación conjunta de las CC de la señal. Aparece marcado con un asterisco verde el lóbulo lateral CC
<input type="button" value="Visualizar espectro códigos"/>	Representación conjunta de las FFT de la señal


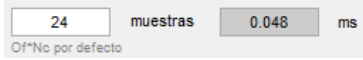
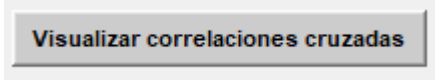
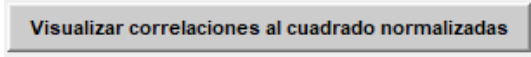

Caso 2. Señal recibida

Nomenclatura	Definición	Observaciones
Señal recibida	Equivale a la suma de todos los códigos generados para la emisión	Cada código se ha repetido $Nrep$ veces
CC	Función de correlación cruzada	Véase (34) para $m \neq s$
FFT	Función transformada de Fourier de la señal recibida	
CC Señal recibida, C_i	CC entre la señal recibida y los códigos C_i generados	$1 \leq i \leq Nemo$
SMR	<i>Sidelobe to Mainlobe Ratio</i> . Relación pico máximo CC entre lóbulo lateral CC de cada código	Se van a dar $Nrep$ picos de correlación. Para el cálculo, se divide la señal CC en $Nrep$ ventanas y se calcula cada SMR. El lóbulo lateral se obtiene anulando el pico principal y una zona $Nc \cdot Of$ alrededor, y se vuelve a calcular el valor máximo de la señal. El resultado es la media de todos ellos con una desviación típica
SMR total	Máximo de entre los SMR de cada código	

Función correlación cruzada

(34)

$$X_{r,Cs}[\tau] = \begin{cases} \sum_{l=0}^{L-1-\tau} r[l]C_s[l+\tau], & 0 \leq \tau \leq L-1 \\ \sum_{l=0}^{L-1+\tau} r[l-\tau]C_s[l], & 1-L \leq \tau \leq 0 \\ 0, & |\tau| \geq L \end{cases}$$

Opciones	Definición
	<p>Selección de la CC o del SMR de código C_i que se desee visualizar en pantalla ($1 \leq i \leq Nemo-1$)</p>
	<p>Modificación del valor de exclusión alrededor del pico principal de cada ventana para el cálculo del SMR. Por defecto es $Nc \cdot Of$</p>
	<p>Representación conjunta de la CC de la señal recibida con cada código C_i ($1 \leq i \leq Nemo-1$). Aparecen indicadas las $Nrep$ ventanas y marcado con un asterisco rojo el pico principal CC y con un asterisco verde el lóbulo lateral CC</p>
	<p>Representación conjunta de las CC de la señal recibida con cada código C_i ($1 \leq i \leq Nemo-1$) divididas entre su pico principal CC y elevadas el cuadrado. Aparecen indicadas las $Nrep$ ventanas y marcado con un asterisco rojo el pico principal CC y con un asterisco verde el lóbulo lateral CC</p>
	<p>Representación conjunta de las CC de la señal recibida con cada código C_i ($1 \leq i \leq Nemo-1$) divididas entre su pico principal CC y junto con su envolvente (Transformada de Hilbert)</p>

Caso 3. Simulación paramétrica

Representa el RMSE para cada código C_i ($1 \leq i \leq Nemo$) en función de los valores configurados de la simulación paramétrica del ruido en la ventana de Configuración efectos añadidos. Este parámetro tiende a cero cuanto mayor es el ruido.

Ejemplo

Configuración de emisión: Kasami

Entradas

Longitud (L) Tamaño familia (M)

Nº emisores simultáneos (Nemo) Eficiencia energética (η)

Modulación BPSK:

Tipo portadora

Frecuencia portadora (fc) kHz

Frecuencia de muestreo (fse) kHz

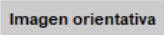
Nº ciclos portadora (Nc)

Factor sobremuestreo (Of)

Duración emisión un código (Te) ms

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms 

Número de repeticiones (Nrep)

Modelo transductor

Salidas

Fig. 180. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Configuración de la emisión

Efectos

Añadir efecto:

- Ruido blanco gaussiano
- Ruido impulsivo
- Efecto cerca-lejos
- Multicamino
- Efecto Doppler
- Simulación paramétrica

Ruido impulsivo

SNR dB

Duración % respecto a la duracion de la emisión

Ubicación

Fig. 181. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Configuración de efectos añadidos

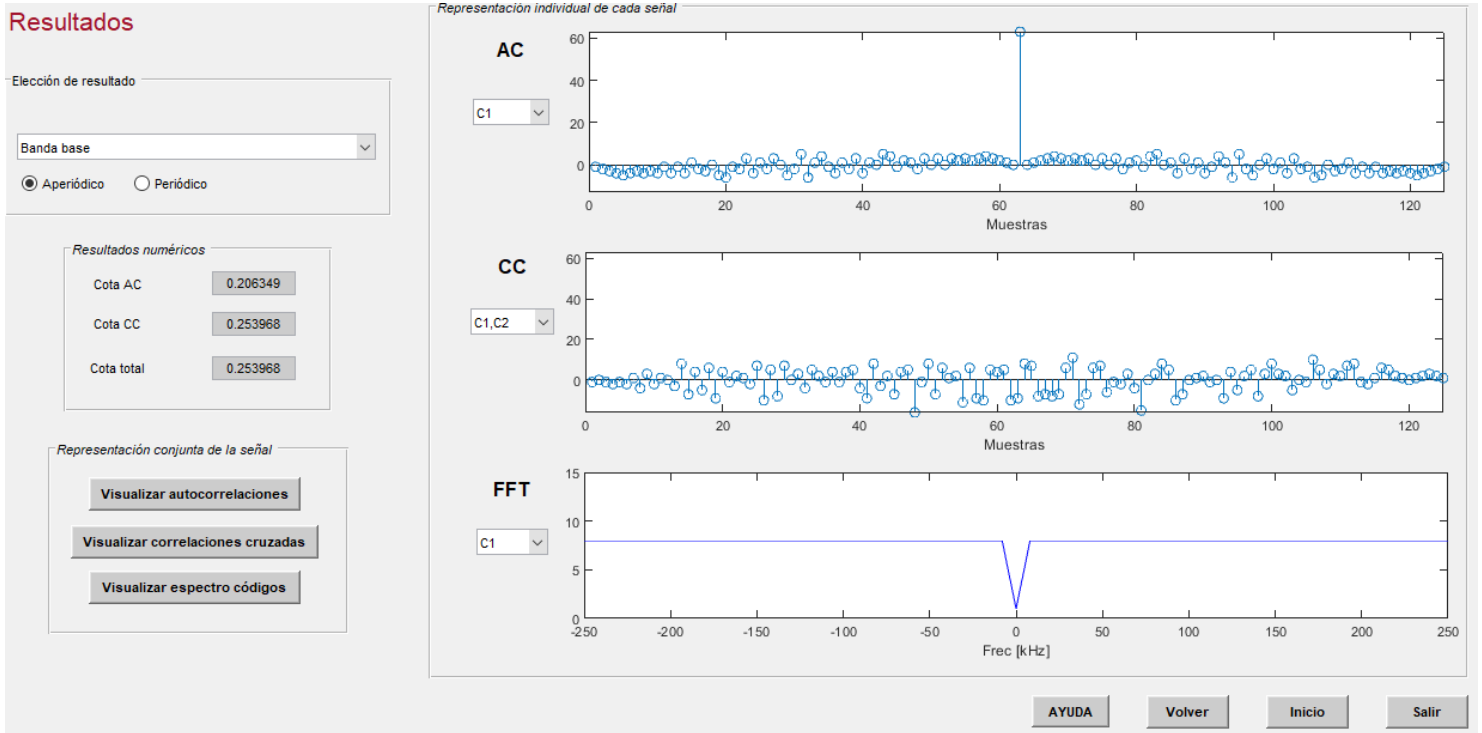


Fig. 182. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: banda base aperiódica

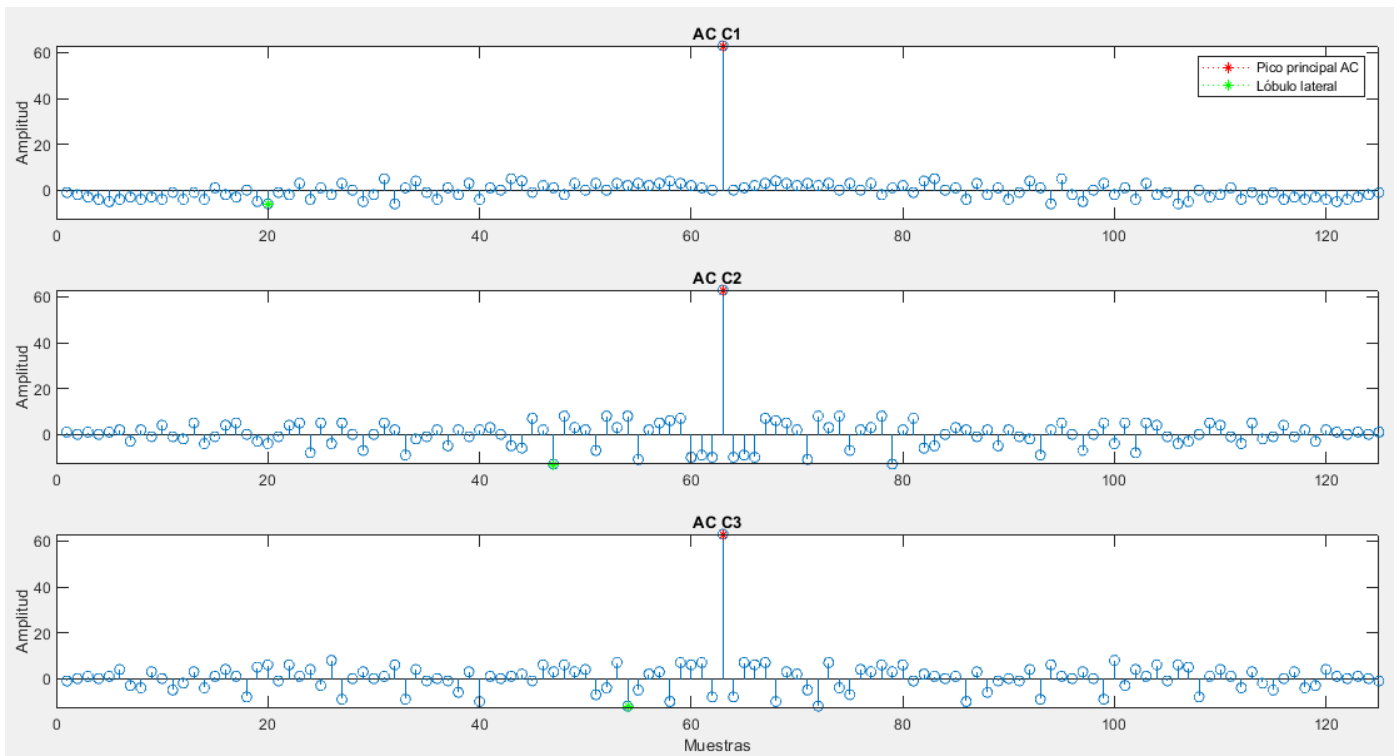


Fig. 183. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC banda base aperiódica

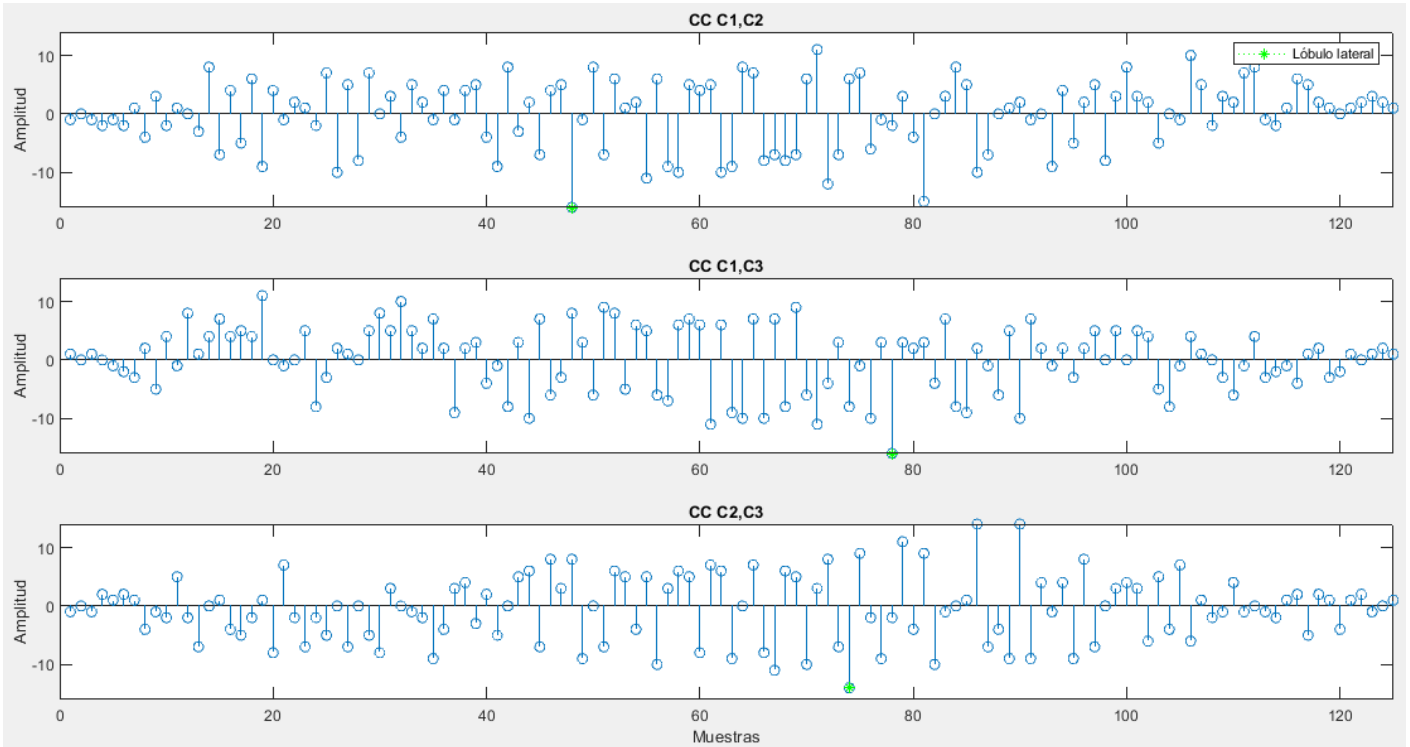


Fig. 184. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC banda base aperiódica

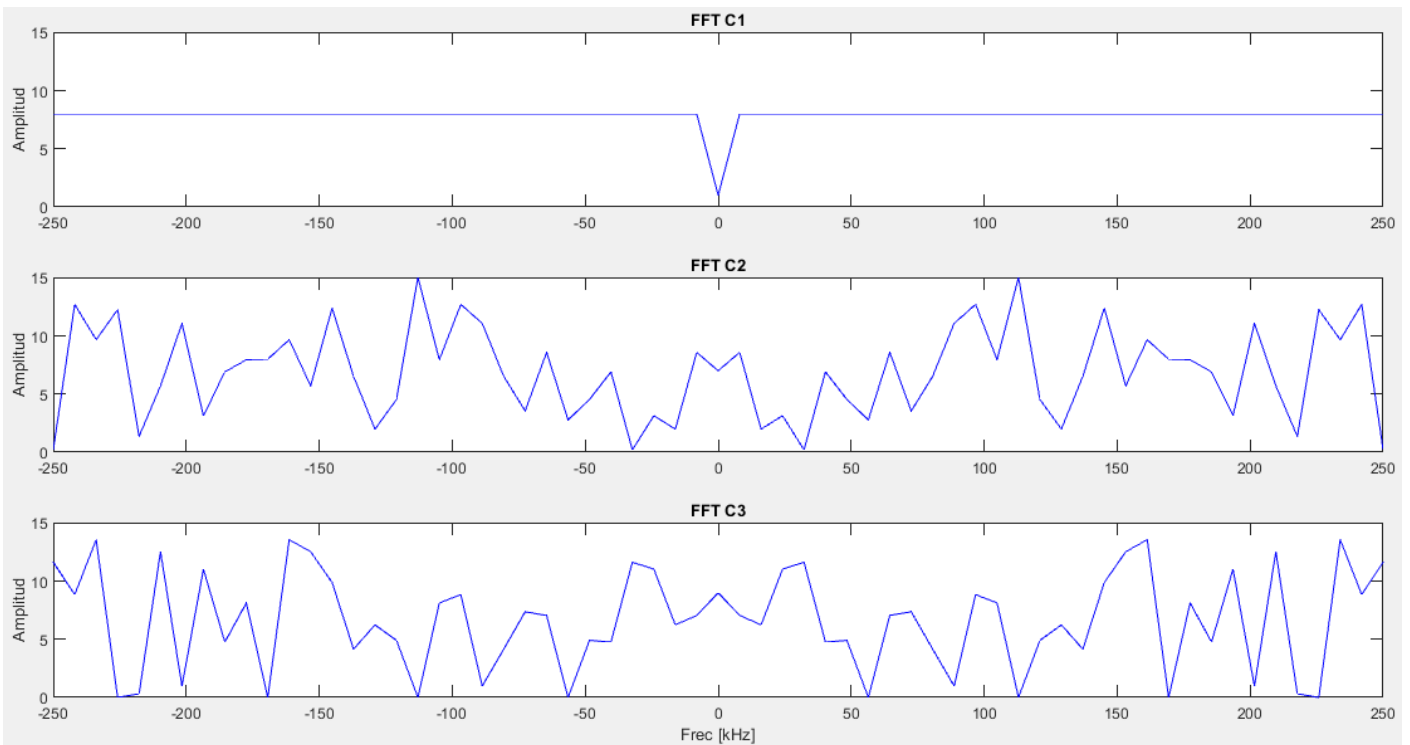


Fig. 185. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: FFT banda base

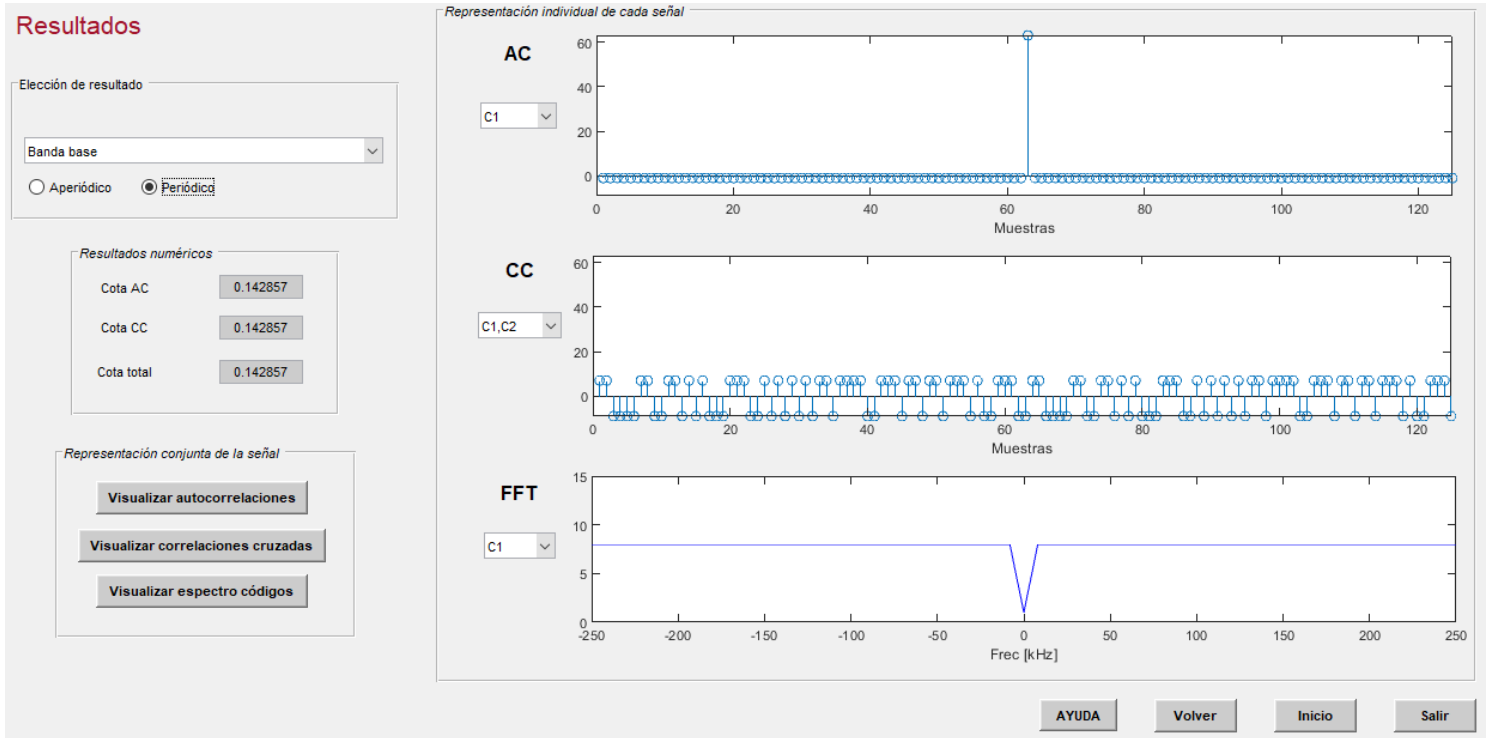


Fig. 186. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: banda base periódica

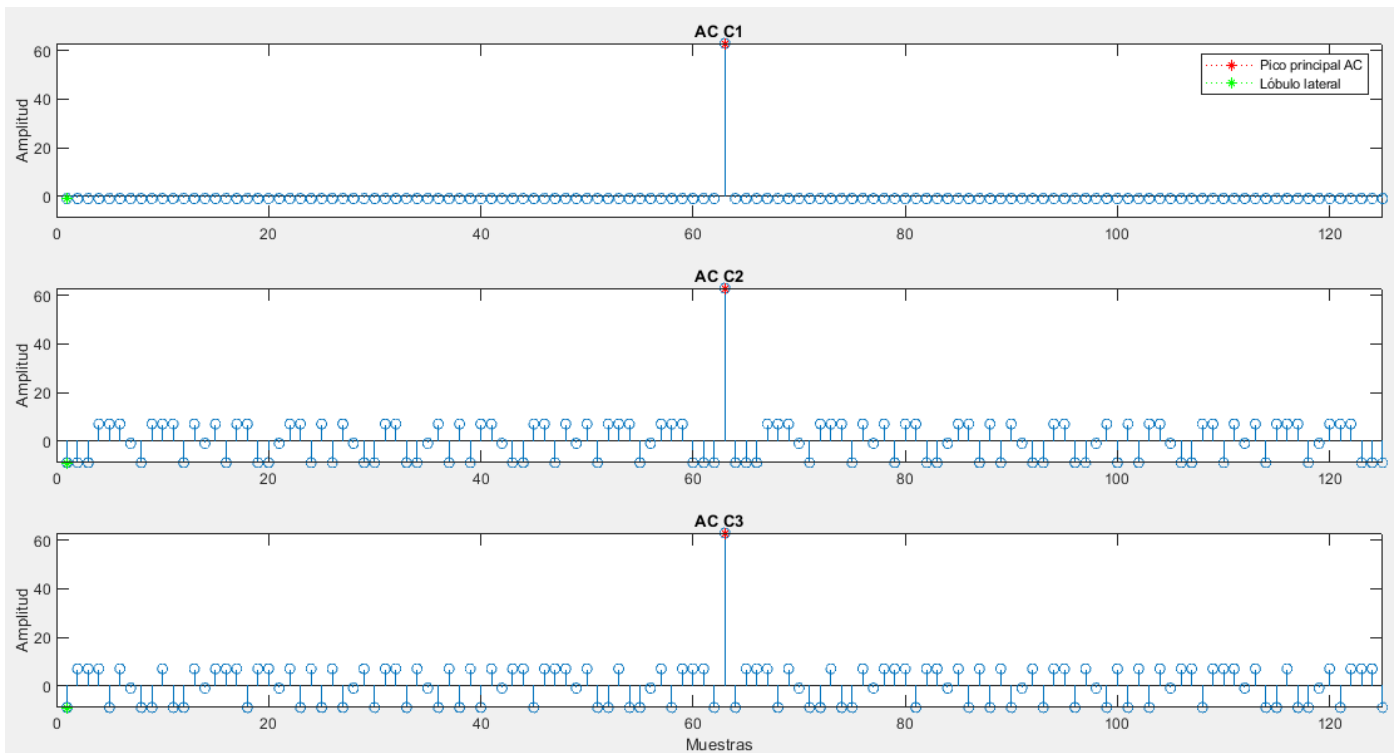


Fig. 187. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC banda base periódica

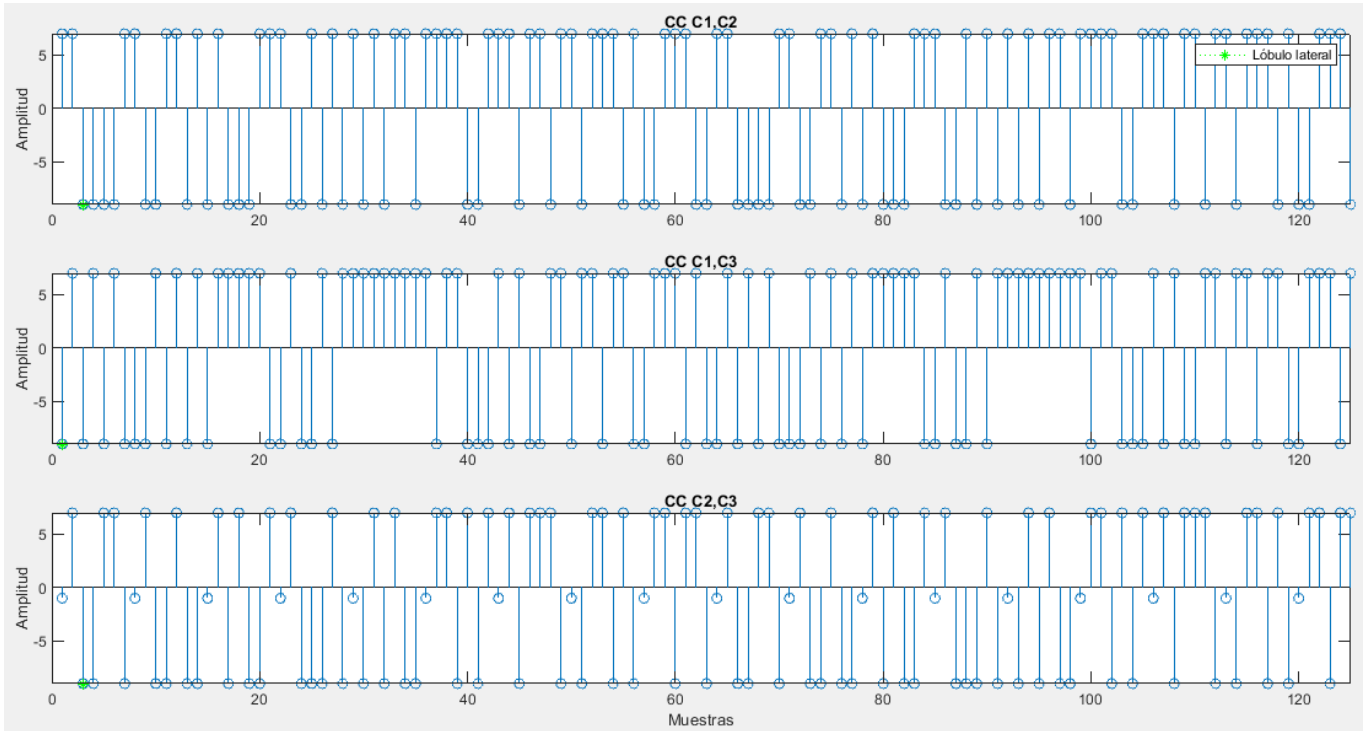


Fig. 188. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC banda base periódica

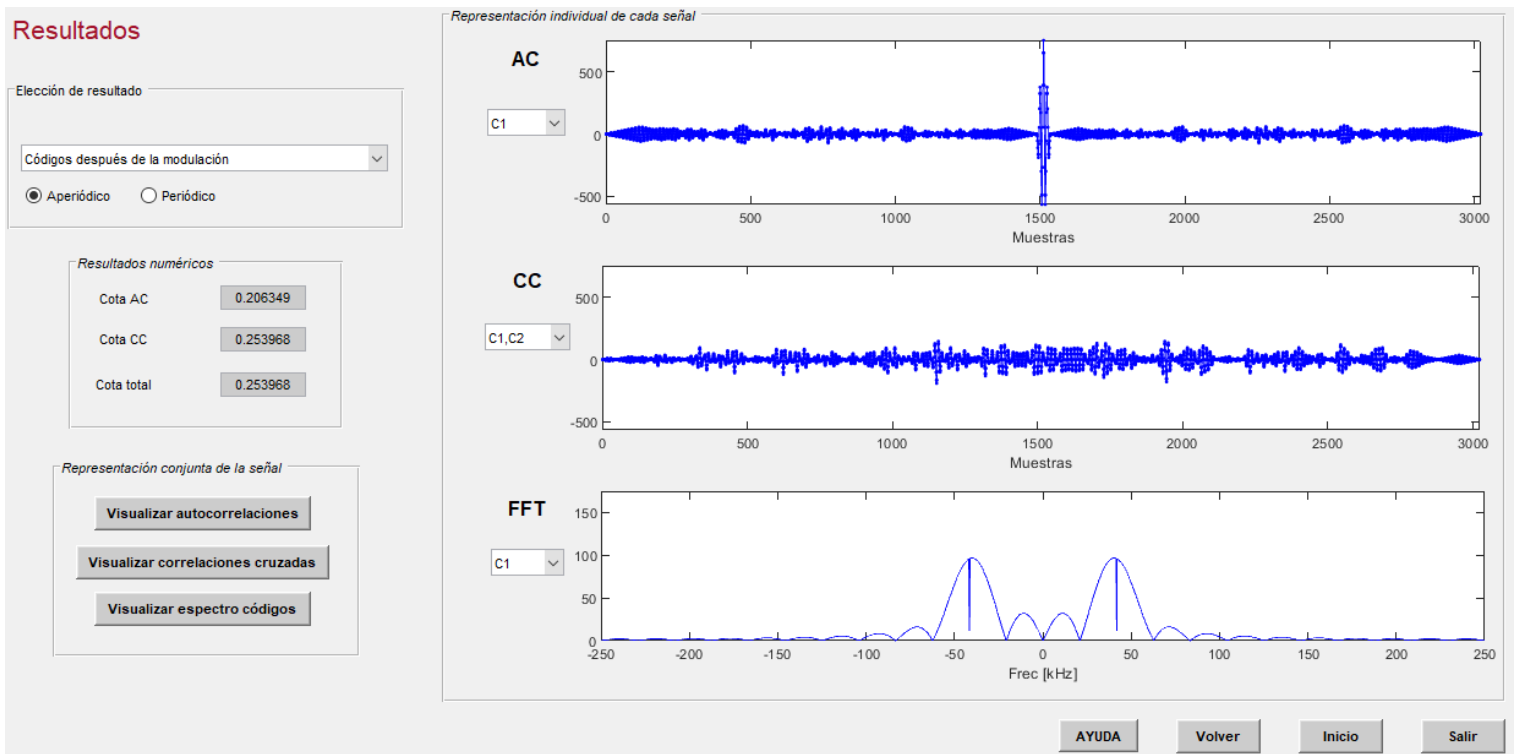


Fig. 189. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada aperiódica

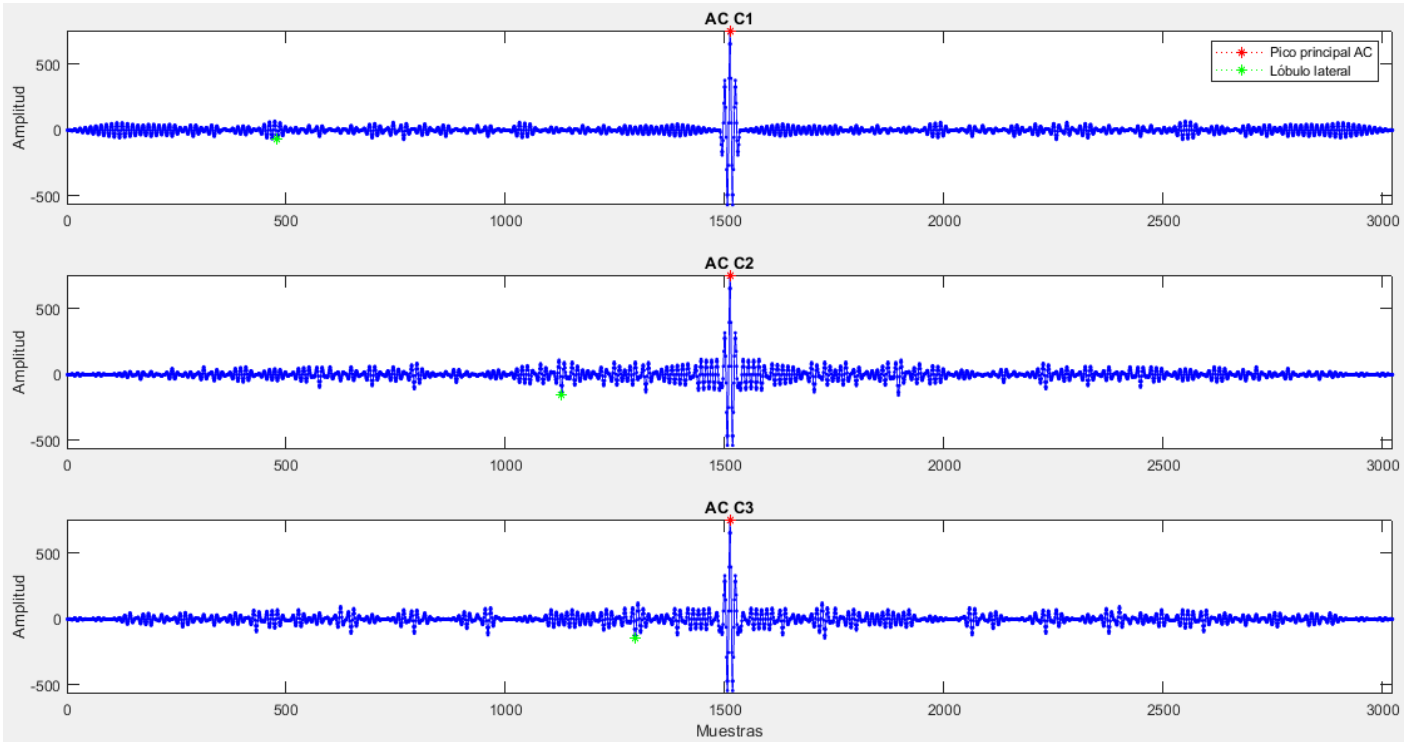


Fig. 190. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada aperiódica

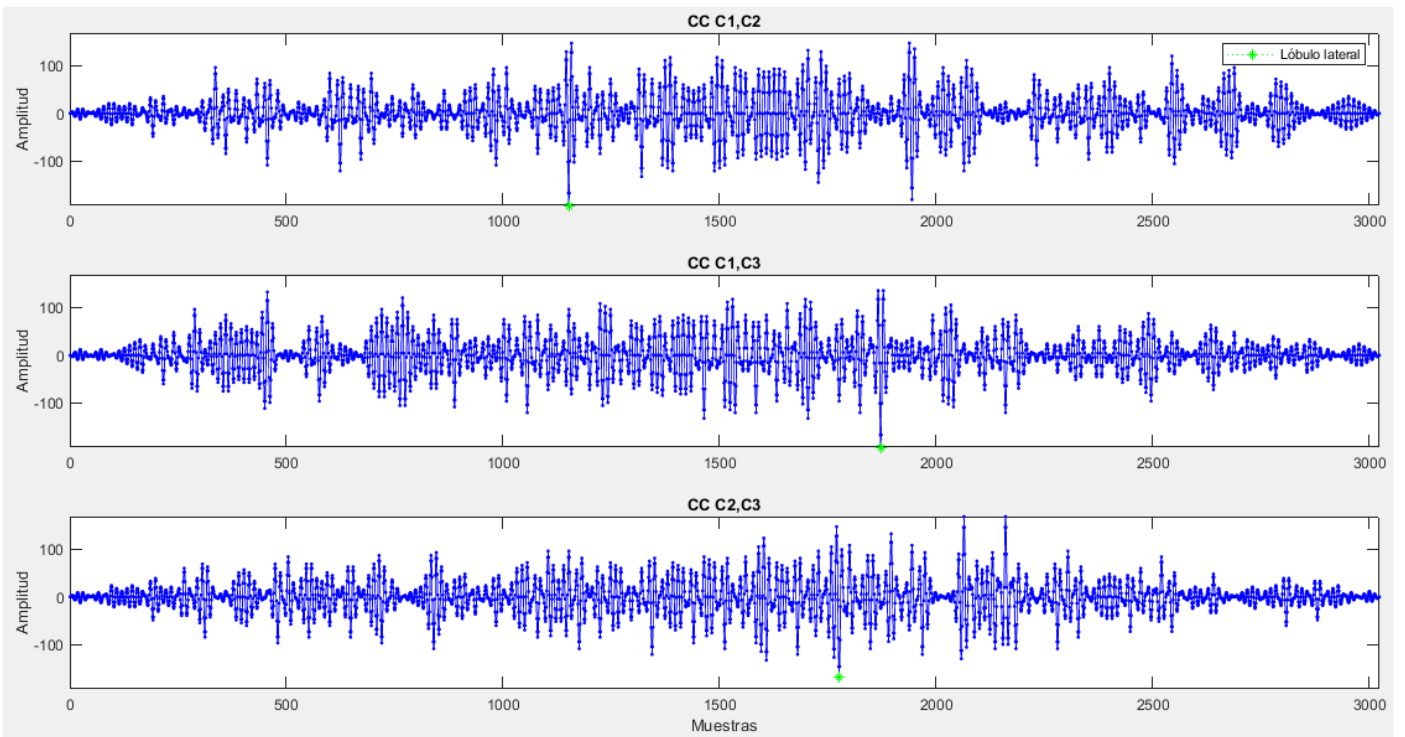


Fig. 191. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada aperiódica

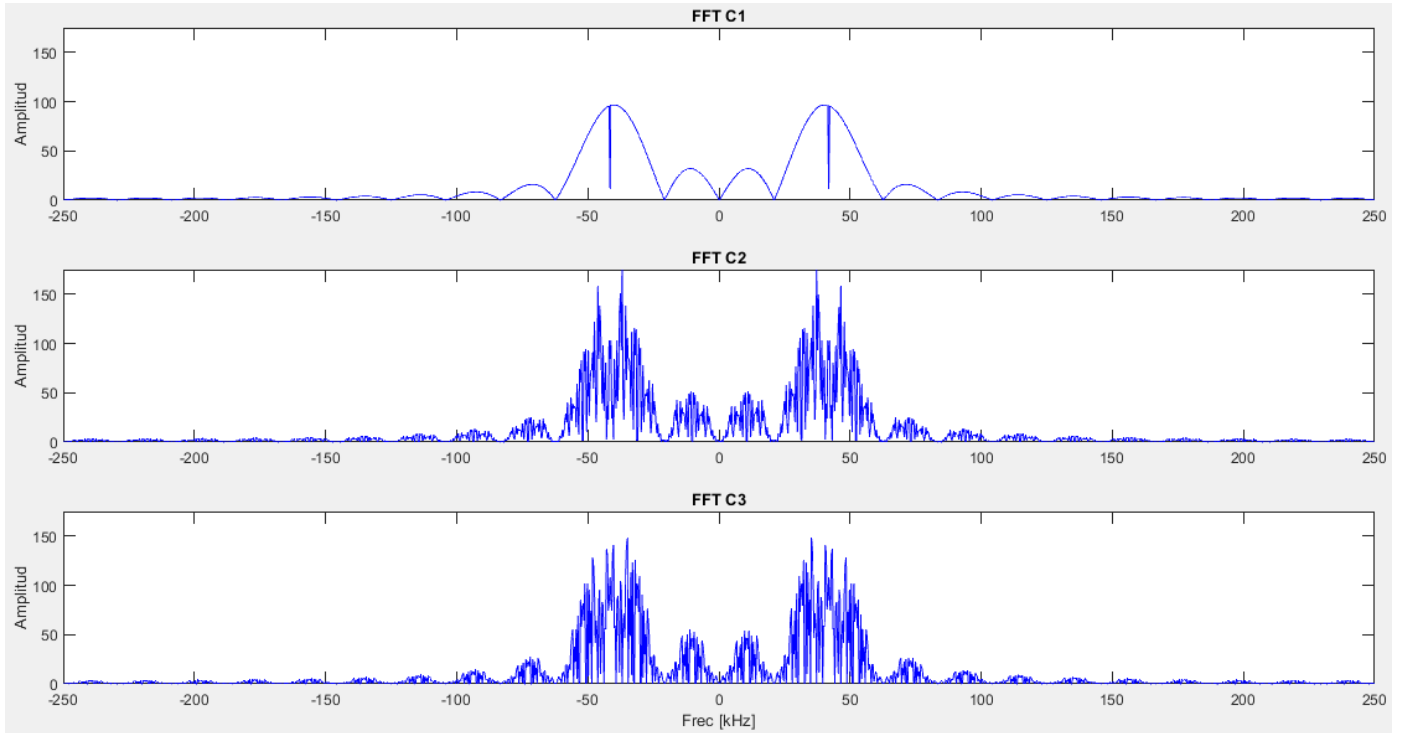


Fig. 192. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: FFT modulada

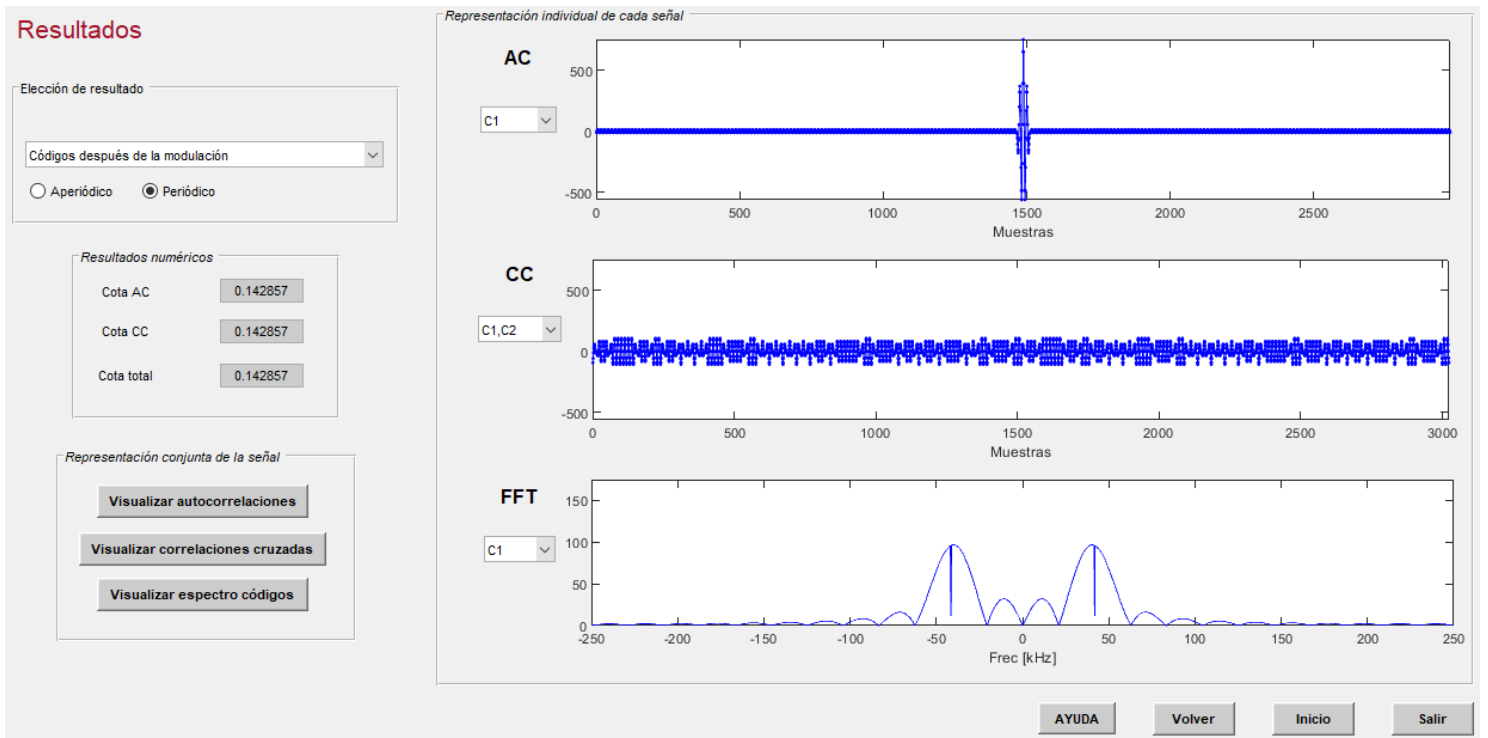


Fig. 193. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada periódica

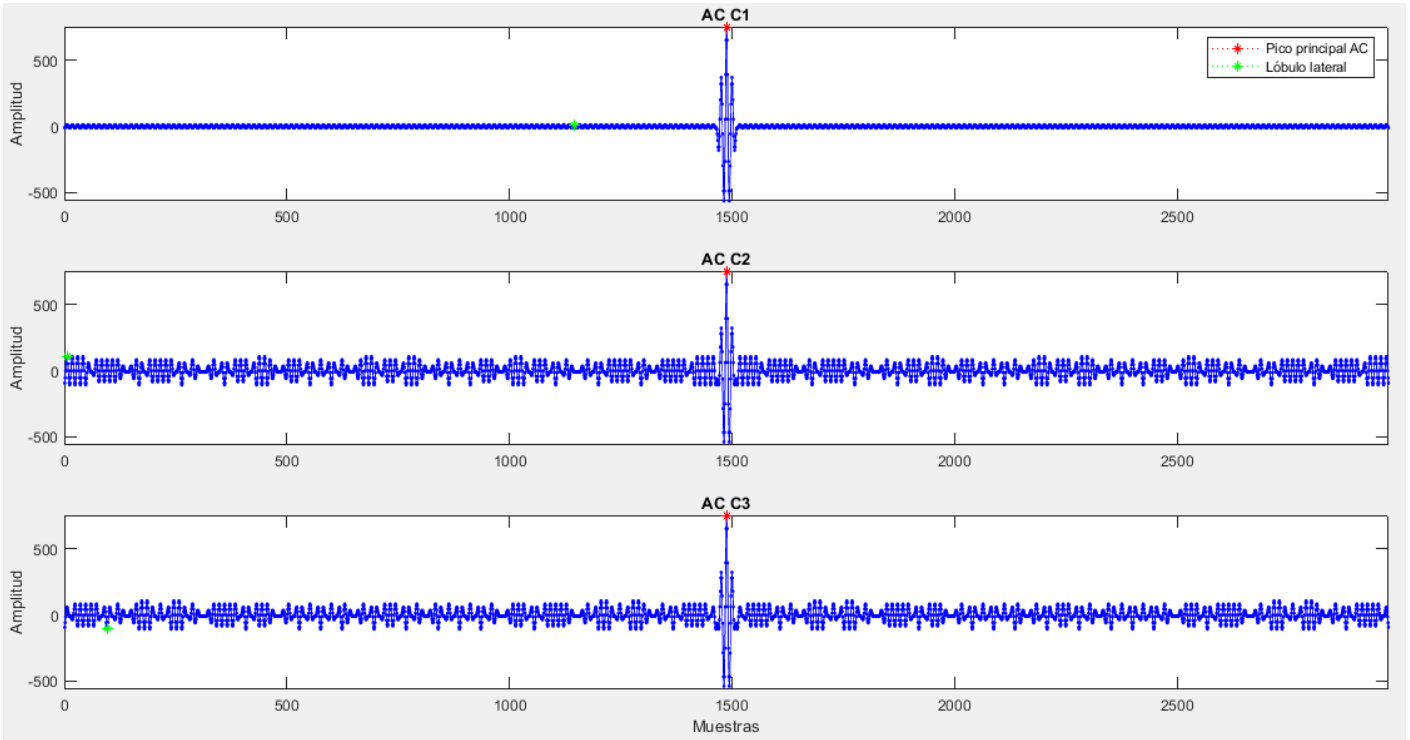


Fig. 194. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada periódica

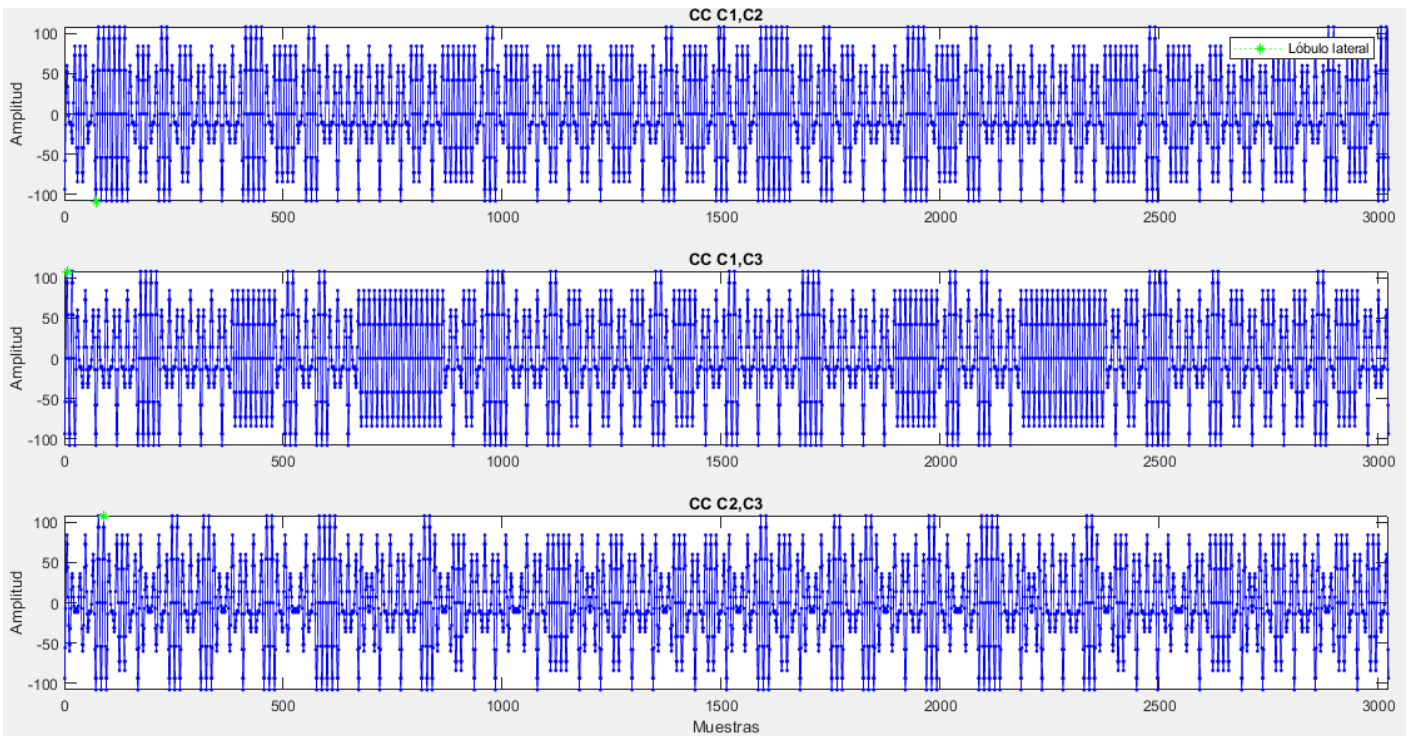


Fig. 195. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada periódica

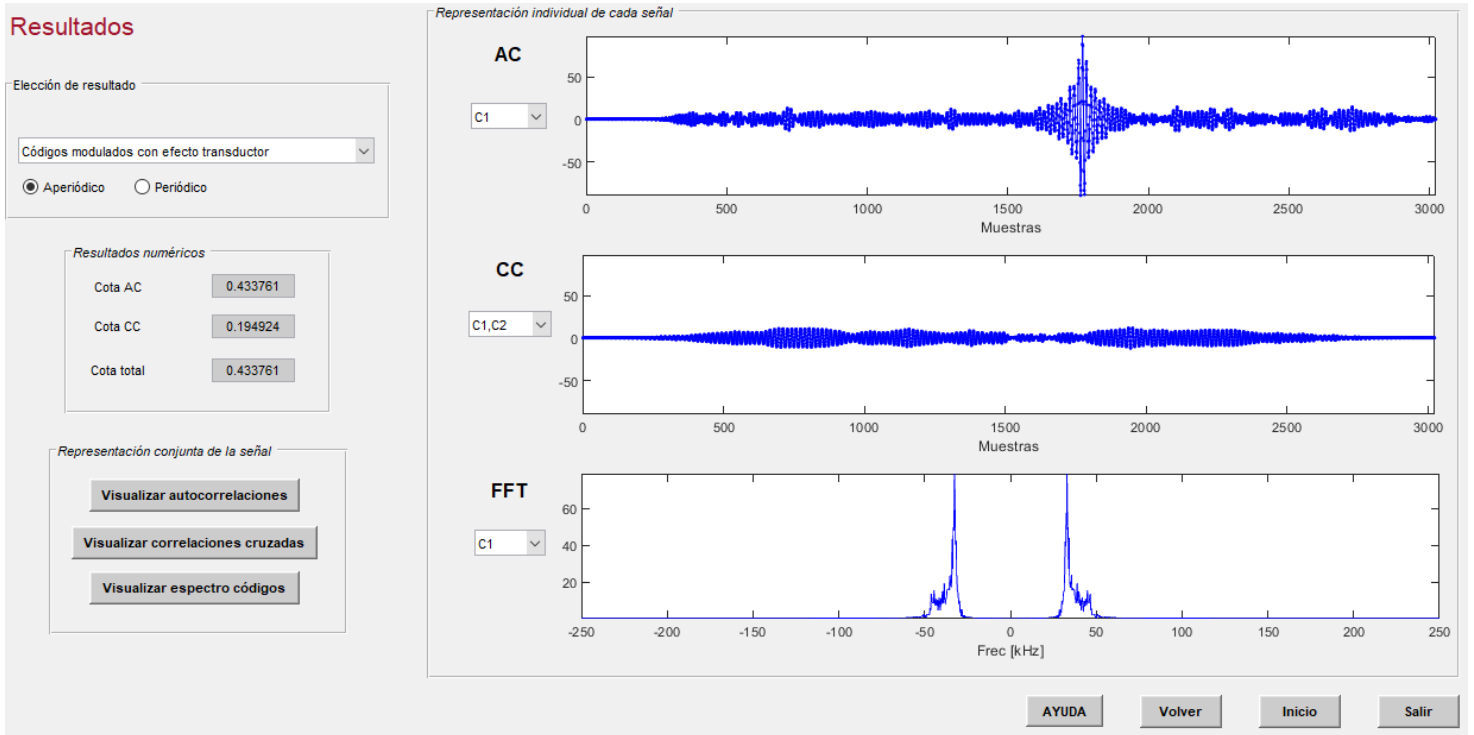


Fig. 196. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada con efecto transductor aperiódica

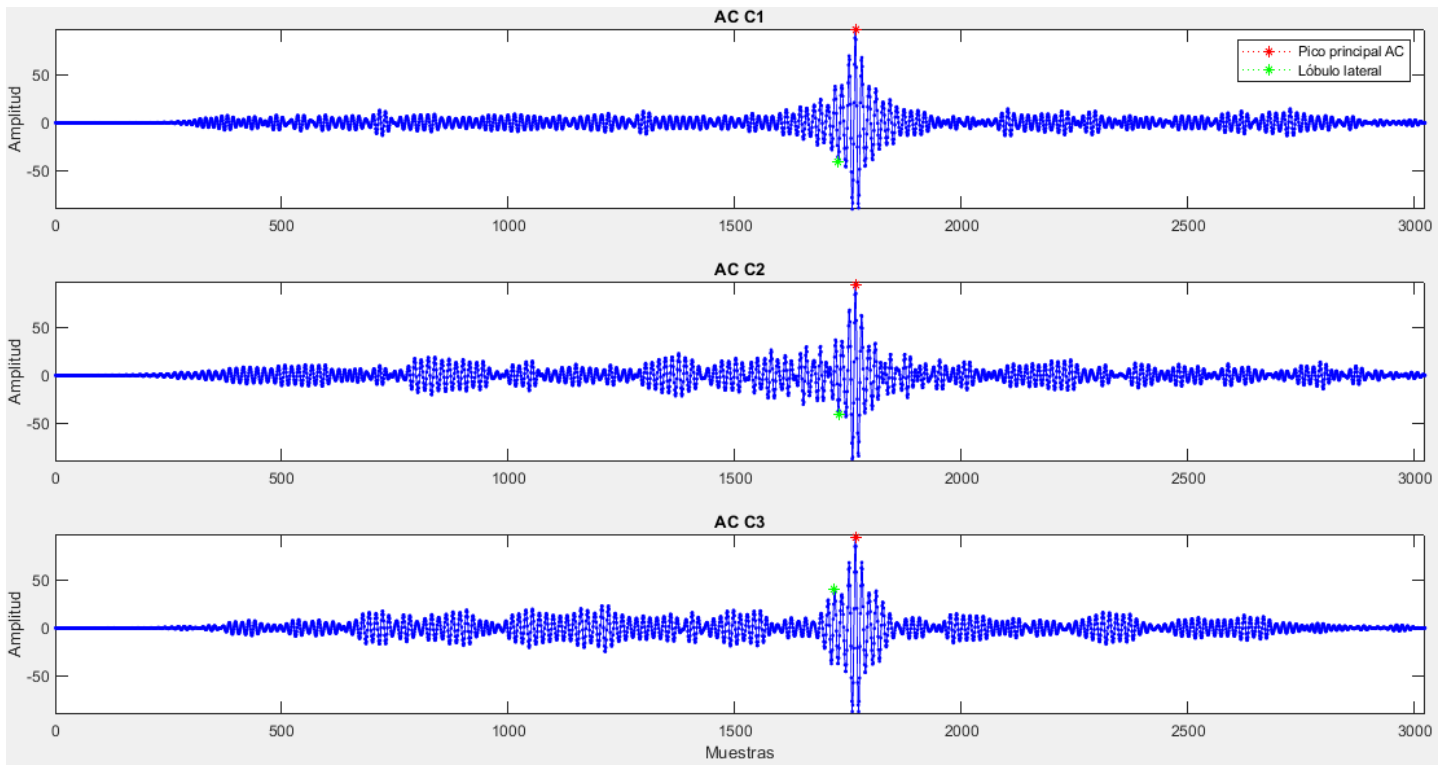


Fig. 197. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada con efecto transductor aperiódica

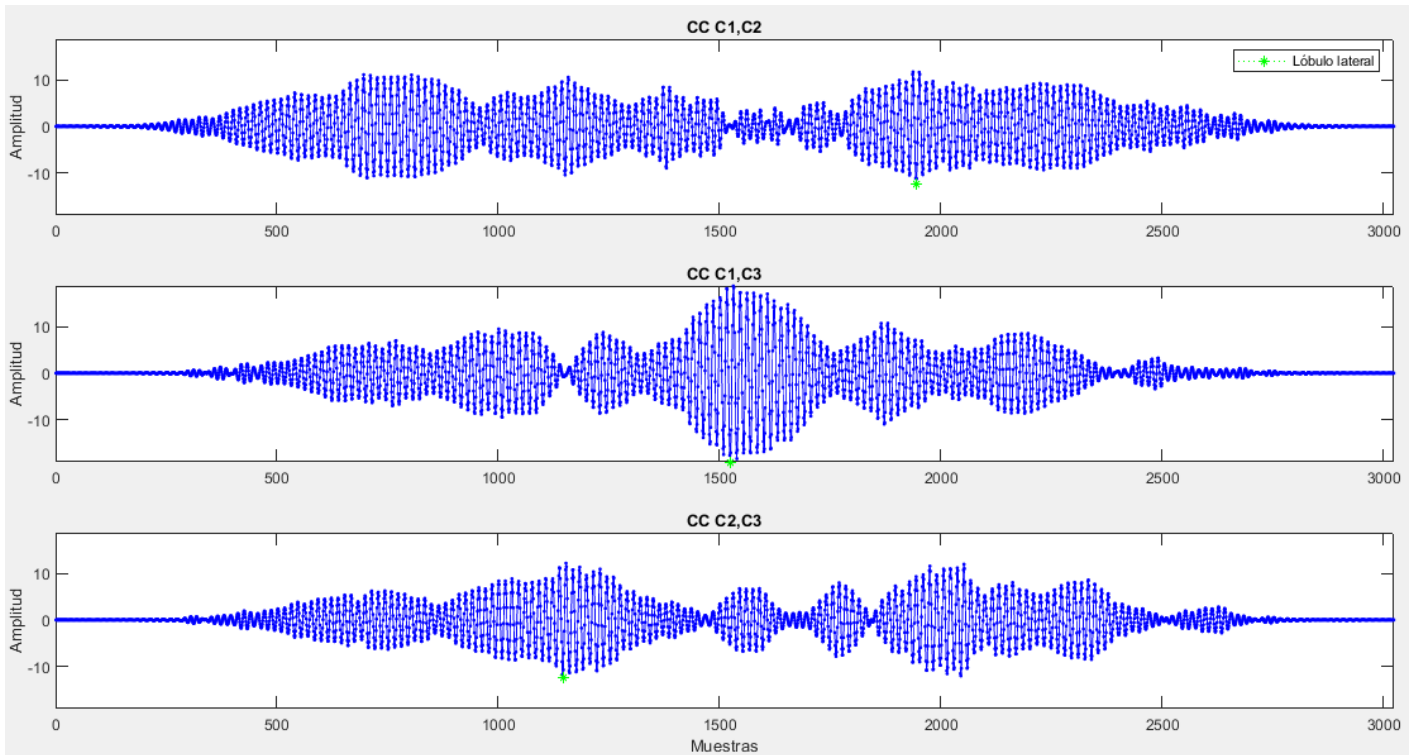


Fig. 198. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada con efecto transductor aperiódica

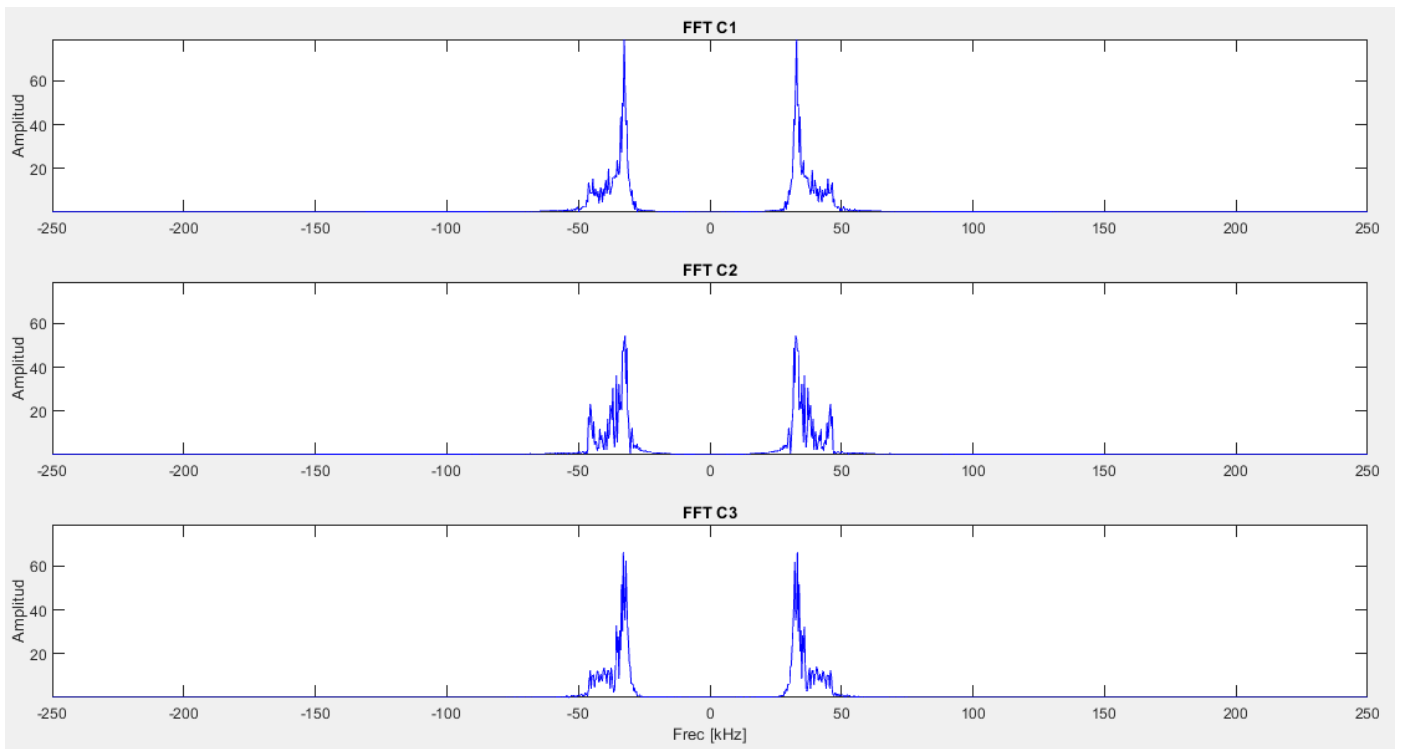


Fig. 199. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: FFT modulada con efecto transductor

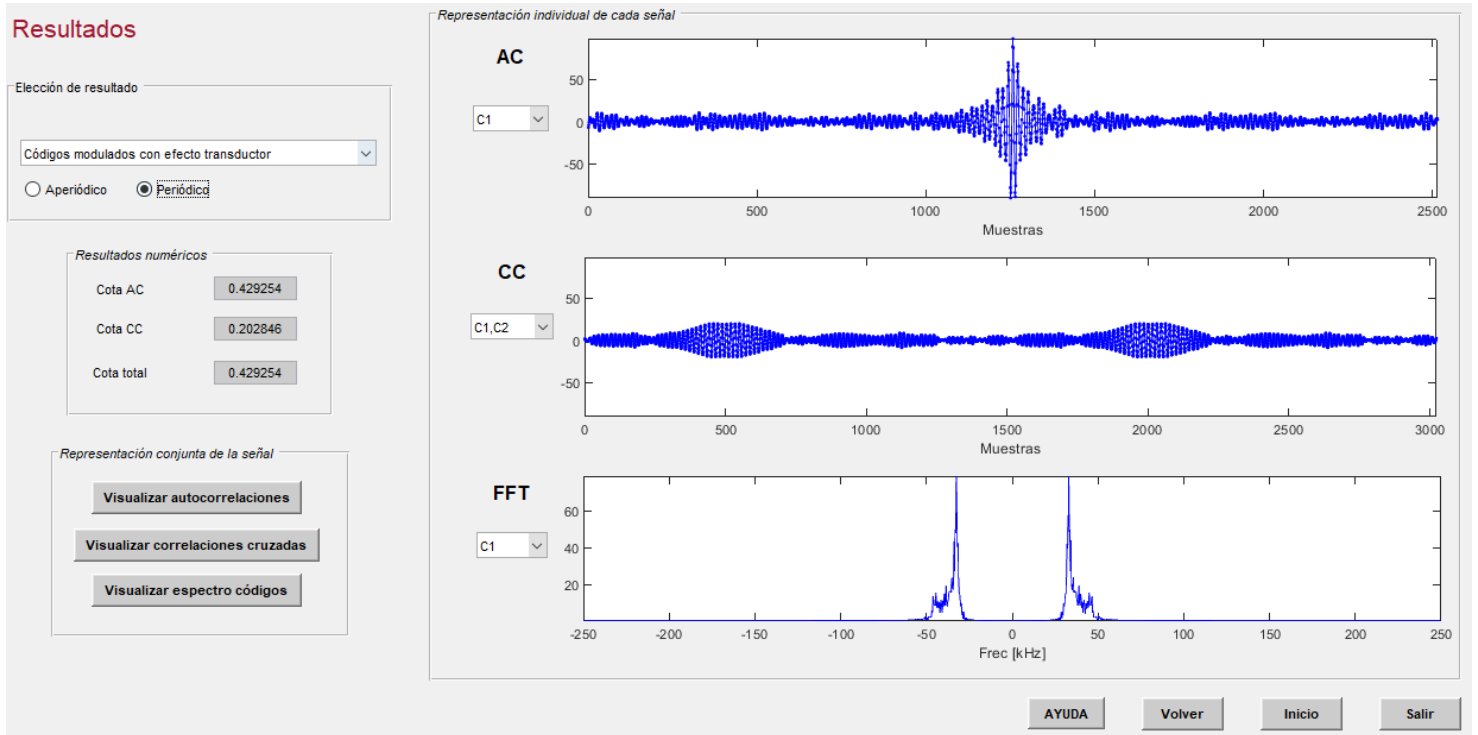


Fig. 200. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: modulada con efecto transductor periódica

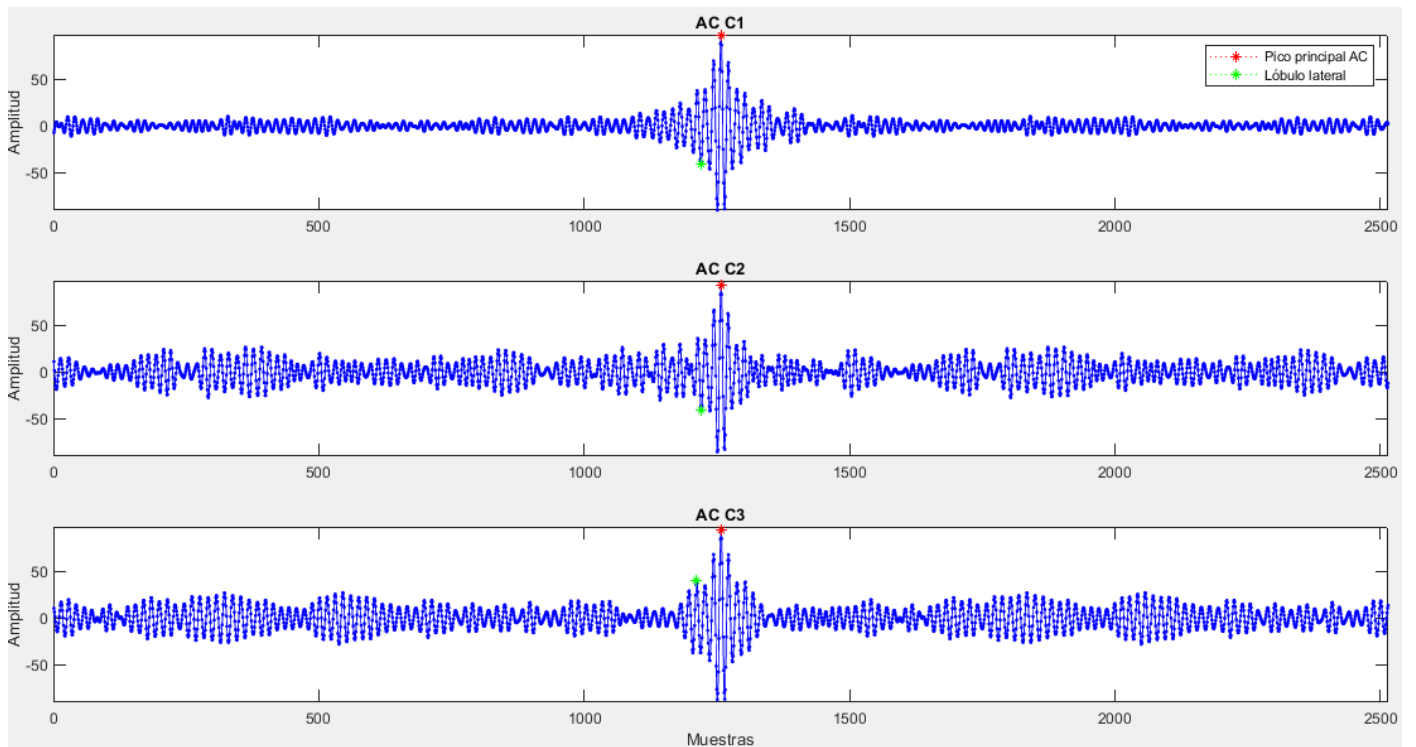


Fig. 201. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: AC modulada con efecto transductor periódica

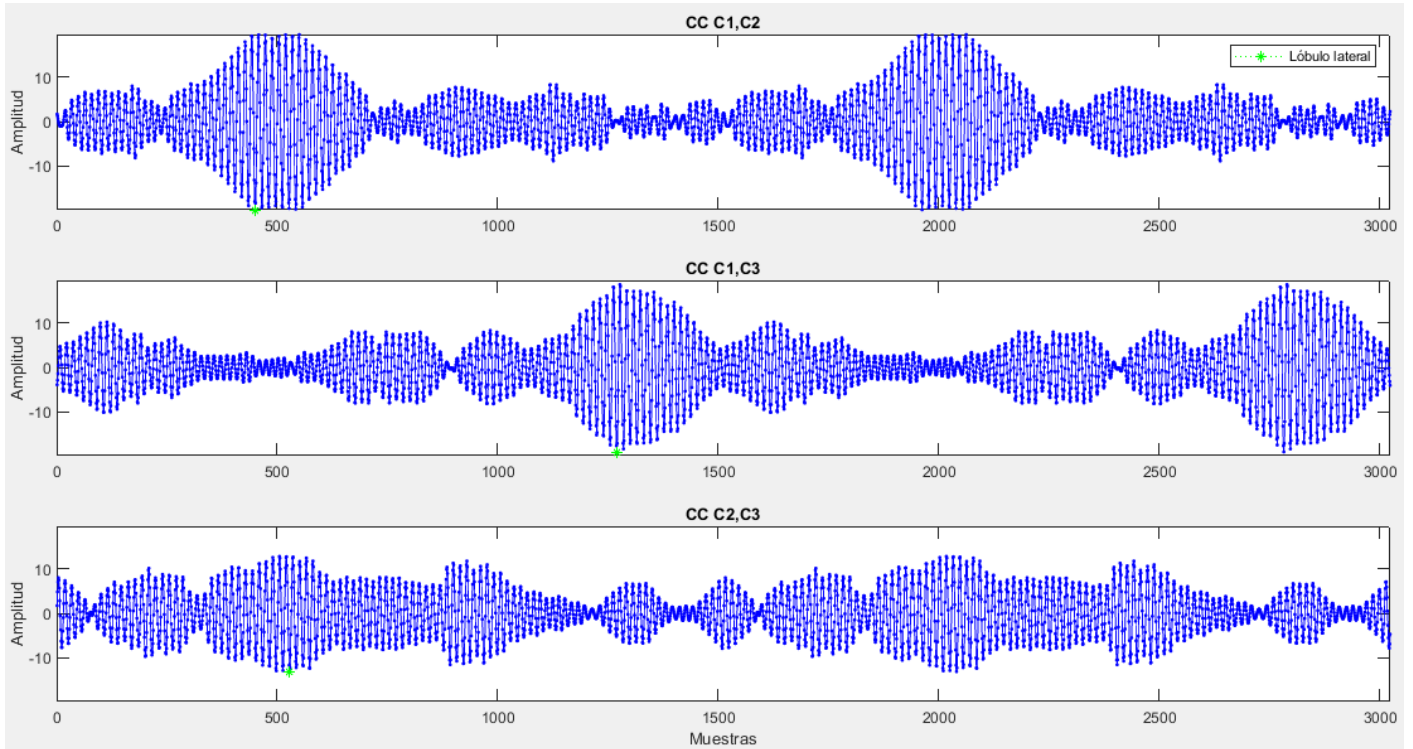


Fig. 202. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC modulada con efecto transductor periódica

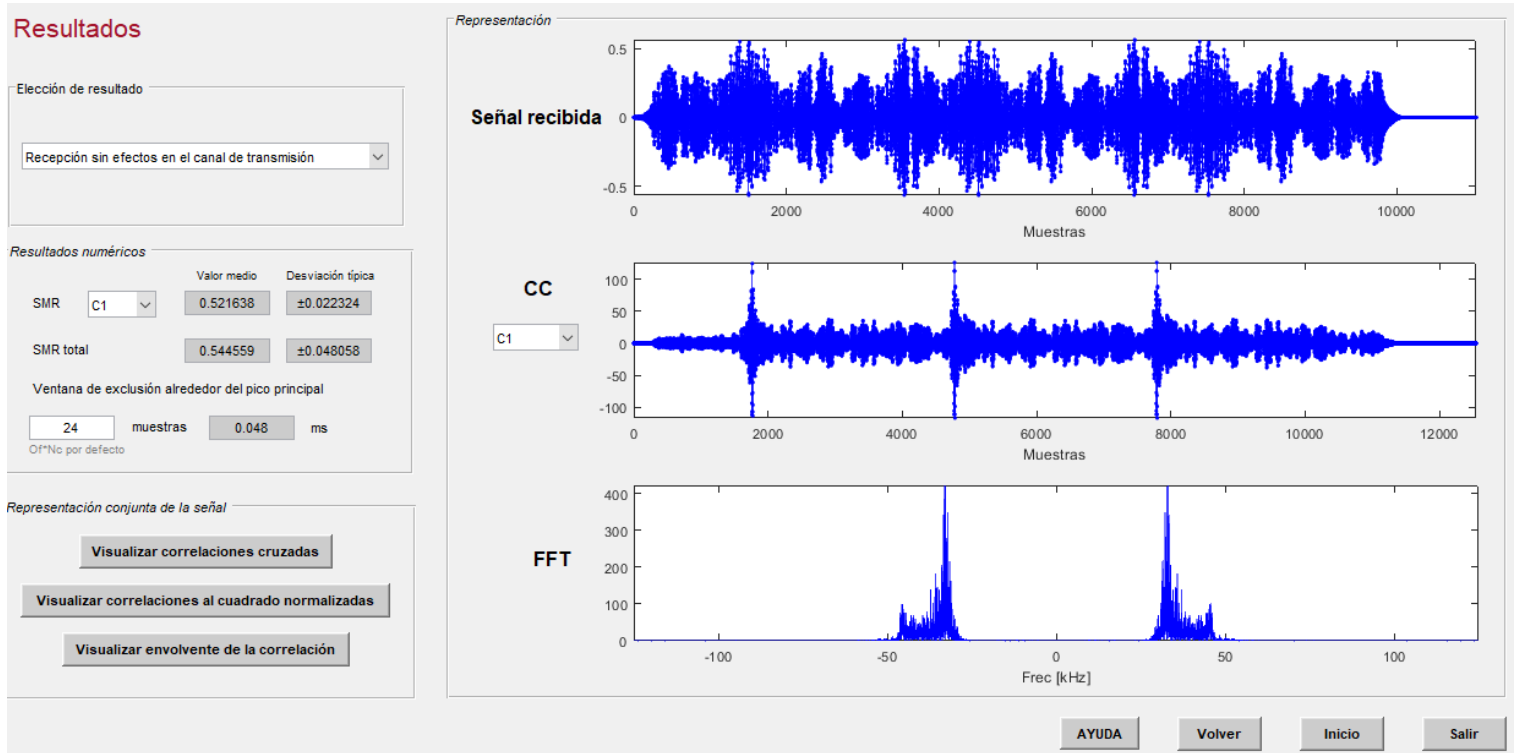


Fig. 203. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Señal recibida sin efectos canal

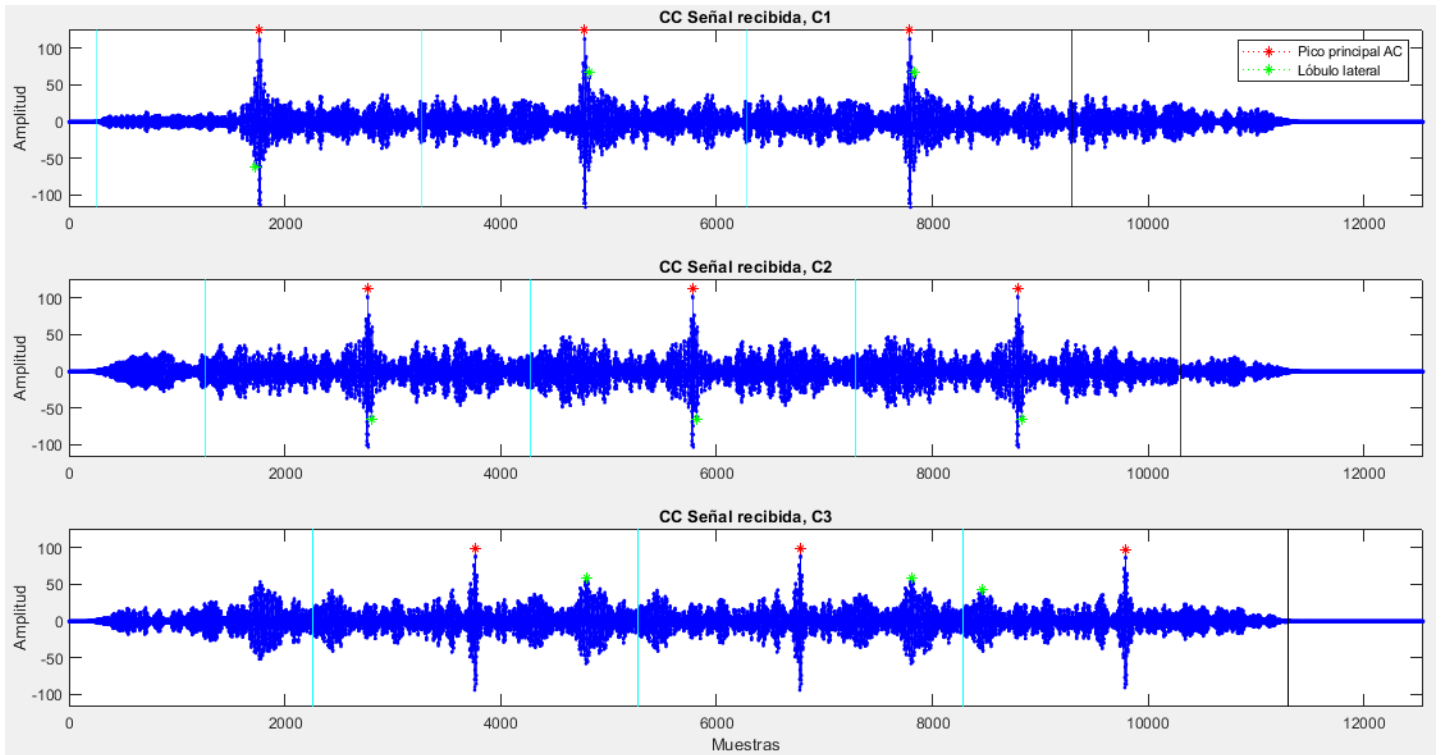


Fig. 204. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC Señal recibida sin efectos canal

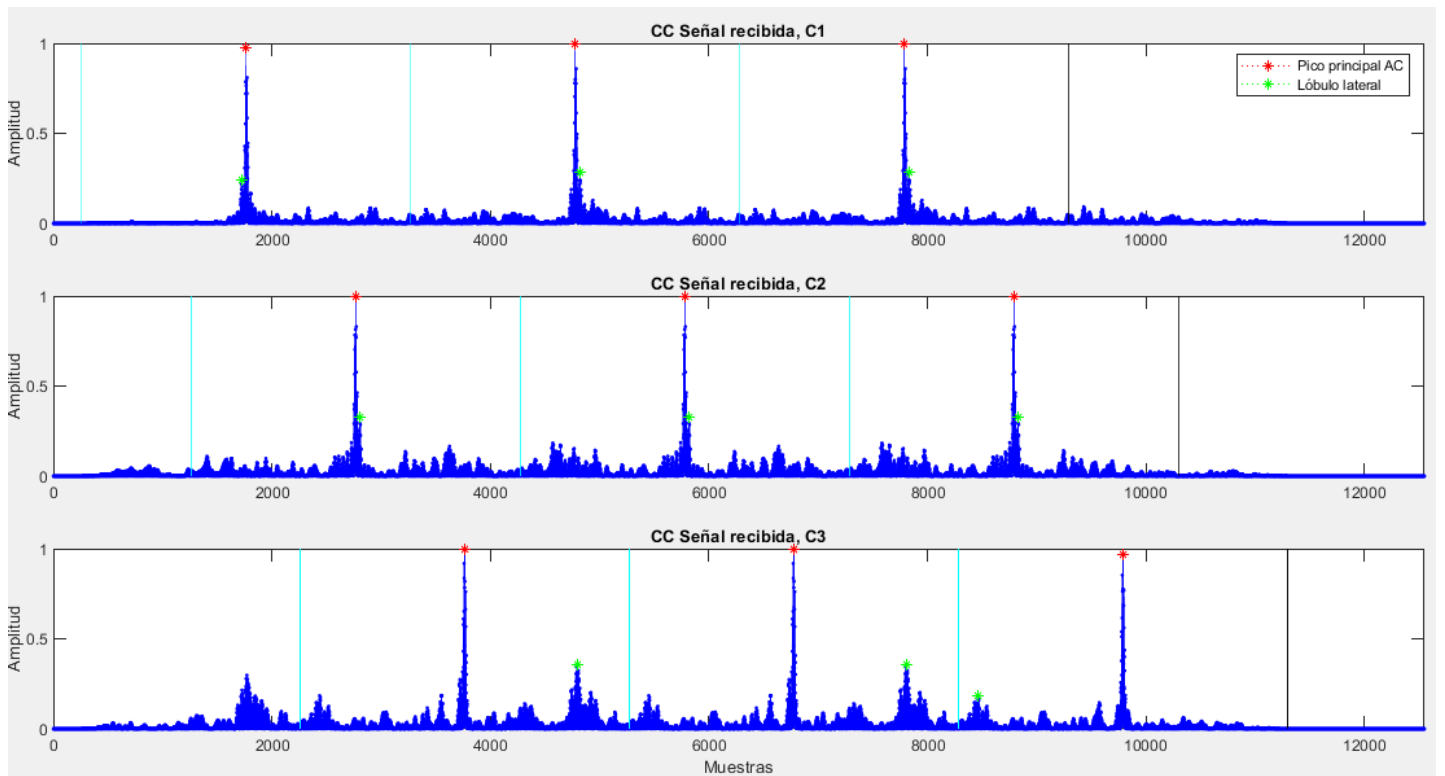


Fig. 205. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC Normalizada al cuadrado Señal recibida sin efectos canal

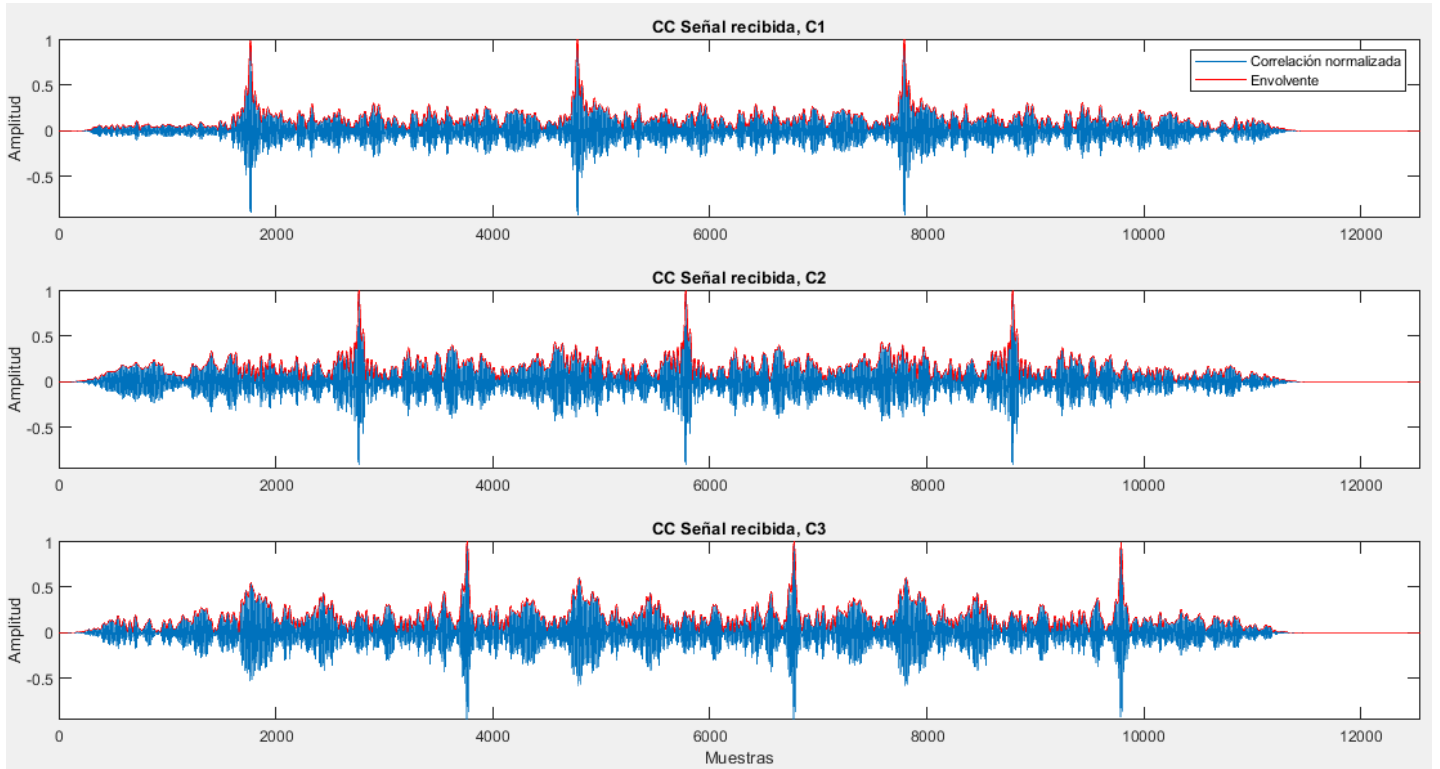


Fig. 206. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Envoltente Señal recibida sin efectos canal

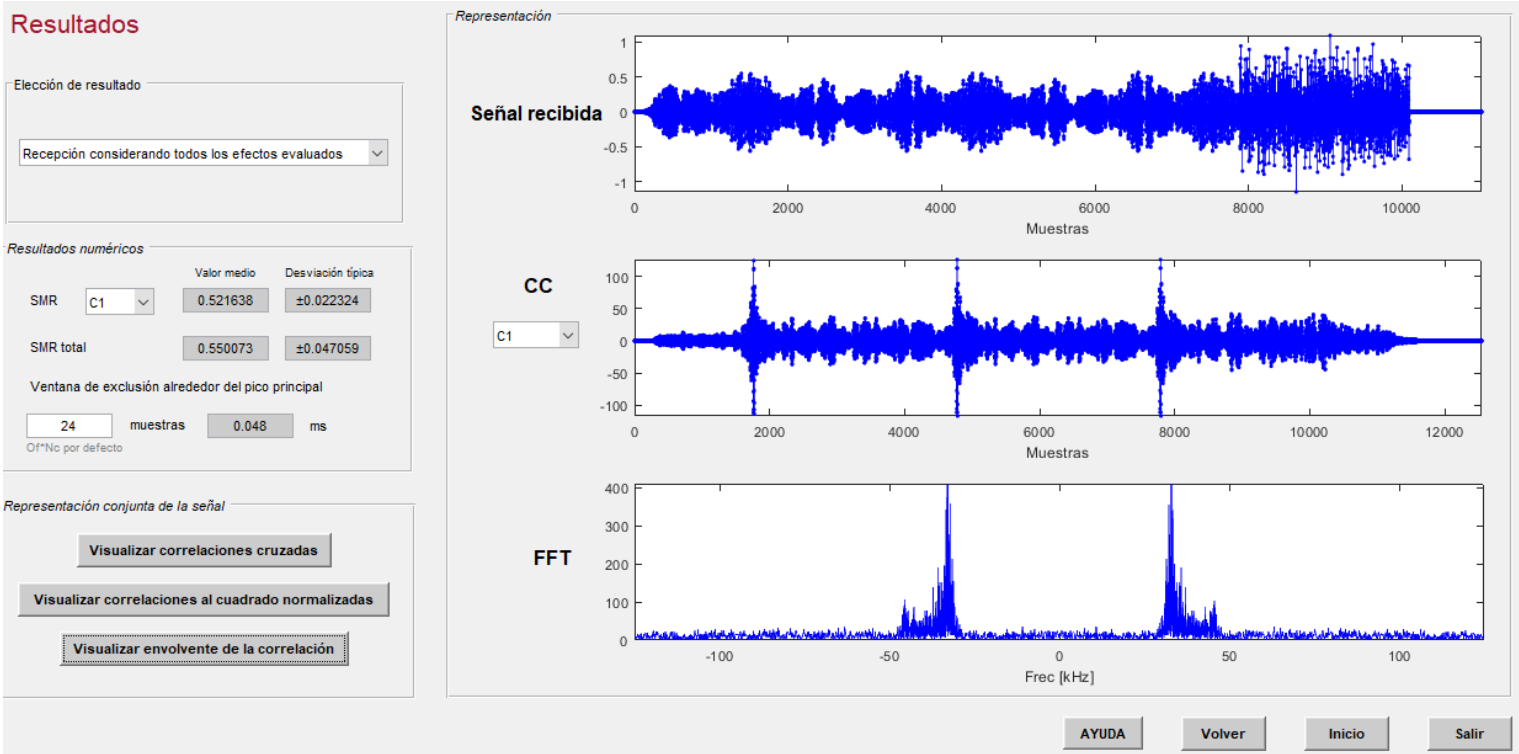


Fig. 207. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Señal recibida con efectos canal

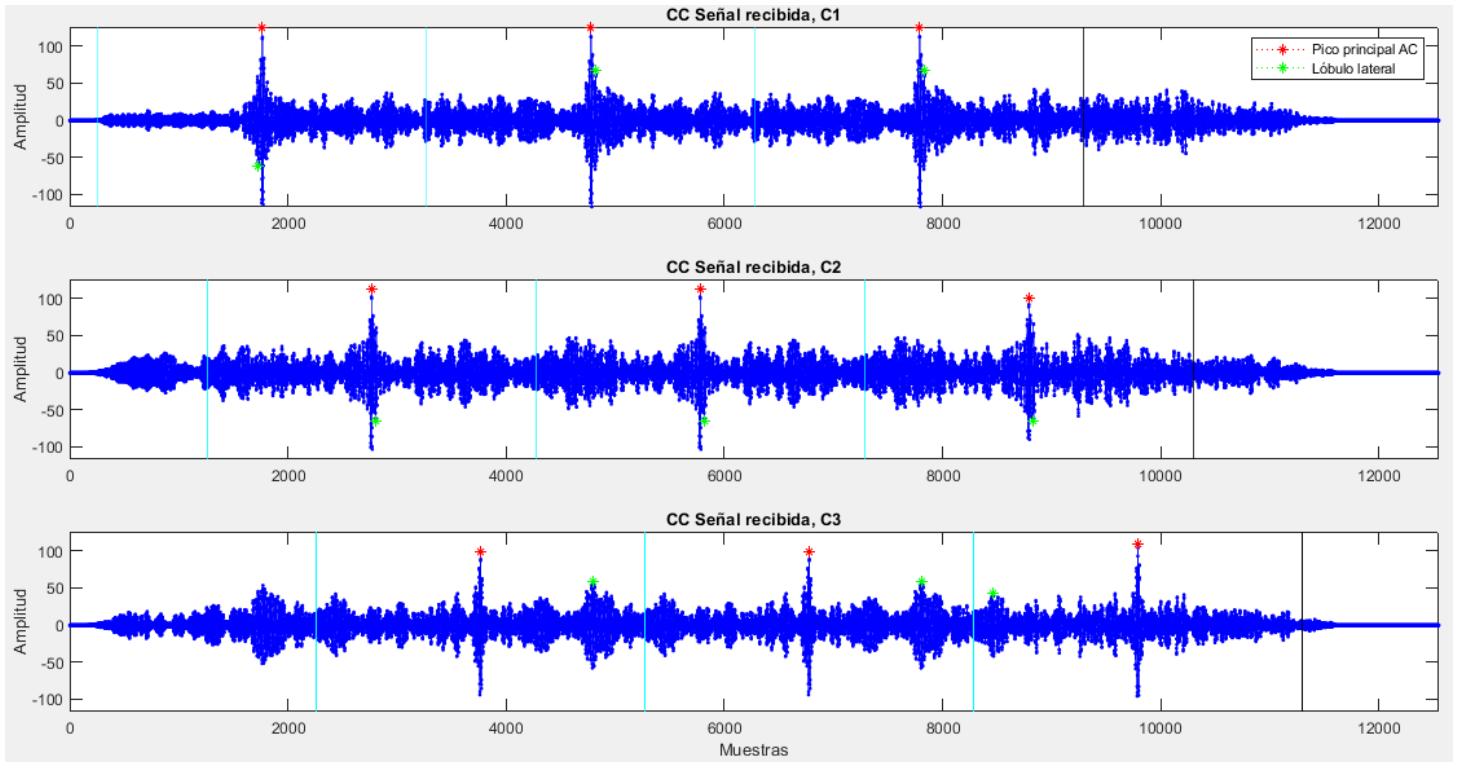


Fig. 208. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: CC Señal recibida con efectos canal

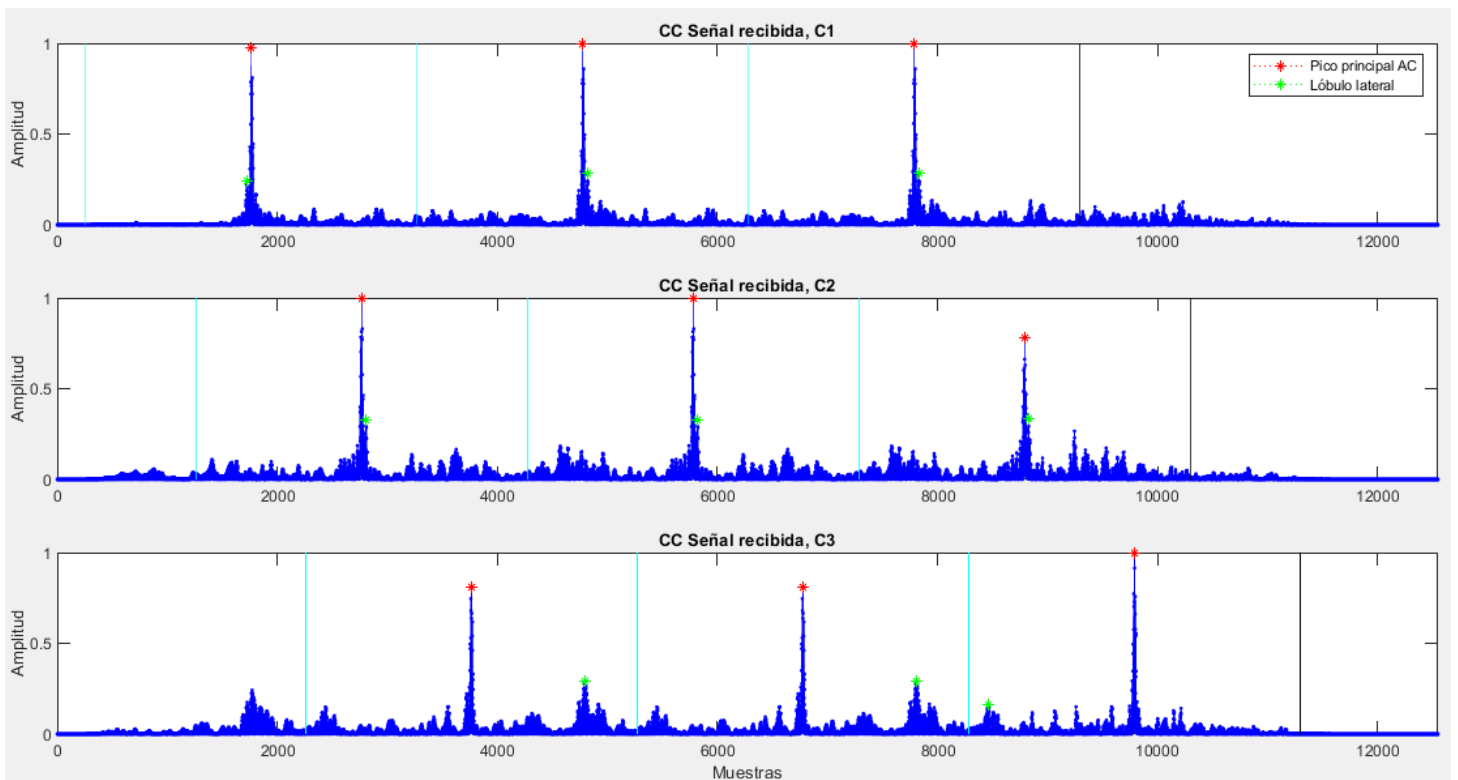


Fig. 209. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Normalizada al cuadrado Señal recibida con efectos canal

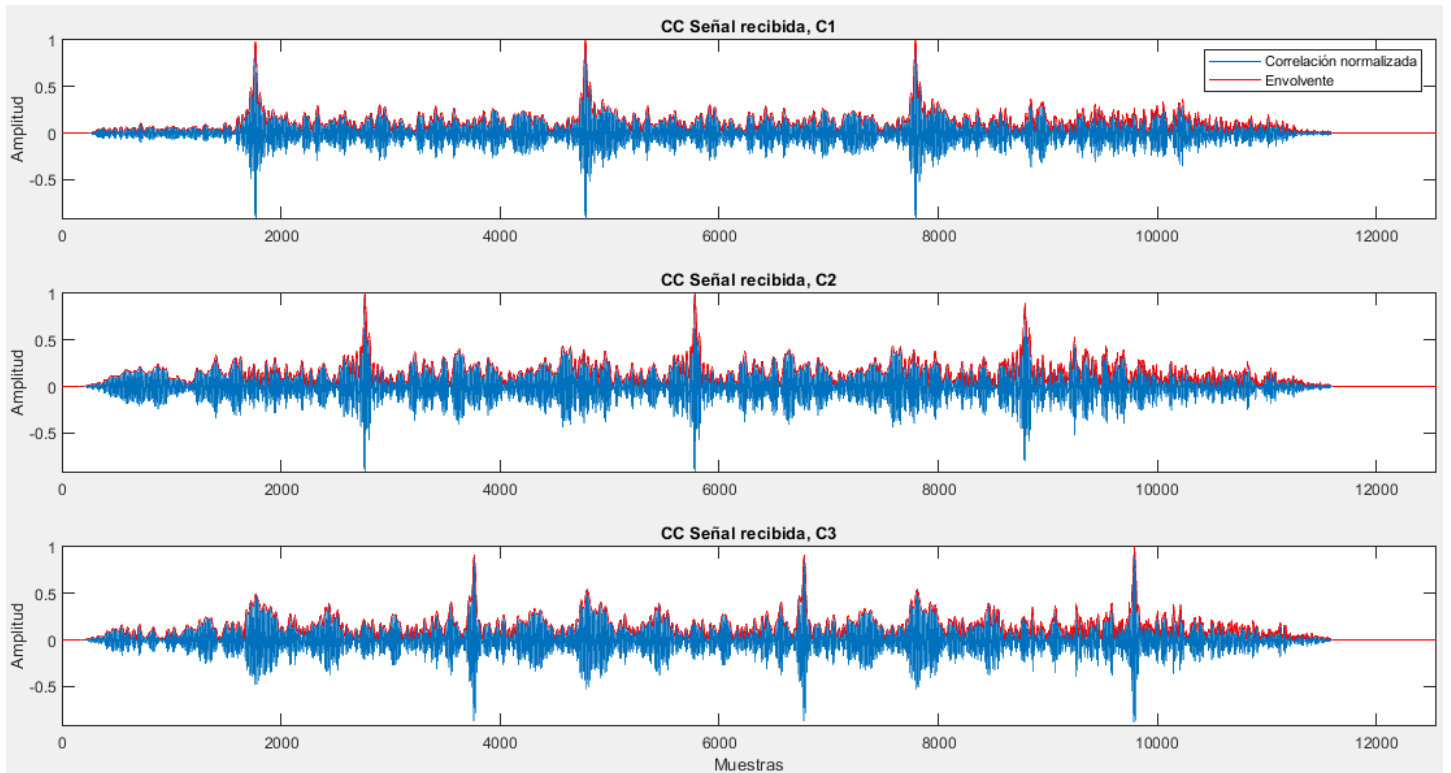


Fig. 210. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Envolvente Señal recibida con efectos canal

Efectos

Añadir efecto:

Simulación paramétrica

Simulación paramétrica

SNR desde dB

SNR hasta dB

Salto

Valores por defecto Borrar AYUDA Volver Siguiente

Fig. 211. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Configuración efectos añadidos (2)

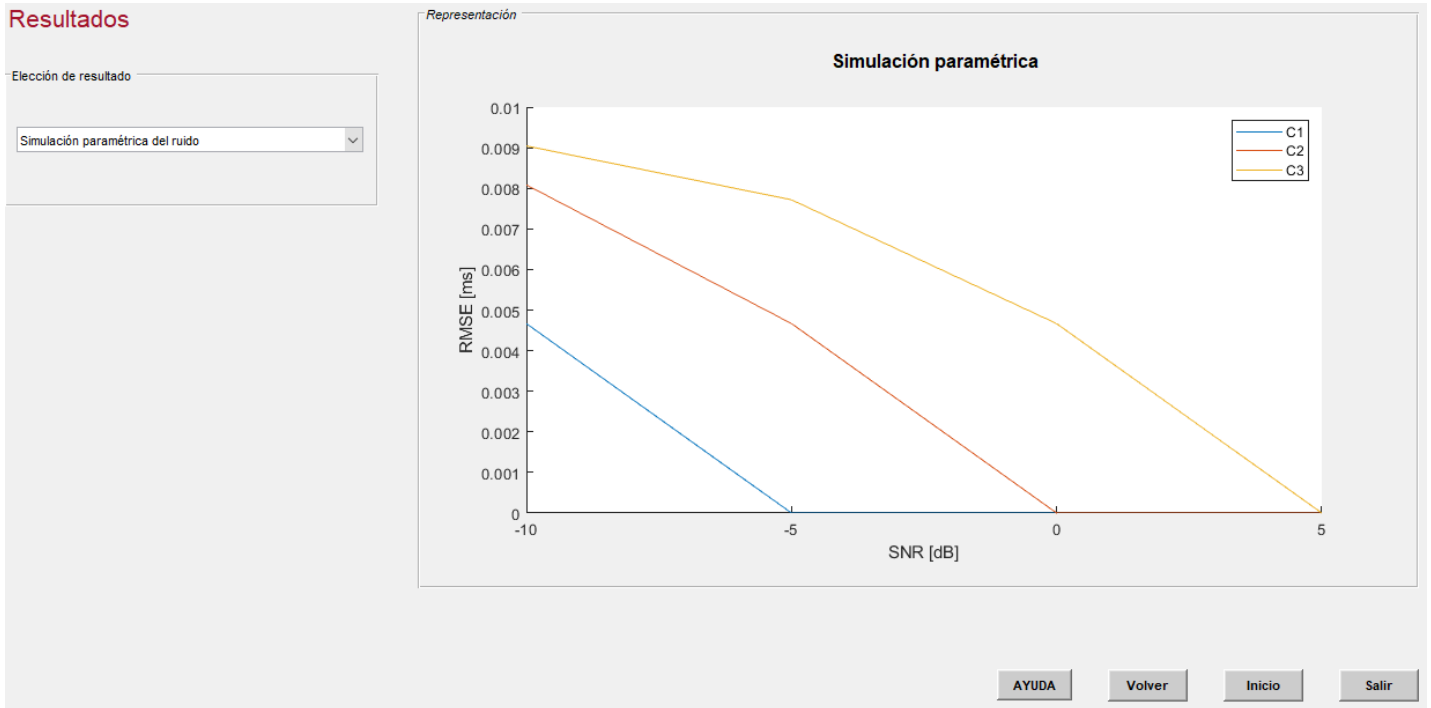


Fig. 212. Manual de Usuario: Ventana de Resultados secuencias Kasami: Ejemplo: Simulación paramétrica del ruido

8.6.3. Resultados secuencias CSS

Frente al resto de lenguajes de codificación, las secuencias CSS tienen una opción más de selección para la obtención de resultados. Esta opción no se agrupa en ninguno de los tres casos citados anteriormente y se denomina banda base ideal (Caso 0) que equivale al análisis de la señal en banda base antes de reordenar las M secuencias.

Caso 0. Banda base sin reordenar secuencias y sin modular

Nomenclatura	Definición	Observaciones
SACF	Suma de las AC de las secuencias de un código	
SCCF	Suma de las CC de las secuencias de un código con otro	
AC $C_i s_k$	AC de la secuencia S_k del código C_i	$1 \leq i \leq Nemo, 1 \leq k \leq M$
CC $C_i, C_j s_k$	CC de la secuencia S_k del código C_i con la secuencia S_k con el código C_j	$1 \leq i \leq Nemo-1, 2 \leq j \leq Nemo, 1 \leq k \leq M$
FFT $C_i s_k$	FFT de la secuencia S_k del código C_i	$1 \leq i \leq Nemo, 1 \leq k \leq M$
SACF C_i	SACF del código C_i	$1 \leq i \leq Nemo-1, 2 \leq j \leq Nemo$
SCCF C_i, C_j	SCCF del código C_i con el código C_j	$1 \leq i \leq Nemo-1, 2 \leq j \leq Nemo$

Opciones particulares	Definición
<input type="text" value="S1"/>	Selección de la FFT de la secuencia S_k del código C_i que se desee visualizar en pantalla ($1 \leq i \leq Nemo-1$, $1 \leq k \leq M$)
<input type="button" value="Visualizar autocorrelaciones"/> <input type="text" value="C1"/>	Representación conjunta de las AC del código que se seleccione a la derecha junto con su SACF correspondiente. Aparece marcado con un asterisco rojo el pico principal AC y con un asterisco verde el lóbulo lateral AC
<input type="button" value="Visualizar correlaciones cruzadas"/> <input type="text" value="C1,C2"/>	Representación conjunta de las CC entre los códigos que se seleccione a la derecha junto con su SCCF correspondiente. Aparece marcado con un asterisco verde el lóbulo lateral CC
<input type="button" value="Visualizar espectro códigos"/> <input type="text" value="C1"/>	Representación conjunta de las FFT del código seleccionado a la derecha

Ejemplo.

Configuración de emisión: CSS

Entradas

Tamaño familia (M) Concatenado Entrelazado

Longitud (L) Eficiencia energética (η)

Nº emisores simultáneos (Nemo)

Modulación BPSK:

Tipo portadora kHz

Frecuencia portadora (fc) kHz

Frecuencia de muestreo (fse) Factor sobremuestreo (Of) ms

Nº ciclos portadora (Nc) Duración emisión un código (Te)

Emisión:

Espaciado entre emisiones consecutivas (Tgap) ms

Retardo entre emisores (Tshift) ms

Número de repeticiones (Nrep)

Modelo transductor

Salidas

C1

Fig. 213. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: Configuración de la emisión

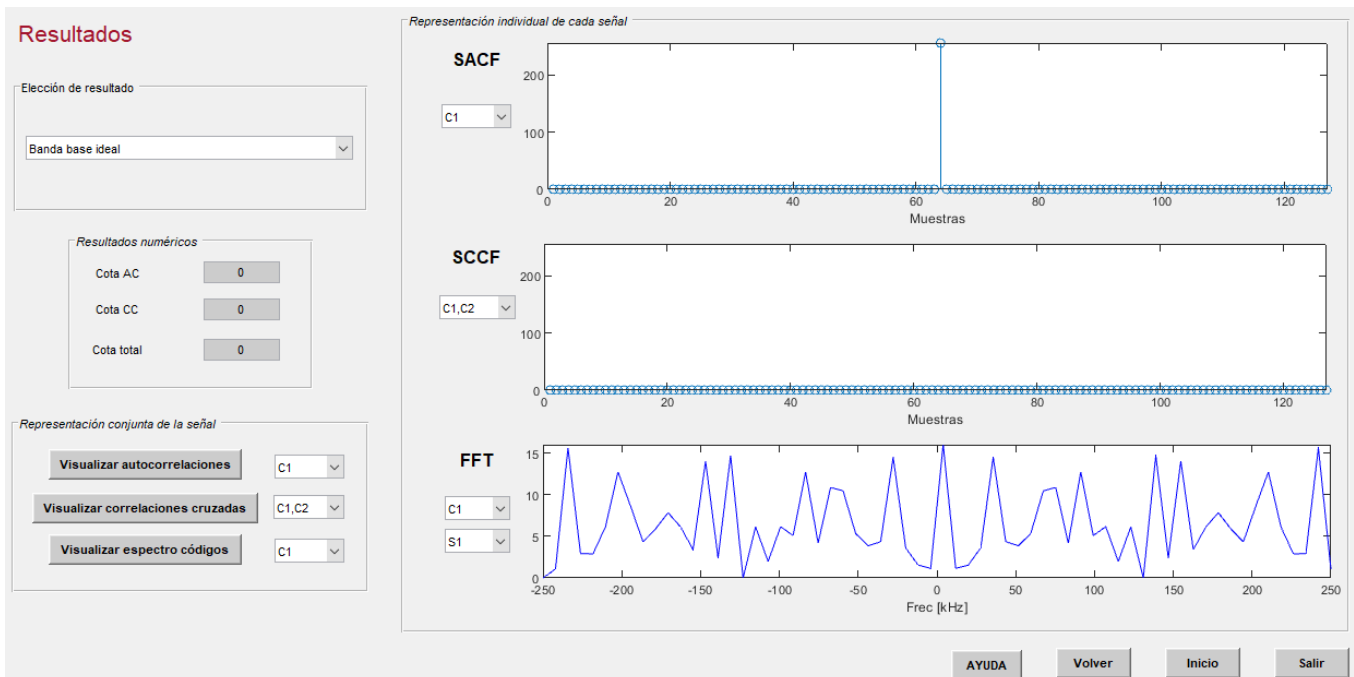


Fig. 214. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: Banda base ideal

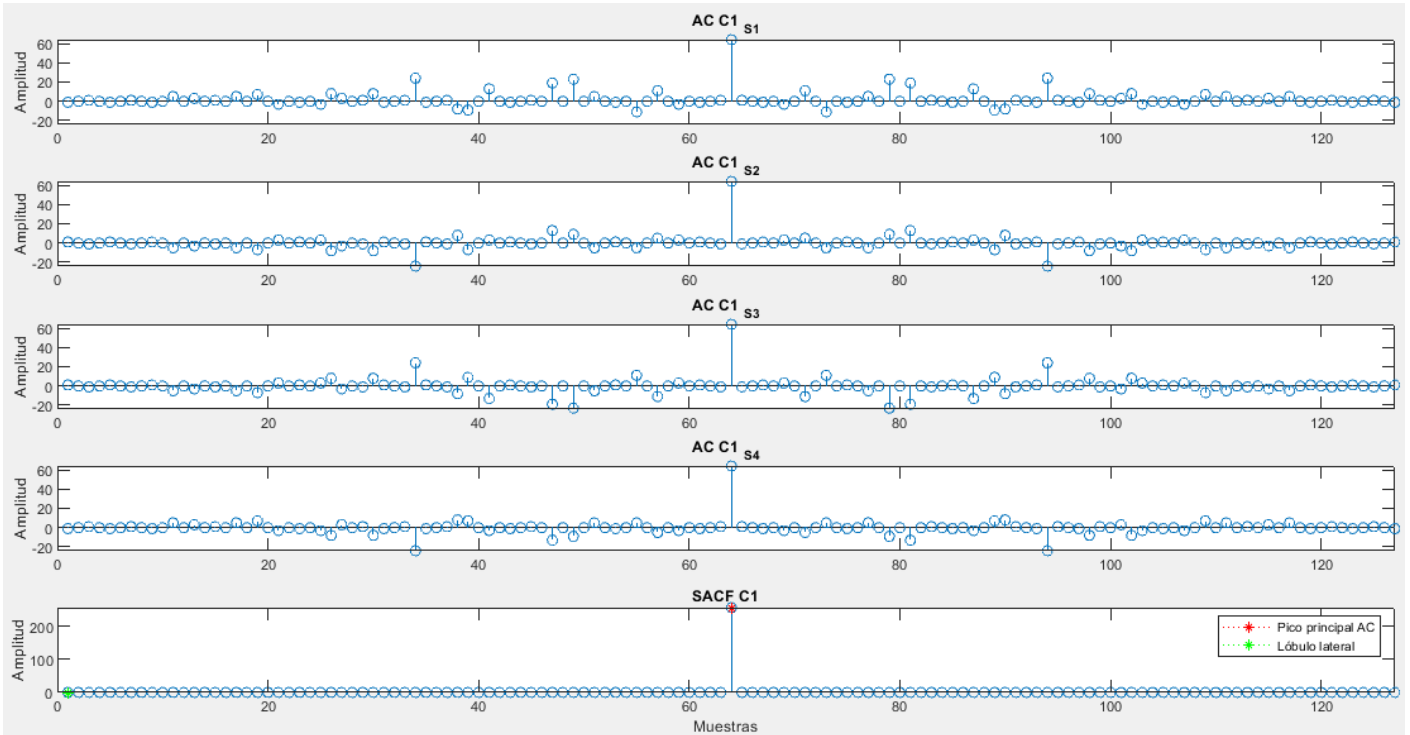


Fig. 215. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: AC Banda base ideal

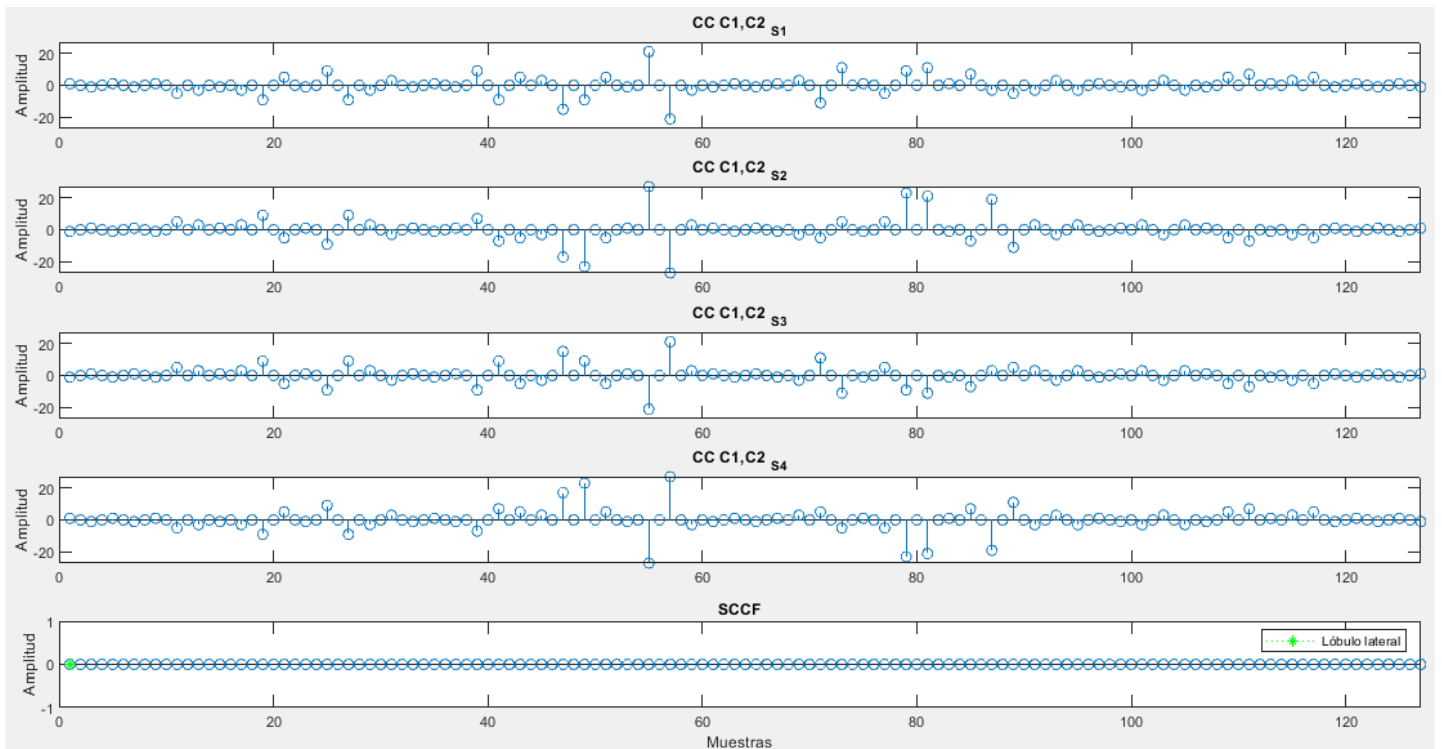


Fig. 216. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: CC Banda base ideal

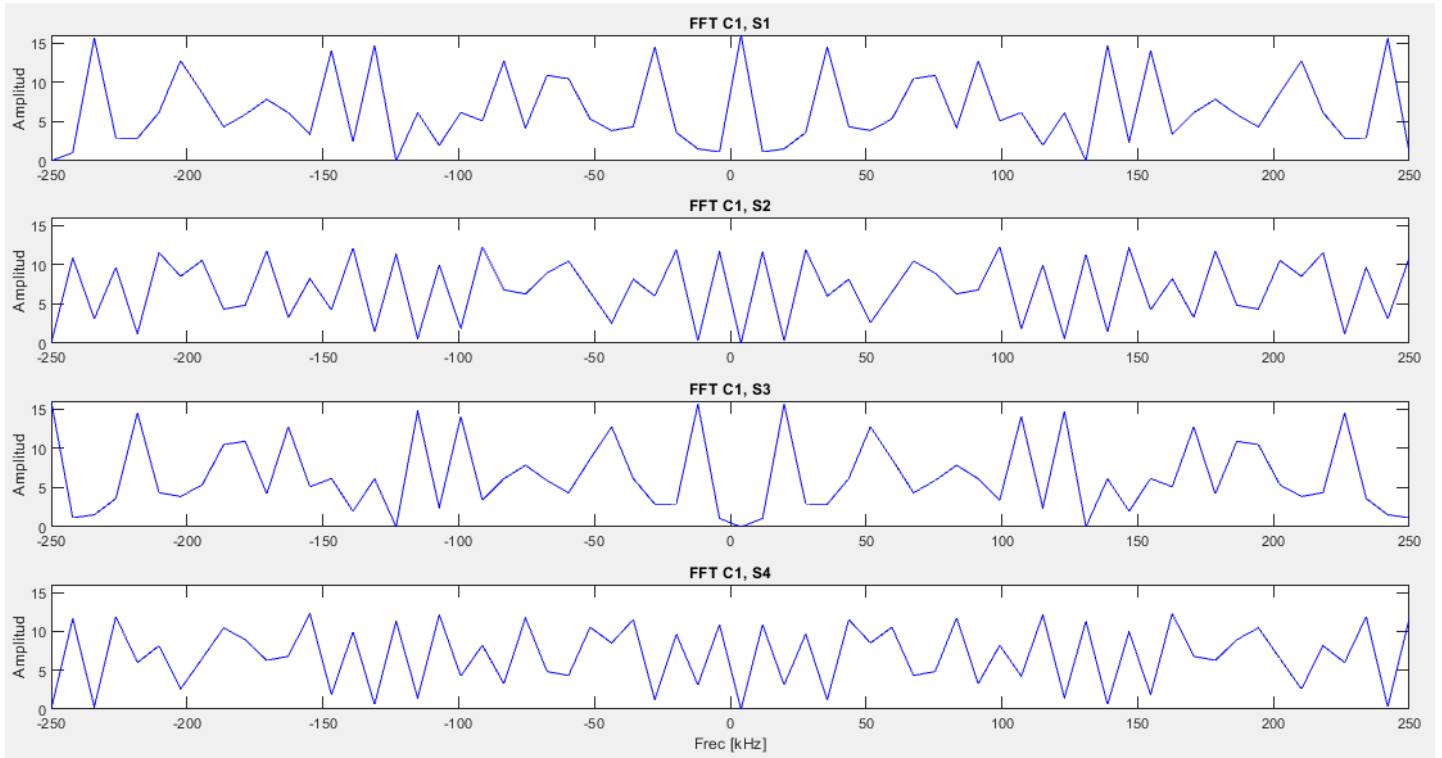


Fig. 217. Manual de Usuario: Ventana de resultados secuencias CSS: Ejemplo: FFT Banda base ideal

8.6.4. Resultados secuencias LS

Opciones particulares	Definición
<input checked="" type="radio"/> Dentro de la ZCZ	Realiza el cálculo de cotas dentro de la ZCZ
<input type="radio"/> Fuera de la ZCZ	Realiza el cálculo de cotas considerando toda la señal

Ejemplo

Configuración de emisión: LS

Entradas

Tamaño familia (M)	<input type="text" value="8"/>	Zona correlación cero (ZCZ)	<input type="text" value="31"/>
Longitud parejas Golay Base (Lo)	<input type="text" value="16"/>	Longitud código (L)	<input type="text" value="143"/>
Nº emisores simultáneos (Nemo)	<input type="text" value="4"/>	Eficiencia energética (η)	<input type="text" value="0.895105"/>

Modulación BPSK:

Tipo portadora	<input type="text" value="Sinusoidal"/>		
Frecuencia portadora (fc)	<input type="text" value="41.6667"/>	kHz	
Frecuencia de muestreo (fse)	<input type="text" value="500"/>	kHz	Factor sobremuestreo (Of)
Nº ciclos portadora (Nc)	<input type="text" value="2"/>		<input type="text" value="12"/>
		Duración emisión un código (Te)	<input type="text" value="6.86399"/>
		ms	

Emisión:

Espaciado entre emisiones consecutivas (Tgap)	<input type="text" value="10"/>	ms	
Retardo entre emisores (Tshift)	<input type="text" value="1"/>	ms	<input type="button" value="Imagen orientativa"/>
Número de repeticiones (Nrep)	<input type="text" value="4"/>		
Modelo transductor	<input type="text" value="Sí"/>		

Salidas

Fig. 218. Manual de Usuario: Ventana de resultados secuencias LS: Ejemplo: Configuración de la emisión

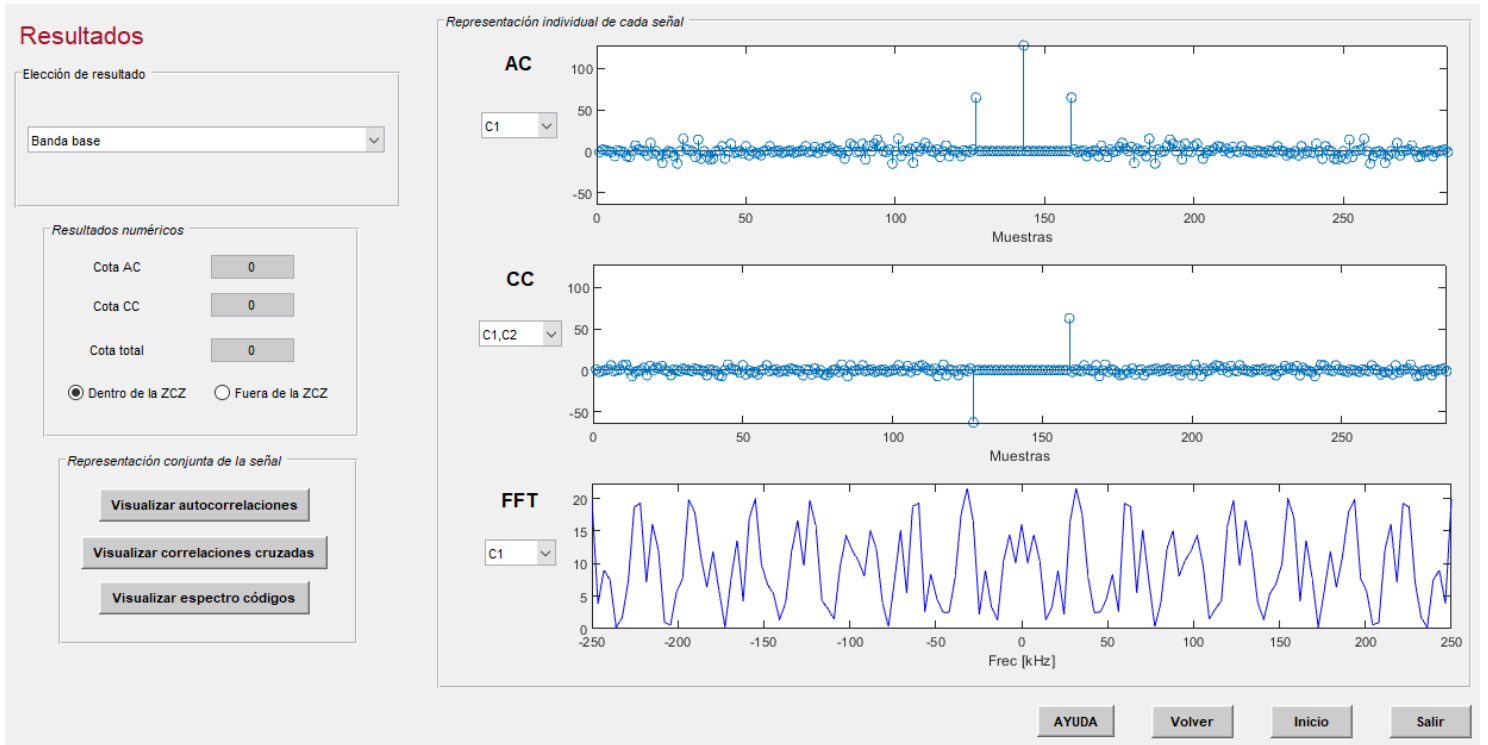


Fig. 219. Manual de Usuario: Ventana de resultados secuencias LS: Ejemplo: Banda base dentro de la ZCZ

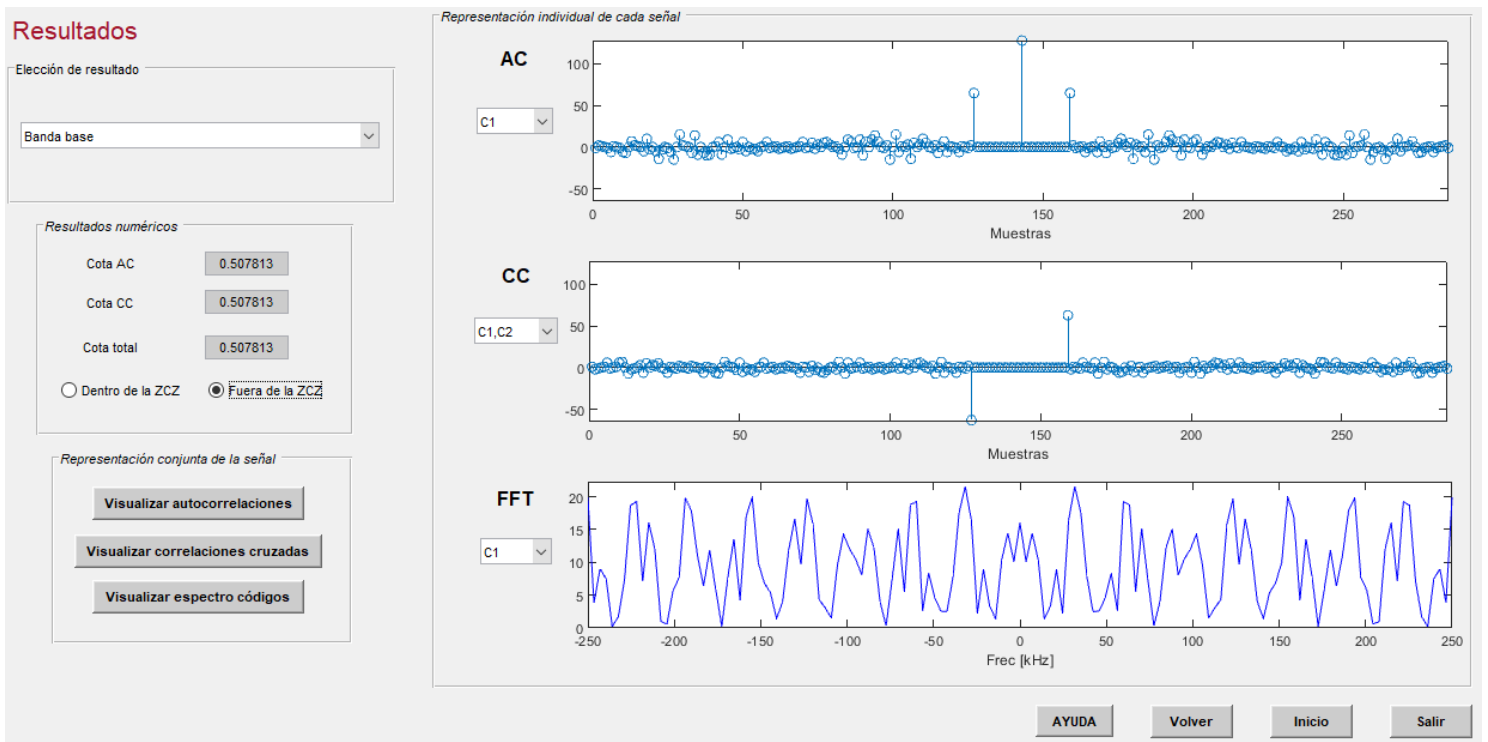


Fig. 220. Manual de Usuario: Ventana de resultados secuencias LS: Ejemplo: banda base fuera de la ZCZ

8.6.5. Resultados secuencias Zadoff-Chu

La visualización de resultados de los códigos Zadoff-Chu es similar a la de las Kasami. La única peculiaridad que tienen es que el resultado de las FFT (Fig. 131) en **banda base** son otras señales complejas de módulo distinto a uno.

Ejemplo

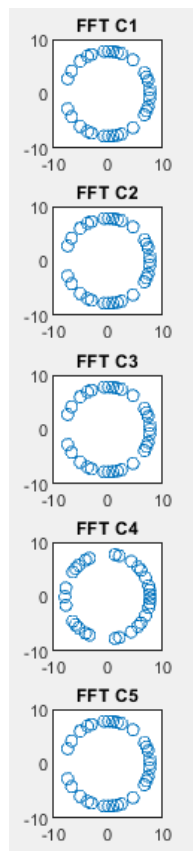


Fig. 221. Manual de Usuario: Ventana de resultados secuencias Zadoff-Chu: Ejemplo: FFT banda base

8.7. Ventana de salida

Cabe la posibilidad de ampliar este proyecto realizando pruebas experimentales a través de la herramienta Locate-Us cuyo proyecto trata del Desarrollo de un demostrador online para Sistemas de Posicionamiento Local Ultrasónicos [CHG⁺19a] [CHG⁺19b].

Las restricciones que existen para poder realizar estas pruebas aparecen en la figura 222 la cual representa la última pantalla que aparece en este TFG.



Fig. 222. Ventana 5 TFG: Locate-US

9. BIBLIOGRAFÍA

- [BBC⁺14] P. Banelli, S. Buzzi, G. Colavolpe, A. Modenini, F. Rusek, A. Ugolini, "Modulation Formats and Waveforms for 5G Networks: Who Will be the Heir of OFDM?", IEEE Signal Processing Magazine, pp 80-93, Nov. 2014
- [CHG⁺19a] F. Ciudad, Á. Hernández, D. Gualda, E. Aparicio-Esteve, "Desarrollo de un demostrador online para Sistemas de Posicionamiento Local Ultrasónicos", 26ª edición del Seminario Anual de Automática, Electrónica Industrial e Instrumentación, 2019. I.S.B.N. : 978-84-17171-50-6
- [CHG⁺19b] F. Ciudad, Á. Hernández, D. Gualda, E. Aparicio-Esteve, "Design and Implementation of a Remotely Configurable Ultrasonic Local Positioning Systems", tenth edition of the International conference on Indoor Positioning and Indoor Navigation, 2019. Aceptado para publicación.
- [Chu72] D. Chu, "Polyphase codes with good periodic correlation properties" IEEE Transactions on Information Theory, vol. 18, no. 4, pp. 531-532, 1972
- [FD96] Pingzhi Fan, Michael Darnell, "Sequence design for communications applications", John Wiley & Sons Inc. 1996
- [Gar13] E. García, "Efficient Complementary Sequences-Based Architectures and their Application to Ranging Measurements", Tesis doctoral, Universidad de Alcalá, Abril 2013
- [GPÁ⁺15] Enrique García, José A. Paredes, Fernando J. Álvarez, M. Carmen Pérez, Juan Jesús García, "Spreading sequences in active sensing: A review", Signal Processing, vol. 106, pp. 88-105, 2015
- [Kas68] T. Kasami. "Weight distribution formula for some class of cyclic codes". Technical report R-285, Coordinated Science Lab. University of Illinois, 1968
- [MAT19] MathWorks (26 de Junio de 2019), "Creación de apps con interfaces gráficas de usuario en MATLAB", recuperado de <https://es.mathworks.com/discovery/matlab-gui.html>
- [MPU⁺18] Santiago Murano, M^a Carmen Pérez, Jesús Ureña, Chris J. Bleakley, Carlos De Marziani, "Comparison of Zadoff-Chu Encoded Modulation Schemes in an Ultrasonic Local Positioning System", Proceedings of the

2018 International Conference on Indoor Positioning and Indoor Navigation, September 2018

- [Pér09] M. C. Pérez, “Generación y correlación eficiente de códigos binarios derivados de conjuntos de secuencias complementarias para sistemas ultrasónicos”, tesis doctoral, Universidad de Alcalá, 2009
- [PUH⁺09] M. C. Pérez, J. Ureña, A. Hernández, A. Jiménez, F. J. Álvarez, C. De Marziani, “Performance Comparison of Different Codes in an Ultrasonic Local Positioning System using DS-CDMA”, 6th IEEE International Symposium on Intelligent Signal Processing, Budapest (Hungría), 2009
- [SBH01] S. Stanzak, H. Boche y M. Haardt, “Are LAS-codes a miracle?” Proc. of IEEE Global Telecommunications Conference (GLOBECOM’2001), vol. 1, pp. 589-593, San Antonio (EEUU), noviembre, 2001
- [TL72] C. C. Tsens y C. L. Liu, “Complementary Sets of Sequences” IEEE Transactions on Information Theory. IT-18 (5):644-652, 1972
- [Wel74] L. Welch, “Lower bounds on the maximum cross correlation of signals”, Information Theory, IEEE Transactions on, 20(3):397–399, 1974.

10. ANEXO: CÓDIGOS

Se incluyen en esta memoria los script más relevantes de la aplicación

10.1. Cálculo de la Cota AC para secuencias Kasami

```

function cota_ac=crear_cota_ac(code)

%Crear la Cota AC para los distintos casos: banda base, modulación y
%transductor

Nemo=evalin('base','Nemo');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
delay=evalin('base','delay');
L=evalin('base','L');
aperiodica=evalin('base','aperiodica');
switch code
  case 1
    x=evalin('base','codigo_kasami_sin_modular');
  case 2
    x=evalin('base','codigo_kasami_modulado');
    L=L*Nc*Of; %en muestras
  case 3
    x=evalin('base','codigo_kasami_transductor');
    y=evalin('base','codigo_kasami_modulado');
    L=L*Nc*Of; %en muestras
end

if aperiodica==0
  for i=1:Nemo
    z(i,:)=repmat(x(i,:),1,3);
  end
else
  z=x;
end

if code==1 %fórmula para el caso banda base
  for i=1:Nemo
    ac(i,:)=conv(z(i,:),fliplr(x(i,:)));
    if aperiodica==0
      iz=L+1;
      de=3*L-1;
      aux(i,:)=ac(i,(iz:de));
      [value(i,:),pos(i,:)]=max(aux(i,:));
      ac_aux(i,:)=aux(i,:);
      ac_aux(i,pos(i,:))=0;
      [value_lob(i,:),pos_lob(i,:)]=max(abs(ac_aux(i,:)));
    else
      [value(i,:),pos(i,:)]=max(ac(i,:));
      ac_aux(i,:)=ac(i,:);
      ac_aux(i,pos(i,:))=0;
      [value_lob(i,:),pos_lob(i,:)]=max(abs(ac_aux(i,:)));
    end
  end
end
  
```

```

        cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
    end
    cota_ac=max(cota_ac_aux);
end
if code==2
    for i=1:Nemo
        ac(i,:)=conv(z(i,:),fliplr(x(i,:)));
        if aperiodica==0
            iz=L+1+Nc*Of;
            de=3*L-1-Nc*Of;
            aux(i,:)=ac(i,(iz:de));
            [value(i,:),pos(i,.)]=max(aux(i,:));
            ac_aux(i,:)=aux(i,:);
            ac_aux(i,(pos(i,:)-
round(Of*Nc):pos(i,.)+round(Of*Nc)))=zeros(1,2*Of*Nc+1);
            %ac_aux(i,1:Of*Nc)=zeros(1,Of*Nc);
            %ac_aux(i,length(ac_aux(i,:))-
Of*Nc:length(ac_aux(i,:)))=zeros(1,Of*Nc+1);
            [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
        else
            [value(i,:),pos(i,.)]=max(abs(ac(i,:)));
            ac_aux(i,:)=ac(i,:);
            ac_aux(i,(pos(i,:)-
round(Of*Nc):pos(i,.)+round(Of*Nc)))=zeros(1,2*Of*Nc+1);
            [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
        end
        cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
    end
    cota_ac=max(cota_ac_aux);
end
if code==3
    for i=1:Nemo
        ac(i,:)=conv(z(i,:),fliplr(y(i,:)));
        if aperiodica==0
            iz=L+delay+1+delay;
            de=3*L+delay-1-delay;
            aux(i,:)=ac(i,(iz:de));
            [value(i,:),pos(i,.)]=max(aux(i,:));
            ac_aux(i,:)=aux(i,:);
            ac_aux(i,[pos(i,.)-
round(Of*Nc):pos(i,.)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
            %ac_aux(i,1:Of*Nc)=zeros(1,Of*Nc);
            %ac_aux(i,length(ac_aux(i,:))-
Of*Nc:length(ac_aux(i,:)))=zeros(1,Of*Nc+1);
            [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
        else
            [value(i,:),pos(i,.)]=max(abs(ac(i,:)));
            ac_aux(i,:)=ac(i,:);
            ac_aux(i,[pos(i,.)-
round(Of*Nc):pos(i,.)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
            [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
        end
        cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
    end
    cota_ac=max(cota_ac_aux);
end
end

```



```

if aperiodica==1
    assignin('base','ac',ac);
else
    assignin('base','ac',aux);
end
assignin('base','cota_ac',cota_ac);
assignin('base','value',value);
assignin('base','pos',pos);
assignin('base','value_lob',value_lob);
assignin('base','pos_lob',pos_lob);
  
```

10.2. Cálculo de la Cota AC para las secuencias CSS

```

function cota_ac=crear_cota_ac(code)

Nemo=evalin('base','Nemo');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
delay=evalin('base','delay');
L=evalin('base','L');
aperiodica=evalin('base','aperiodica');
M=evalin('base','M');
switch code
    case 1
        conca=evalin('base','conca');
        if conca==1
            x=concatenar;
        else
            x=entrelazar;
        end
    case 2
        x=evalin('base','codigo_CSS_modulado');
        L=L*M*Nc*Of; %en muestras
    case 3
        x=evalin('base','codigo_CSS_transductor');
        y=evalin('base','codigo_CSS_modulado');
        L=L*M*Nc*Of; %en muestras
    case 4
        x=evalin('base','codigo_CSS_sin_modular');
        M=evalin('base','M');
end

if aperiodica==0
    for i=1:Nemo
        if code==4
            for j=1:M %repito 3 veces cada secuencia de cada código
                z(j,:,i)=repmat(x(j,:,i),1,3);
            end
        else
            z(i,:)=repmat(x(i,:),1,3);
        end
    end
else
    z=x;
end
  
```

```

if code==1 %fórmula para el caso banda base
  for i=1:Nemo
    ac(i,:)=conv(z(i,:),fliplr(x(i,:)));
    if aperiodica==0
      iz=L*M+1;
      de=3*L*M-1;
      aux(i,:)=ac(i,(iz:de));
      [value(i,:),pos(i,)] =max(aux(i,:));
      ac_aux(i,:)=aux(i,:);
      ac_aux(i,pos(i,:))=0;
      [value_lob(i,:),pos_lob(i,)] =max(abs(ac_aux(i,:)));
    else
      [value(i,:),pos(i,)] =max(ac(i,:));
      ac_aux(i,:)=ac(i,:);
      ac_aux(i,pos(i,:))=0;
      [value_lob(i,:),pos_lob(i,)] =max(abs(ac_aux(i,:)));
    end
    cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
  end
  cota_ac=max(cota_ac_aux);
end
if code==2
  for i=1:Nemo
    ac(i,:)=conv(z(i,:),fliplr(x(i,:)));
    if aperiodica==0
      iz=L+1+Nc*Of;
      de=3*L-1-Nc*Of;
      aux(i,:)=ac(i,(iz:de));
      [value(i,:),pos(i,)] =max(aux(i,:));
      ac_aux(i,:)=aux(i,:);
      ac_aux(i,[pos(i,)-
round(Of*Nc):pos(i,)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
      [value_lob(i,:),pos_lob(i,)] =max(abs(ac_aux(i,:))); %Max
lob. lateral
    else
      [value(i,:),pos(i,)] =max(abs(ac(i,:)));
      ac_aux(i,:)=ac(i,:);
      ac_aux(i,[pos(i,)-
round(Of*Nc):pos(i,)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
      [value_lob(i,:),pos_lob(i,)] =max(abs(ac_aux(i,:))); %Max
lob. lateral
    end
    cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
  end
  cota_ac=max(cota_ac_aux);
end
if code==3
  for i=1:Nemo
    ac(i,:)=conv(z(i,:),fliplr(y(i,:)));
    if aperiodica==0
      iz=L+delay+1+delay;
      de=3*L+delay-1-delay;
      aux(i,:)=ac(i,(iz:de));
      [value(i,:),pos(i,)] =max(aux(i,:));
      ac_aux(i,:)=aux(i,:);
      ac_aux(i,[pos(i,)-
round(Of*Nc):pos(i,)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
      [value_lob(i,:),pos_lob(i,)] =max(abs(ac_aux(i,:))); %Max
lob. lateral
    else
      [value(i,:),pos(i,)] =max(abs(ac(i,:)));
  
```

```

        ac_aux(i,:)=ac(i,:);
        ac_aux(i,[pos(i,:)-
round(Of*Nc):pos(i,)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
        [value_lob(i,:),pos_lob(i,)] = max(abs(ac_aux(i,:))); %Max
lob. lateral
    end
        cota_ac_aux(i,:)=value_lob(i,)/value(i,:);
    end
    cota_ac=max(cota_ac_aux);
end

if code==4
    ac=zeros(Nemo,2*length(z(1,:,1))-1);
    for i=1:Nemo
        for j=1:M
            ac_parcial(j,:,i)=xcorr2(z(j,:,i),x(j,:,i));
            ac(i,:)=ac(i,)+ac_parcial(j,:,i);
        end
        if aperiodica==0
            medio=round(length(ac)/2);
            [valor,duracion]=max(ac(i,:));
            L=round((duracion));
            aux(i,:)=ac(i,((medio-L+1):(medio+L-1)));
            [value(i,:),pos(i,)] = max(aux(i,:));
            ac_aux(i,:)=aux(i,:);
            ac_aux(i,pos(i,:))=0;
            [value_lob(i,:),pos_lob(i,)] = max(abs(ac_aux(i,:)));
            assignin('base','desp',duracion);
        else
            [value(i,:),pos(i,)] = max(ac(i,:));
            ac_aux(i,:)=ac(i,:);
            ac_aux(i,pos(i,:))=0;
            [value_lob(i,:),pos_lob(i,)] = max(abs(ac_aux(i,:)));
        end
        cota_ac_aux(i,:)=value_lob(i,)/value(i,:);
    end
    assignin('base','ac_parcial',ac_parcial);
    cota_ac=max(cota_ac_aux);
end

if aperiodica==1
    assignin('base','ac',ac);
else
    assignin('base','ac',aux);
end
assignin('base','cota_ac',cota_ac);
assignin('base','value',value);
assignin('base','pos',pos);
assignin('base','value_lob',value_lob);
assignin('base','pos_lob',pos_lob);

```

10.3. Cálculo de la Cota AC para secuencias LS

```

function cota_ac=crear_cota_ac(code)

Nemo=evalin('base','Nemo');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
dentro=evalin('base','dentro');
W=evalin('base','W');
L=evalin('base','L');
switch code
  case 1
    x=evalin('base','codigo_LS_sin_modular');
  case 2
    x=evalin('base','codigo_LS_modulado');
    W=W*Nc*Of; %en muestras
    L=L*Nc*Of; %en muestras
  case 3
    x=evalin('base','codigo_LS_transductor');
    y=evalin('base','codigo_LS_modulado');
    W=W*Nc*Of;
    L=L*Nc*Of; %en muestras
end

if code==1 %fórmula para el caso banda base
  for i=1:Nemo
    ac(i,:)=conv(x(i,:),fliplr(x(i,:)));
    [value(i,:),pos(i,:)]=max(ac(i,:));
    ac_aux(i,:)=ac(i,:);
    ac_aux(i,pos(i,:))=0;
    if dentro==1
      izquierda=pos(i,)-W;
      derecha=pos(i,)+W;

[value_lob(i,:),pos_lob(i,:)]=max(abs(ac_aux(i,izquierda:derecha)));
      pos_lob(i,:)=pos_lob(i,)+izquierda-1;
    else
      [value_lob(i,:),pos_lob(i,:)]=max(abs(ac_aux(i,:)));
    end
    end
    cota_ac_aux(i,:)=value_lob(i,)/value(i,);
  end
  cota_ac=max(cota_ac_aux);
end

if code==2
  for i=1:Nemo
    ac(i,:)=conv(x(i,:),fliplr(x(i,:)));
    [value(i,:),pos(i,:)]=max(abs(ac(i,:)));
    if dentro==1
      izquierda=pos(i,)-W-1;
      derecha=length(ac(i,))-pos(i,)-W;
      ac_aux(i,:)=[zeros(1,izquierda) ac(i,pos(i,))-
W:pos(i,)+W) zeros(1,derecha)];
    else
      ac_aux(i,:)=ac(i,);
    end
  end
end

```

```

    end
    ac_aux(i, [pos(i, :)-
round(Of*Nc):pos(i, :)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
    [value_lob(i, :), pos_lob(i, :)] = max(abs(ac_aux(i, :))); %Max lob.
  lateral
    cota_ac_aux(i, :)=value_lob(i, :)/value(i, :);
  end
  cota_ac=max(cota_ac_aux);
end

if code==3
  for i=1:Nemo
    ac(i, :)=conv(x(i, :),fliplr(y(i, :)));
    [value(i, :), pos(i, :)] = max(abs(ac(i, :)));
    if dentro==1
      izquierda=pos(i, :)-W-1;
      derecha=length(ac(i, :))-pos(i, :)-W;
      ac_aux(i, :)= [zeros(1, izquierda) ac(i, pos(i, :)-
W:pos(i, :)+W) zeros(1, derecha)];
    else
      ac_aux(i, :)=ac(i, :);
    end
    ac_aux(i, (pos(i, :)-
round(Of*Nc):pos(i, :)+round(Of*Nc)))=zeros(1,2*Of*Nc+1);
    [value_lob(i, :), pos_lob(i, :)] = max(abs(ac_aux(i, :))); %Max lob.
  lateral
    cota_ac_aux(i, :)=value_lob(i, :)/value(i, :);
  end
  cota_ac=max(cota_ac_aux);
end

assignin('base', 'ac', ac);
assignin('base', 'cota_ac', cota_ac);
assignin('base', 'value', value);
assignin('base', 'pos', pos);
assignin('base', 'value_lob', value_lob);
assignin('base', 'pos_lob', pos_lob);

```

10.4. Cálculo de la Cota AC para secuencias Zadoff-Chu

```

function cota_ac=crear_cota_ac(code)

Nemo=evalin('base', 'Nemo');
Nc=evalin('base', 'Nc');
Of=evalin('base', 'Of');
delay=evalin('base', 'delay');
L=evalin('base', 'L');
aperiodica=evalin('base', 'aperiodica');
switch code
  case 1
    x=evalin('base', 'codigo_zc_sin_modular');
  case 2
    x=evalin('base', 'codigo_zc_modulado');

```

```

    L=L*Nc*Of; %en muestras
  case 3
    x=evalin('base','codigo_zc_transductor');
    y=evalin('base','codigo_zc_modulado');
    L=L*Nc*Of; %en muestras
  end

  if aperiodica==0
    for i=1:Nemo
      z(i,:)=repmat(x(i,:),1,3);
    end
  else
    z=x;
  end

  if code==1 %fórmula para el caso banda base
    for i=1:Nemo
      ac(i,:)=xcorr2(z(i,:),x(i,:));
      ac(i,:)=abs(ac(i,:));
      if aperiodica==0
        iz=L+1;
        de=3*L-1;
        aux(i,:)=ac(i,(iz:de));
        [value(i,:),pos(i,.)]=max(aux(i,:));
        ac_aux(i,:)=aux(i,:);
        ac_aux(i,pos(i,:))=0;
        [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:)));
      else
        [value(i,:),pos(i,.)]=max(ac(i,:));
        ac_aux(i,:)=ac(i,:);
        ac_aux(i,pos(i,:))=0;
        [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:)));
      end
      cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
    end
    cota_ac=max(cota_ac_aux);
  end
  if code==2
    for i=1:Nemo
      ac(i,:)=conv(z(i,:),fliplr(x(i,:)));
      if aperiodica==0
        iz=L+1+Nc*Of;
        de=3*L-1-Nc*Of;
        aux(i,:)=ac(i,(iz:de));
        [value(i,:),pos(i,.)]=max(aux(i,:));
        ac_aux(i,:)=aux(i,:);
        ac_aux(i,[pos(i,.)-
round(Of*Nc):pos(i,.)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
        [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
      else
        [value(i,:),pos(i,.)]=max(abs(ac(i,:)));
        ac_aux(i,:)=ac(i,:);
        ac_aux(i,[pos(i,.)-
round(Of*Nc):pos(i,.)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
        [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
      end
      cota_ac_aux(i,:)=value_lob(i,:)/value(i,:);
    end
    cota_ac=max(cota_ac_aux);
  end

```

```

end
if code==3
  for i=1:Nemo
    ac(i,:)=conv(z(i,:),fliplr(y(i,:)));
    if aperiodica==0
      iz=L+delay+1+delay;
      de=3*L+delay-1-delay;
      aux(i,:)=ac(i,(iz:de));
      [value(i,:),pos(i,.)]=max(aux(i,:));
      ac_aux(i,:)=aux(i,:);
      ac_aux(i,[pos(i,:)-
round(Of*Nc):pos(i,.)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
      [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
    else
      [value(i,:),pos(i,.)]=max(abs(ac(i,:)));
      ac_aux(i,:)=ac(i,:);
      ac_aux(i,[pos(i,:)-
round(Of*Nc):pos(i,.)+round(Of*Nc)])=zeros(1,2*Of*Nc+1);
      [value_lob(i,:),pos_lob(i,.)]=max(abs(ac_aux(i,:))); %Max
lob. lateral
    end
    cota_ac_aux(i,:)=value_lob(i,.) / value(i,.);
  end
  cota_ac=max(cota_ac_aux);
end

if aperiodica==1
  assignin('base','ac',ac);
else
  assignin('base','ac',aux);
end
assignin('base','cota_ac',cota_ac);
assignin('base','value',value);
assignin('base','pos',pos);
assignin('base','value_lob',value_lob);
assignin('base','pos_lob',pos_lob);

```

10.5. Cálculo de la Cota CC de las secuencias Kasami

```

function cota_cc=crear_cota_cc(code)

Nemo=evalin('base','Nemo');
value=evalin('base','value');
aperiodica=evalin('base','aperiodica');
delay=evalin('base','delay');
L=evalin('base','L');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
if Nemo==1
  cc=0;
  cota_cc=0;
  aux=0;

```

```

value_lob_cc=0;
pos_lob_cc=0;
else
  switch code
    case 1
      x=evalin('base','codigo_kasami_sin_modular');
    case 2
      x=evalin('base','codigo_kasami_modulado');
      L=L*Nc*Of;
    case 3
      x=evalin('base','codigo_kasami_transductor');
      y=evalin('base','codigo_kasami_modulado');
      L=L*Nc*Of;
  end

if aperiodica==0
  for i=1:Nemo
    k(i,:)=repmat(x(i,:),1,3);
  end
else
  k=x;
end

if code==1
  pico_AC_ideal=sum(abs(x(1,:)));%Pico AC ideal
  z=1;
  for i=1:Nemo
    for j=i+1:Nemo
      cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
      if aperiodica==0
        iz=L+1;
        de=3*L-1;
        aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
      else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
      end
      z=z+1;
    end
  end

  pico_lat_cc=max(value_lob_cc);
  cota_cc=pico_lat_cc/pico_AC_ideal;
end
if code==2
  z=1;
  for i=1:Nemo
    for j=i+1:Nemo
      cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
      if aperiodica==0
        iz=L+1;
        de=3*L-1;
        aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
      else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
    end
  end
end

```



```

        end
        z=z+1;
    end
    end
    pico_lat_cc=max(value_lob_cc);
    cota_cc=pico_lat_cc/max(value);
end
if code==3
    z=1;
    for i=1:Nemo
        for j=i+1:Nemo
            cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
            if aperiodica==0
                iz=L+delay+1;
                de=3*L+delay-1;
                aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
            else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
            end
            value_ac=max(value(i,:),value(j,:));
            cota_cc_aux(z,:)=value_lob_cc(z,:)/value_ac;
            z=z+1;
        end
    end
    cota_cc=max(cota_cc_aux);
end
end

if aperiodica==0
    assignin('base','cc',aux);
else
    assignin('base','cc',cc);
end
assignin('base','cota_cc',cota_cc);
assignin('base','value_lob_cc',value_lob_cc);
assignin('base','pos_lob_cc',pos_lob_cc);

```

10.6. Cálculo de la Cota CC de las secuencias CSS

```

function cota_cc=crear_cota_cc(code)

Nemo=evalin('base','Nemo');
value=evalin('base','value');
aperiodica=evalin('base','aperiodica');
delay=evalin('base','delay');
M=evalin('base','M');
L=evalin('base','L');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
if Nemo==1
    cc=0;
    cota_cc=0;

```

```

aux=0;
value_lob_cc=0;
pos_lob_cc=0;
else
  switch code
    case 1
      conca=evalin('base','conca');
      if conca==1
        x=concatenar;
      else
        x=entrelazar;
      end
    case 2
      x=evalin('base','codigo_CSS_modulado');
      L=L*M*Nc*Of;
    case 3
      x=evalin('base','codigo_CSS_transductor');
      y=evalin('base','codigo_CSS_modulado');
      L=L*M*Nc*Of;
    case 4
      x=evalin('base','codigo_CSS_sin_modular');
      M=evalin('base','M');
  end

if aperiodica==0
  for i=1:Nemo
    if code==4
      for j=1:M %repito 3 veces cada secuencia de cada código
        k(j,:,i)=repmat(x(j,:,i),1,3);
      end
    else
      k(i,:)=repmat(x(i,:),1,3);
    end
  end
else
  k=x;
end

if code==1
  pico_AC_ideal=sum(abs(x(1,:)));%Pico AC ideal
  z=1;
  for i=1:Nemo
    for j=i+1:Nemo
      cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
      if aperiodica==0
        iz=L+1;
        de=3*L-1;
        aux(z,:)=cc(z,(iz:de));
      end
    end
  end

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
else
  [value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
end
  z=z+1;
end

end
pico_lat_cc=max(value_lob_cc);
cota_cc=pico_lat_cc/pico_AC_ideal;
end

```

```

if code==2
  z=1;
  for i=1:Nemo
    for j=i+1:Nemo
      cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
      if aperiodica==0
        iz=L+1;
        de=3*L-1;
        aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
      else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
      end
      z=z+1;
    end
  end
  pico_lat_cc=max(value_lob_cc);
  cota_cc=pico_lat_cc/max(value);
end
if code==3
  z=1;
  for i=1:Nemo
    for j=i+1:Nemo
      cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
      if aperiodica==0
        iz=L+delay+1;
        de=3*L+delay-1;
        aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
      else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
      end
      value_ac=max(value(i,:),value(j,:));
      cota_cc_aux(z,:)=value_lob_cc(z,:)/value_ac;
      z=z+1;
    end
  end
  cota_cc=max(cota_cc_aux);
end
end
if code==4
  L=evalin('base','L');
  z=1;
  for i=1:Nemo
    %cc(z,:)=zeros(z,2*L-1);
    for n=i+1:Nemo
      prueba=0;
      for j=1:M
        cc_parcial(z,:,j)=xcorr2(k(j,:,i),x(j,:,n));
%correlacion cc de la secuencia j de cada código
        %cc(z,:)=cc(z,:)+cc_parcial(z,:,j);
        prueba=prueba+cc_parcial(z,:,j);
      end
      cc(z,:)=prueba;
      [value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
      z=z+1;
    end
  end

```

```

    end
    pico_lat_cc=max(value_lob_cc);
    cota_cc=pico_lat_cc/max(value);
    if Nemo~=1
      assignin('base','cc_parcial',cc_parcial);
    end
  end

if aperiodica==0
  assignin('base','cc',aux);
else
  assignin('base','cc',cc);
end
assignin('base','cota_cc',cota_cc);
assignin('base','value_lob_cc',value_lob_cc);
assignin('base','pos_lob_cc',pos_lob_cc);

```

10.7. Cálculo de la Cota CC de las secuencias LS

```

function cota_cc=crear_cota_cc(code)

Nemo=evalin('base','Nemo');
value=evalin('base','value');
dentro=evalin('base','dentro');
W=evalin('base','W');
L=evalin('base','L');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
if Nemo==1
  cc=0;
  cota_cc=0;
  aux=0;
  value_lob_cc=0;
  pos_lob_cc=0;
else
  switch code
    case 1
      x=evalin('base','codigo_LS_sin_modular');
    case 2
      x=evalin('base','codigo_LS_modulado');
      W=W*Nc*Of; %en muestras
      L=L*Nc*Of;
    case 3
      x=evalin('base','codigo_LS_transductor');
      y=evalin('base','codigo_LS_modulado');
      W=W*Nc*Of;
      L=L*Nc*Of;
  end

  if code==1
    pico_AC_ideal=sum(abs(x(1,:)));%Pico AC ideal
    z=1;
    for i=1:Nemo
      for j=i+1:Nemo
        cc(z,:)=conv(x(i,:),fliplr(x(j,:)));

```

```

        if dentro==1
            pos=round(length(cc(z,:))/2);
            izquierda=pos-W;
            derecha=pos+W;

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,izquierda:derecha)));
            pos_lob_cc(z,:)=pos_lob_cc(z,)+izquierda-1;
        else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
            end
            z=z+1;
        end

        end
        pico_lat_cc=max(value_lob_cc);
        cota_cc=pico_lat_cc/pico_AC_ideal;
    end

    if code==2
        z=1;
        for i=1:Nemo
            for j=i+1:Nemo
                cc(z,:)=conv(x(i,:),fliplr(x(j,:)));
                if dentro==1
                    pos=round(length(cc(z,:))/2);
                    izquierda=pos-W-1;
                    derecha=length(cc(z,:))-pos-W;
                    ac_aux(z,:)=zeros(1,izquierda) cc(z,pos-W:pos+W)
zeros(1,derecha)];

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(ac_aux(z,:)));
                else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
                    end
                    z=z+1;
                end
            end
            pico_lat_cc=max(value_lob_cc);
            cota_cc=pico_lat_cc/max(value);
        end

        if code==3
            z=1;
            for i=1:Nemo
                for j=i+1:Nemo
                    cc(z,:)=conv(x(i,:),fliplr(x(j,:)));
                    if dentro==1
                        pos=round(length(cc(z,:))/2);
                        izquierda=pos-W-1;
                        derecha=length(cc(z,:))-pos-W;
                        ac_aux(z,:)=zeros(1,izquierda) cc(z,pos-W:pos+W)
zeros(1,derecha)];

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(ac_aux(z,:)));
                    else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
                        end
                        value_ac=max(value(i,:),value(j,:));
                    end
                end
            end
        end
    end

```

```

        cota_cc_aux(z,:)=value_lob_cc(z,:)/value_ac;
        z=z+1;
    end
end
cota_cc=max(cota_cc_aux);
end
end
assignin('base','cc',cc);
assignin('base','cota_cc',cota_cc);
assignin('base','value_lob_cc',value_lob_cc);
assignin('base','pos_lob_cc',pos_lob_cc);

```

10.8. Cálculo de la Cota CC de las secuencias Zadoff-Chu

```

function cota_cc=crear_cota_cc(code)

Nemo=evalin('base','Nemo');
value=evalin('base','value');
aperiodica=evalin('base','aperiodica');
delay=evalin('base','delay');
L=evalin('base','L');
Nc=evalin('base','Nc');
Of=evalin('base','Of');
if Nemo==1
    cc=0;
    cota_cc=0;
    aux=0;
    value_lob_cc=0;
    pos_lob_cc=0;
else
    switch code
        case 1
            x=evalin('base','codigo_zc_sin_modular');
        case 2
            x=evalin('base','codigo_zc_modulado');
            L=L*Nc*Of;
        case 3
            x=evalin('base','codigo_zc_transductor');
            y=evalin('base','codigo_zc_modulado');
            L=L*Nc*Of;
    end

if aperiodica==0
    for i=1:Nemo
        k(i,:)=repmat(x(i,:),1,3);
    end
else
    k=x;
end

if code==1 %banda base
    pico_AC_ideal=sum(abs(x(1,:)));%Pico AC ideal
    z=1;
    for i=1:Nemo

```

```

    for j=i+1:Nemo
      %cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
      cc(z,:)=xcorr2(k(i,:),x(j,:));
      cc(z,:)=abs(cc(z,:));
      if aperiodica==0
        iz=L+1;
        de=3*L-1;
        aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
      else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
      end
      z=z+1;
    end

    end
    pico_lat_cc=max(value_lob_cc);
    cota_cc=pico_lat_cc/pico_AC_ideal;
  end
  if code==2
    z=1;
    for i=1:Nemo
      for j=i+1:Nemo
        cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
        if aperiodica==0
          iz=L+1;
          de=3*L-1;
          aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
        else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
        end
        z=z+1;
      end
    end
    pico_lat_cc=max(value_lob_cc);
    cota_cc=pico_lat_cc/max(value);
  end
  if code==3
    z=1;
    for i=1:Nemo
      for j=i+1:Nemo
        cc(z,:)=conv(k(i,:),fliplr(x(j,:)));
        if aperiodica==0
          iz=L+delay+1;
          de=3*L+delay-1;
          aux(z,:)=cc(z,(iz:de));

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(aux(z,:)));
        else

[value_lob_cc(z,:),pos_lob_cc(z,:)]=max(abs(cc(z,:)));
        end
        value_ac=max(value(i,:),value(j,:));
        cota_cc_aux(z,:)=value_lob_cc(z,:)/value_ac;
        z=z+1;
      end
    end
  end

```

```

        end
        cota_cc=max(cota_cc_aux);
    end
end

if aperiodica==0
    assignin('base','cc',aux);
else
    assignin('base','cc',cc);
end
assignin('base','cota_cc',cota_cc);
assignin('base','value_lob_cc',value_lob_cc);
assignin('base','pos_lob_cc',pos_lob_cc);

```

10.9. Calcular SMR

```

function [SMR_avg_code,SMR_std_code]=crear_SMR

Nemo=evalin('base','Nemo');
fse=evalin('base','fse');
Tgap=evalin('base','Tgap');
Tshift=evalin('base','Tshift');
Of=evalin('base','Of');
Nc=evalin('base','Nc');
L=evalin('base','L');
Te=L*Nc*Of; %en muestras;Tgap=Tgap*fse;Tshift=Tshift*fse; %los paso a
muestras
Tgap=Tgap*fse;
Tshift=Tshift*fse; %los paso a muestras
cambia_muestra=evalin('base','cambia_muestra');
if cambia_muestra==0
    muestras=Of*Nc; %esto puede variar
else
    muestras=evalin('base','muestras');
end
Nrep=evalin('base','Nrep');
delay=evalin('base','delay'); %retardo provocado por el transductor
transductor=evalin('base','transductor');
selector=evalin('base','selector');
if selector==4
    code=evalin('base','received_signal_sin_efectos');
else
    code=evalin('base','received_signal_con_efectos');
end
codigo_zc_modulado=evalin('base','codigo_zc_modulado');

for i=1:Nemo %Generacion de la correlacion cruzada de los codigos y la
señal recibida
    ac(i,:)=xcorr2(code,codigo_zc_modulado(i,:));
end
assignin('base','ac',ac);

if transductor~=2 %anulo el delay
    delay=0;
end

```



```

for j=1:Nemo %Códigos
  for i=1:Nrep %Ventanas
    pico(j,i)=delay+Te+(i-1)*(Te+Tgap)+(j-1)*Tshift;
    if (pico(j,i)-round((Te+Tgap)/2))<0
      inicio_ventana(j,i)=1;
    else
      inicio_ventana(j,i)=(pico(j,i)-round((Te+Tgap)/2));
    end
    fin_ventana(j,i)=(pico(j,i)+round((Te+Tgap)/2));
    ac_ventana=ac(j,(inicio_ventana(j,i):fin_ventana(j,i)));
    [value(j,i),pos(j,i)]=max(abs(ac_ventana));
    ac_ventana=[ac_ventana(1:(pos(j,i)-muestras)
zeros(1,2*muestras+1)
ac_ventana(pos(j,i)+muestras:length(ac_ventana))];
    [value_lob_cc(j,i),pos_lob_cc(j,i)]=max(abs(ac_ventana));
    dif(j,i)=pos_lob_cc(j,i)-pos(j,i);
    SMR_aux(j,i)=value_lob_cc(j,i)/value(j,i); %filas=ventana
, columnas=codigo
    pos(j,i)=pos(j,i)+inicio_ventana(j,i)-1;
    pos_lob_cc(j,i)=pos_lob_cc(j,i)+inicio_ventana(j,i)-1;
  end
  SMR_avg_code(j,:)=mean(SMR_aux(j,:));
  SMR_std_code(j,:)=std(SMR_aux(j,:));
end

assignin('base','pos',pos);
assignin('base','value_lob_cc',value_lob_cc);
assignin('base','pos_lob_cc',pos_lob_cc);
assignin('base','value',value);
assignin('base','dif',dif);
assignin('base','SMR_avg_code',SMR_avg_code);
assignin('base','SMR_std_code',SMR_std_code);
assignin('base','inicio_ventana',inicio_ventana);
assignin('base','fin_ventana',fin_ventana);

```

10.10. Simulación paramétrica del ruido

```

function crear_s_parametrica

Nrep=evalin('base','Nrep');
Nemo=evalin('base','Nemo');
SNR_desde=evalin('base','SNR_desde');
SNR_hasta=evalin('base','SNR_hasta');
salto=evalin('base','salto');
delay=evalin('base','delay');
codigo_zc_modulado=evalin('base','codigo_zc_modulado');
codigo_zc_emitido=evalin('base','codigo_zc_emitido');
fse=evalin('base','fse');
Tshift=evalin('base','Tshift');
Tgap=evalin('base','Tgap');
Te=evalin('base','Te');
if Nemo==1
    received_signal_sin_efectos=codigo_zc_emitido;
else
    received_signal_sin_efectos=sum(codigo_zc_emitido);

```

```

end
SNR=SNR_desde:salto:SNR_hasta;

for j=1:Nemo %Códigos

ac_ideal(j,:)=conv(received_signal_sin_efectos,fliplr(codigo_zc_modula
do(j,:)));
    for i=1:Nrep %Ventanas
        pico(j,i)=round(delay+fse*Te+(i-1)*fse*(Te+Tgap)+(j-
1)*fse*Tshift);
        inicio_ventana(j,i)=pico(j,i)-fse*((Te/2)+(Tgap/2));
        fin_ventana(j,i)=pico(j,i)+fse*((Te/2)+(Tgap/2));

ac_ventana_ideal=ac_ideal(j,(inicio_ventana(j,i):fin_ventana(j,i)));
        [value_ideal(i,j),pos_ideal(i,j)]=max(abs(ac_ventana_ideal));
        pos_ideal(i,j)=pico(j,i);
    end
end

for k=1:length(SNR)

    if Nemo==1
        signal_parametrica(k,:)=codigo_zc_emitido;
    else
        signal_parametrica(k,:)=sum(codigo_zc_emitido);
    end

signal_parametrica(k,:)=awgn(signal_parametrica(k,:),SNR(k),'measured'
);
    for i=1:Nemo %Generacion de la correlacion cruzada de los codigos
y la señal recibida

ac_real(i,k,:)=conv(signal_parametrica(k,:),fliplr(codigo_zc_modulado(
i,:)));
    end
    for j=1:Nemo %Códigos
        RMSE(j,k,:)=0;
        for i=1:Nrep %Ventanas
            pico(j,i)=round(delay+fse*Te+(i-1)*fse*(Te+Tgap)+(j-
1)*fse*Tshift);
            inicio_ventana(j,i)=pico(j,i)-fse*((Te/2)+(Tgap/2));
            fin_ventana(j,i)=pico(j,i)+fse*((Te/2)+(Tgap/2));

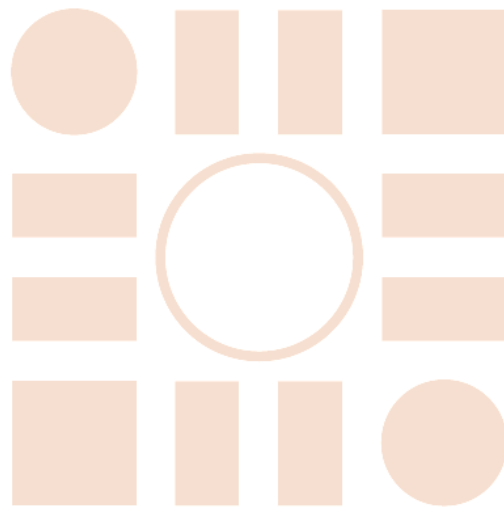
ac_ventana_real=ac_real(j,k,(inicio_ventana(j,i):fin_ventana(j,i)));

[value_real(i,j,k),pos_real(i,j,k)]=max(abs(ac_ventana_real));
            %dif(i,j,k)=pico(j,i)-pos_real(i,j,k); %esto es para saber
donde está cada pico real en la ventana
            dif(i,j,k)=pico(j,i)-pos_real(i,j,k)-
inicio_ventana(j,i)+1;
            pos_real(i,j,k)=pico(j,i)+dif(i,j,k);
            RMSE(j,:)=RMSE(j,:)+(pos_ideal(i,j)-pos_real(i,j,k))^2;
        end
    end
end
for i=1:Nemo
    RMSE(i,:)=sqrt(RMSE(i,:))/(Nrep*fse);
    p(i,:)=plot(SNR,RMSE(i,:));
    ylabel('RMSE [ms]')
    xlabel('SNR [dB]')
end

```

```
    hold on
end
for i=1:Nemo %crear palabra para la leyenda
    if Nemo < 10
        palabra(i,:)=['C',num2str(i)]; %2 caracteres
    else
        if i < 10
            palabra(i,:)=['C',num2str(i),' ']; %3 caracteres
        else
            palabra(i,:)=['C',num2str(i)]; %3 caracteres
        end
    end
end
legend(p,palabra)
```

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR



Universidad
de Alcalá