



UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

**“DISEÑO Y DESARROLLO DEL SISTEMA DE
ENTRENAMIENTO DREAMBLUE UTILIZANDO
LA METODOLOGÍA SPIES”**

TESIS

**PARA OBTENER EL TÍTULO DE
INGENIERO EN ELECTRÓNICA**

PRESENTA

EDGAR MANUEL CANO CRUZ

DIRECTOR DE TESIS

ING. HERIBERTO ILDEFONSO HERNÁNDEZ MARTÍNEZ

HUAJUAPAN DE LEÓN, OAX.; DICIEMBRE DE 2010

**Tesis presentada el 10 de Diciembre de 2010
ante los siguientes sinodales:**

**M.C. José Antonio Moreno Espinosa
Dr. Iván Antonio García Pacheco
C. Dr. Alejandro Ernesto Ramírez González**

Director de Tesis:

Ing. Heriberto Ildefonso Hernández Martínez

Dedicatoria

A mi madre Yolanda.

Manuel.

Agradecimientos

Agradezco a Heriberto I. Hernández Martínez por aportar sus conocimientos y experiencias, y sobre todo por brindarme su amistad y paciencia a lo largo del trabajo de tesis.

A los profesores José Antonio Moreno Espinosa, Iván Antonio García Pacheco y Alejandro Ernesto Ramírez González, cuyas sugerencias y observaciones contribuyeron a mejorar el documento de tesis.

A mis amigos: Iván Antonio García Pacheco por su contribución en la metodología para el desarrollo del trabajo de tesis; José Antonio Juárez Abad por la imprescindible aportación en la investigación y desarrollo del módulo DTE; Antonio Orantes Molina por su comprensión y motivación en la investigación de nuevas tecnologías; José Miguel Torres Garfías y Amilcar Díaz Martínez por sus valiosas contribuciones en la revisión al sistema final; Noé Coronel García por su apoyo en la elaboración del video tutorial para el uso del sistema DreamBlue y a Dante Raúl Vásquez Hernández por su ayuda desinteresada y darme las facilidades para realizar las impresiones del documento de tesis.

Quiero otorgar un agradecimiento especial a Carmen Torres Martínez por apoyarme y conseguirme un espacio de trabajo en la realización de mi trabajo de tesis.

Evelyn Gabriela Hernández gracias por tú amor y apoyo en todo momento.

Manuel.

Índice

Dedicatoria.....	v
Agradecimientos	vii
Índice	ix
Lista de tablas	xiii
Lista de figuras.....	xvii
Resumen.....	xxiii
Abstract.....	xxv
1. Introducción	1
1.1. Tecnologías inalámbricas de corto alcance	1
1.2. Planteamiento del problema.....	4
1.3. Antecedentes	5
1.4. Justificación	5
1.5. Hipótesis	5
1.6. Objetivos.....	5
1.6.1. Objetivo general.....	5
1.6.2. Objetivos específicos	5
1.7. Estructura de la tesis	5
2. Tecnología Bluetooth.....	7
2.1. Topología de Bluetooth	7
2.2. Arquitectura de protocolos Bluetooth.....	9
2.3. Núcleo de protocolos Bluetooth	10
2.3.1. Radio Bluetooth	10
2.3.1.1. Ensanchado del espectro	10
2.3.2. Banda base	12
2.3.2.1. Formato de los paquetes BB_PDU	14
2.3.2.2. Operación de búsqueda.....	16

2.3.2.3. Operación de invitación.....	17
2.3.2.4. Tipos de enlaces.....	18
2.3.3. LMP.....	18
2.3.3.1. Formato de los mensajes LM_PDU.....	19
2.3.4. L2CAP.....	22
2.3.4.1. Formato de los paquetes L2CAP.....	23
2.3.5. HCI.....	25
2.3.5.1. Formato de los paquetes de órdenes HCI.....	25
2.3.5.2. Formato de los paquetes de eventos HCI.....	28
2.3.5.3. Formato de los paquetes de datos HCI.....	29
2.3.6. SDP.....	29
2.3.6.1. Formato de los paquetes SDP_PDU.....	30
2.4. Capa de reemplazo de cables.....	31
2.4.1. RFCOMM.....	31
2.4.1.1. Formato de los paquetes RFCOMM.....	32
2.5. Capa de protocolos de telefonía.....	33
2.5.1. Protocolo TCS-BIN.....	33
2.5.2. Órdenes AT.....	34
2.5.2.1. Formatos de los paquetes de órdenes AT.....	34
3. Diseño y Desarrollo del Sistema DreamBlue.....	37
3.1. Metodología SPIES.....	37
3.2. Especificación de requerimientos.....	40
3.2.1. Módulo DTE.....	42
3.2.2. Módulo DCE.....	44
3.2.3. Modos de operación.....	47
3.2.3.1. Modo de Órdenes AT.....	48
3.2.3.2. Modo de Datos.....	59
3.3. Diseño del Sistema.....	62
3.3.1. Diagramas de estructura.....	62
3.3.2. Diagramas de secuencia.....	66
3.3.3. Diagramas de máquinas de estado.....	86
3.4. Desarrollo del Sistema.....	90
3.4.1. Desarrollo electrónico.....	90
3.4.1.1. Desarrollo del módulo DTE.....	90
3.4.1.2. Desarrollo del módulo DCE.....	92
3.4.2. Desarrollo del software.....	94
3.5. Integración.....	96

3.5.1. Integración del módulo DTE	96
3.5.2. Integración del módulo DCE	96
3.5.3. Integración del nodo DreamBlue	97
4. Resultados	99
4.1. Verificación del sistema DreamBlue	100
4.2. Entrega y mantenimiento	107
5. Conclusiones y trabajos futuros	109
Bibliografía	111
Sitios URL	112
A. Anexo DTE	A-1
A.1. Pantalla GLCD	A-1
A.2. Teclado matricial	A-3
A.3. MCU dsPIC30F4013	A-4
B. Anexo DCE	B-1
B.1. Adaptador de RS232 OEMII	B-1
B.2. Dispositivo OEMSPA13i	B-2
C. Anexo órdenes AT	C-1
C.1. Configuración del Nodo	C-1
C.2. Gestión de la Conexión	C-6
C.3. Servicios Suplementarios	C-8
D. Manual de usuario	D-1
D.1. Operaciones básicas sobre los nodos DreamBlue	D-1
D.1.1. Suministro de energía a los nodos DreamBlue	D-1
D.1.2. Operaciones sobre el teclado	D-2
D.1.3. Configuración de la velocidad de transmisión	D-3
D.1.4. Activación del modo conectable	D-4
D.1.5. Activación del modo detectable	D-4
D.1.6. Configuración del enlace	D-5
D.1.7. Establecimiento de seguridad entre los nodos DreamBlue	D-5
D.1.7.1 Activación del modo de emparejamiento	D-5
D.1.7.2 Escritura del código PIN	D-6
D.1.7.3 Unión entre los nodos DreamBlue	D-6
D.1.7.4 Activación del modo seguro	D-7
D.1.8. Sondeo de dispositivos	D-7
D.1.9. Envío de mensaje entre los nodos DreamBlue	D-7
D.2. Solución de errores	D-8

Lista de tablas

Tabla 1.1. Comparación entre las tecnologías ZigBee, Bluetooth y Wi-Fi (IEEE 802.11b).....	3
Tabla 2.1. Evolución de la especificación Bluetooth.....	8
Tabla 2.2. Modos de modulación utilizados en Bluetooth [21].....	11
Tabla 2.3. Características de la transmisión síncrona y asíncrona [21].....	18
Tabla 2.4. Formato de mensajes LM_PDU [2].....	19
Tabla 2.5. Definiciones de los CID [2].....	23
Tabla 2.6. Valores de PSM [2].	24
Tabla 2.7. Datos de las órdenes de señalización L2CAP.....	25
Tabla 2.8. Tipos de órdenes HCI [2].	27
Tabla 2.9. Códigos y parámetros de los eventos HCI [2].	28
Tabla 2.10. Valores para las banderas PB y BC [2].	30
Tabla 2.11. Tipos de SDP_PDU [18].	30
Tabla 2.12. Formato de órdenes AT [8].....	34
Tabla 2.13. Formatos de respuestas a órdenes AT [8].....	35
Tabla 2.14. Tipos de datos usados en las órdenes AT [8].	35
Tabla 3.1 Componentes de las interfaces del módulo DTE.....	42
Tabla 3.2. Componentes de las interfaces del módulo DCE.	44
Tabla 3.3. Combinaciones y significado del LED RGB.....	44
Tabla 3.4. Estructura de modelado del sistema DreamBlue.	47
Tabla 3.5. Métodos no visibles.	48
Tabla 3.6. Métodos y órdenes empleados en el subsistema Configuración del Nodo.	52
Tabla 3.7. Métodos y órdenes empleados por el subsistema Gestión de la Conexión.	55
Tabla 3.8. Métodos y órdenes empleados por el subsistema Servicios Suplementarios.	59
Tabla 3.9. Métodos empleados por el subsistema Gestión de Mensajes.	62
Tabla 3.10. Componentes del módulo DTE.	91
Tabla 3.11. Componentes del módulo DCE.	92

Tabla 3.12. Componentes del dispositivo OEMSPA13i.	92
Tabla 3.13. Componentes del adaptador de RS232 OEMII.	93
Tabla 3.14. Componentes SW empleados.	94
Tabla 4.1. Versiones de actualización del sistema DreamBlue.	107
Tabla 4.2. Costos de las herramientas HW y SW para cada nodo DreamBlue.	107
Tabla A.1. Asignación de terminales para la pantalla GLCD [26].	A-1
Tabla A.2. Características eléctricas de la pantalla GLCD [26].	A-2
Tabla A.3. Características mecánicas de la pantalla GLCD.	A-3
Tabla A.4. Asignación de terminales para teclado matricial [URL7].	A-3
Tabla A.5. Características mecánicas del teclado matricial [URL7].	A-4
Tabla A.6. Principales características del dsPIC30F4013.	A-5
Tabla A.7. Asignación de terminales utilizadas en el MCU dsPIC30F4013.	A-6
Tabla B.1. Terminales del conector DB9 del módulo adaptador de RS232 OEM II [8].	B-1
Tabla B.2. Características mecánicas del adaptador RS232 OEMII [8].	B-2
Tabla B.3. Asignación de las terminales para los conectores J1 y J2 [8].	B-2
Tabla B.4. Características de la antena interna del dispositivo OEMSPA13i.	B-4
Tabla B.5. Consumo de corriente del dispositivo OEMSPA13i [8].	B-5
Tabla C.1. Método Escribir_Configuracion_RS232 (AT*ARMS=).	C-1
Tabla C.2. Método Lectura_Configuracion_RS232 (AT*ARMS?).	C-2
Tabla C.3. Método Escritura_Modo_Conectable (AT*AGCM=).	C-2
Tabla C.4. Método Lectura_Modo_Conectable (AT*AGCM?).	C-3
Tabla C.5. Método Escritura_Modo_Detectable (AT*AGDM=).	C-3
Tabla C.6. Método Lectura_Modo_Detectable (AT*AGDM?).	C-3
Tabla C.7. Método Escritura_Nodo_Local (AT*AGLN=).	C-3
Tabla C.8. Método Leer_Nombre_Local (AT*AGLN?).	C-4
Tabla C.9. Método Escritura_COD_Local (AT*AGLC=).	C-4
Tabla C.10. Método Leer_COD_Local (AT*AGLC?).	C-5
Tabla C.11. Método Leer_BD_ADDR (AT*AILBA?).	C-5
Tabla C.12. Método Restauracion_Nodo (AT&F).	C-5
Tabla C.13. Método Escribir_Perfil_Servidor (AT*ADDSP=).	C-6
Tabla C.14. Método Escribir_Enlace_Multipunto (AT*ADWM=).	C-6
Tabla C.15. Método Leer_Enlace_Multipunto (AT*ADWM?).	C-7
Tabla C.16. Método Escribir_RolMaestroEsclavo (AT*AGMSP=).	C-7
Tabla C.17. Método Conectar_A_Servicio_Serial (AT*ADCP=).	C-7
Tabla C.18. Método Escribir_Perfil_Cliente (AT*ADDCP=).	C-8
Tabla C.19. Método Escribir_Modo_Ahorro (AT*AMPMP=).	C-8
Tabla C.20. Método Leer_Modo_Ahorro (AT*AMPMP?).	C-9
Tabla C.21. Método Sondeo (AT*AGI=).	C-9

Tabla C.22. Método Descubrir_Dispositivos (AT*AGDD=).....	C-9
Tabla C.23. Método Escribir_Modo_Emparejamiento (AT*AGPM=).....	C-10
Tabla C.24. Método Leer_Modo_Emparejamiento (AT*AGPM?).....	C-10
Tabla C.25. Método Escribir_Codigo_PIN (AT*AGFP=).....	C-10
Tabla C.26. Método Leer_Codigo_PIN (AT*AGFP?).....	C-11
Tabla C.27. Método Union (AT*AGPM=).....	C-11
Tabla C.28. Método Desunir (AT*AGPM?).....	C-11
Tabla C.29. Método Leer_Uniones (AT*AGBD?).....	C-11
Tabla C.30. Método Escribir_Modo_Seguridad (AT*AGSM=).....	C-12
Tabla C.31. Método Leer_Modo_Seguridad (AT*AGSM?).....	C-12

Lista de figuras

Figura 1.1. Clases de tecnologías inalámbricas de corto alcance [24].	2
Figura 1.2. Comparación ZigBee, Bluetooth y IEEE 802.11b [24].	3
Figura 1.3. Diagrama general del sistema DreamBlue.	5
Figura 2.1. Topologías Bluetooth [2].	8
Figura 2.2. Arquitectura de protocolos Bluetooth [2].	9
Figura 2.3. Diagrama de un dispositivo Bluetooth con módulo firmware.	10
Figura 2.4. Temporización de las ranuras por paquete [2].	12
Figura 2.5. Estados y subestados de un dispositivo Bluetooth [2].	13
Figura 2.6. Formato de la dirección de un dispositivo Bluetooth [2].	13
Figura 2.7. Formato general de los BB_PDU [2].	14
Figura 2.8. Campos del código de acceso en un BB_PDU [2].	14
Figura 2.9. Campos de la cabecera en un BB_PDU [2].	15
Figura 2.10. Tiempo para paquetes de 1, 3 ó 5 ranuras de longitud [2].	15
Figura 2.11. Campos de la carga en un BB_PDU [2].	16
Figura 2.12. Procedimiento de búsqueda.	17
Figura 2.13. Procedimiento de invitación.	17
Figura 2.14. Mensajes entre dos LM por medio del LMP [2].	19
Figura 2.15. Formato de mensajes LM_PDU [2].	20
Figura 2.16. Diferentes estados de conexión [21].	21
Figura 2.17. L2CAP reside dentro de la capa de enlace de datos.	22
Figura 2.18. Formato de los paquetes L2CAP_PDU [2].	23
Figura 2.19. Paquetes L2CAP_PDU con un canal ACL [2].	24
Figura 2.20. Formato de un paquete de señalización SCO [2].	24
Figura 2.21. Formato de órdenes L2CAP de señalización [2].	25
Figura 2.22. Intercambio de información entre el huésped y el HC [2].	26
Figura 2.23. Formato de los paquetes de órdenes HCI [2].	26

Figura 2.24. Formato de los paquetes de eventos HCI [2].	28
Figura 2.25. Formato de los paquetes de datos ACL [2].	29
Figura 2.26. Formato de los paquetes de datos SCO [2].	29
Figura 2.27. Formato de los paquetes SDP_PDU [18].	31
Figura 2.28. Formato de los paquetes RFCOMM [18].	32
Figura 2.29. Subcampo Dirección [18].	33
Figura 2.30. Control telefónico sobre órdenes AT y sobre el protocolo TCS-BIN.	33
Figura 2.31. Formato de las órdenes AT.	34
Figura 3.1. Relación entre las capas y las fases de SPIES [12].	38
Figura 3.2. Secuencia de las fases de SPIES [12].	40
Figura 3.3. Diagrama de requerimientos generales del sistema DreamBlue.	41
Figura 3.4. Diagrama de caso de uso general para el sistema DreamBlue.	41
Figura 3.5. Diagrama de caso de uso para nodo del sistema DreamBlue.	42
Figura 3.6. Diagrama de caso de uso para el módulo DTE.	43
Figura 3.7. Diagrama de caso de uso para la UI_DTE.	43
Figura 3.8. Diagrama de caso de uso para COM_DTE.	43
Figura 3.9 Diagrama de caso de uso para el módulo DTE.	45
Figura 3.10. Diagrama de caso de uso para la UI_DCE.	45
Figura 3.11. Diagrama de caso de uso para el LED RGB.	46
Figura 3.12. Diagrama de caso de uso para la COM_DCE.	46
Figura 3.13. Diagrama de caso de uso para la CINP.	46
Figura 3.14. Diagrama de caso de uso para los modos de operación.	47
Figura 3.15. Diagrama de caso de uso para el modo de Órdenes AT.	49
Figura 3.16. Diagrama de caso de uso para el subsistema Configuración del Nodo.	50
Figura 3.17. Diagrama de caso de uso para la generalización Configuración RS232.	51
Figura 3.18. Diagrama de caso de uso para la generalización Modo Conectable.	51
Figura 3.19. Diagrama de caso de uso para la generalización Modo Detectable.	51
Figura 3.20. Diagrama de caso de uso para la generalización Nombre y Clase.	52
Figura 3.21. Diagrama de caso de uso para la generalización Restaurar Nodo.	52
Figura 3.22. Diagrama de caso de uso para el subsistema Gestión de la Conexión.	54
Figura 3.23. Diagrama de caso de uso para la generalización Modo Servidor.	54
Figura 3.24. Diagrama de caso de uso para la generalización Modo Cliente.	55
Figura 3.25. Diagrama de caso de uso para el subsistemas Servicios Suplementarios.	57
Figura 3.26. Diagrama de caso de uso para la generalización Modo Ahorro.	57
Figura 3.27. Diagrama de caso de uso para la generalización Modo Sondeo.	58
Figura 3.28. Diagrama de caso de uso para la generalización Modo Seguridad.	58
Figura 3.29. Intercambio entre modos de operación de los nodos.	59
Figura 3.30. Diagrama de caso de uso para el Modo de Datos.	59

Figura 3.31. Diagrama de caso de uso para el subsistemas Gestión de Mensajes.	60
Figura 3.32. Diagrama de caso de uso para la generalización Modo de Recepción.	61
Figura 3.33. Diagrama de caso de uso para la generalización Modo de Transmisión.	61
Figura 3.34. Diagrama de estructura del nodo.	62
Figura 3.35. Diagrama de estructura general de cada nodo DreamBlue.	63
Figura 3.36. Diagrama de estructura del subsistema Configuración del Nodo.	64
Figura 3.37. Diagrama de estructura del subsistema Gestión de la Conexión.	64
Figura 3.38. Diagrama de estructura del subsistema Servicios Suplementarios.	65
Figura 3.39. Diagrama de estructura del subsistema Gestión de Mensajes.	65
Figura 3.40. Diagrama de secuencia para el Escenario 1.	66
Figura 3.41. Diagrama de secuencia para el Escenario 2.	67
Figura 3.42. Diagrama de secuencia para el Escenario 3.	68
Figura 3.43. Diagrama de secuencia para el Escenario 4.	69
Figura 3.44. Diagrama de secuencia para el Escenario 5.	70
Figura 3.45. Diagrama de secuencia para el Escenario 6.	71
Figura 3.46. Diagrama de secuencia para el Escenario 7.	72
Figura 3.47. Diagrama de secuencia para el Escenario 8.	73
Figura 3.48. Diagrama de secuencia para el Escenario 9.	74
Figura 3.49. Diagrama de secuencia para el Escenario 10.	75
Figura 3.50. Diagrama de secuencia para el Escenario 11.	76
Figura 3.51. Diagrama de secuencia para el Escenario 12.	77
Figura 3.52. Diagrama de secuencia para el Escenario 13.	78
Figura 3.53. Diagrama de secuencia para el Escenario 14.	79
Figura 3.54. Diagrama de secuencia para el Escenario 15.	80
Figura 3.55. Diagrama de secuencia para el Escenario 16.	81
Figura 3.56. Diagrama de secuencia para el Escenario 17.	82
Figura 3.57. Diagrama de secuencia para el Escenario 18.	83
Figura 3.58. Diagrama de secuencia para el Escenario 19.	84
Figura 3.59. Diagrama de secuencia para el Escenario 20.	85
Figura 3.60. Máquina de estados para elegir modo de operación.	86
Figura 3.61. Máquina de estados dentro del subsistema Configuración del Nodo.	86
Figura 3.62. Máquina de estados para el establecimiento de una conexión serial.	87
Figura 3.63. Máquina de estados dentro de la generalización Modo Servidor.	87
Figura 3.64. Máquina de estados dentro del subsistema Servicios Suplementarios.	87
Figura 3.65. Máquina de estados para ingresar al modo de operación de Órdenes AT.	88
Figura 3.66. Máquina de estados para ingresar al modo de configuración RS232.	88
Figura 3.67. Máquina de estados de establecimiento de una conexión serial como Servidor.	89
Figura 3.68. Máquina de estados para el envío de un mensaje.	89

Figura 3.69. Componentes de la UI_DTE	91
Figura 3.70. Componentes de la COM_DTE.	92
Figura 3.71. Componentes de la CINP [8].	93
Figura 3.72. Componentes de la COM_DCE y de la UI_DCE [8].....	93
Figura 3.73. Interfaz del programa Rhapsody.	94
Figura 3.74. . Entorno de desarrollo mikroC for dsPIC.....	95
Figura 3.75. Entorno de diseño y para imágenes GLCD Bitmap generator.	95
Figura 3.76. Interfaz del programador MASTER-PROG.....	95
Figura 3.77. Diagrama de integración del módulo DTE.	96
Figura 3.78. Diagrama de integración del módulo DCE [8].....	97
Figura 3.79. Diagrama de integración del nodo DreamBlue.	97
Figura 4.1. Prototipo final para los nodos del sistema DreamBlue.	99
Figura 4.2. Menú principal de los modos de operación de los nodos DreamBlue.	100
Figura 4.3. Modo de operación de Órdenes AT.	100
Figura 4.4. Modo de operación de Datos.....	101
Figura 4.5. Velocidades de transmisión de los nodos.....	101
Figura 4.6. Respuesta de lectura de la velocidad de transmisión.	101
Figura 4.7. Generalización Nombre y Clase.....	102
Figura 4.8. Escritura de nombre local al nodo.....	102
Figura 4.9. Sondeo de dispositivos.	102
Figura 4.10. Prueba de descubrimiento de dispositivos Bluetooth.....	103
Figura 4.11. Sondeo ejecutado por el teléfono celular Motorola A1200.....	103
Figura 4.12. Configuración del enlace entre los nodos.	103
Figura 4.13. Generalización modo de seguridad.....	104
Figura 4.14. Establecimiento del modo de emparejamiento.....	104
Figura 4.15. Establecimiento del código PIN.....	104
Figura 4.16. Establecimiento del proceso de unión.....	105
Figura 4.17. Establecimiento del modo seguro sobre los nodos.....	105
Figura 4.18. Escritura de un mensaje de texto.....	105
Figura 4.19. Lectura de un mensaje de texto.	106
Figura 4.20. Estados de operación de los nodos DreamBlue.	106
Figura A.1. Pantalla GLCD.	A-2
Figura A.2. Dimensiones externas de la pantalla GLCD [26].....	A-2
Figura A.3. Teclado matricial de 4x4 interruptores.....	A-3
Figura A.4. Dimensiones externas del teclado matricial [URL7].	A-4
Figura A.5. MCU dsPIC30F4013 [19].	A-4
Figura A.6. Distribución de terminales para el MCU dsPIC30F4013 [19].	A-5
Figura B.1 Conectores J1, J2 y J3 del dispositivo OEMSPA13i[8].	B-2

Figura B.2. Componentes del dispositivo OEMSPA13i [8].	B-4
Figura D.1. Suministro de energía de los nodos.	D-1
Figura D.2. Alimentación de los módulos del nodo.	D-2
Figura D.3. Selección de opciones.	D-2
Figura D.4. Acceso y selección de escritura de caracteres.	D-3
Figura D.5. Configuración de la velocidad.	D-3
Figura D.6. Activación del modo conectable.	D-4
Figura D.7. Activación del modo detectable.	D-4
Figura D.8. Configuración de enlace.	D-5
Figura D.9. Activación del modo de emparejamiento.	D-5
Figura D.10. Escritura del código PIN.	D-6
Figura D.11. Escritura de la Bd_Addr para el proceso de unión.	D-6
Figura D.12. Activación del modo seguro.	D-7
Figura D.13. Sondeo de dispositivos.	D-7
Figura D.14. Envío de mensajes de texto entre los nodos DreamBlue.	D-8
Figura D.15. Error interno del nodo DreamBlue.	D-9
Figura D.16. Error de escritura en orden AT.	D-9
Figura D.17. Restauración del módulo DTE.	D-9

Resumen

La tecnología inalámbrica Bluetooth define un conjunto de especificaciones para realizar comunicaciones de corto alcance, con bajo consumo de energía y seguridad en el establecimiento del enlace entre dispositivos, incluso en áreas con actividad electromagnética. Bluetooth utiliza la comunicación serial para transmitir datos en forma binaria mediante un enlace de radiofrecuencia con la finalidad de sustituir el cableado entre dispositivos por ondas de radio.

El sistema DreamBlue tiene como principal objetivo ser una herramienta de entrenamiento para el protocolo de comunicaciones Bluetooth. El sistema se diseñó a través de una metodología para el desarrollo de sistemas empujados con enfoque en la mejora de procesos (SPIES).

Como resultado de este trabajo de tesis se logró un modelado basado en técnicas de cómputo, obteniéndose un sistema flexible destinado al empleo e investigación de nuevas tecnologías inalámbricas.

Abstract

Bluetooth wireless technology defines a set of specifications for short-range communication with low power consumption and security in the exchange of data, even in areas with electromagnetic activity. Bluetooth uses serial communication to transmit binary data via a wireless link, in order to replace the wiring between the devices through radio waves.

The DreamBlue's goal is to be a tool for training Bluetooth protocol. The DreamBlue system was designed by a development methodology for embedded systems with process improvement (SPIES).

As result of this thesis, we have a flexible system based in computational techniques and with the possibility to reuse in the future.

1. Introducción

La interconexión de dispositivos digitales demanda una gran cantidad de cables y de diversos tipos y tamaños de conectores, lo que resulta impráctico y molesto para el usuario final. Con el surgimiento de la tecnología inalámbrica es posible que dichos dispositivos puedan comunicarse a través de ondas de radio [16].

En los últimos años, el creciente desarrollo de tecnologías de comunicaciones inalámbricas ha generado la necesidad de estudiarlas y comprender su funcionamiento. El protocolo de comunicaciones inalámbricas Bluetooth¹ ha apoyado el crecimiento de las tecnologías destinadas a redes de área personal (PAN, *Personal Area Network*), las cuales se definen como redes de corto alcance (aproximadamente 10 m) y de propósito individual [15].

Actualmente, Bluetooth considera un bajo consumo de energía siendo su objetivo principal la transferencia de voz y datos. Bluetooth es una de las tecnologías inalámbricas de corto alcance más rentables para el diseño de pequeños dispositivos portátiles que regularmente son alimentados por baterías [17].

1.1. Tecnologías inalámbricas de corto alcance

De acuerdo a [24], las tecnologías inalámbricas de corto alcance se dividen en:

- *Redes inalámbricas de área local (WLAN, Wireless Local Area Network)*: Las WLAN son un reemplazo o extensión de las redes de área local (LAN, *Local Area Network*).
- *Redes inalámbricas de área personal (WPAN, Wireless Personal Area Network)*: Las WPAN han sido creadas para proporcionar un bajo consumo de potencia entre dispositivos portátiles inalámbricos. Las WPAN se dividen en 3 clases (Figura 1.1):
 - *WPAN de alta velocidad (HR, High-rate)*.
 - *WPAN de velocidad media (MR, Medium-rate)*.
 - *WPAN de baja velocidad (LR, Low-rate)*.

¹ De aquí en adelante se nombrará a la tecnología inalámbrica Bluetooth simplemente como Bluetooth, de la misma manera a la tecnología inalámbrica ZigBee como ZigBee y a la tecnología inalámbrica Wi-Fi como Wi-Fi.

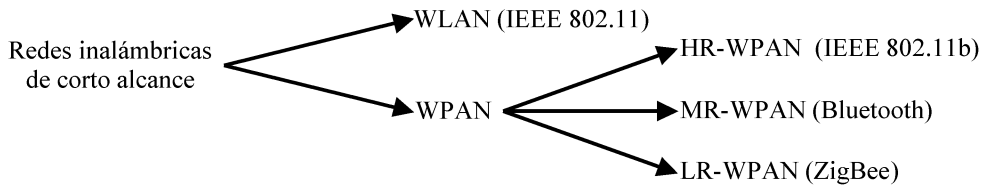


Figura 1.1. Clases de tecnologías inalámbricas de corto alcance [24].

Las tecnologías inalámbricas de corto alcance más importantes en la banda de operación libre ISM (*Industrial, Scientific and Medical*) enfocadas al bajo consumo de potencia, baja complejidad de adaptación en proyectos y bajo costo son (Figura 1.2):

- **ZigBee:** ZigBee™ es considerada una de las tecnologías más importantes en cuanto a implementaciones de bajo consumo de energía a bajas velocidades de transmisión. El estándar fue desarrollado por el grupo de trabajo *ZigBee Alliance*, organización destinada a controlar y supervisar los productos basados en la especificación ZigBee.

El objetivo principal de la comunicación con ZigBee es el intercambio de datos u órdenes entre dispositivos, tales como sensores de temperatura o humedad. Los dispositivos basados en ZigBee operan en las bandas de frecuencia de 860 MHz, 915 MHz y 2.4 GHz [9].

- **Bluetooth:** Bluetooth™ hace referencia a una especificación abierta para la tecnología aplicada en la transmisión de voz y datos con el objetivo principal de permitir la comunicación ininterrumpida de voz y datos a través de enlaces inalámbricos de corto alcance entre dispositivos estacionarios y portátiles (teléfonos celulares, PDAs, teclados inalámbricos, etc.). La frecuencia de operación se encuentra en el rango de los 2.4 a 2.48 GHz con una potencia de salida típica entre 20 dbm y 30 dbm (Versión 1.1 y Versión 1.2 respectivamente).

Las especificaciones de Bluetooth son realizadas por el grupo especial de interés (SIG, *Special Interest Group*) [URL4].

- **IEEE 802.11b:** Wi-Fi™ (IEEE 802.11b) pertenece a la familia del estándar IEEE 802.11² que opera en la banda de frecuencia a 2.4 GHz y es comúnmente utilizada para ofrecer servicio de acceso inalámbrico a Internet (oficinas, escuelas, lugares públicos, etc.). Las redes IEEE 802.11b tienen una potencia de salida típica entre 12 a 18 dbm.

El grupo de trabajo *Wi-Fi Alliance* es el encargado de certificar la interoperabilidad entre dispositivos basados en el estándar 802.11 [16].

La Tabla 1.1 muestra la comparación entre dichas tecnologías inalámbricas para dispositivos portátiles.

² La familia del estándar IEEE 802.11 considera las versiones IEEE 802.11b (frecuencia de operación a 2.4 GHz, alcance hasta 100 m y velocidad de transferencia máxima de 11 Mbps), IEEE 802.11g (frecuencia de operación a 2.4 GHz, alcance hasta 100 m y velocidad de transferencia máxima de 54 Mbps) y IEEE 802.11n (frecuencia de operación a 2.4 GHz, alcance hasta 100 m y velocidad de transferencia máxima de 540 Mbps); sin embargo únicamente IEEE 802.11b se considera una HR-WPAN de bajo consumo de potencia.

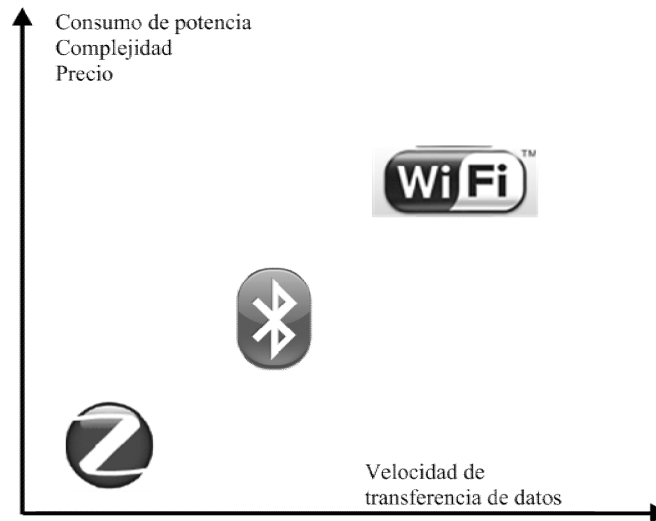


Figura 1.2. Comparación ZigBee, Bluetooth y IEEE 802.11b [24].

Tabla 1.1. Comparación entre las tecnologías ZigBee, Bluetooth y Wi-Fi (IEEE 802.11b).

Tecnología	Velocidad	Rango	Frecuencia	Potencia	Ejemplos de Aplicaciones
ZigBee	20 a 250 Kbps	2 a 100 m	860 MHz, 915 MHz y 2.4 GHz (ISM)	0 a 4 dbm	Redes de sensores inalámbricos
Bluetooth	1 a 3 Mbps	10 a 100 m	2.4 GHz (ISM)	0 a 20 dbm	Transmisión de voz entre un teléfono celular y una computadora
Wi-Fi	1 a 11 Mbps	30 a 100 m	2.4 GHz (ISM)	12 a 18 dbm	Conexión de internet inalámbrica

Una de las principales ventajas de Bluetooth sobre otras tecnologías inalámbricas, es la mejora de ultra bajo consumo de potencia que se estableció en la especificación Bluetooth v4.0 [2], dicha mejora prolonga el ciclo de vida en baterías para relojes digitales, controles remotos, sensores de asistencia médica, etc. La nueva especificación de Bluetooth permite el estudio de nuevas tecnologías enfocadas al procesamiento de señales biomédicas (*Body LAN*) de pacientes para su monitorización remota [22].

Un parámetro importante respecto a la elección de una tecnología inalámbrica para dispositivos portátiles y de bajo consumo es la resistencia a susceptibilidad de ruido. Ante el problema originado por dispositivos con uso de canales fijos de radio, así como aparatos que utilizan la misma banda de operación ISM, Bluetooth emplea la técnica por saltos de frecuencia adaptables (AFH, *Adaptive Frequency Hopping*), que es un mecanismo disponible dentro de su núcleo de protocolos creada para coexistir con entornos de trabajo reales [25].

El reciente estudio del SIG, para emplear normas mundiales en el consumo de energía de forma inteligente (*Smart Energy*), propone el uso de medidores inteligentes (*Smart Meters*) con los cuales se puede monitorear una red de sensores inalámbricos o productos electrodomésticos vía Bluetooth [20]. El SIG está trabajando para asegurar que esta nueva tecnología sea rentable, confiable y segura en redes de área domésticas (HAN, *Home Area Network*) [3]. En HAN y en el campo de la instrumentación médica, Bluetooth presenta una ventaja contra ZigBee, debido a que esta última ha reportado ser extremadamente vulnerable a interferencias ocasionadas por dispositivos Wi-Fi en su vecindad [1], [14].

Otra de las principales ventajas de Bluetooth es la compatibilidad para aplicaciones de bajo costo en teléfonos celulares [17], lo cual garantiza el establecimiento de Bluetooth en el mercado para realizar aplicaciones de sensores inteligentes de forma remota [13], [27].

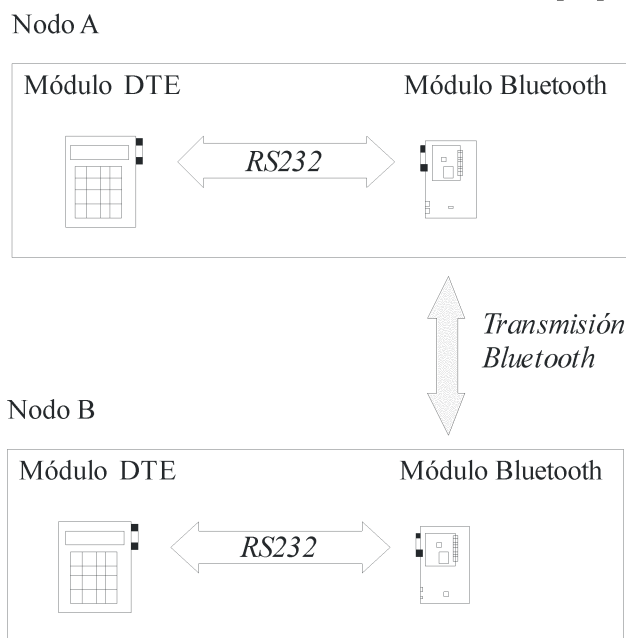
1.2. Planteamiento del problema

El presente trabajo de tesis propone el diseño y desarrollo del Sistema DreamBlue como Herramienta de apoyo al Proceso de Enseñanza-Aprendizaje del Protocolo de Comunicaciones Bluetooth.

El sistema DreamBlue está formado por dos nodos comunicados mediante el protocolo Bluetooth (Figura 1.3); cada nodo consta de un Equipo Terminal de Datos (DTE, *Data Terminal Equipment*) como interfaz con el usuario (UI, *User Interface*) y un módulo Bluetooth³, o Equipo de Terminación del Circuito de Datos (DCE, *Data Circuit-terminating Equipment*), que realiza la conexión Bluetooth para formar una piconet. La comunicación entre el módulo DTE y el módulo DCE se lleva a cabo por medio del protocolo RS232.

El módulo DCE es un dispositivo de fabricación original (OEM, *Original Equipment Manufacturer*) de segunda generación de la firma connectBlue [URL1] y se encarga de transmitir o recibir información en forma serial mediante el protocolo Bluetooth.

El módulo DTE consiste de un teclado matricial de 4x4 interruptores y una pantalla de cristal líquido para gráficos (GLCD, *Graphic Liquid Cristal Display*), medios por los cuales el usuario configura cada nodo y visualiza los mensajes enviados entre estos respectivamente. La configuración de cada nodo se establece a través de un conjunto de órdenes AT⁴ (*Command Attention*) enviadas del módulo DTE al módulo DCE. La implementación del firmware dentro de cada módulo DTE se realiza en el microcontrolador dsPIC30F4013 [19].



³ De ahora en adelante se llamará módulo DCE al módulo Bluetooth.

⁴ En cada dispositivo Bluetooth existe un conjunto de órdenes AT destinado al control y configuración de dichos dispositivos [27].

Figura 1.3. Diagrama general del sistema DreamBlue.

Durante el diseño y desarrollo del sistema DreamBlue se emplea la Metodología de Diseño para Sistemas Empotrados con enfoque en la Mejora del Proceso del Software (SPIES, *Software Process Improvement for Embedded Systems*) [12].

1.3. Antecedentes

Entre las líneas de investigación del Cuerpo Académico de Redes de Instrumentación del Instituto de Electrónica y Mecatrónica (IEM) de la UTM, se encuentra el estudio de protocolos de comunicaciones industriales y de instrumentación electrónica, dando como resultado trabajos realizados en el campo de la instrumentación programable [21] y en el de las comunicaciones industriales [5].

1.4. Justificación

Debido al creciente avance de Bluetooth y la tendencia a eliminar el cableado entre dispositivos portátiles, es importante el aprendizaje del funcionamiento del protocolo de comunicaciones Bluetooth con la finalidad de configurar y enlazar dispositivos que soporten dicha tecnología. Cabe señalar que el mercado ofrece una gran variedad de sistemas y dispositivos Bluetooth, sin embargo no existen herramientas destinadas a su estudio a nivel arquitectura de protocolos.

1.5. Hipótesis

Con el uso de la metodología SPIES durante el diseño y desarrollo del sistema DreamBlue se obtendrá el diseño y modelado de una herramienta de entrenamiento robusta para el Protocolo de Comunicaciones Bluetooth.

1.6. Objetivos

1.6.1. Objetivo general

Diseñar y desarrollar un sistema educativo como herramienta de entrenamiento del protocolo de comunicaciones Bluetooth utilizando la metodología SPIES.

1.6.2. Objetivos específicos

Para cumplir con el objetivo general planteado, se proponen los siguientes objetivos específicos:

- Realizar un marco teórico de la tecnología de comunicaciones Bluetooth.
- Utilizar la metodología SPIES durante el diseño y desarrollo del sistema.
- Diseñar y desarrollar dos nodos de comunicación Bluetooth configurables.
- Configurar una picored con base en los módulos desarrollados.
- Realizar pruebas para asegurar la confiabilidad en el funcionamiento de los módulos elaborados.

1.7. Estructura de la tesis

El presente trabajo de tesis está organizado en cinco capítulos:

El capítulo 1 presenta una introducción al trabajo de tesis.

El capítulo 2 presenta el marco teórico del protocolo de comunicaciones Bluetooth.

El capítulo 3 describe el uso de la metodología SPIES para el diseño e implementación del sistema DreamBlue.

El capítulo 4 muestra los resultados obtenidos de las pruebas realizadas al sistema.

El capítulo 5 plantea las conclusiones y trabajos futuros.

Por último se presentan las referencias bibliográficas y los anexos referentes a los módulos DTE y DCE, a las órdenes AT utilizadas por el sistema DreamBlue y un manual de usuario.

2. Tecnología Bluetooth

Bluetooth define un conjunto de especificaciones para realizar comunicaciones de corto alcance con un bajo consumo de energía, con la finalidad de sustituir el cableado entre dispositivos por ondas de radio; así mismo Bluetooth permite eliminar la mayoría de conexiones cableadas entre dispositivos electrónicos personales [21].

Bluetooth permite el intercambio seguro de datos con una velocidad de transmisión alrededor de 1 Mbps, incluso en áreas con actividad electromagnética; y utiliza la comunicación serial para transmitir datos en forma binaria mediante un enlace de radiofrecuencia (RF). Este modo de transmisión permite tanto comunicación de datos asíncrona como síncrona [15].

A partir de la especificación Bluetooth versión 1.0 publicada en julio de 1999, se han realizado sucesivas actualizaciones como muestra la Tabla 2.1 [URL1].

2.1. Topología de Bluetooth

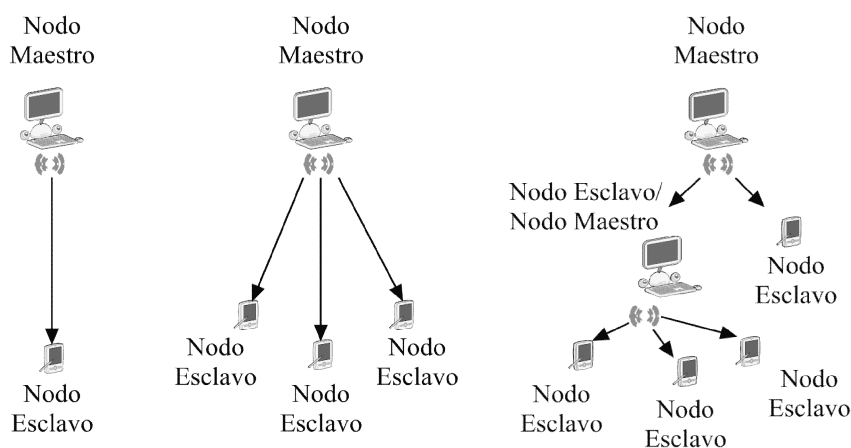
La topología Bluetooth está diseñada bajo el modelo de red de pares (P2P, *peer-to-peer*). Cuando un dispositivo se encuentra dentro del rango de otro dispositivo, automáticamente se forma un enlace de comunicaciones, sin embargo la comunicación no se establece de manera espontánea sino dependiendo de la configuración se pueden aceptar o no las conexiones [21].

Bluetooth permite formar redes *ad-hoc*⁵ punto a punto o redes con conexiones punto a multipunto con los dispositivos Bluetooth que se encuentran dentro del rango de cobertura, éstos pueden añadirse o suprimirse de forma dinámica sin necesidad de reconfigurar la red. La formación de una red con dos o más dispositivos Bluetooth en un rango de cobertura se conoce como picored. La unión de varias picoredes para formar configuraciones *ad hoc* se denomina red dispersa (*scatternet*), la cual permite la comunicación e intercambio de datos en configuraciones flexibles (Figura 2.1). Cada picored se gestiona en forma individual y para ello cuenta con un dispositivo que funciona como nodo Maestro (*master*) de la picored, cuya principal función es gestionar el tráfico de un canal, mientras que los otros dispositivos de la picored funcionan como nodos Esclavo (*slaves*).

⁵ Una red *ad-hoc* es aquella, especialmente inalámbrica, en la que no existe un nodo central, sino que todos los dispositivos están en igualdad de condiciones.

Tabla 2.1. Evolución de la especificación Bluetooth.

Versión	Fecha	Velocidad Máxima	Descripción
1.0	Julio 1999	1 Mbps	Primera versión de la especificación Bluetooth
1.1	Febrero 2001	1 Mbps	Solución de errores de la especificación 1.0 Añade el Indicador de Calidad de Señal Recibida (RSSI)
1.2	Abril 2000	1 Mbps	Implementa la técnica de salto en frecuencia para mejorar la resistencia a interferencias Introduce el tipo de enlace para aplicaciones de audio extendido (eSCO) mejorando la calidad de transmisión de voz Mejoras sobre la interfaz del controlador del huésped (HCI) para sincronizaciones más rápidas entre dispositivos
2.0	Diciembre 2006	3 Mbps	Incorpora la tecnología EDR incrementando la velocidad de transmisión de datos hasta 3 Mbps Incorpora el mecanismo para la reducción de consumo de energía
2.1	Marzo 2007	3Mbps	Mejora sobre la seguridad Bluetooth a través de una simple clave de acceso compuesta de seis dígitos, más fuerte que el código PIN 16 dígitos Incorporación del servicio " <i>Man in the Middle</i> ", para proteger la información de capturas de terceros
3.0	Abril 2009	24 Mbps	Incorporación de la Capa de Adaptación del Protocolo (PAL) ofreciendo mayor rendimiento de transferencia de datos a velocidades aproximadas a 24 Mbps (<i>high speed</i>) Mejora de consumo de potencia en dispositivos móviles Compatibilidad Bluetooth con Wi-Fi para lograr mayor velocidad de transmisión de datos en teléfonos portátiles inteligentes
4.0	Diciembre 2009	24 Mbps	Mejora sobre el consumo de energía permitiendo transmisión de datos hasta 100 m a un bajo costo de energía Modo de ultra bajo consumo para baterías de litio tipo moneda Interoperabilidad entre múltiples proveedores



a) Punto a punto

b) Punto a multipunto

c) Red dispersa

Figura 2.1. Topologías Bluetooth: a) Punto a punto, b) Punto a multipunto y c) Red dispersa [2].

Un dispositivo Bluetooth puede configurarse únicamente como nodo Maestro o como nodo Esclavo en cualquier momento dentro de una picored. El nodo Maestro es el que inicia la transmisión de datos, mientras que el nodo Esclavo es quien responde a la petición del nodo Maestro. El nodo Maestro en una picored determina el reloj de operación y la secuencia de saltos en frecuencia que se utilizará durante la transmisión.

Una picored se forma a partir de dos y hasta ocho dispositivos acoplados, en donde uno de los dispositivos actúa como nodo Maestro y el resto como nodos Esclavo. En una red dispersa un dispositivo puede estar configurado como nodo Esclavo en una picored y puede fungir como nodo Maestro en otra picored.

2.2. Arquitectura de protocolos Bluetooth

La arquitectura de protocolos Bluetooth se compone de protocolos definidos específicamente para Bluetooth, y de protocolos adoptados con la finalidad de facilitar la adaptación de aplicaciones y asegurar su funcionalidad. La arquitectura se divide en cuatro capas principales (Figura 2.2): Núcleo de Protocolos Bluetooth, Capa de Reemplazo de Cables, Capa de Control de Telefonía y Capa de Protocolos Adoptados [2].

A continuación se describen los protocolos de las capas utilizadas en el presente trabajo de tesis⁶.

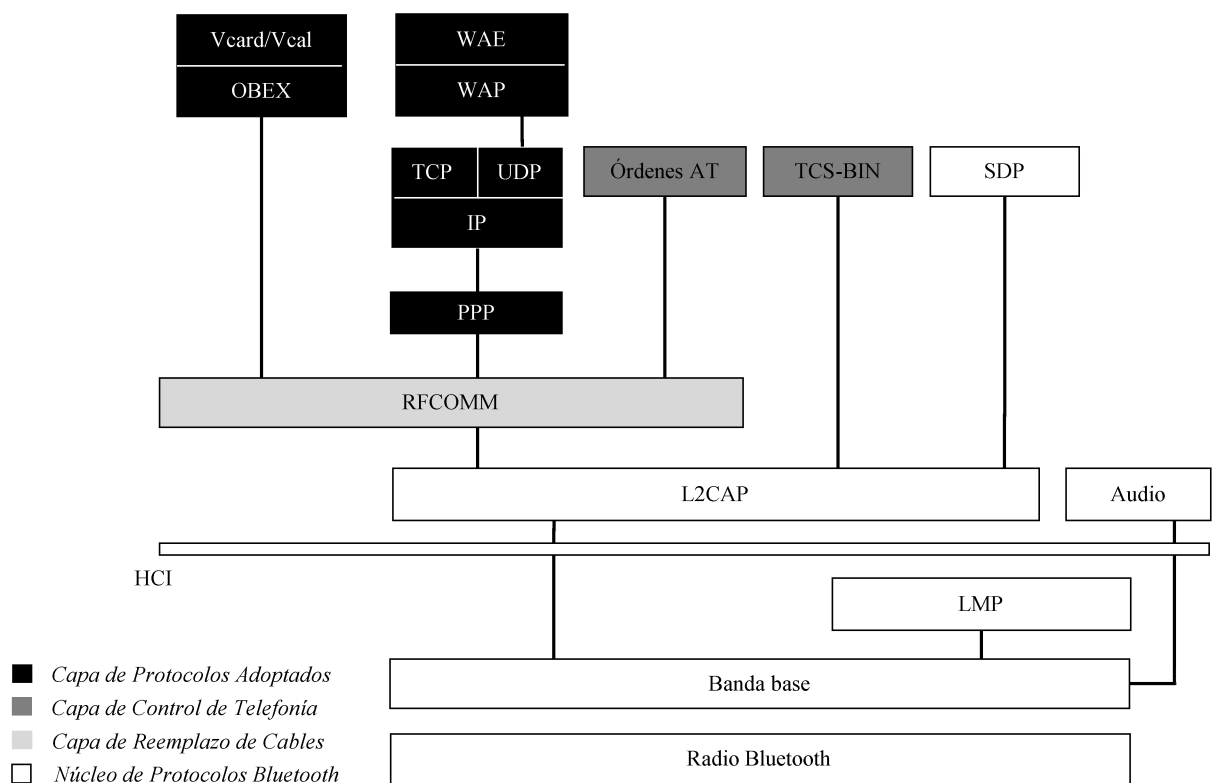


Figura 2.2. Arquitectura de protocolos Bluetooth [2].

⁶ No todas las aplicaciones hacen uso de la arquitectura completa de protocolos Bluetooth.

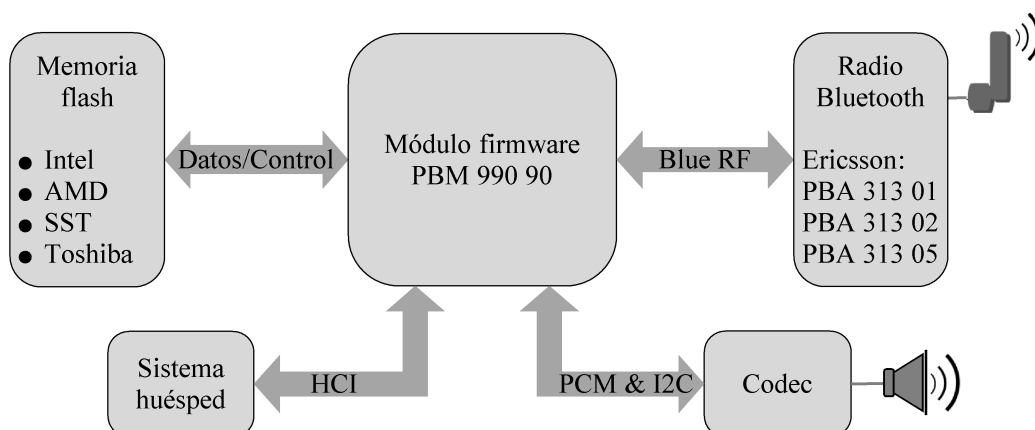


Figura 2.3. Diagrama de un dispositivo Bluetooth con módulo firmware.

2.3. Núcleo de protocolos Bluetooth

La capa del núcleo de protocolos Bluetooth se encuentra en todos los dispositivos Bluetooth dentro de un módulo firmware y se encarga de realizar las funciones de bajo nivel (Figura 2.3).

Esta capa de protocolos soporta la transmisión asíncrona de datos y la transmisión síncrona de voz o ambas. Las aplicaciones de audio en la capa tienen una prioridad mayor para mantener la Calidad de Servicio (QoS, *Quality Of Service*), por lo que la transmisión de audio evita los protocolos adoptados y se procesa directamente por la banda base.

Los principales protocolos definidos para esta capa son: Radio Bluetooth, Banda Base, LMP, L2CAP y SDP.

2.3.1. Radio Bluetooth

El protocolo de radio Bluetooth define el uso de transceptores (*transceivers*), los cuales consideran las emisiones dentro y fuera de la banda de operación, la frecuencia, la interferencia de canales, el método de modulación, etc. Un transceptor Bluetooth es un sistema de radio de espectro ensanchado por salto de frecuencia (FHSS, *Frequency Hopping Spread Spectrum*).

2.3.1.1. Ensanchado del espectro

La técnica de codificación digital conocida como espectro ensanchado⁷ (*spread spectrum*) toma una señal de banda estrecha y la propaga sobre una porción disponible en la banda de RF, incrementando el número de bits a transmitir y expandiendo el ancho de banda. El nodo receptor hace uso del mismo código de propagación que el nodo transmisor para coordinar y compactar la señal expandida y llevarla a su forma original. De esta manera se obtiene una transmisión robusta con mayor precisión al momento de reconstruir la señal por el nodo receptor.

Además, la señal propagada tiene menor densidad de potencia, proporcionando resistencia contra elementos como:

⁷ La técnica de espectro ensanchado también se conoce como espectro esparcido, espectro disperso, o SS por sus siglas en inglés.

- *Interferencia (Interference)*: Condición en la cual la transmisión es interrumpida por fuentes externas, como ruido emitido por dispositivos electromecánicos.
- *Obstrucción (Jamming)*: Condición en la cual una señal más fuerte obstruye a una señal más débil.
- *Trayectoria múltiple (Multipath)*: Condición en la que una señal se distorsiona al ser reflejada por algún objeto sólido.
- *Intercepción (Interception)*: Condición en la que usuarios no autorizados capturan la señal.

Bluetooth utiliza la técnica de salto de frecuencia (*frequency hopping*), en donde el transmisor sigue una orden de saltos por medio de una secuencia pseudoaleatoria establecida. La especificación Bluetooth define un rango de 1,600 saltos por segundo en estado conectado a través de 79 canales en la banda de los 2.4 GHz, con un ancho de banda de 1 MHz y con una potencia de transmisión de 1 W (30 dBm).

Los radios Bluetooth utilizan la técnica AFH para evitar interferencia con otros dispositivos en la misma banda de operación. El algoritmo AFH cambia dinámicamente la secuencia de saltos de frecuencia restringiendo el número de canales de cada nodo Bluetooth, permitiendo que ciertos canales permanezcan disponibles para su uso [25].

En una picored cada dispositivo tiene un reloj interno o nativo (*native clock*), el cual determina el tiempo y los saltos usados por el transceptor. El reloj nativo del nodo Maestro, determina el tiempo y los saltos de frecuencia en un canal dentro de una picored, así el reloj nativo de cada nodo Esclavo suma un tiempo de compensación (*offset*) para coincidir con el reloj nativo del nodo Maestro.

La especificación de Bluetooth define dos modos de modulación para transmitir información, cuyas características lista la Tabla 2.2 [2].

Tabla 2.2. Modos de modulación utilizados en Bluetooth [21].

Modos de modulación	Modulación	Velocidad	Descripción
Velocidad Básica (<i>Basic Rate</i>)	GFSK	1 Mbps	Modulación obligatoria que minimiza la complejidad del transceptor
Velocidad de Datos Mejorada (<i>Enhanced Data Rate</i>)	PSK	2 Mbps con $\pi/4$ -DQPSK 3 Mbps con 8DPSK	Modulación opcional que utiliza dos variantes de la modulación PSK

En la mayoría de sistemas de comunicaciones se requiere que la información se transmita en ambas direcciones (*full-duplex*). Los sistemas de comunicaciones RF permiten dos formas básicas para realizar comunicaciones *full-duplex*: dúplex por división de frecuencia (FDD, *Frequency Division Duplexing*) y dúplex por división de tiempo (TDD, *Time Division Duplexing*).

Las reglas para compartir la interfaz aérea por diferentes dispositivos Bluetooth se basan en la técnica TDD, la cual utiliza un canal para recepción y transmisión de forma alternada, mediante una antena que divide el tiempo para cada una de dichas funciones. El canal se divide en ranuras de 625 μ s, donde las ranuras son enumeradas de acuerdo al reloj del nodo Maestro en una picored. Todas las operaciones de transmisión o recepción de datos ocurren cada 1, 3 ó 5 ranuras (*slot*) por paquete [16].

Cuando un nodo Maestro desea transmitir un paquete a un nodo Esclavo, ambos nodos deben sintonizar el mismo canal de radio. El nodo Esclavo sincroniza la secuencia de sus saltos utilizando el reloj del nodo Maestro.

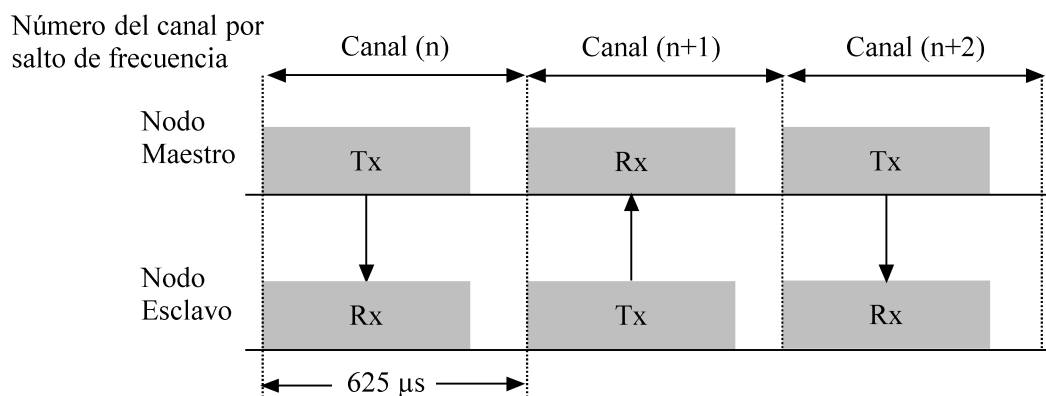


Figura 2.4. Temporización de las ranuras por paquete [2].

Los dispositivos Bluetooth saltan de frecuencia una vez por cada paquete (ya sea en una operación de transmisión o de recepción) permitiendo a los dispositivos utilizar el ancho de banda disponible en la banda ISM a través de los 79 canales diferentes con 1 MHz de separación entre éstos, ésta es la característica fundamental de los sistemas Bluetooth [25] (Figura 2.4).

2.3.2. Banda base

La banda base define los procesos de búsqueda y enlace de los dispositivos, tipos de enlace, determina los roles del nodo Maestro y del nodo Esclavo, así como las secuencias de los saltos en frecuencia a utilizar. El nodo Maestro dentro de la banda base define:

- La secuencia de los saltos de frecuencia utilizados por los miembros activos.
- La frecuencia de espera.
- La frecuencia de operación.
- Qué nodo Esclavo debe transmitir y qué nodo Esclavo será el siguiente en hacerlo.
- El tipo de paquete a transmitir.

La banda base utiliza 5 tipos diferentes de canales lógicos para el control y flujo de información: Canal de Control de Enlaces (LC, *Link Control*), canal de Administración de Enlaces (LM, *Link Manager*), canal Asíncrono de Usuario (UA, *User Asynchronous*), canal Isócrono de Usuario (UI, *User Isochronous*) y canal Síncrono de Usuario (US, *User Isochronous*).

La banda base utiliza el canal LC para administrar los estados de operación de un dispositivo Bluetooth. Existen tres estados de operación principales y siete subestados (Figura 2.5). Los estados de operación principales son:

- *Conectado (connected)*: Un dispositivo Bluetooth se encuentra en estado conectado cuando es miembro de una picored.
- *Espera (standby)*: Se dice que un dispositivo Bluetooth se encuentra en estado de espera cuando no pertenece a ninguna picored (estado por defecto para todo dispositivo Bluetooth).
- *Estacionario (park)*: Estado de operación utilizado cuando un dispositivo no necesita participar en una picored, pero permanece sincronizado a ésta para su posterior ingreso.

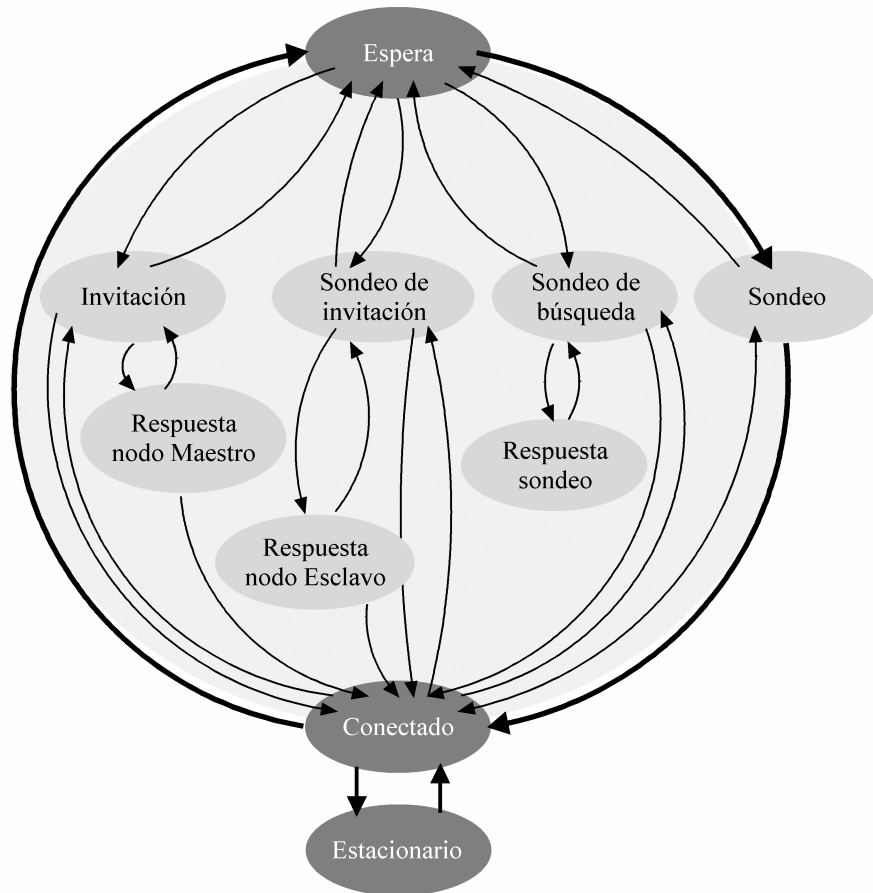


Figura 2.5. Estados y subestados de un dispositivo Bluetooth [2].

Existen tres elementos principales asociados al intercambio de paquetes de banda base entre dispositivos Bluetooth:

- *Dirección del dispositivo Bluetooth (BD_ADDR, Bluetooth Device Address):* La dirección del dispositivo Bluetooth es única para cada dispositivo y está formada por 48 bits (Figura 2.6). La BD_ADDR se divide en tres partes: dirección baja (LAP, Lower Address Part), dirección alta (UAP, Upper Address Part) y dirección no significativa (NAP, NonSignificat Address Part)⁸.
- *Reloj del dispositivo Bluetooth:* El reloj dentro de un dispositivo Bluetooth decide cuándo un dispositivo puede transmitir o cuándo escuchar, a qué frecuencia transmitir o recibir, qué tipo de paquetes se recibirán o se transmitirán. El reloj se compone de 28 bits y se incrementa cada 312.5 μ s, representado una frecuencia de operación de 3.2 KHz.



Figura 2.6. Formato de la dirección de un dispositivo Bluetooth [2].

⁸ Los 8 bits de la UAP y los 16 bits de la NAP constituyen el identificador único de la organización (OUI, Organization Unique Identifier), los cuales son asignados por la autoridad regulatoria y la LAP es asignada por el SIG.

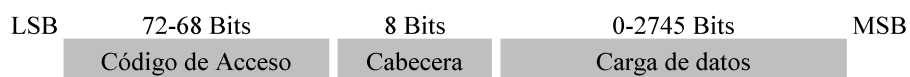


Figura 2.7. Formato general de los BB_PDU [2].

- *Secuencia de salto de frecuencia*: Los dispositivos Bluetooth utilizan secuencias para comunicarse en la misma frecuencia de operación. Para ello cuentan con un módulo seleccionador de frecuencias (FSM, *Frequency Selection Module*) que genera el orden de los saltos. Existen seis tipos de secuencias para los saltos, cinco tipos de secuencia para el sistema base y un tipo de secuencia AFH [2].

2.3.2.1. Formato de los paquetes BB_PDU

El formato general de los paquetes usados por la banda base para transmitir datos (BB_PDU, *Baseband Packet Data Unit*) se compone de tres campos: Código de Acceso (*Access Code*), Cabecera (*Header*) y Carga (*Payload*). La Figura 2.7 muestra el formato de un paquete BB_PDU [2].

2.3.2.1.1. Código de acceso

Cada paquete BB_PDU empieza con un código de acceso para identificar y sincronizar el flujo de bits en una picored. El número de bits del código de acceso varía dependiendo de si se utiliza la cabecera del paquete BB_PDU (72 bits) o no (68 bits), en este último caso no se realiza el rastreo. El código se compone de los campos (Figura 2.8):

- *Preámbulo (Preamble)*: Indica la llegada de un paquete al receptor.
- *Palabra de sincronización (sync)*: Se usa para sincronizar el tiempo con el receptor.
- *Rastreo (trailer)*: Se adjunta a la palabra de sincronización si el siguiente campo del código de acceso es la cabecera del paquete.

La función del código de acceso puede cambiar dependiendo de la operación del dispositivo Bluetooth. Existen tres tipos de código de acceso:

- *Código de Acceso del Canal (CAC)*: Identifica a una picored mediante la LAP de la dirección del nodo Maestro de la picored, antes de que algún dato sea transmitido sobre una frecuencia dada por la secuencia de salto de canal (*channel-hopping*). Este código se incluye en todos los paquetes de intercambio en un canal de una picored. Todos los paquetes enviados en la misma picored empiezan con el mismo código de acceso del canal.
- *Código de Acceso del Dispositivo (DAC)*: Este código se usa para realizar indicaciones especiales como invitaciones y respuestas a las invitaciones. La transmisión en una frecuencia dada por la secuencia de saltos de invitación, y su respuesta, son precedidas por el DAC indicado en la LAP de la dirección del dispositivo invitado.
- *Código de Acceso de Búsqueda (IAC)*: Código utilizado para generar la secuencia de saltos de búsqueda. Existen 64 IAC divididos en dos tipos:
 - *Búsqueda general*: El IAC general o GIAC se asocia a los 64 bits reservados de la LAP y es común para todos los dispositivos en el rango de cobertura de la picored.



Figura 2.8. Campos del código de acceso en un BB_PDU [2].



Figura 2.9. Campos de la cabecera en un BB_PDU [2].

- *Búsqueda dedicada*: Son los 63 IACs restantes, conocidos como DIACs (*Dedicated IACs*) y se asignan a un grupo de unidades Bluetooth que comparten características dentro de un rango de cobertura⁹.

2.3.2.1.2. Cabecera

Si se usa, la cabecera tiene una longitud de 18 bits que se transmite tres veces y contiene la información del control de enlace (LC). La cabecera consiste de seis campos (Figura 2.9):

- *Dirección de miembros activos (AM_ADDR, Active Member Address)*: Campo de 3 bits usado para distinguir entre los miembros activos que participan en una picored. A cada nodo Esclavo activo se le asigna una dirección temporal de tres bits como identificador y se utiliza en los paquetes enviados entre el nodo Maestro y el nodo Esclavo.
- *Tipo (Type)*: Este campo consta de 4 bits y especifica tanto el tipo de paquete, como el tiempo que ocuparán dichos paquetes (1, 3 ó 5 ranuras de longitud) (Figura 2.10).
- *Flujo (Flow)*: Este campo consta de un bit y se usa para controlar el flujo de paquetes sobre enlaces ACL. Cuando se llena la memoria temporal (*buffer*) de un enlace ACL, se regresa una indicación de paro (*stop*) a la mitad de la transmisión y cuando la memoria temporal del receptor se vacía, se regresa una indicación de disponibilidad (*go*).
- *Solicitud Automática de Repetición (ARQ, Automatic Repeat Request)*: Campo de un bit usado para indicar al dispositivo transmisor que los datos llegaron con éxito. La recepción completa se verifica mediante un código de redundancia cíclica (CRC)¹⁰. La notificación de retorno se puede considerar una señal de reconocimiento positiva (ACK) o negativa (NAK) según sea el caso.

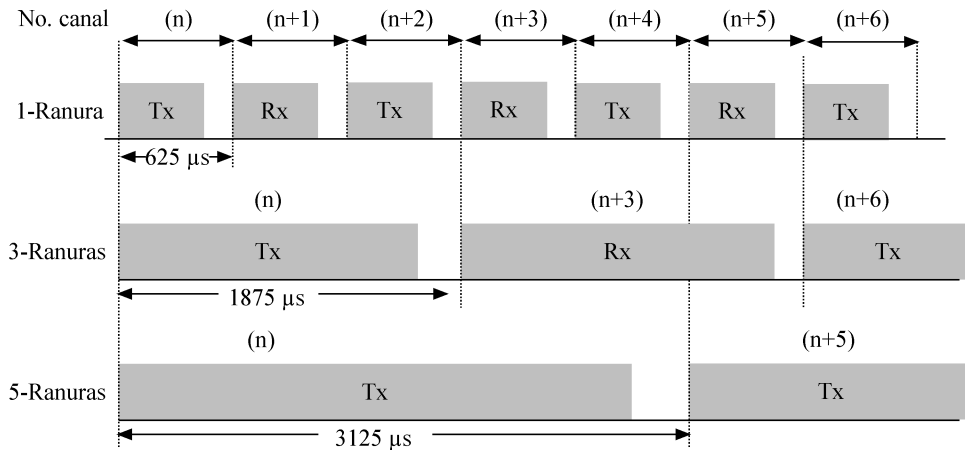


Figura 2.10. Tiempo para paquetes de 1, 3 ó 5 ranuras de longitud [2].

⁹ La especificación de Bluetooth no define la utilización de los IAC, solo recomienda que deben funcionar como un mecanismo de filtrado para identificar subconjuntos predefinidos de dispositivos Bluetooth.

¹⁰ El método CRC verifica la exactitud de los datos mediante una función matemática determinando un valor numérico en función del contenido del paquete, el valor es comparado con otro valor generado por la función del dispositivo receptor. Si coinciden se regresa una señal ACK, de lo contrario se regresa una señal NACK.

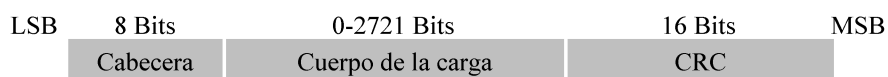


Figura 2.11. Campos de la carga en un BB_PDU [2].

- *Secuencia Numérica (SEQN, Sequence Number)*: Campo de un bit que proporciona una secuencia numérica y ordena el flujo de paquete de datos al llegar a un dispositivo receptor.
- *Cabecera de Verificación de Errores (HEC, Header Error Check)*: Este campo consta de 8 bits y se usa para verificar la integridad de la cabecera. El HEC se genera a partir del polinomio generador $G_{HEC}=x^8+x^7+x^5+x^2+x+1$.

2.3.2.1.3. Carga

La parte final del formato del paquete general es la carga (*Payload*) y consta de dos tipos de campos: campo para voz (síncrono) y campo para datos (asíncrono).

Los paquetes ACL contienen el campo para datos, los paquetes SCO contienen el campo para voz y los paquetes DV contienen ambos.

El campo de datos consiste de tres segmentos (Figura 2.11):

- *Cabecera de la Carga (Payload Header)*: Este campo contiene los paquetes de datos con uno o dos bytes de longitud y especifica los canales lógicos, el control del flujo sobre estos canales y el indicador de longitud de la carga. El indicador proporciona el número de bytes que contiene el cuerpo de la carga, que pueden ser uno o dos bytes.
- *Cuerpo de la Carga (Payload Body)*: Contiene la información del usuario.
- *Generación del código CRC*: Después de que el generador del código CRC ha sido inicializado, se calcula el código de verificación de redundancia cíclica de 16 bits sobre la información a transmitir.

2.3.2.2. Operación de búsqueda

En una operación de búsqueda, un dispositivo adquiere información referente a la identidad de otros dispositivos dentro de su rango de cobertura. Los dispositivos utilizan el GIAC para generar la secuencia de saltos de búsqueda, enseguida un dispositivo selecciona una nueva frecuencia en la cual realizará una búsqueda cada 312.5 μ s. Mientras que los dispositivos buscados ejecutan sondeos de búsqueda seleccionando una nueva frecuencia cada 1.28s.

El estado de búsqueda está compuesto de varios subestados, donde el nodo Maestro ejecuta el subestado de búsqueda y transmite paquetes de búsqueda, los cuales son recibidos por los nodos Esclavo si se encuentran en el subestado de sondeo de búsqueda (*inquiry scan*), si es así, los nodos Esclavo entran en el subestado de respuesta de búsqueda y responden a la búsqueda (*inquiry response*) realizada por el nodo Maestro (Figura 2.12).

En la búsqueda se utiliza un mecanismo de retroalimentación para evitar colisiones ocasionadas por la respuesta simultánea de múltiples nodos Esclavo a un mismo paquete de búsqueda ID (*inquiry ID*). El nodo Esclavo puede permanecer en estado de espera, estado conectado o subestado de búsqueda.

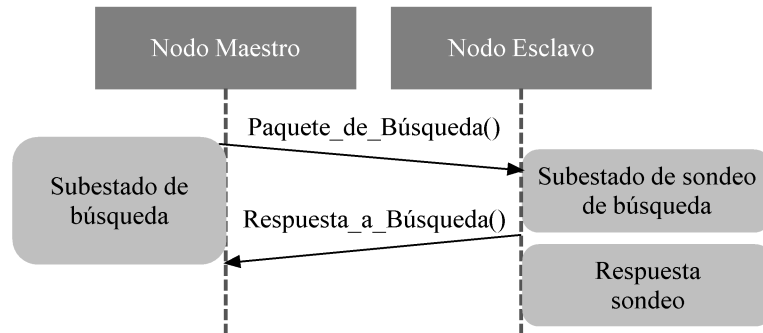


Figura 2.12. Procedimiento de búsqueda.

2.3.2.3. Operación de invitación

En una operación de invitación un dispositivo invita a otro a formar una piconet, así mismo el estado de invitación está compuesto de varios subestados, donde el nodo Maestro ejecuta la invitación y la respuesta del nodo Maestro (*master response*), mientras que el nodo Esclavo ejecuta el sondeo de invitación (*page scan*) y la respuesta del nodo Esclavo (*slave response*).

En el estado de invitación un nodo Maestro utiliza la dirección Bluetooth y el reloj del nodo Esclavo para enviarle la invitación a través de un paquete BB_PDU. El nodo Maestro envía la invitación cada 312.5 μ s. El nodo Esclavo debe encontrarse en el subestado de sondeo de invitación para recibir el paquete de invitación del nodo Maestro. El nodo Esclavo selecciona una frecuencia cada 1.28 s para entrar al subestado de sondeo de invitación. Durante la operación de invitación se utiliza una secuencia de salto de invitación. La transmisión contiene únicamente el código de acceso DAC del nodo Esclavo, este BB_PDU se conoce como paquete ID del nodo Esclavo (*Slave ID Packet*). El nodo Esclavo responde enviando el paquete ID del nodo Esclavo para avisarle al nodo Maestro que ha recibido la invitación. Por último, el nodo Maestro entra en el subestado de respuesta del nodo Maestro, en donde transmite la AM_ADDR del nodo Esclavo dentro de un paquete de Sincronización de Salto de Frecuencia (FHS, *Frequency Hopping Synchronization*), el cual permite al nodo Esclavo unirse a la piconet del nodo Maestro y entrar en estado de conectado (Figura 2.13).

Durante el proceso de invitación, el nodo Maestro asigna una AM_ADDR de 3 bits al nuevo nodo Esclavo. La AM_ADDR toma el valor de 1 a 7 y es única para cada nodo Esclavo activo en una piconet. El valor 0 está reservado para transmisiones de difusión del nodo Maestro a todos los nodos Esclavo en la piconet.

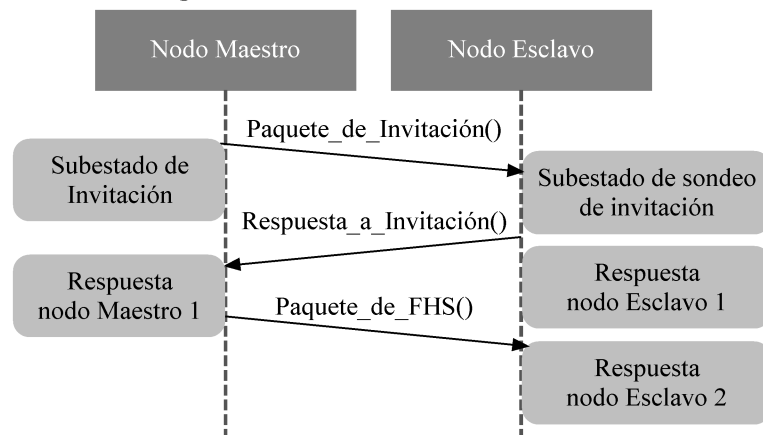


Figura 2.13. Procedimiento de invitación.

Para que un dispositivo se encuentre en estado conectado, debe pasar del estado de búsqueda al estado de invitación.

2.3.2.4. Tipos de enlaces

Bluetooth soporta dos tipos de enlaces para la transmisión de voz y datos:

- *Enlace de conexión asíncrona (ACL, Asynchronous Connection Link)*: Tipo de enlace usado en la transmisión de datos; soporta conexiones punto a multipunto y conmutación de circuitos; detecta errores en el dispositivo receptor enviando una notificación en la cabecera del paquete de retorno, con la finalidad de reenviar únicamente el dato perdido [2].
- *Enlace síncrono orientado a conexión (SCO, Synchronous Connection-Oriented)*: Este tipo de enlace se utiliza para la transmisión de voz y tráfico multimedia. El enlace SCO proporciona conexiones punto a punto y conmutación de circuitos, ambas usadas para un ancho de banda reservado.

La Tabla 2.3 muestra las características de la transmisión síncrona y asíncrona para cada tipo de enlace.

Tabla 2.3. Características de la transmisión síncrona y asíncrona [21].

Transmisión	Sincronización	Velocidad máxima de transmisión
Asíncrona	Mediante bits de inicio y de paro.	Asimétrica: 723.2 Kbps en una dirección (longitud de los paquetes de 5 ranuras) y 57.6 Kbps en dirección opuesta (longitud de los paquetes a 1 ranura)
Síncrona	Mediante señales de sincronización <i>sync</i> de inicio y de paro.	Simétrica: 433.9 Kbps en cualquier dirección

La transmisión síncrona es más eficiente que la asíncrona en el uso del ancho de banda debido a la ausencia de los bits de inicio y de paro.

Existen tres canales síncronos de 64 Kbps dedicados a la transmisión de voz, en donde los paquetes no pueden ser retransmitidos ya que el resultado sería un retraso que interrumpe la interacción entre el receptor y el transmisor.

2.3.3. LMP

El Protocolo de Administración de Enlace (LMP, *Link Manager Protocol*) se encarga de administrar la configuración y control entre dispositivos Bluetooth, incluyendo el tamaño de los paquetes, establecimiento de la seguridad del enlace, claves de encriptación, modos de potencia, ciclo de trabajo de los radios Bluetooth y estados de conexión de los dispositivos dentro una piconet.

Cuando dos dispositivos Bluetooth se encuentran dentro de un rango de alcance, la comunicación punto a punto se realiza mediante mensajes enviados y recibidos a través del LMP entre dos Administradores de Enlaces (LM, *Link Manager*) (Figura 2.14).

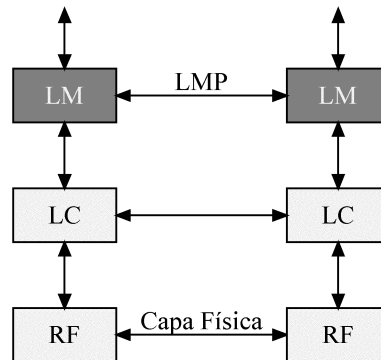


Figura 2.14. Mensajes entre dos LM por medio del LMP [2].

El LM es un *software* que trabaja sobre un microprocesador dentro de una unidad Bluetooth y sirve para descubrir dispositivos, montar enlaces, incluir seguridad y encriptación de datos, y generar, intercambiar y verificar las claves de encriptación. El LM también se encarga de controlar los modos de ahorro de energía ajustando los ciclos de trabajo del radio Bluetooth. El LMP no carga datos de la aplicación, únicamente se basa en los datos de control de las capas superiores y del intercambio de mensajes LM_PDU entre LM. Los mensajes LM_PDU tienen una alta prioridad, por lo que si es necesario enviar un mensaje LM_PDU, éste no será retrasado por el tráfico de L2CAP. El tiempo en recibir un BB_PDU, conteniendo un LMP_PDU, y enviar otro BB_PDU, con una respuesta válida LMP_PDU, es menor a 30 segundos, en caso de exceder este tiempo el enlace se considera como “muerto”.

2.3.3.1. Formato de los mensajes LM_PDU

Los campos de los mensajes LM_PDU son (Figura 2.15):

- *Identificador de transacción (TID, Transaction identifier)*: Campo de un bit utilizado por el LMP para identificar si la transacción entre mensajes LM_PDU ha sido iniciada por el nodo Maestro (TID igual a 0) o por el nodo Esclavo (TID igual a 1).
- *Código de operación*: Este campo consta de 7 bits y se utiliza para identificar el tipo de LM_PDU.
- *Parámetro 0 hasta N*: Representan el conjunto de parámetros específicos de los mensajes LM_PDU, en donde cada parámetro consta de 1 byte.

En la especificación de Bluetooth existen 55 tipos de LM_PDU, cada uno definido para una función específica. Existen dos respuestas LM_PDU generales: `LMP_accepted` y `LMP_not_accepted` que se usan para responder otros mensajes. La Tabla 2.4 muestra dos ejemplos del formato de mensajes LM_PDU.

Tabla 2.4. Formato de mensajes LM_PDU [2].

LMP PDU	Op code	Bytes	Tipo	Dirección	Parámetros
<code>LMP_accepted</code>	3	2	Obligatorio	M ⇌ E	Codigo_operacion
<code>LMP_not_accepted</code>	4	3	Obligatorio	M ⇌ E	Codigo_operacion, Motivo



Figura 2.15. Formato de mensajes LM_PDU [2].

2.3.3.1.1. Seguridad

Para establecer la seguridad de los paquetes de información se utiliza un procedimiento de autenticación, el cual se basa en un esquema de intercambio de mensajes de respuesta (*challenge-response*), donde un verificador (*verifier*) envía un número aleatorio a un pretendiente (*claimant*) a descubrir mediante la LM_PDU `LMP_au_rand`. El pretendiente utiliza el número aleatorio y regresa una respuesta que contiene su BD_ADDR y una clave mediante la LM_PDU `LMP_sres`. Si el resultado es el esperado por el verificador, el pretendiente es considerado como un dispositivo autenticado. En caso contrario, termina la conexión enviando el mensaje LM_PDU `LMP_detach`.

2.3.3.1.2. Emparejamiento

El proceso de emparejamiento genera la clave de inicialización utilizada para autenticar dispositivos y crea una clave de enlace permanente. Si no existe una clave de enlace para un dispositivo, se envía una LM_PDU `LMP_au_rand` y el pretendiente responde con una LM_PDU `LMP_not_accepted`, indicando que la clave está perdida o no ha sido creada.

Cuando dos dispositivos no tienen una clave de enlace común, entonces se debe generar una clave basada en un Número de Identificación Personal (PIN, *Personal Identification Number*) y en número aleatorio. Cuando el verificador envía el mensaje `LMP_in_rand` al pretendiente, se genera una clave de 128 bits de longitud. Después de una autenticación exitosa, se genera la clave, la cual se usará en las autenticaciones entre dos dispositivos para todas las conexiones subsecuentes hasta que sea necesario cambiar la clave.

2.3.3.1.3. Encriptación del enlace

La encriptación se usa para proteger la información del flujo de datos y se utiliza cuando se ha establecido un proceso de autenticación. El tamaño máximo para la clave de encriptación es de 128 bits.

El nodo Maestro y el nodo Esclavo se ponen de acuerdo si usar o no la encriptación. En caso de que así sea, se determina el tamaño de la clave de encriptación. El nodo Maestro envía el mensaje `LMP_encryption_key_size_req`. Cuando empieza un proceso de encriptación, el nodo Maestro reparte un número aleatorio y procesa la clave de encriptación mediante la LM_PDU `LMP_start_encryption_req` que incluye el número aleatorio para todos los nodos Esclavos en una picored.

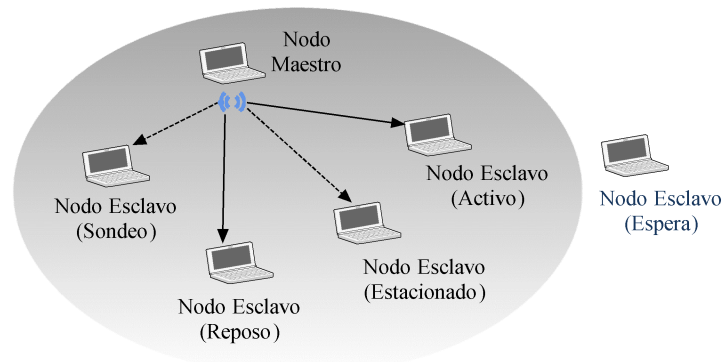


Figura 2.16. Diferentes estados de conexión [21].

2.3.3.1.4. Enlaces SCO

La conexión inicial entre dos dispositivos es sobre un enlace ACL y una vez que se ha reconocido el enlace ACL, se pueden establecer uno o más enlaces SCO mediante la LM_PDU `LMP_SCO_link_req`.

Para establecer un enlace SCO, el nodo Maestro envía una petición con los parámetros para especificar los tiempos, el tipo de paquete y la codificación que se usará sobre el enlace SCO. Existen tres tipos de formatos de codificación de voz: μ -Law, A-Law y CVSD.

Las ranuras de tiempo usadas para enlaces SCO se determinan por tres parámetros controlados por el nodo Maestro: Intervalo SCO, retardo SCO y una bandera que indica cómo se calcula la primera ranura SCO.

2.3.3.1.5. Control de potencia

El control de potencia se usa para limitar y optimizar el consumo de energía de los dispositivos Bluetooth y cubrir niveles de interferencia. Cuando un dispositivo mide la fuerza de la señal recibida, regresa un reporte para solicitar que la potencia aumente o disminuya, lo anterior se realiza mediante el Indicador de Fuerza de la Señal Recibida (RSSI, *Received Signal Strength Indicator*). El ajuste de la potencia es útil cuando dos dispositivos se encuentran muy cerca físicamente y no se requieren altos niveles de potencia en la transmisión.

Los dispositivos en un estado conectado pueden ahorrar energía o atender otros procesos tales como la participación en una red dispersa, además de notificar el establecimiento de un enlace entre dispositivos [21]. Cuando un dispositivo Bluetooth entra al estado de conectado, puede ingresar a cuatro modos de consumo de energía (Figura 2.16):

- *Modo de Reposo (Hold Mode)*: Normalmente se accede a este modo cuando no se necesita enviar datos durante un tiempo prolongado. Durante este periodo, el transceptor puede apagarse para ahorrar energía. El modo de reposo también se usa si un dispositivo espera unirse a otra piconet, o si espera descubrir o ser descubierto por otro dispositivo Bluetooth.
- *Modo de Sondeo (Sniff Mode)*: Proporciona un mecanismo para reducir el tiempo en activo de un dispositivo, mejorando de esta manera el consumo de energía. Para entrar en el estado de ahorro de energía, tanto el nodo Maestro como el nodo Esclavo establecen un intervalo de sondeo y un intervalo de compensación, los cuales establecen las ranuras de tiempo para el modo.

- *Modo Estacionario (Park Mode)*: Se utiliza si un nodo Esclavo no necesita participar en una conexión, pero debe permanecer sincronizado. El dispositivo abandona su AM_ADDR, no obstante se le concede una dirección única estacionaria (PM_ADDR, *Parked Member Address*) y posteriormente puede ser reactivado por el nodo Maestro.
- *Modo Activo (Active Mode)*: En el modo activo, tanto el nodo Maestro y el nodo Esclavo participan en la conexión. El nodo Esclavo escucha todos los paquetes provenientes del nodo Maestro. En este modo se obtienen respuestas más rápidas, sin embargo, representa el modo con mayor consumo de energía.

2.3.3.1.6. Establecimiento de la conexión

Cuando un dispositivo desea establecer una conexión involucrando capas por encima del LM, envía la LM_PDU *LMP_host_connection_req*. Cuando el mensaje es recibido por el dispositivo opuesto, se informa al huésped (*host*) sobre la conexión en curso. El dispositivo remoto puede aceptar o rechazar la petición de conexión enviando las LM_PDU *LMP_accepted* o *LMP_not_accepted* según sea el caso. Si la LM_PDU *LMP_host_connection_req* es aceptada, los procedimientos de seguridad son llamados (emparejamiento, autenticación y encriptación). Cuando un dispositivo no necesita otro procedimiento de configuración del enlace, envía la LM_PDU *LMP_setup_complete*. Una vez que ambos dispositivos se hayan enviado la LM_PDU *LMP_setup_complete*, es posible transmitir el primer paquete sobre un canal lógico diferente del LMP.

2.3.4. L2CAP

El protocolo de Control de Enlace Lógico y Adaptación de Protocolos (L2CAP¹¹, *Logical Link Control and Adaptation Protocol*) cuenta con el Servicios de Multiplexado de Protocolos (PSM, *Protocol Service Multiplexer*), para comunicarse con la capa de protocolos adoptados y con el protocolo de órdenes AT de la capa de control de telefonía; así mismo L2CAP ofrece el servicio de segmentación y reensamblado de paquetes (SAR, *Packet Segmentation And Reassembly*), y transmisión de información con Calidad de Servicio (QoS, *Quality of Service*). L2CAP se encuentra por encima del protocolo de banda base y reside dentro de la capa de enlace de datos (*data link*) del modelo de referencia OSI (Figura 2.17).

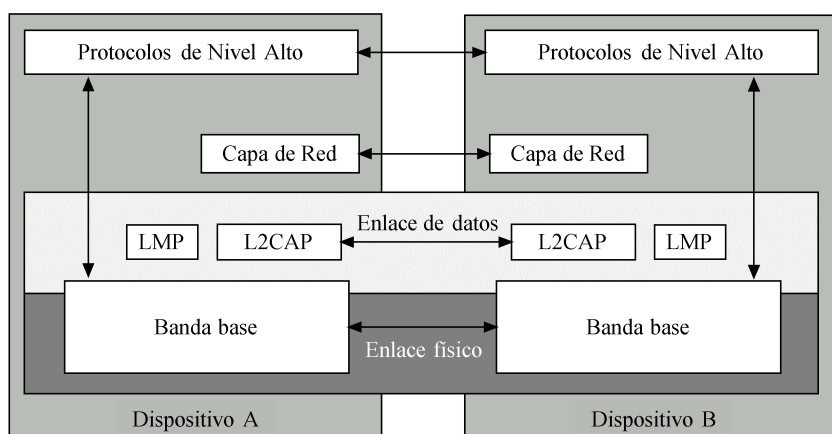


Figura 2.17. L2CAP reside dentro de la capa de enlace de datos.

¹¹ El prefijo “L2” en L2CAP hace referencia a la capa 2 del modelo OSI [2].

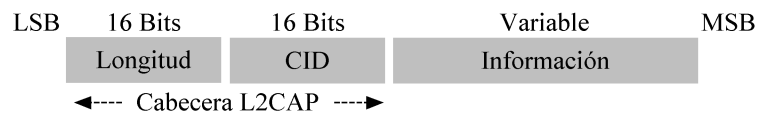


Figura 2.18. Formato de los paquetes L2CAP_PDU [2].

El tamaño de una BB_PDU es limitado y no es suficiente para comunicar paquetes de mayor longitud que manejan los protocolos de niveles superiores [21]. L2CAP permite a los protocolos de niveles superiores transmitir y recibir paquetes de datos de hasta 64 Kbytes de longitud. La especificación de L2CAP define que solo pueden existir enlaces ACL y no más de un enlace entre dos dispositivos.

2.3.4.1. Formato de los paquetes L2CAP

Siguiendo el modelo de comunicación basado en canales, en donde cada canal representa un flujo de datos entre entidades L2CAP de diferentes dispositivos remotos, los canales pueden ser orientados a conexión (*Connection Oriented*) o sin conexión (*Connectionless*) [21].

2.3.4.1.1. Canal orientado a conexión

La Figura 2.18 muestra el formato de los paquetes L2CAP_PDU con un canal orientado a conexión. Los campos que constituyen un paquete L2CAP_PDU son:

- *Longitud*: Indica el tamaño en bytes que tiene la información del paquete, sin incluir la longitud de la cabecera de L2CAP.
- *Identificador del Canal (CID, Channel Identifier)*: Contiene el valor local que representa al extremo final (*endpoint*) de un canal lógico en un dispositivo remoto.
- *Información*: Puede contener la carga recibida o transmitida de protocolos de niveles superiores o datos propios de L2CAP dependiendo del valor de CID. La Tabla 2.5 contiene los valores y descripción de los CID.

Tabla 2.5. Definiciones de los CID [2].

CID	Descripción
0x0000	Identificador nulo
0x0001	Canal de señalización
0x0002	Canal sin conexión
0x0003-0x0030	Reservados para uso futuro
0x0040-0xFFFF	Asignación dinámica

El identificador nulo se define como un identificador ilegal y por lo tanto no debe ser usado por los dispositivos. Los valores de CID de 0x0001 hasta 0x0030 son valores reservados para funciones específicas de L2CAP.

Para transmitir datos sobre canales orientados a conexión, éstos deben ser asignados dinámicamente y configurados a través del envío de paquetes de señalización entre los dispositivos transmisores. Después de haberse establecido los canales del dispositivo local y del dispositivo remoto, se puede comenzar a transmitir información.

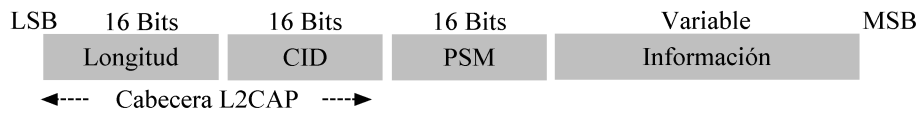


Figura 2.19. Paquetes L2CAP_PDU con un canal ACL [2].

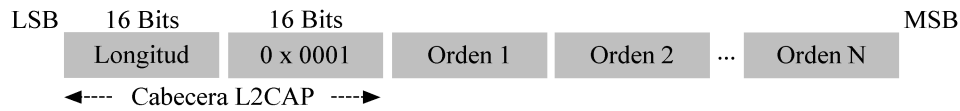


Figura 2.20. Formato de un paquete de señalización SCO [2].

2.3.4.1.2. Canal sin conexión

El canal sin conexión se utiliza para transmitir información a un grupo de dispositivos sin que se haya establecido una conexión previamente. Un canal representa a un grupo de dispositivos que asegura que los datos enviados lleguen a todos los miembros del grupo. Sin embargo, existen inconvenientes como la ausencia de QoS y al hecho de que no existe verificación sobre confirmación de los datos transmitidos. La Figura 2.19 muestra el formato de los paquetes L2CAP_PDU con un canal sin conexión.

En cada paquete L2CAP con un canal ACL, cuenta con el PSM. Los valores de PSM se dividen en dos rangos, el primer rango es definido por el SIG y el segundo se asigna dinámicamente y se usa en conjunción con el protocolo SDP. La Tabla 2.6 muestra los posibles valores del campo PSM.

Tabla 2.6. Valores de PSM [2].

PSM	Protocolos
0x0001	Protocolo de Servicios de Descubrimiento (SDP)
0x0003	RFCOMM
0x0005	Protocolo de Control Telefónico
0x0007	TCS-BIN-CORDLESS
0x000F	Protocolo de Encapsulamiento de Red Bluetooth (BNEP)
0x1001-0xFFFF	Asignados dinámicamente

2.3.4.1.3. Canal de Señalización

El canal de señalización (0x0001) genera canales de datos orientados a conexión y establece sus características. Los paquetes de señalización se usan para transmitir órdenes de señalización entre L2CAP de diferentes dispositivos (Figura 2.20), y mientras no se exceda el tamaño máximo del paquete, se pueden enviar múltiples órdenes L2CAP en dicho paquete.

El formato de las órdenes L2CAP de señalización se compone de los siguientes campos (Figura 2.21):

- **Código:** Contiene un valor de 1 byte que identifica el tipo de orden de señalización.
- **Identificador:** Campo de 1 byte que se usa para hacer coincidir una solicitud con su respuesta. Los dispositivos solicitantes envían este campo y los dispositivos que responden usan el mismo valor en su respuesta.
- **Longitud:** Campo de 2 bytes que indica el número de bytes que tiene la información contenida en el campo de datos.

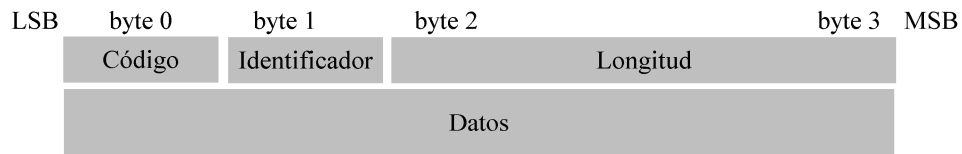


Figura 2.21. Formato de órdenes L2CAP de señalización [2].

- *Datos*: Información específica de las órdenes necesarias para su ejecución por los dispositivos replicadores. Para algunas órdenes este campo no se requiere.

La Tabla 2.7 lista los datos que transportan las órdenes de señalización L2CAP.

Tabla 2.7. Datos de las órdenes de señalización L2CAP.

Código	Descripción	Datos
0x00	Reservado	-
0x01	Rechazo de orden	Motivo, Dato
0x02	Solicitud de conexión PSM	PSM, Origen_CID
0x03	Respuesta de conexión	Destino_CID, Origen_CID, Resultado, Estado
0x04	Solicitud de configuración	Destino_CID, Banderas, Opciones
0x05	Respuesta de configuración	Origen_CID, Banderas, Resultado, Configuración
0x06	Solicitud de desconexión	Destino_CID, Origen_CID
0x07	Respuesta de desconexión	Destino_CID, Origen_CID
0x08	Solicitud de eco	Dato
0x09	Respuesta de eco	Dato
0x0A	Solicitud de información	Tipo_de_información
0x0B	Respuesta de información	Dato, Tipo_de_información, Resultado

2.3.5. HCI

La Interfaz del Controlador de Huésped (HCI, *Host Controller Interface*) proporciona una interfaz de órdenes HCI para la comunicación entre el dispositivo y el firmware del módulo Bluetooth. Las órdenes HCI son utilizadas por el huésped (*host*) y las envía al Controlador del Huésped (HC, *Host Controller*), quien interpreta la orden HCI y la dirige a los componentes apropiados (radio Bluetooth, banda base, administrador de enlace, etc.) por medio de una interfaz física llamada transporte del huésped (*host transport*), la cual puede ser mediante los protocolos USB (*Universal Serial Bus*), RS-232 o UART [18].

En respuesta a las solicitudes enviadas por el huésped, el HC envía información relacionada con las órdenes HCI en forma de eventos HCI. Además, los eventos HCI son también utilizados para notificar al huésped cuando algo ocurre (Figura 2.22).

Para HCI existen diferentes formatos de paquetes dependiendo del tipo de información a transportar. Los paquetes pueden ser de órdenes HCI, eventos HCI o datos [18].

2.3.5.1. Formato de los paquetes de órdenes HCI

Los paquetes de órdenes HCI cuentan con un determinado número de parámetros dependiendo de su función (Figura 2.23):

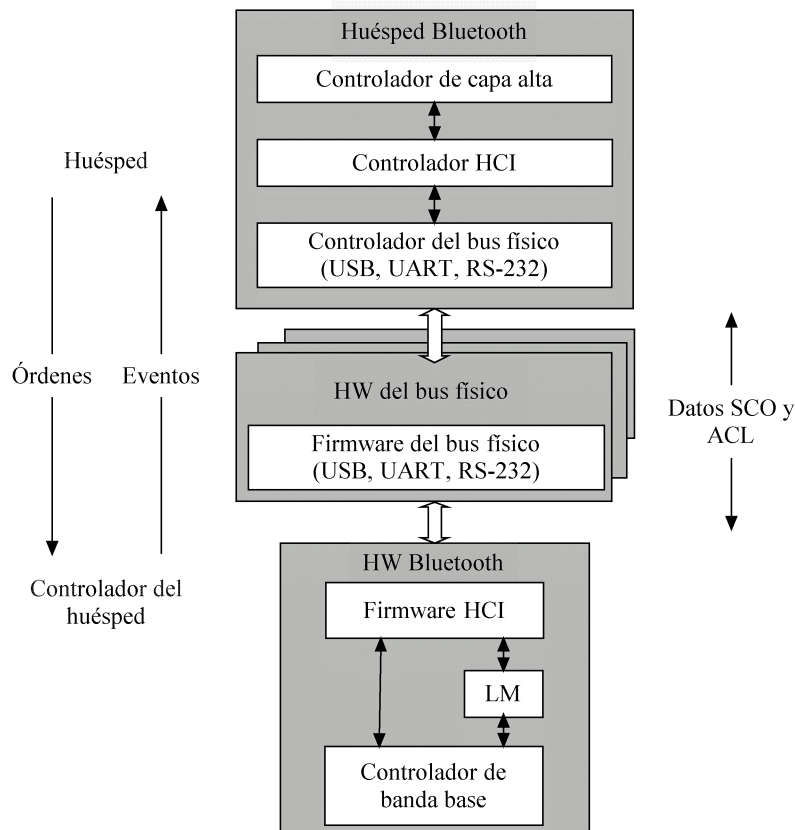


Figura 2.22. Intercambio de información entre el huésped y el HC [2].

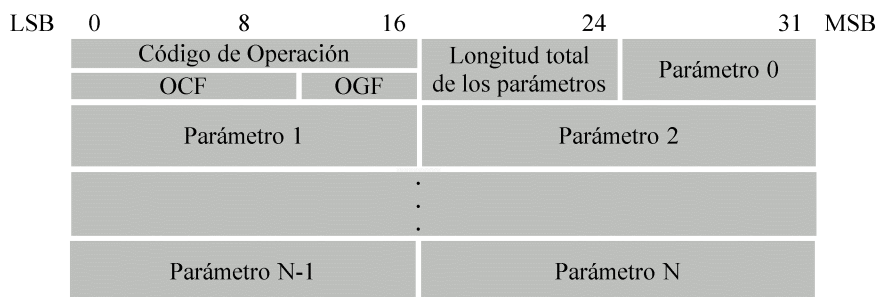


Figura 2.23. Formato de los paquetes de órdenes HCI [2].

- **Código de operación (OpCode):** A cada orden HCI se le asigna un campo de 2 bytes para su identificación. Este campo se divide en dos subcampos:
 - **Campo Orden del código de operación (OCF, OpCode Command Field):** Identifica el tipo de orden HCI y ocupa 10 bits del campo Código de operación.
 - **Campo Grupo del código de operación (OGF, OpCode Group Field):** Su función consiste en identificar el grupo al que pertenece la orden HCI y ocupa los 6 bits más significativos del campo Código de operación.
- **Longitud total de los parámetros:** Campo de un byte de longitud que indica la cantidad de bytes que utilizan todos los parámetros contenidos dentro del paquete.
- **Parámetro 0 hasta N:** Representan el conjunto de parámetros específicos de la orden HCI, en donde cada parámetro consta de 2 bytes.

En la Tabla 2.8 se muestran algunos ejemplos de los diferentes tipos de órdenes, también se incluye la información solicitada por algunas órdenes HCI en forma de parámetros. Los parámetros son devueltos al huésped a través del evento Orden Completa (*Command Complete*) cuando el HC procesa correctamente la orden HCI. En caso de que ocurra un error en la ejecución de la orden HCI, el evento HCI Estado de la Orden (*Command Status*) indica el código del error correspondiente [18].

Tabla 2.8. Tipos de órdenes HCI [2].

Orden HCI	OCF	Parámetros	Parámetros devueltos
Órdenes HCI de control del enlace (OGF=0x01)			
Sondeo	0x0001	LAP, Longitud_Sondeo, Num_Respuestas	-
Cancelar_Sondeo	0x0002	-	Estado
Modo_Sondeo_Periodico	0x0003	Longitud_Max_Periodo, Longitud_Min_Periodo, LAP, Longitud_Sondeo, Num_Respuestas	Estado
Crear_Conexion	0x0005	BD_ADDR, Tipo_paquete, Repetir_Modo_Escucha_Invitacion, Modo_Escucha_Invitacion, Desfase_Reloj, Permitir_Intercambio_Rol	-
Desconectar	0x0006	Manejador_Conexion, Razon	-
Órdenes de políticas del enlace (OGF=0x02)			
Modo_Reposo	0x0001	Manejador_Conexion, Intervalo_Max_Reposo, Intervalo_Min_Reposo	-
Modo_Sondeo	0x0003	Manejador_Conexion, Intervalo_Max_Sondeo, Intervalo_Min_Sondeo, Atencion_Sondeo, Tiempo_Espera_Sondeo	-
Salir_Modo_Sondeo	0x0004	Manejador_Conexion	-
Modo_Estacionario	0x0005	Manejador_Conexion, Intervalo_Max_Estacionario, Intervalo_Min_Estacionario	-
Salir_Modo_Estacionar	0x0006	Manejador_Conexion	-
Órdenes del HC y banda base (OGF=0x03)			
Restaurar	0x0003	-	Estado
Crear_Nueva_Clave	0x000B	-	Estado
Leer_Clave_Enlace	0x000D	BD_ADDR, Leer_Todas_Banderas	Estado, Num_Max_Claves, Leer_Num_Claves
Escribir_Clave_Enlace	0x0011	Num_Claves_A_Escribir, Clave_Enlace[i], BD_ADDR[i]	Estado, Num_Claves_Escritas
Borrar_Clave_Enlace	0x0012	BD_ADDR, Borrar_Todas_Banderas	Estado, Num_Claves_Borradas
Cambiar_Nombre_Local	0x0013	Nombre	Estado
Órdenes HCI de información (OGF=0x04)			
Leer_Informacion_Version_Local	0x0001	-	Estado, Version_HCI, Revision_HCI, Version_LMP, Fabrican-

			te, Subversion_LMP
Leer_Codigo_Pais	0x0007	-	Estado, Codigo_Pais
Leer_BD_ADDR	0x0009	-	Estado, BD_ADDR
Órdenes HCI de estado de la conexión (OGF=0x05)			
Leer_Fallo_Contacto	0x0001	Manejador_Conexion	Estado, Manejador_Conexion, Fallo_Contacto
Leer_RSSI	0x0005	Manejador_Conexion	Estado, Manejador_Conexion, RSSI

2.3.5.2. Formato de los paquetes de eventos HCI

La Figura 2.24 muestra el formato de los eventos HCI en unidades de bits. Al igual que las órdenes HCI, la longitud del paquete de cada evento HCI es variable.

El primer campo de los paquetes de eventos establece un Código del evento HCI único de longitud de un byte, mediante el cual se identifica el tipo de evento. A continuación está el campo Longitud total de los parámetros que indica la cantidad total de bytes de los parámetros correspondientes al evento. Por último, los campos Parámetro 0 hasta el N representan cada uno de los parámetros del evento. En la Tabla 2.9 se presentan los códigos y parámetros correspondientes a los eventos HCI.

Tabla 2.9. Códigos y parámetros de los eventos HCI [2].

Evento HCI	Código	Parámetros
Sondeo_Completo	0x01	Estado
Resultado_Sondeo	0x02	Num_Respuestas, BD_ADDR[i], Repetir_Modo_Escucha_Invitacion[i], Periodo_Modo_Escucha_Invitacion[i], Modo_Escucha_Invitacion[i], Clase_de_Dispositivo[i], Desfase_Reloj[i]
Conexion	0x03	Estado Completo, Manejador_Conexion, BD_ADDR, Tipo_Enlace, Modo_Encryptacion
Solicitud_Conexion	0x04	BD_ADDR, Clase_de_Dispositivo, Tipo_Enlace
Desconexion_Completa	0x05	Estado, Manejador_Conexion, Razon
Autenticacion_Completa	0x06	Estado, Manejador_Conexion
...
Prueba_Logo_Bluetooth	0xFE	-
Campo_Especifico_Proveedor	0xFF	-

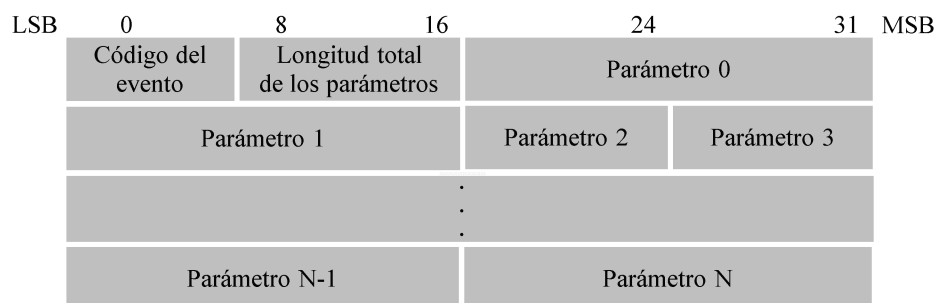


Figura 2.24. Formato de los paquetes de eventos HCI [2].

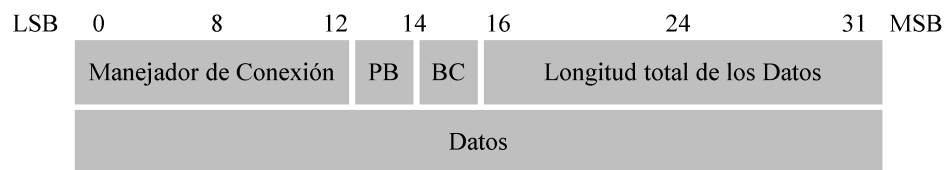


Figura 2.25. Formato de los paquetes de datos ACL [2].

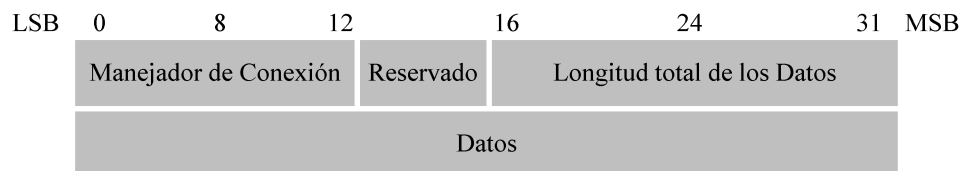


Figura 2.26. Formato de los paquetes de datos SCO [2].

2.3.5.3. Formato de los paquetes de datos HCI

Durante el establecimiento de algún tipo de conexión entre dos dispositivos Bluetooth, ACL o SCO, se tienen diferentes campos dependiendo del tipo de paquete:

- *Manejador de Conexión (Connection Handle)*: Tanto en paquetes ACL como en SCO, el manejador de conexión establece un identificador único de 12 bits de longitud que sólo conocen el huésped y el HC; y en ambos paquetes de datos, ACL (Figura 2.25) y SCO (Figura 2.26), el Manejador de Conexión es el primer campo a ser transmitido.
- *Límite del Paquete (PB, Packet Boundary) y Difusión (BC, Broadcast)*: En los paquetes ACL las banderas PB y BC ocupan 2 bits cada una. Los valores para las banderas y su significado se encuentran descritos en la Tabla 2.10.
- *Reservado*: En los paquetes SCO, estos bits están destinados para uso futuro.
- *Longitud Total de los Datos*: Este campo se encuentra en ambos tipos de paquetes de datos e indica la longitud total que comprenden los datos.
- *Datos*: Contiene información de otros protocolos como L2CAP. En los paquetes ACL dicha información consiste en datos y en los paquetes SCO consiste en voz.

2.3.6. SDP

El Protocolo de Descubrimiento de Servicio (SDP, *Service Discovery Protocol*) brinda los medios para la localización de servicios y atributos proporcionados por un dispositivo Bluetooth. Los servicios son entidades que pueden proveer información, ejecutar una acción o controlar un recurso en nombre de otra entidad, también pueden ser implementados como software, hardware o firmware. Los atributos definen las características de un servicio y por lo general incluyen el tipo o clase de servicio, y la información necesaria del mecanismo o protocolo para utilizar el servicio.

En las transacciones SDP se involucra una aplicación Servidor y una aplicación Cliente situadas en dispositivos remotos. La aplicación Servidor cuenta con una lista de registros en donde se encuentran almacenados los servicios y sus atributos, mientras que la aplicación Cliente tiene la capacidad de recuperar esta información. El flujo de la información entre la aplicación Servidor y la aplicación Cliente SDP se basa en el modelo de envío de solicitudes, hechas por la aplicación Cliente, y de las respuestas realizadas por la aplicación Servidor, que se transmiten por medio de SDP_PDU [2].

Tabla 2.10. Valores para las banderas PB y BC [2].

Valor	Descripción
Bandera PB	
00	Primer paquete del mensaje de capa superior
01	Continuar la fragmentación del paquete del mensaje de capa superior
10	Primer paquete del mensaje de capa superior rechazable
11	Paquete completo de L2CAP
Bandera BC en paquetes transmitidos del huésped al HC	
00	Sin difusión. Solo punto a punto
01	Difusión activa: El paquete es enviado a todos los nodos Esclavo activos y por nodos Esclavo en modo de sondeo
10	Difusión en picored: El paquete es enviado a todos los nodos Esclavo activos y en modo estacionario
11	Reservado para uso futuro
Bandera BC en paquetes transmitidos desde el HC al huésped	
00	Punto a punto
01	Paquete recibido por un nodo Esclavo que no se encuentre en modo estacionario
10	Paquete recibido por el nodo Esclavo en modo estacionario
11	Reservado para uso futuro

2.3.6.1. Formato de los paquetes SDP_PDU

La estructura de cada SDP_PDU contiene los siguientes campos (Figura 2.27):

- **Identificador del tipo de SDP_PDU (PDU ID):** Determina el significado del paquete y los parámetros que lo conforman.
- **ID de Transacción:** Almacena el valor de una solicitud SDP_PDU. Este valor se usa para emparejar respuestas a solicitudes SDP_PDU y cuenta con 2 bytes de longitud.
- **Longitud de Parámetros:** Contiene el número de bytes de los parámetros del SDP_PDU que son representados por los campos Parámetro 1... N. En la Tabla 2.11 se muestran los tipos de SDP_PDU con sus parámetros correspondientes.

Tabla 2.11. Tipos de SDP_PDU [18].

PDU ID	Descripción	Parámetros
0x00	Reservado	-
0x01	Respuesta del error	Codigo_del_Error, Informacion_del_Error
0x02	Solicitud de búsqueda de servicios	Servicio_de_Busqueda_de_Patrones, Maximo_Servicio_del_Registro, Continuacion_de_Estado
0x03	Respuesta de búsqueda de servicios	Servicios_Totales_del_Registro, Servicio_Actual_del_Registro, Lista_de_Servicios_del_Registro, Continuacion_de_Estado
0x04	Solicitud de atributos del servicio	Servicio_del_Registro, Atributos_Maximos_en_Bytes, Lista_de_Identidad_de_Atributos, Continuacion_de_Estado

0x05	Respuesta de atributos del servicio	Lista_de_Bytes_de_Atributos, Lista_de_Atributos, Continuacion_de_Estado
0x06	Solicitud de búsqueda de servicios y atributos	Servicio_de_Busqueda_de_Patrones, Atributos_Maximos_en_Bytes, Lista_de_Identidad_de_Atributos, Continuacion_de_Estado
0x07	Respuesta de búsqueda de servicios y atributos	Lista_de_Atributos_en_Bytes, Listas_de_Atributos, Continuacion_de_Estado
0x08 a 0xFF	Reservados	-

2.4. Capa de reemplazo de cables

En la capa de reemplazo de cables solo se encuentra un protocolo encargado de sustituir el cableado entre dispositivos Bluetooth mediante la emulación de una conexión serial.

2.4.1. RFCOMM

El protocolo de Comunicación por Radiofrecuencia (RFCOMM¹², *Radio Frequency Communication*) hace referencia a un modelo inalámbrico (RF) de un puerto COM virtual, y el SIG lo ha definido como un puerto serie para facilitar la migración de aplicaciones modeladas para ambientes seriales cableados a comunicaciones seriales inalámbricas.

RFCOMM utiliza una conexión L2CAP para iniciar un enlace lógico serie entre dos dispositivos. Solamente se permite una conexión RFCOMM entre dos dispositivos en un tiempo dado, no obstante, la conexión puede estar multiplexada para dar soporte a múltiples conexiones seriales entre los dispositivos. El primer Cliente RFCOMM establece la conexión RFCOMM sobre L2CAP y los demás usuarios de la conexión existente pueden hacer uso de las capacidades de multiplexación de RFCOMM para establecer nuevos canales sobre el enlace existente; y el último usuario en dejar el enlace serie RFCOMM deberá terminar la conexión RFCOMM.

La especificación permite hasta 60 enlaces lógicos seriales multiplexados sobre una conexión RFCOMM. RFCOMM soporta dos tipos de dispositivos:

- Punto final de comunicación (*communication endpoint*), p.e. PC y periféricos.
- Punto intermedio de comunicación (*communication midpoint*), p. e. un módem.

RFCOMM cumple un ambiente multiplexado, donde cada canal serie puede verse como una entidad independiente, con su propio conjunto de señales de datos y de control con la finalidad de definir un canal de control específico por medio del cual la información se transmita como datos. La velocidad de transmisión de la comunicación inalámbrica Bluetooth se determina por los tipos de paquetes y estructuras enviadas sobre la interfaz aérea [18].

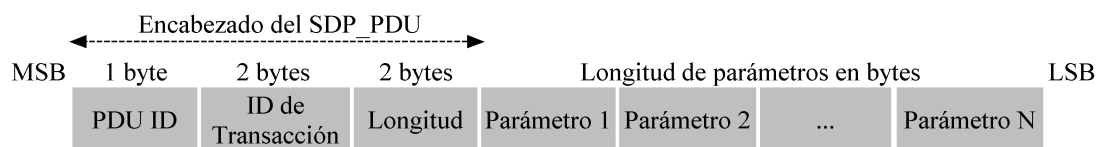


Figura 2.27. Formato de los paquetes SDP_PDU [18].

¹² El protocolo de emulación de comunicación serial RFCOMM está basado en el estándar 07.10 del ETSI (*European Telecommunications Standards Institute's Technical Standard*).

2.4.1.1. Formato de los paquetes RFCOMM

RFCOMM maneja diferentes tipos de paquetes en sus transmisiones:

- Establecimiento del Modo Asíncrono Balanceado (SABM, *Set Asynchronous Balanced Mode*).
- Respuesta de Reconocimiento No enumerado (UA, *Unnumbered Acknowledgement*).
- Respuesta de Modo Desconectado (DM, *Disconnected Mode*).
- Orden Desconectar (DISC, *Disconnect*).
- Orden/Respuesta de Información No enumerada con Encabezado de Verificación (UIH, *Unnumbered Information with Header Check*).

La Figura 2.28 muestra el formato de los paquetes RFCOMM. A continuación se describe cada uno de los campos del paquete RFCOMM:

- *Dirección*: este campo se compone de los bits (Figura 2.29):
 - *Dirección Extendida (EA, Extend Address)*: Se utiliza para extender la longitud del campo Dirección. Cuando EA es 1, significa que el byte que lo contiene es el último, mientras que cuando EA es 0 indica que el campo siguiente también pertenece al campo Dirección.
 - *Orden/Respuesta (C/R, Command/Response)*: Identifica al paquete como una orden o una respuesta. Los valores de C/R dependen del tipo de paquete.
 - *Identificador de Conexión de Enlace de Datos (DLCI, Data Link Connection Identifier)*: Cada conexión establecida entre una aplicación Servidor y un Cliente se identifica por un número de 6 bits (D1,..., D6). Los valores de DLCI se encuentran en el rango del 2 al 61, el DLCI con valor de 0 está reservado para el control de multiplexión. No se utilizan los valores 1, 62 y 63.
- *Control*: Contiene el tipo de paquete utilizado para solicitar órdenes relacionadas con la configuración de la conexión RFCOMM (velocidad de transmisión, control del flujo, bits de paro, etc.) y para finalizar respuestas a estas solicitudes.
- *Indicador de longitud*: Establece, en 7 ó 15 bits, la longitud de los datos contenidos en el campo Información. En el primer byte se encuentra un bit EA que indica si el campo Indicador de longitud está constituido por 1 (EA = 1) ó 2 bytes (EA = 0).
- *Información*: En este campo se almacenan los datos de los protocolos de capas superiores u órdenes en caso de que el canal sea de control (DLCI igual a 0). El campo Información únicamente se encuentra en los paquetes UIH.
- *Secuencia de Verificación de Trama (FCS, Frame Checking Sequence)*: Se almacena un valor utilizado para verificar la integridad de la trama. En el cálculo de este valor se requieren diferentes campos dependiendo del tipo de trama. Para las tramas UIH se requieren los campos Dirección y Control.



Figura 2.28. Formato de los paquetes RFCOMM [18].

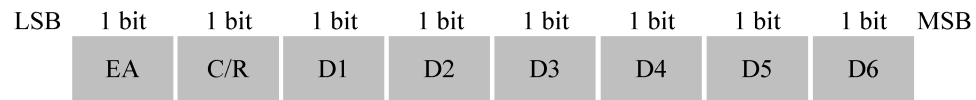


Figura 2.29. Subcampo Dirección [18].

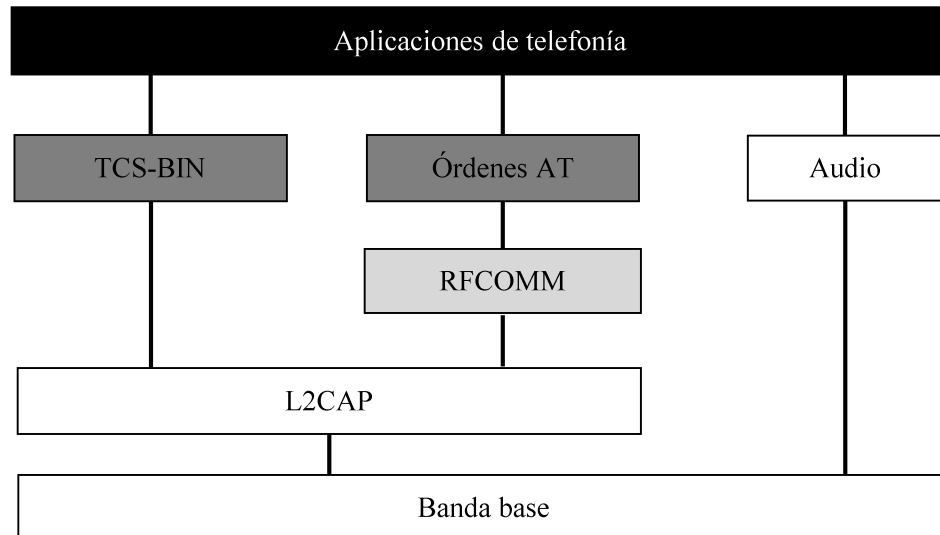


Figura 2.30. Control telefónico sobre órdenes AT y sobre el protocolo TCS-BIN.

2.5. Capa de protocolos de telefonía

La capa de protocolos de telefonía se basa en el protocolo de Control de Telefonía de Especificación Binaria (TCS-BIN¹³, *Telephony Control Specification Binary*) o mediante la ejecución de órdenes AT a través del puerto serie RFCOMM (Figura 2.30).

2.5.1. Protocolo TCS-BIN

El protocolo binario de control telefónico basado en paquetes fue adaptado de una especificación existente ITU-T (*International Telecommunications Union-Telecommunication*), y es un protocolo que define el control de llamadas de voz y datos entre dispositivos.

A principio de 1999 se llegó a la conclusión de que el futuro de las aplicaciones de control telefónico estaba orientado al estilo TCS-BIN, así mismo se observó que TCS-BIN proporcionaba todas las funciones necesarias para todos los perfiles basados en la telefonía.

TCS-BIN se puede usar para controlar llamadas de voz y datos. Cuando se realiza una llamada de voz, el elemento de audio de la arquitectura de protocolos se usa para cargar el contenido; en el caso de llamadas de datos, éstos pueden cargarse sobre las capas de transporte de la pila. TCS-BIN también define un método sin conexión (TCS-BIN sin conexión) para que los dispositivos intercambien información de señalización de llamadas sin tener una conexión de llamada establecida [21].

¹³ TCS-BIN está basado en la Recomendación Q.931 realizada por la ITU-T (*International Telecommunication Union-Telecommunications*).

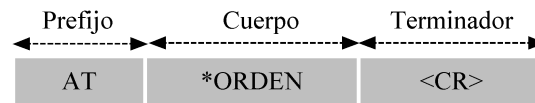


Figura 2.31. Formato de las órdenes AT.

2.5.2. Órdenes AT

Las órdenes AT¹⁴ están enfocadas al control y configuración de módems. Una orden AT se envía desde un módulo DTE hacia un módulo DCE para controlar funciones y servicios sobre una interfaz de red (en Bluetooth, la red inalámbrica) [28].

El conjunto de órdenes AT fue desarrollado en 1977 por Dennis Hayes como un interfaz de comunicación con un módem para configurarlo y proporcionarle instrucciones, tales como marcar un número de teléfono. Más adelante, fueron las compañías Microcomm y US Robotics quienes continuaron desarrollando y expandiendo el conjunto de órdenes AT [URL4].

La telefonía móvil GSM (*Global System for Mobile*) también ha adoptado como estándar dichas órdenes para comunicarse con sus terminales. De esta forma, todos los teléfonos móviles GSM¹⁵ poseen un conjunto de órdenes AT específico que sirve de interfaz para configurar y proporcionar instrucciones a los teléfonos móviles.

Las órdenes AT entre dispositivos Bluetooth tienen la función de establecer enlaces sobre un puerto serie, configurar parámetros de enlace, descubrir dispositivos, etc.

El presente trabajo de tesis se basa en la codificación y decodificación de órdenes AT sobre dispositivos remotos.

2.5.2.1. Formatos de los paquetes de órdenes AT

Los paquetes de órdenes AT se componen por los siguientes campos (Figura 2.31):

- *Prefijo*: El prefijo en cualquier orden es “AT” o “at”.
- *Cuerpo*: El cuerpo es una cadena de caracteres en el rango de 032-255 en formato ASCII.
- *Terminador*: El terminador es un retorno de carro <CR> (ASCII 013).

La Tabla 2.12 muestra los formatos para el ingreso de órdenes AT¹⁶ de lectura y escritura.

Tabla 2.12. Formato de órdenes AT [8].

Tipo de orden AT	Formato
Orden de lectura sin parámetros	AT<orden>?<CR>
Orden de escritura sin parámetros	AT<orden><CR>
Orden de escritura con parámetros	AT<orden>=<parámetro1>, ..., <parámetroN>, <CR>

Cuando un dispositivo Bluetooth recibe una orden AT, envía una contestación por medio de mensajes. La Tabla 2.13 lista los diferentes formatos de mensajes de respuesta a órdenes AT.

¹⁴ Las órdenes AT se basan en la Recomendación V.250 de la ITU-T.

¹⁵ En los terminales GSM el conjunto de órdenes AT puede encontrarse en la documentación técnica.

¹⁶ Los caracteres ‘?’ (ASCII 063) y ‘=’ (ASCII 061) sobre el cuerpo de las órdenes AT, indican que las órdenes son de lectura o de escritura respectivamente.

Tabla 2.13. Formatos de respuestas a órdenes AT [8].

Tipo de respuesta a orden AT	Formato
Mensaje final de éxito	<CR><LF>OK<CR><LF>
Mensaje con parámetros intermedios	<CR><LF><respuesta>:<parámetro1>, ..., <parámetroN> <CR><LF>OK<CR><LF>
Mensaje de error	<CR><LF>ERROR<CR><LF>

En los mensajes de respuesta, el carácter de salto de línea <LF> (ASCII 010) solo se utiliza para dar formato a las órdenes AT, mientras que se ignoran los caracteres <BS> (ASCII 008). La Tabla 2.14 lista los tipos de datos utilizados por las órdenes AT.

Tabla 2.14. Tipos de datos usados en las órdenes AT [8].

Tipo de dato	Descripción	Ejemplo
Cadena ¹⁷	Uno o más caracteres en formato ASCII (8 bits por carácter). Las cadenas deben de estar delimitadas por comillas	"D" "NODO A"
Entero	Consiste en una secuencia de caracteres en el rango del 0 a 9 (ASCII del 048 al 057). Las constantes numéricas se representan en formato decimal	0123
Enumerador	El tipo enumerador, es en realidad un tipo de dato entero; sin embargo, todos sus posibles valores son detallados al valor del enumerador	1. Modo a 2. Modo b 3. Modo c
Dirección del dispositivo Bluetooth	Tipo de dato utilizado para representar la dirección de un dispositivo Bluetooth. La dirección Bluetooth consiste de seis campos y dos valores hexadecimales por campo.	112233AABBCC 00803719c44f

En el Anexo órdenes AT se describen los parámetros y formatos de todas las órdenes AT utilizadas en el presente trabajo de tesis.

¹⁷ Si en la cadena se utilizan comillas, éstas deben representarse por "\22". De la misma forma si se usa el caracter de diagonal invertida, deberá representarse por "\5c".

3. Diseño y Desarrollo del Sistema DreamBlue

En este capítulo se presenta el diseño y desarrollo del sistema DreamBlue con base en la metodología SPIES.

3.1. Metodología SPIES

La metodología de desarrollo para sistemas empotrados SPIES consiste de un lenguaje de modelado para especificar relaciones entre componentes, elementos y relaciones de un sistema en un conjunto de procesos (actividades, productos, entradas, salidas, métricas, criterios de entrada, criterios de salida, funciones y más) que le indican al desarrollador qué partes del lenguaje utilizar, y cómo y cuándo usarlas. SPIES especifica un conjunto integrado de actividades para guiar al desarrollador en todos los ciclos de vida de los procesos involucrados [12].

SPIES utiliza el Lenguaje Unificado de Modelado (UML, *Unified Modeling Language*) para representar las fases que integran la metodología [4], [23]. El enfoque iterativo de la metodología asegura que el sistema sea probado en cada fase y no hasta el final. La metodología contiene un repositorio de conocimientos compuesto de plantillas para cada actividad, las cuales gestionan toda la información sobre el proyecto para, de esta forma, diseñar sistemas robustos, capaces y seguros. El proceso de desarrollo de la metodología, tiene un enfoque *top-down* con énfasis en la mejora continua a lo largo del tiempo de desarrollo [10].

SPIES recomienda para cada actividad el uso de una herramienta SW específica como Rhapsody [URL3] para la modelación de requerimientos, la validación del sistema y demás actividades. SPIES utiliza la idea básica del Proceso en Equipo para el desarrollo de Software (TSP, *Team Software Process*), ya que establece la documentación que guía al desarrollador durante la gestión del proyecto.

La metodología SPIES se ha diseñado a partir de las áreas de procesos de la especificación CMMI-DEV v1.2 Nivel 2 [6], dicho nivel ha sido adaptado para establecer un conjunto de requerimientos para el diseño de sistemas empotrados.

La metodología SPIES está compuesta por tres capas y ocho fases, donde las capas son dependientes entre sí y se relacionan de forma progresiva. La Figura 3.1 muestra la relación entre las capas y las fases de SPIES [12].

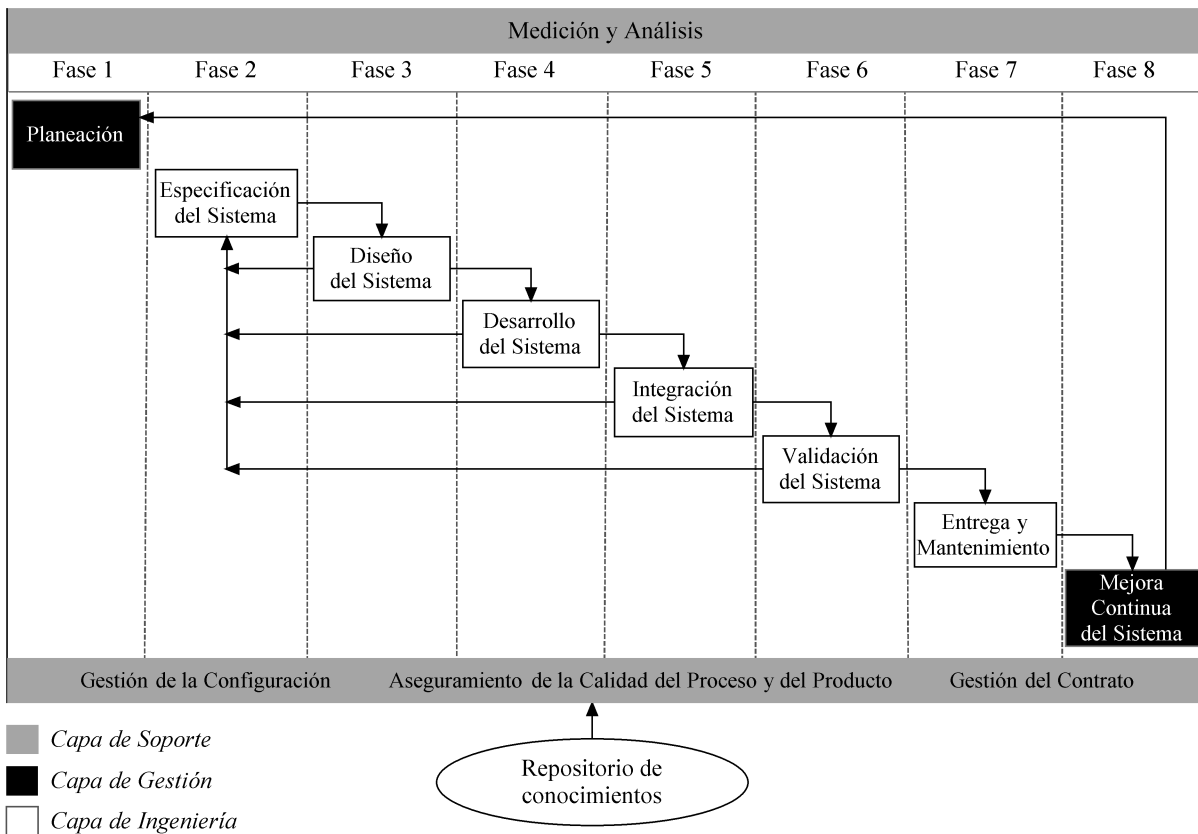


Figura 3.1. Relación entre las capas y las fases de SPIES¹⁸ [12].

A continuación, se describen las capas de la metodología:

- *Capa de gestión:* Proporciona los procesos necesarios para controlar la totalidad del desarrollo del proyecto. La capa está compuesta por dos fases: Planeación y Mejora Continua del Sistema.
- *Capa de ingeniería:* Proporciona los procesos relacionados con el desarrollo completo del sistema. Esta capa está compuesta por las siguientes fases: Especificación de Requerimientos, Diseño del Sistema, Desarrollo del Sistema, Integración del Sistema, Validación del Sistema, y Liberación y Mantenimiento del Sistema.
- *Capa de soporte:* Proporciona ayuda para lograr la calidad esperada a través de actividades establecidas para la configuración, gestión de contratos y la medición y análisis continuos de las fases de la metodología. La capa está compuesta por 4 partes:
 - *Gestión de la Configuración:* Se proporcionan datos de configuración e información actual del sistema a los desarrolladores y usuarios finales.
 - *Aseguramiento de la Calidad del Proceso y del Sistema:* Se proporciona una perspectiva sobre el cumplimiento de los requerimientos del sistema para asegurar la calidad del sistema.

¹⁸ En el documento de especificaciones de la metodología SPIES las fases hacen referencia al producto (Especificación del Producto, Diseño del Producto, etc.).

- *Gestión del Contrato*: Se establece la negociación de los términos y condiciones para garantizar su cumplimiento en los contratos, así como la documentación y aceptar los cambios que puedan surgir en su aplicación o ejecución.
- *Medición y Análisis*: La medición y análisis es un requisito para todas las fases de la metodología y asegura una mejora continua del sistema. La medición se usa para dos propósitos: Evaluar la calidad del sistema empotrado y Evaluar las relaciones de los procesos del sistema.
- *Repositorio de conocimientos*: El repositorio se encarga de gestionar toda la información sobre el proyecto para el diseño del sistema.

La Figura 3.2 muestra la secuencia de las fases de SPIES:

1. *Planeación*: En esta fase se crea una guía para establecer las actividades del proyecto considerando las especificaciones y las tareas que deba cumplir el sistema. Se determinan los recursos para la elaboración del proyecto, se establece un calendario de actividades, se identifican y analizan los riesgos del proyecto, y se establece el dispositivo en el cual se va a desarrollar el sistema.
2. *Especificación de Requerimientos*: En esta fase se capturan los requerimientos de desarrollo y de administración del sistema. La descripción de los requerimientos se realiza mediante diagramas de caso de uso y diagramas de requerimientos en UML.
3. *Diseño del Sistema*: En esta fase el diseño del sistema se realiza mediante diseños funcionales con base en la fase de requerimientos. Los diseños funcionales examinan las operaciones del sistema sin considerar detalles sobre técnicas en la implementación. Los diseños funcionales se representan con: diagramas de estructura, diagramas de secuencia y diagramas de máquinas de estado para modelar el funcionamiento completo del sistema.
4. *Desarrollo del Sistema*: La fase de desarrollo del sistema se compone de los siguientes procesos:
 - a. *Desarrollo de otras tecnologías*: Proceso en el cual se elaboran prácticas para desarrollar sistemas empotrados sin dependencias tecnológicas.
 - b. *Desarrollo electromecánico*: Proceso en el que se construyen actividades para manejar y sincronizar elementos de electricidad y elementos mecánicos.
 - c. *Desarrollo automatizado*: Proceso donde se elaboran prácticas para integrar todos los componentes del sistema en un modelo unificado y realizar simulaciones completas del sistema.
 - d. *Desarrollo electrónico*: Proceso donde se diseña y desarrolla el sistema electrónico mediante el uso de herramientas de análisis y construcción del HW. Así mismo se elige el dispositivo en donde se implementará el sistema [11].
 - e. *Desarrollo del software*: Proceso donde se construye el software que controlará el sistema; posteriormente, se refina o extiende el modelado del sistema independientemente de la plataforma de diseño, de esta forma, es posible programar código más eficientemente sobre la tarjeta que realizará la ejecución del sistema.

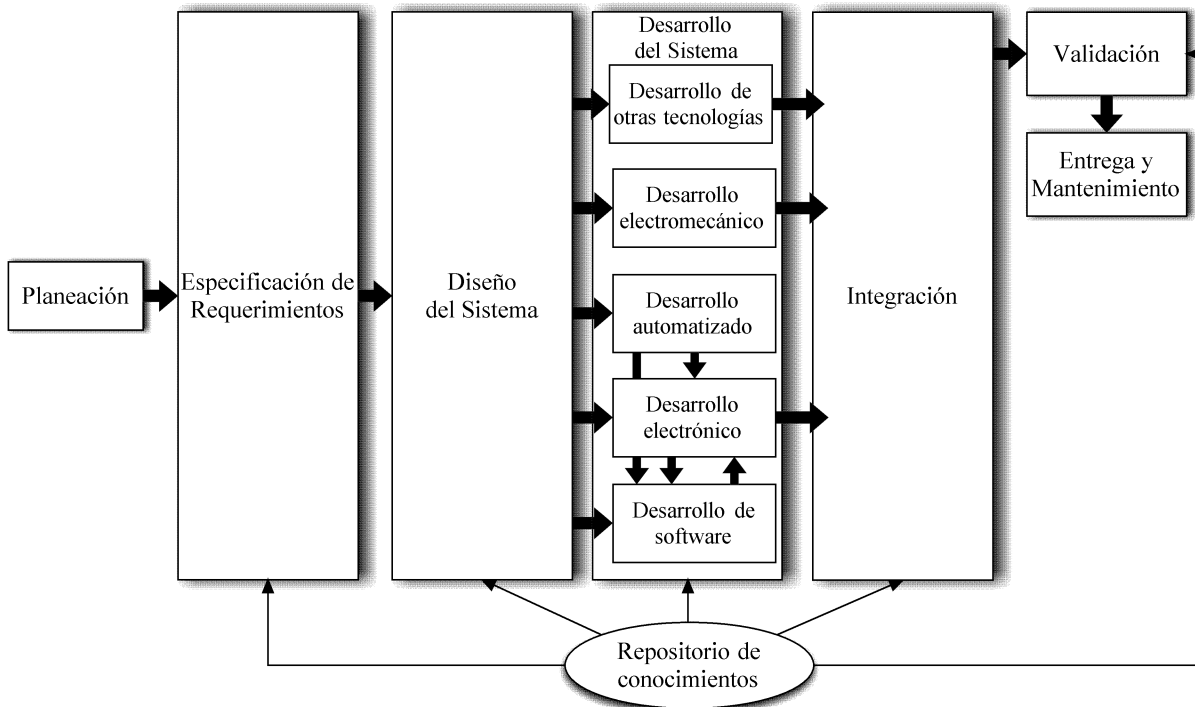


Figura 3.2. Secuencia de las fases de SPIES [12].

5. *Integración*: En esta fase se proporcionan actividades para integrar los componentes en varios pasos antes de que se evalué el sistema. La fase de integración de software y hardware se realiza mediante pruebas de depuración.
6. *Validación del Sistema*: En esta fase se realizan pruebas de campo antes de que el sistema empotrado esté listo para su liberación y su posterior mantenimiento.
7. *Entrega y Mantenimiento del Sistema*: En esta fase los desarrolladores deben usar la documentación ya existente del sistema para entender el funcionamiento y darle mantenimiento.
8. *Mejora Continua del Sistema*: Última fase donde se realizan mejoras continuas del producto, y de ser necesario, regresar a la fase de especificación de requerimientos.

A continuación se describen las fases definidas por la metodología para el diseño y construcción del sistema DreamBlue¹⁹.

3.2. Especificación de requerimientos

El propósito de la fase de especificación de requerimientos de la metodología SPIES es producir y analizar tanto los requerimientos del sistema como los requerimientos de los componentes del sistema [12]. La Figura 3.3 muestra el diagrama de requerimientos generales del sistema DreamBlue.

El diagrama de caso de uso general del sistema se forma por dos objetos Nodo y por dos actores: Usuario y Picored (Figura 3.4).

¹⁹ Durante el diseño y desarrollo del sistema DreamBlue solo se emplearon las fases correspondientes a la capa de ingeniería de la metodología SPIES.

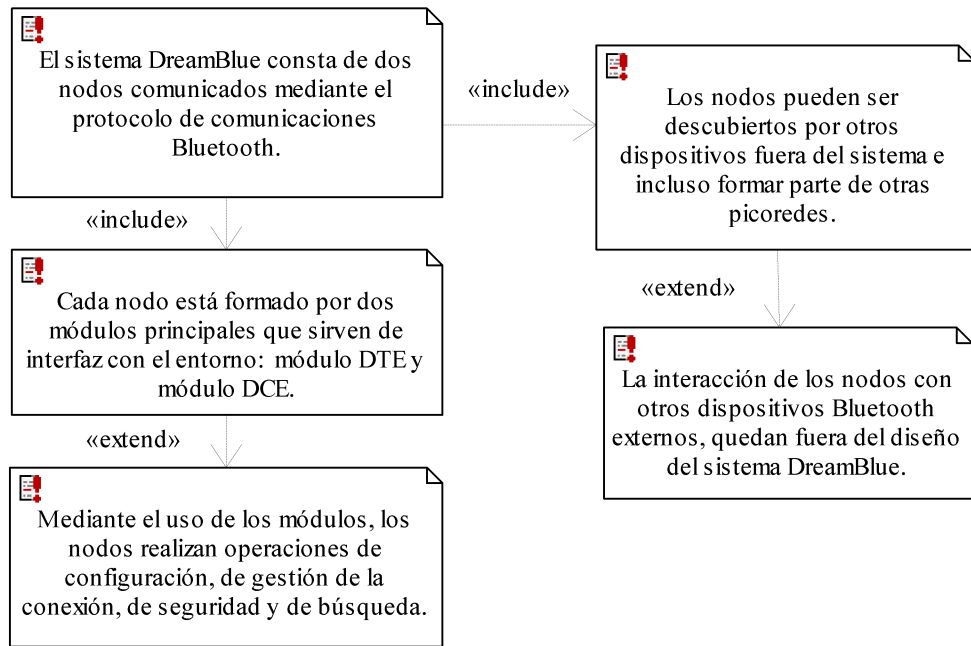


Figura 3.3. Diagrama de requerimientos generales del sistema DreamBlue.

Los dos actores Usuario de la Figura 3.4 pueden ser dos personas diferentes o puede tratarse de una sola persona²⁰; así mismo, la caja System Boundary Box representa la funcionalidad total del sistema DreamBlue que encierra a los casos de uso para el manejo de los nodos²¹.

Dentro del diagrama de caso de uso para el nodo (Figura 3.5) se incluyen los casos de uso para los módulos DTE y DCE. La especialización Modos de Operación representa los diferentes modos de operación en que puede funcionar el nodo, éstos dependen de la interacción entre los módulos DTE y DCE. La Figura 3.5 muestra la interacción que existe entre los actores del sistema dentro de un nodo y un tercer actor RS232²².

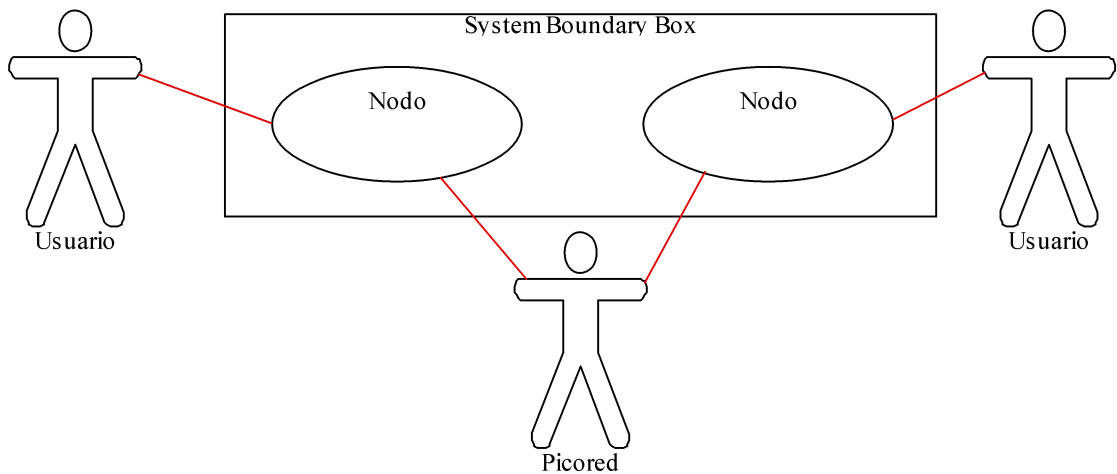


Figura 3.4. Diagrama de caso de uso general para el sistema DreamBlue.

²⁰ En los diagramas de caso de uso no existe distinción entre usuarios o nodos.

²¹ Ambos nodos del sistema tienen el mismo modelado de diagramas de casos de uso.

²² El actor RS232 sólo simboliza la comunicación interna entre los módulos de un nodo.

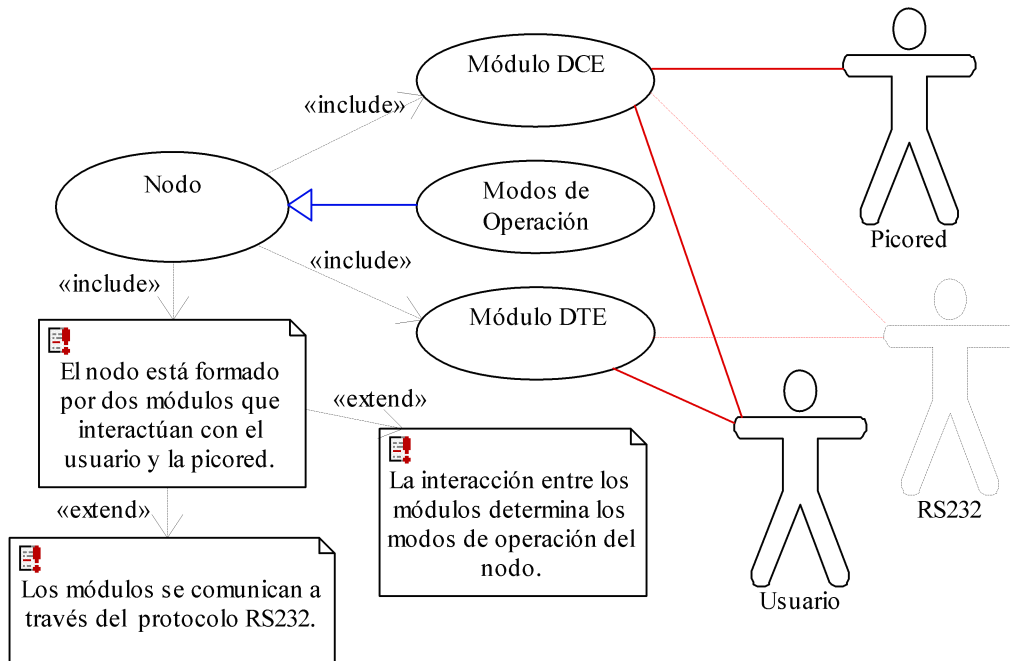


Figura 3.5. Diagrama de caso de uso para nodo del sistema DreamBlue.

3.2.1. Módulo DTE

El módulo DTE permite al usuario configurar y acceder a las diferentes funciones del nodo. Algunas de las actividades que puede realizar el módulo DTE son:

- Interactuar con el usuario a través de un teclado y una pantalla.
- Comunicarse por medio del protocolo RS232 con el módulo DCE.
- Reiniciar el módulo DTE.

El módulo DTE posee dos interfaces para comunicarse con su entorno:

- *Interfaz de usuario del módulo DTE (UI_DTE)*: Establece la comunicación entre el nodo y el usuario.
- *Interfaz RS232 del módulo DTE (COM_DTE)*: Comunica y acopla al módulo DTE con el módulo DCE.

La Tabla 3.1 expone los componentes y funciones de cada interfaz del módulo DTE.

Tabla 3.1 Componentes de las interfaces del módulo DTE.

Interfaz	Componente	Función
UI_DTE	Pantalla GLCD	Medio por donde el usuario visualiza las operaciones del nodo
	Teclado matricial	Medio por donde el usuario selecciona opciones del menú, ingresa parámetros de configuración e introduce mensajes de texto
	Interruptor de reinicio	Reinicia el módulo DTE
COM_DTE	Transceptor RS232	Ajuste de niveles de voltaje entre el módulo DCE y el módulo DTE
	Conector DB9 hembra	Conector por donde el módulo DTE se ensambla con el módulo DCE

La Figura 3.6 muestra el diagrama de caso de uso para el módulo DTE con sus respectivas interfaces. La Figura 3.7 muestra el diagrama de caso de uso para la UI_DTE, el cual incluye los componentes pertenecientes a la UI_DTE. La Figura 3.8 muestra el diagrama de caso de uso para la COM_DTE con sus respectivos componentes.

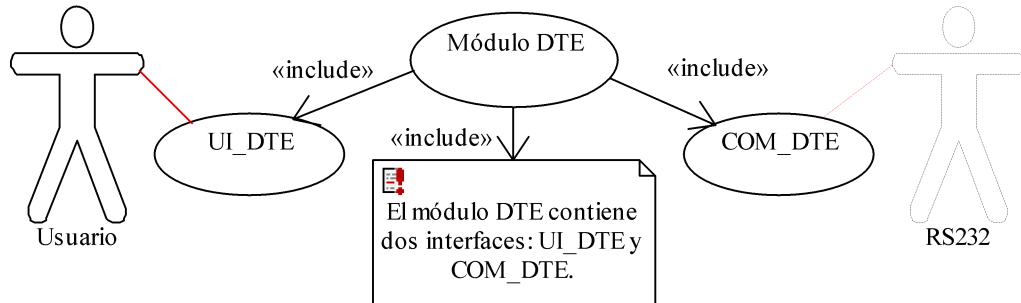


Figura 3.6. Diagrama de caso de uso para el módulo DTE.

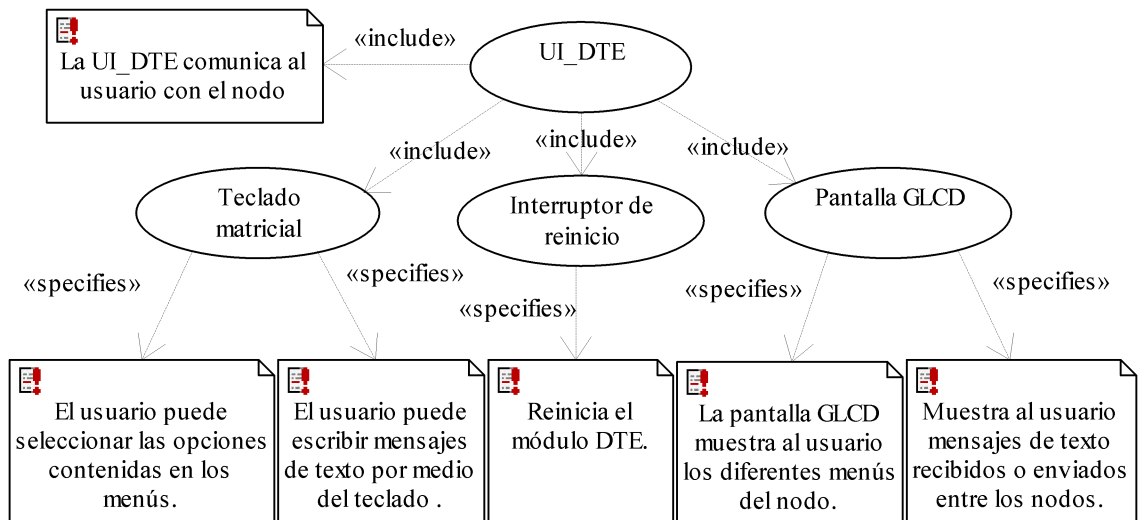


Figura 3.7. Diagrama de caso de uso para la UI_DTE.

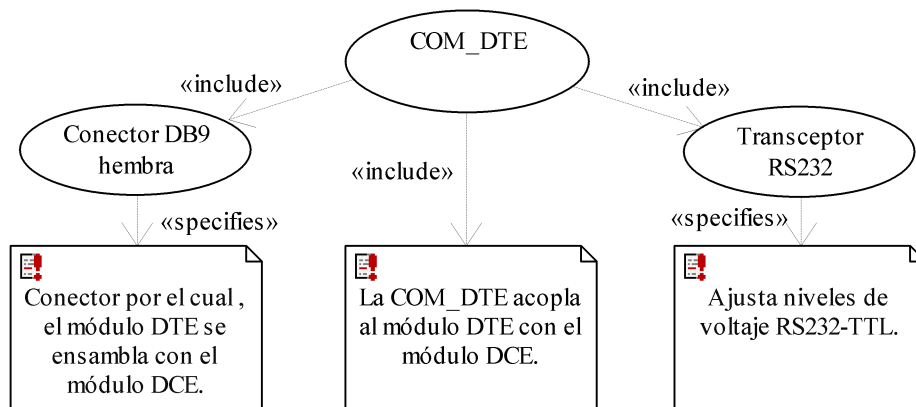


Figura 3.8. Diagrama de caso de uso para COM_DTE.

3.2.2. Módulo DCE

El módulo DCE permite al nodo comunicarse con una picored (o picoredes). Algunas de las actividades que puede realizar el módulo DCE son:

- Comunicarse por medio del protocolo RS232 con el módulo DTE.
- Recibir y transmitir datos a través de la banda ISM a 2.4 GHz.
- Mostrar el estado de operación actual del nodo a través de un LED RGB.
- Restablecer las funciones y parámetro a sus valores por defecto del nodo.

El módulo DCE cuenta con tres interfaces para comunicarse con su entorno:

- *Interfaz de usuario del módulo DCE (UI_DCE)*: Mediante la UI_DCE el usuario puede observar el modo de operación del nodo o restablecer la configuración por defecto del módulo DCE.
- *Interfaz RS232 del módulo DCE (COM_DCE)*: Acopla al módulo DCE con el módulo DTE.
- *Interfaz de comunicación del nodo con la picored (CINP)*: Mediante la CINP el nodo se comunica con la picored (o picoredes).

La Tabla 3.2 lista las funciones de los componentes de las interfaces del módulo DCE.

Tabla 3.2. Componentes de las interfaces del módulo DCE.

Interfaz	Componentes	Función
UI_DCE	LED RGB	Verificación del modo de operación del nodo
	Interruptor de restauración	Reestablece los parámetros del módulo DCE a sus valores por defecto
COM_DCE	Conector DB9 macho	Ensambla al módulo DCE con el módulo DTE
CINP	Transceptor RS232	Ajusta las operaciones de transmisión y recepción a niveles de voltaje RS232
	Radio Bluetooth	Permite transmitir datos en la banda ISM a 2.4 GHz
	Controlador de banda base	Accede al medio y establece los enlaces entre los nodos
	Antena Bluetooth	Recibe y transmite ondas electromagnéticas a través del aire

El LED RGB de la UI_DCE indica al usuario el modo de operación actual del nodo, en la Tabla 3.3 se muestran todas las combinaciones y sus respectivos significados.

Tabla 3.3. Combinaciones y significado del LED RGB.

Color	Descripción
Verde	El nodo se encuentra en modo de datos y no se intenta establecer alguna conexión
Naranja	El nodo se encuentra en modo de operación de Órdenes AT
Púrpura	Intento de conexión en progreso
Azul	Existe una conexión activa
Azul intermitente	Existe una conexión activa, y además, se reciben o transmiten datos a través del aire
Rojo intermitente	Desbordamiento del buffer, error de paridad o de trama detectado sobre la UART

La Figura 3.9 muestra el diagrama de caso de uso para el módulo DCE, el cual incluye la interacción entre los actores y las interfaces del módulo DCE.

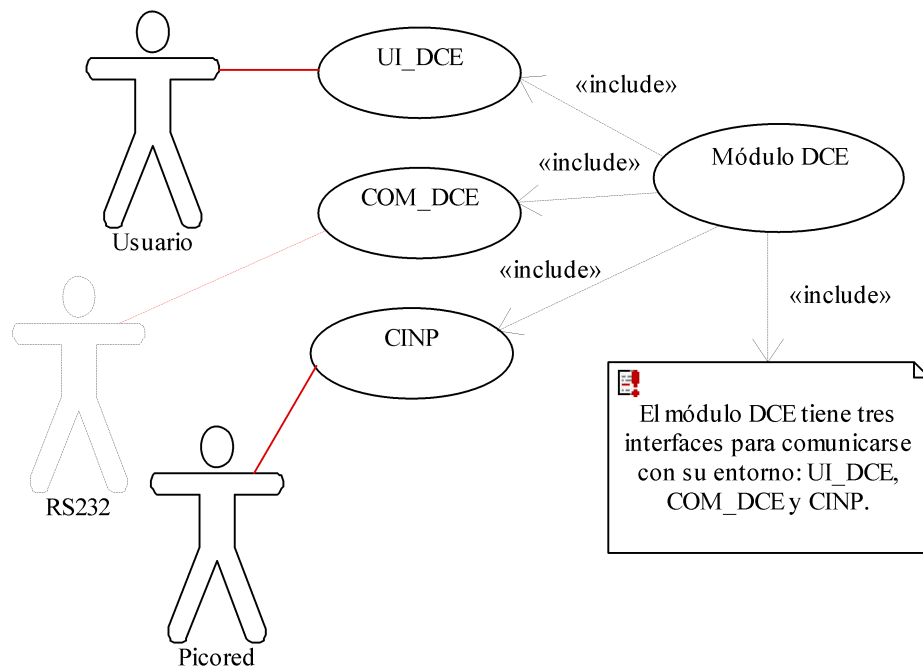


Figura 3.9 Diagrama de caso de uso para el módulo DTE.

En la Figura 3.10 se muestra el diagrama de caso de uso para la UI_DCE.

La Figura 3.11 muestra el diagrama de caso de uso para el LED RGB, en donde los colores del LED RGB indican especializaciones del nodo.

La COM_DCE mediante el uso de un conector DB9 macho, acopla el módulo DCE al módulo DTE. La Figura 3.12 muestra el diagrama de caso de uso para la COM_DCE.

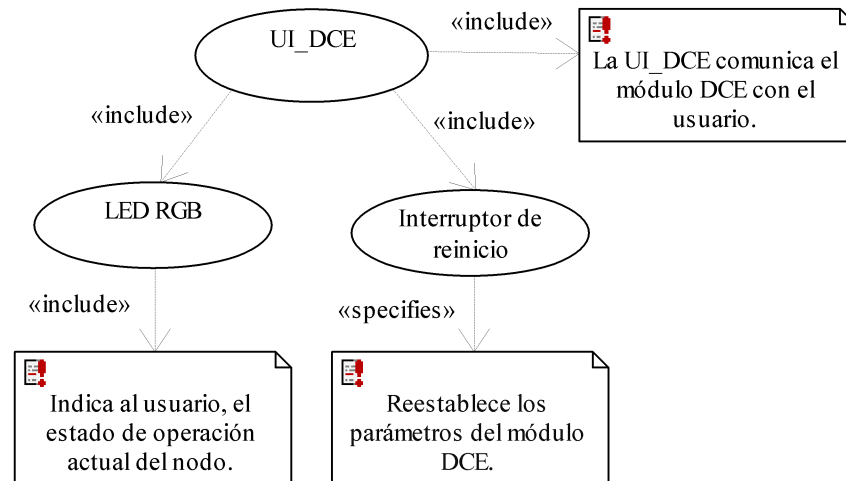


Figura 3.10. Diagrama de caso de uso para la UI_DCE.

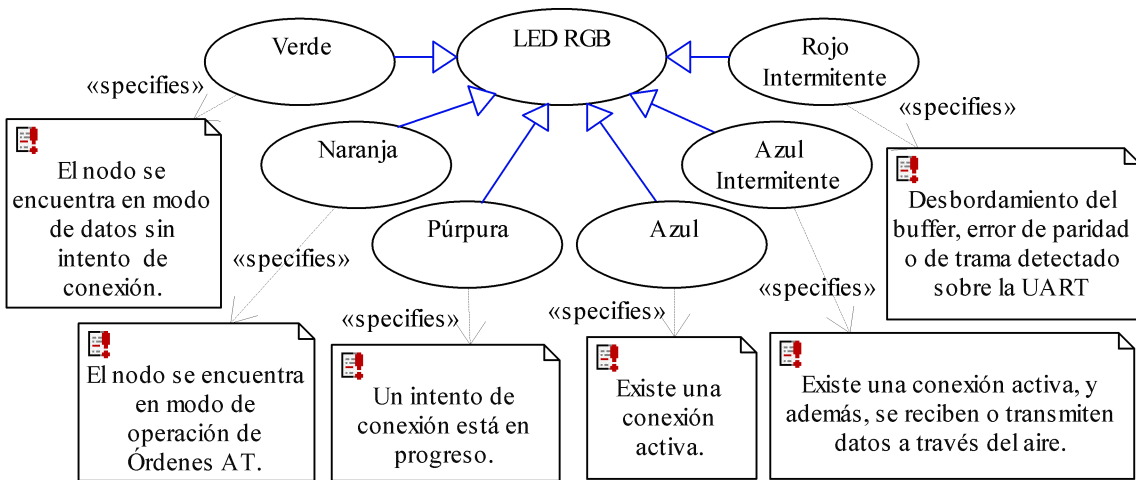


Figura 3.11. Diagrama de caso de uso para el LED RGB.

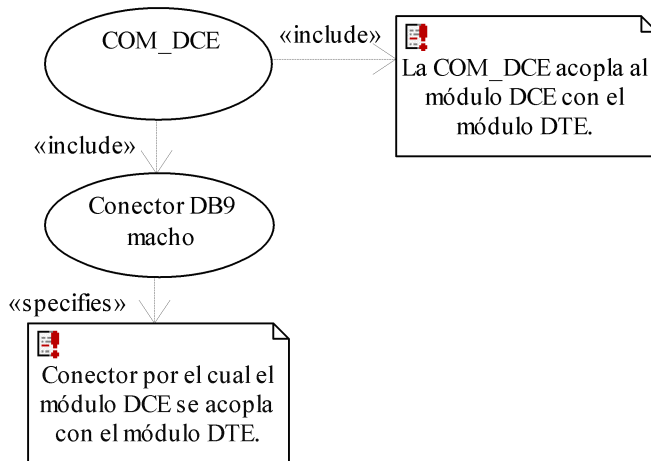


Figura 3.12. Diagrama de caso de uso para la COM_DCE.

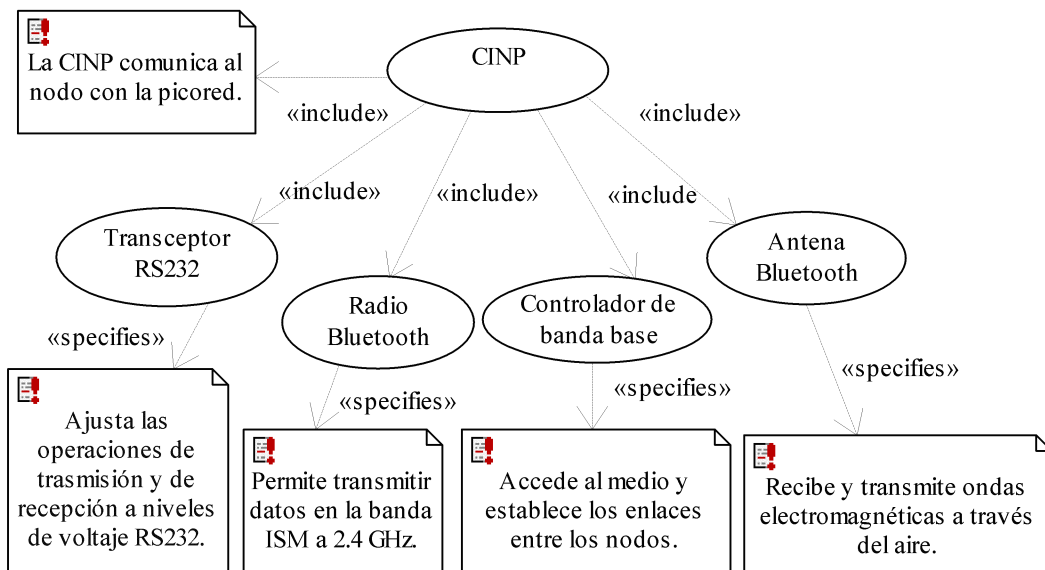


Figura 3.13. Diagrama de caso de uso para la CINP.

Por último, la CINP comunica al nodo con la picored mediante el uso de los componentes: Transceptor RS232, Radio Bluetooth, Controlador de Banda base y Antena Bluetooth. La Figura 3.13 muestra el diagrama de caso de uso para la COM_DCE.

En el anexo DCE se muestran las características eléctricas y mecánicas de los componentes que integran al módulo DCE.

3.2.3. Modos de operación

Los nodos cuentan con dos modos de operación principales (Figura 3.14), con los cuales el usuario decide qué actividades realizarán:

- *Modo de Órdenes AT*²³: Modo de configuración de los nodos.
- *Modo de Datos*: Modo de envío y recepción de mensajes de los nodos.

La Tabla 3.4 muestra la estructura del modelado para los modos de operación de los nodos.

Tabla 3.4. Estructura de modelado del sistema DreamBlue.

Modo de operación	Subsistema	Generalización
Modo de Órdenes AT	Configuración del Nodo	Configuración RS232
		Modo Conectable
		Modo Detectable
		Nombre y Clase
		Restaurar Nodo
	Gestión de la Conexión	Modo Servidor
		Modo Cliente
		Modo Ahorro
	Servicios Suplementarios	Modo Sondeo
		Modo Seguridad
Modo de Datos	Gestión de Mensajes	Modo Transmisión
		Modo Recepción

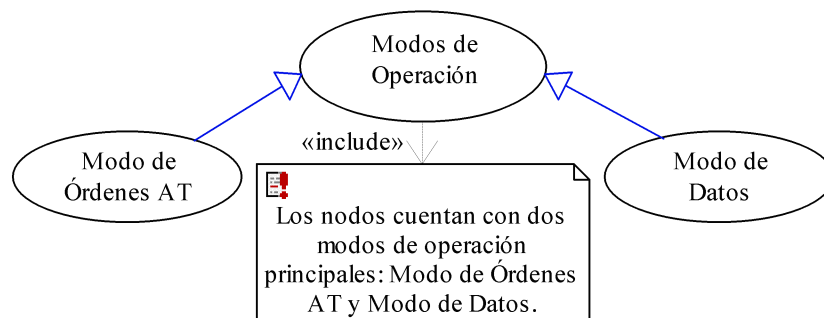


Figura 3.14. Diagrama de caso de uso para los modos de operación.

²³ Se hace la distinción entre Órdenes AT y órdenes AT, la primera hace referencia al modo de operación de los nodos DreamBlue y la segunda al envío de órdenes AT entre los módulos DTE y DCE.

Existen métodos “no visibles” que controlan las operaciones y el acceso a las actividades que realice el usuario sobre los nodos. La Tabla 3.5 muestra algunos de estos métodos utilizados para la gestión de las actividades de los nodos.

Tabla 3.5. Métodos no visibles.

Método no visible	Descripción
Inicia_Interfaces	Inicializa todas las interfaces a ocupar del MCU
Tecla_Opcion	Captura y decodifica la tecla presionada por el usuario
Espera_Opcion	Espera a que el usuario seleccione alguna opción mediante el teclado matricial
Secuencia_Escape	Envía la secuencia de escape para ingresar al modo de operación de Órdenes AT
Orden_AT_Lectura	Captura la orden ingresada por el usuario (y parámetros correspondientes) por medio del teclado
Enviar_Orden_AT	Envía la orden AT ingresada por el usuario del módulo DTE al módulo DCE
Respuesta_Orden_AT	Muestra en la pantalla GLCD la respuesta obtenida por el módulo DCE a petición de una orden AT
Menu_Principal	Muestra el menú principal al usuario
Menu_Modo_Datos	Muestra el menú del modos de operación de Datos
Menu_Modo_Ordenes_AT	Muestra el menú del modo de operación de Órdenes AT
Menu_Configuracion_Nodo	Muestra el menú del subsistema Configuración del Nodo
Menu_Gestion_Conexion	Muestra el menú del subsistema Gestión Conexión
Menu_Servicios_Suplementarios	Muestra el menú del subsistema Servicios Suplementarios
Menu_Gestion_Mensajes	Muestra el menú del subsistema Gestión de Mensajes

A continuación se describen los dos modos de operación de los nodos, así como también los subsistemas y métodos que los conforman.

3.2.3.1. Modo de Órdenes AT

En el modo de operación de Órdenes AT, el usuario configura los nodos mediante el envío de órdenes AT del módulo DTE al módulo DCE.

Para acceder al modo de Órdenes AT en cualquiera de los nodos, el usuario debe ingresar al menú principal y seleccionar el método `Modo_Ordenes_AT`. El nodo espera a que el módulo DTE envíe una secuencia de escape, la cual consiste de tres caracteres consecutivos ‘/’.

El módulo DCE reconoce como válida a la secuencia de escape solo sí:

- Antes y después de introducir la secuencia de escape el nodo no realiza operaciones en un lapso de 1 segundo (*silence*).
- La secuencia completa de escape se envía en menos de 200 ms.

La estructura del modo de Órdenes AT está compuesta por tres subsistemas: Configuración del Nodo, Gestión de la Conexión y Servicios Suplementarios, como muestra la Figura 3.15.

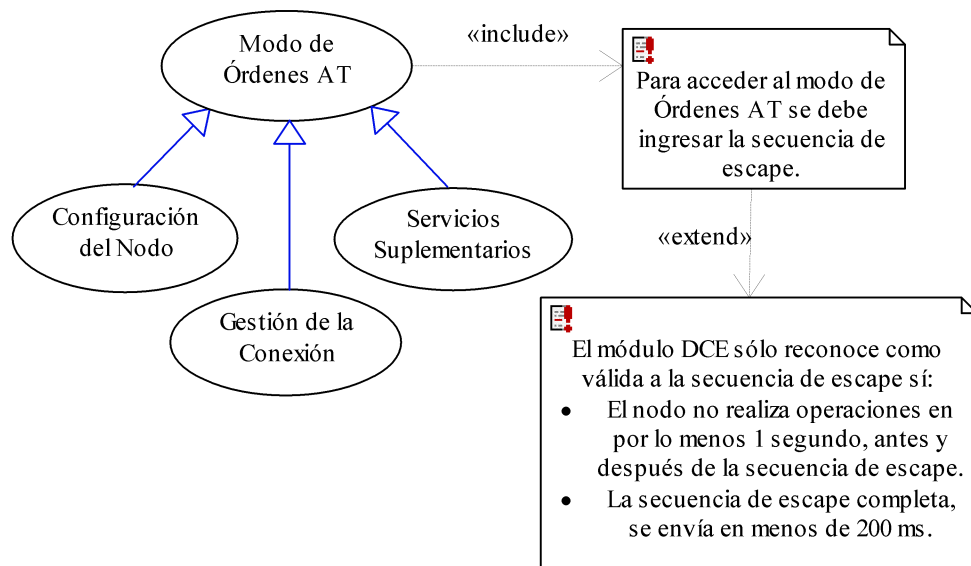


Figura 3.15. Diagrama de caso de uso para el modo de Órdenes AT.

Dentro de los subsistemas, algunos métodos solicitarán al usuario ingresar parámetros referentes a las órdenes AT invocadas. Los caracteres de estos parámetros se introducen con el método Seleccionar (Select) y posteriormente con el método Enviar_Orden_AT (OK), el usuario envía la orden AT del módulo DTE al módulo DCE.

En caso de equivocación durante la escritura de los parámetros, el usuario puede borrar caracteres con el método Borrar (Erase) y por medio del método Cancelar (Cancel), el usuario cancela y regresa al menú donde se encontraba el método seleccionado.

3.2.3.1.1. Subsistema Configuración del Nodo

Dentro del subsistema Configuración del Nodo existen generalizaciones con las que el usuario puede realizar las siguientes actividades sobre cualquiera de los nodos:

- Determinar si los nodos pueden o no ser vistos por otros dispositivos Bluetooth.
- Ajustar los nodos para determinar si pueden o no conectarse a otros dispositivos Bluetooth.
- Designarle a los nodos un nombre local.
- Asociarle a los nodos pertenencia a alguna clase de dispositivo Bluetooth.
- Configurar parámetros RS232 de los nodos.
- Restablecer los nodos a sus configuraciones por defecto.

Las generalizaciones para el subsistema Configuración del Nodo son:

- *Configuración RS232:* Los nodos no permiten la configuración RS232 de forma automática (velocidad de transmisión, bits de datos, bits de stop, etc.). Para ajustar la configuración RS232 de cualquier nodo, el usuario debe usar el método Escribir_Configuracion_RS232 (AT*AMRS=). Si el usuario desea consultar los parámetros actuales de configuración RS232 que existen sobre el nodo, debe utilizar el método Leer_Configuracion_RS232 (AT*AMRS?).

- *Modo Conectable*: El usuario puede determinar por medio del método `Escribir_Modo_Conectable (AT*AGCM=)` si los nodos pueden o no conectarse a otros dispositivo Bluetooth dentro de su vecindad; igualmente, el usuario puede obtener de un nodo su modo de conectividad actual mediante el método `Leer_Modo_Conectable (AT*AGCM?)`.
- *Modo Detectable*: El usuario puede determinar por medio del método `Escribir_Modo_Detectable (AT*AGDM=)`, si los nodos pueden o no ser descubiertos por otros dispositivos Bluetooth dentro de su vecindad; de la misma manera, puede leer de un nodo el modo de descubrimiento actual con el método `Leer_Modo_Detectable (AT*AGDM?)`.
- *Nombre y Clase*: Todos los dispositivos Bluetooth cuentan con un nombre local para identificarse de forma amigable con los usuarios de una picored. Mediante el método `Escribir_Nombre_Local (AT*AGLN=)` el usuario puede cambiar los nombres locales a los nodos. Para obtener el nombre local actual de un nodo, el usuario debe utilizar el método `Leer_Nombre_Local (AT*AGLN?)`.

El usuario puede especificar sobre los nodos la clase local de dispositivo Bluetooth a la que pertenecen²⁴ (dispositivos de audio, dispositivos telefónicos, etc.), esto es posible mediante el método `Escribir_COD_Local (AT*AGLC=)`. Para conocer la clase local de dispositivo Bluetooth que asumen los nodos, el usuario debe utilizar el método `Leer_COD_Local (AT*AGLC?)`.

Por medio del método `Leer_BD_ADDR (AT*AILBA?)` el usuario puede consultar la dirección Bluetooth del nodo.

- *Restaurar Nodo*: Para restablecer la configuración por defecto de los nodos, el usuario debe utilizar el método `Restaurar_Nodo (AT&F)`.

La Figura 3.16 muestra el diagrama de caso de uso para el subsistema Configuración del Nodo. La Figura 3.17 muestra el diagrama de caso de uso para la generalización Configuración RS232 dentro del subsistema Configuración del Nodo. La Figura 3.18 y la Figura 3.19 muestran los diagramas de caso de uso para la generalización Modo Conectable y Modo Detectable respectivamente dentro del subsistema Configuración del Nodo.

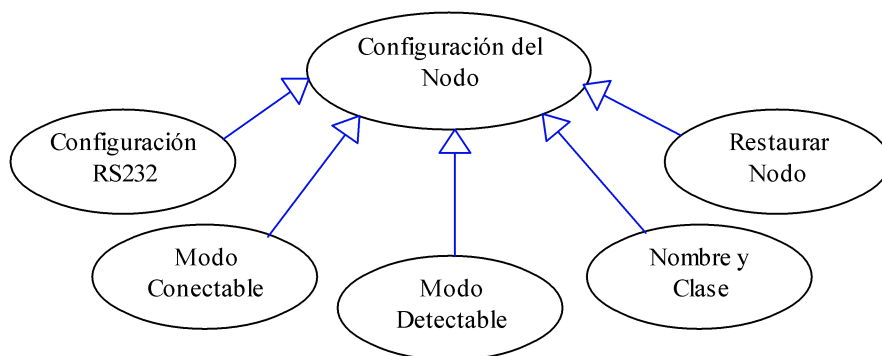


Figura 3.16. Diagrama de caso de uso para el subsistema Configuración del Nodo.

²⁴ Consultar el Anexo órdenes AT para obtener la descripción de los códigos empleados para el tipo de dispositivos Bluetooth (Anexo C).

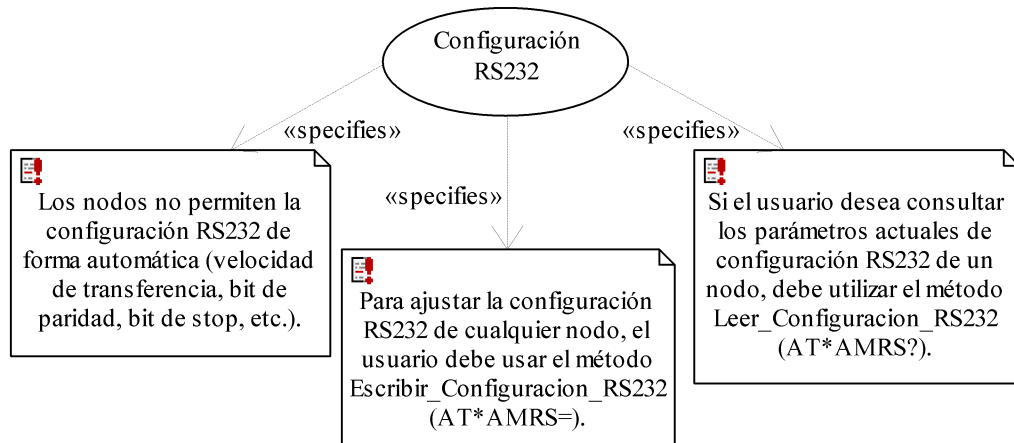


Figura 3.17. Diagrama de caso de uso para la generalización Configuración RS232.

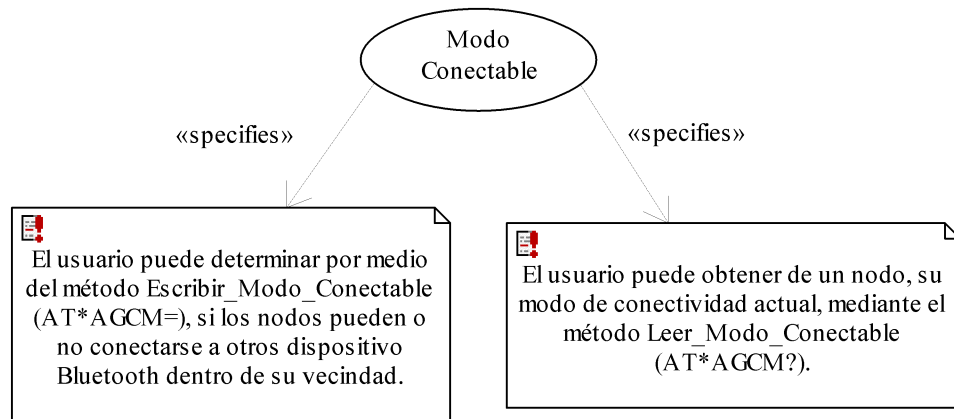


Figura 3.18. Diagrama de caso de uso para la generalización Modo Conectable.

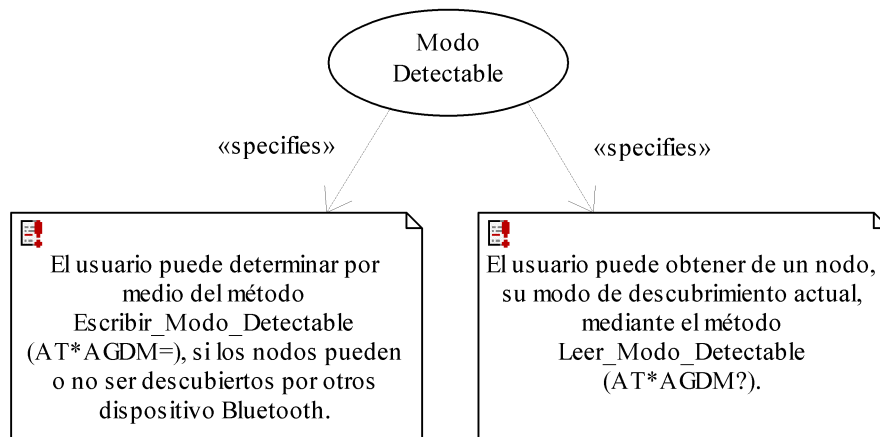


Figura 3.19. Diagrama de caso de uso para la generalización Modo Detectable.

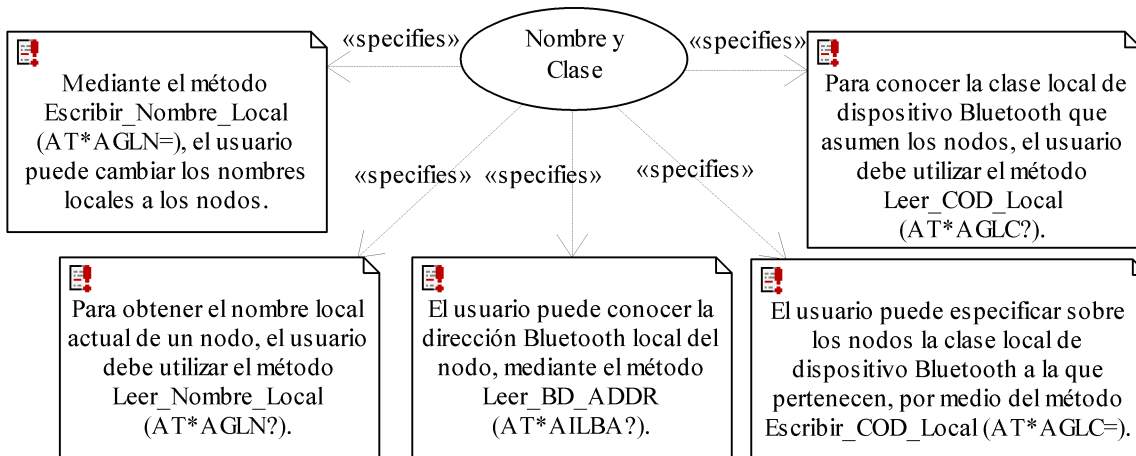


Figura 3.20. Diagrama de caso de uso para la generalización Nombre y Clase.

La Figura 3.20 muestra el diagrama de caso de uso para la generalización Nombre y Clase dentro del subsistema Configuración del Nodo. La Figura 3.21 muestra el diagrama de caso de uso para la generalización Restaurar Nodo dentro del subsistema Configuración del Nodo.

La Tabla 3.6 lista los métodos y órdenes empleados en el subsistema Configuración del Nodo.

Tabla 3.6. Métodos y órdenes empleados en el subsistema Configuración del Nodo.

Generalización	Método	Orden AT
Configuración RS232	Escribir_Configuracion_RS232	AT*AMRS=
	Leer_Configuracion_RS232	AT*AMRS?
Modo Conectable	Escribir_Modo_Conectable	AT*AGCM=
	Leer_Modo_Conectable	AT*AGCM?
Modo Detectable	Escribir_Modo_Detectable	AT*AGDM=
	Leer_Modo_Detectable	AT*AGDM?
Nombre y Clase	Escribir_Nombre_Local	AT*AGLN=
	Leer_Nombre_Local	AT*AGLN?
	Leer_BD_ADDR	AT*AILBA?
	Escribir_COD_Local	AT*AGLC=
	Leer_COD_Local	AT*AGLC?
Restaurar Nodo	Restaurar_Nodo	AT&F

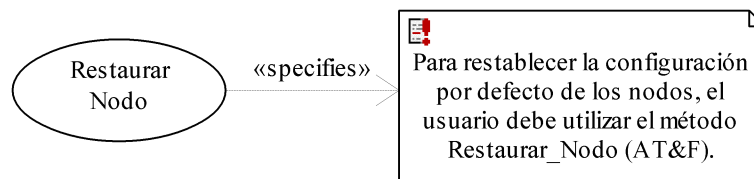


Figura 3.21. Diagrama de caso de uso para la generalización Restaurar Nodo.

3.2.3.1.2. Subsistema Gestión de la Conexión

Mediante el subsistema Gestión de la Conexión el usuario puede realizar las siguientes actividades:

- Crear conexiones seriales entre los nodos.
- Crear conexiones seriales con otros dispositivos remotos.
- Elegir los perfiles de los nodos para el establecimiento de las conexiones.
- Establecer enlaces punto a punto o multipunto.
- Especificar el número de dispositivos a establecer la conexión.

En la creación de conexiones seriales, el usuario debe decidir inicialmente si los nodos actuarán como Clientes o como Servidores. Para establecer conexión entre los nodos, es importante usar el mismo perfil en ambos²⁵; para esto, dentro del subsistema Configuración del Nodo existen las siguientes generalizaciones:

- *Modo Servidor*: Por medio del método `Escribir_Perfil_Servidor (AT*ADDSP=)`, el usuario selecciona el perfil a usar cuando el nodo actúe como Servidor.

Para permitir que varios dispositivos se conecten de forma simultánea al nodo en modo Servidor, el usuario debe habilitar la opción de enlaces multipunto del nodo mediante el método `Escribir_Enlace_Multipunto (AT*ADWM=)`; además, el usuario debe ocupar el método `Escribir_RolMaestroEsclavo (AT*AGMSP=)` para que el nodo intercambie el rol de Maestro a Esclavo cada vez que un dispositivo se conecte a él.

Después de configurar el nodo en modo Servidor, el usuario debe cambiar del modo de operación por Órdenes AT al modo de Datos. La conexión del enlace finalizará si el nodo cambia al modo de Órdenes AT.

- *Modo Cliente*: Existen dos alternativas para crear conexiones en modo cliente:
 - Mediante el método `Conectar_A_Servicio_Serial (AT*ADCP=)` el nodo se conecta a dispositivos remotos que tengan servicios seriales habilitados. Después mover el nodo de modo de Órdenes AT al modo de Datos.

Es posible utilizar el método `Escribir_Enlace_Multipunto (AT*ADWM=)` y conectar el nodo a varios dispositivos.

- Elegir el número de dispositivos a conectarse mediante el método `Escribir_No_Pares_Remotos (AT*ADNRP=)`. Seleccionar uno para enlaces punto a punto o más de uno para enlaces multipunto²⁶. El usuario puede obtener el número máximo de dispositivos soportados para entablar conexión con el nodo mediante el método `Leer_No_Max_Pares_Remotos (AT*ADMRP?)`.

Seleccionar el o los dispositivos a comunicarse con el uso del método `Escribir_Par_Remoto (AT*ADWDRP=)`; enseguida usar el método `Escribir_Perfil_Cliente (AT*ADDCP=)` para seleccionar el perfil a usar cuando el nodo se conecte como Cliente a otro dispositivo remoto. Después de configurar el nodo en modo Cliente,

²⁵ Si el dispositivo remoto a establecer la conexión no pertenece al sistema DreamBlue, el usuario debe verificar qué perfil debe usar en la documentación del dispositivo externo.

²⁶ Debe estar habilitada en el nodo la opción de enlaces multipunto.

el usuario debe cambiar del modo de operación por Órdenes AT al modo de Datos. La conexión del enlace finalizará si el nodo cambia al modo de Órdenes AT.

La Figura 3.22 muestra el diagrama de caso de uso para el subsistema Gestión de la Conexión. La Figura 3.23 muestra el diagrama de caso de uso para la generalización Modo Servidor dentro del subsistema Gestión de la Conexión. La Figura 3.24 muestra el diagrama de caso de uso para la generalización Modo Cliente dentro del subsistema Gestión de la Conexión.

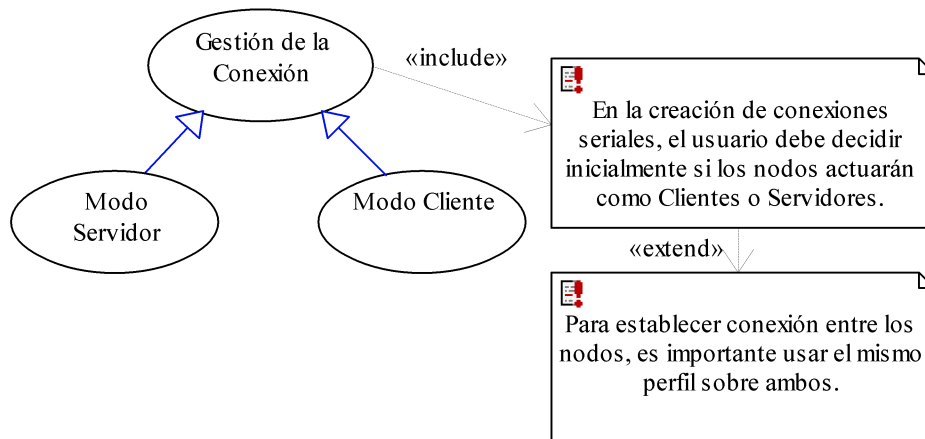


Figura 3.22. Diagrama de caso de uso para el subsistema Gestión de la Conexión.

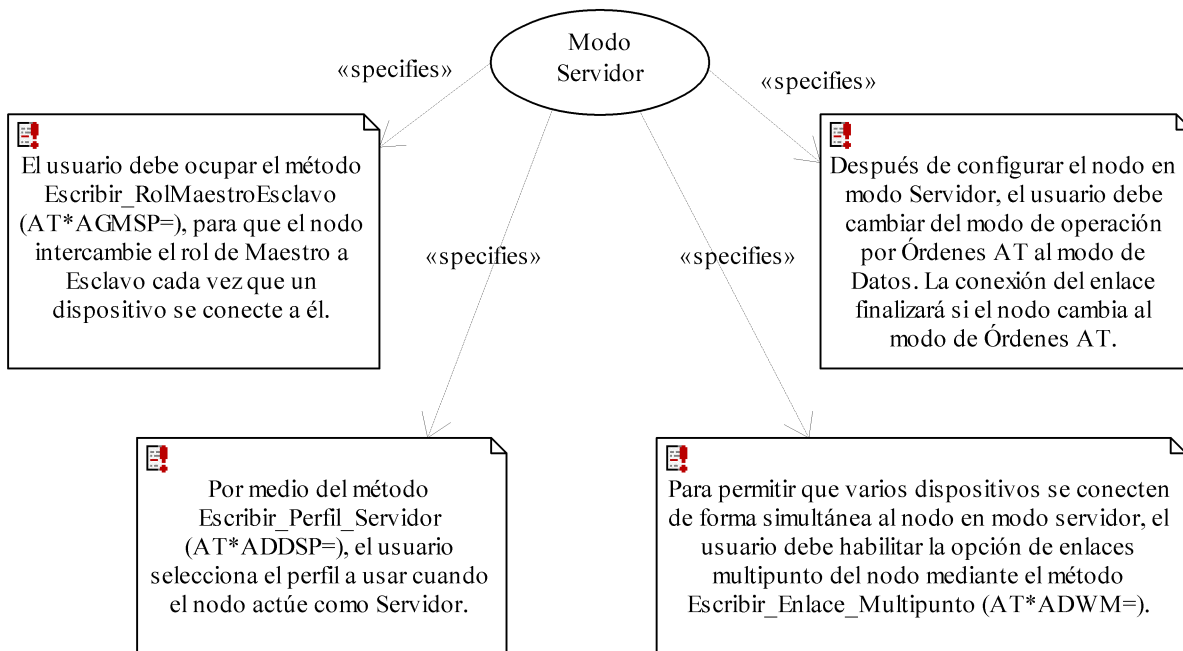


Figura 3.23. Diagrama de caso de uso para la generalización Modo Servidor.

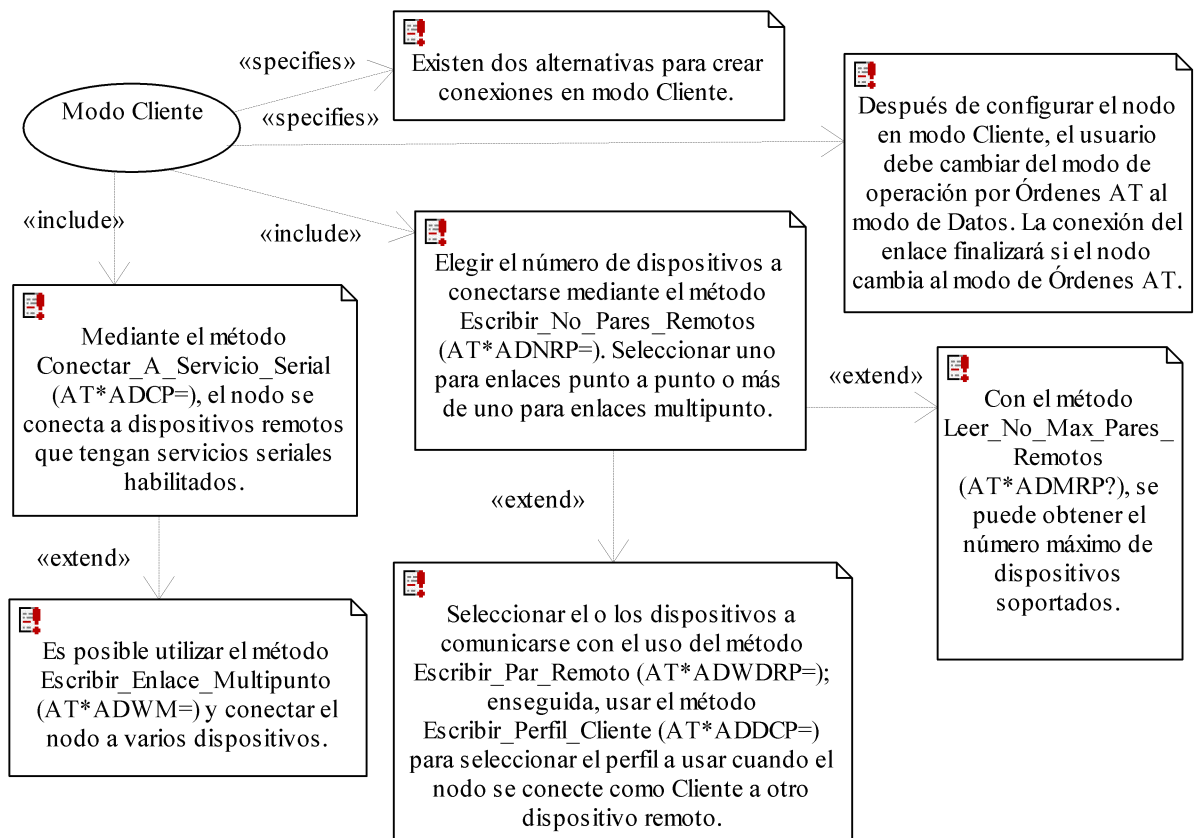


Figura 3.24. Diagrama de caso de uso para la generalización Modo Cliente.

La Tabla 3.7 lista los métodos y órdenes empleados por el subsistema Gestión de la Conexión.

Tabla 3.7. Métodos y órdenes empleados por el subsistema Gestión de la Conexión.

Generalización	Método	Orden AT
Modo Servidor	Escribir_Perfil_Servidor	AT*ADDSP=
	Escribir_Enlace_Multipunto	AT*ADWM=
	Escribir_RolMaestroEsclavo	AT*AGMSP=
Modo Cliente	Conectar_A_Servicio_Serial	AT*ADCP=
	Escribir_Enlace_Multipunto	AT*ADWM=
	Escribir_No_Pares_Remotos	AT*ADNRP=
	Leer_No_Max_Pares_Remotos	AT*ADMRP?
	Escribir_Par_Remoto	AT*ADWDRP=
	Escribir_Perfil_Cliente	AT*ADDCP=

3.2.3.1.3. Subsistema Servicios Suplementarios

El subsistema Servicios Suplementarios permite al usuario realizar las siguientes actividades:

- Controlar el consumo de energía de los nodos.

- Realizar un sondeo para encontrar dispositivos Bluetooth dentro de su vecindad.
- Obtener direcciones, clases y nombres locales de dispositivos Bluetooth dentro de su vecindad.
- Activar la opción de seguridad de los nodos.
- Transmitir y recibir datos entre los nodos en modo de seguridad.

Las generalizaciones correspondientes al subsistema Servicios Suplementarios son:

- *Modo Ahorro:* El usuario debe utilizar el método `Escribir_Modo_Ahorro_Energia` (`AT*AMPM=`) y elegir el modo de ahorro de energía. Si el usuario elige el modo de par²⁷, existen las siguientes restricciones para su uso:
 - No deben existir conexiones activas en el nodo.
 - Cuando un nodo se encuentre en modo de optimización, el usuario no podrá enviar órdenes AT del módulo DTE al módulo DCE.
 - El nodo podrá salir de la opción de optimización si se establece conexión entre los nodos (o algún otro dispositivo remoto).
 - El nodo necesita un lapso de tiempo de por lo menos 10 ms para salir de la opción de optimización de energía.
- *Modo Sondeo:* Los nodos pueden realizar sondeos dentro de su vecindad en busca de dispositivos Bluetooth. Para realizar los sondeos, el usuario cuenta con dos métodos:
 - Mediante el método `Sondeo` (`AT*AGI=`), el nodo adquiere las direcciones Bluetooth y las clases a las que pertenecen de todos los dispositivos dentro de su vecindad.
 - Por medio del método `Descubrir_Dispositivos` (`AT*AGDD=`), el nodo adquiere las direcciones, clases y nombres locales de los dispositivos Bluetooth encontrados dentro de su vecindad.
- *Modo Seguridad:* Los nodos cuentan con soporte de seguridad Bluetooth. La seguridad se basa en la autenticación durante el establecimiento de la conexión y en la encriptación en el envío y recepción de datos. Es posible activar en cada nodo el modo de seguridad y mediante el método `Escribir_Modo_Seguro` (`AT*AGSM=`).

Para transmitir datos entre los nodos con el modo de seguridad habilitado, se tienen las siguientes restricciones:

- Si al menos alguno de los dos nodos tiene la opción de seguridad habilitada se debe activar la seguridad en ambos nodos.
- Se debe establecer una unión entre los nodos; sin embargo para aceptar la unión, primero ambos nodos deben estar en modo de emparejamiento. Con el método `Escribir_Modo_Emparejamiento` (`AT*AGPM=`) se habilita el modo de emparejamiento de los nodos.
- El proceso de unión crea un enlace con clave (Sección 0). Con el método `Escribir_Codigo_PIN` (`AT*AGFP=`) el usuario establece el código PIN a usar entre los nodos.
- Una vez realizado los pasos anteriores, el usuario debe utilizar el método `Union` (`AT*AGB=`). Los nodos pueden guardar varias claves de enlaces para diversos dis-

²⁷ En el Anexo órdenes AT, se muestra la tabla de modos de ahorro de energía para los nodos.

positivos al mismo tiempo. El usuario puede obtener una lista de las claves existentes mediante el método Leer_Dispositivos_Unidos (AT*AGBD?). El usuario puede borrar un dispositivo de la lista de uniones existentes por medio del método Desunir (AT*AGUB=).

La Figura 3.25 muestra el diagrama de caso de uso para el subsistema Servicios Suplementarios. La Figura 3.26 muestra el diagrama de caso de uso para la generalización Modo Ahorro dentro del subsistema Servicios Suplementarios. La Figura 3.27 muestra el diagrama de caso de uso para la generalización Modo Sondeo dentro del subsistema Servicios Suplementarios. La Figura 3.28 muestra el diagrama de caso de uso para la generalización Modo Seguridad dentro del subsistema Servicios Suplementarios.

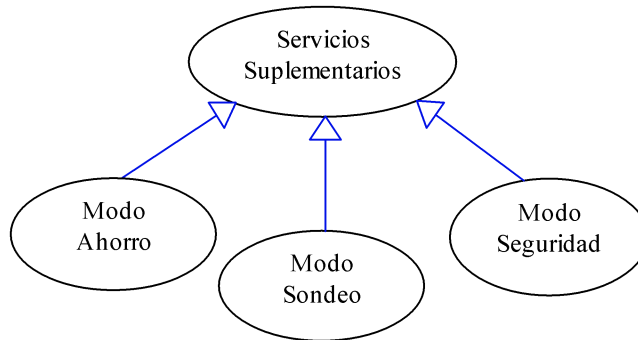


Figura 3.25. Diagrama de caso de uso para el subsistemas Servicios Suplementarios.

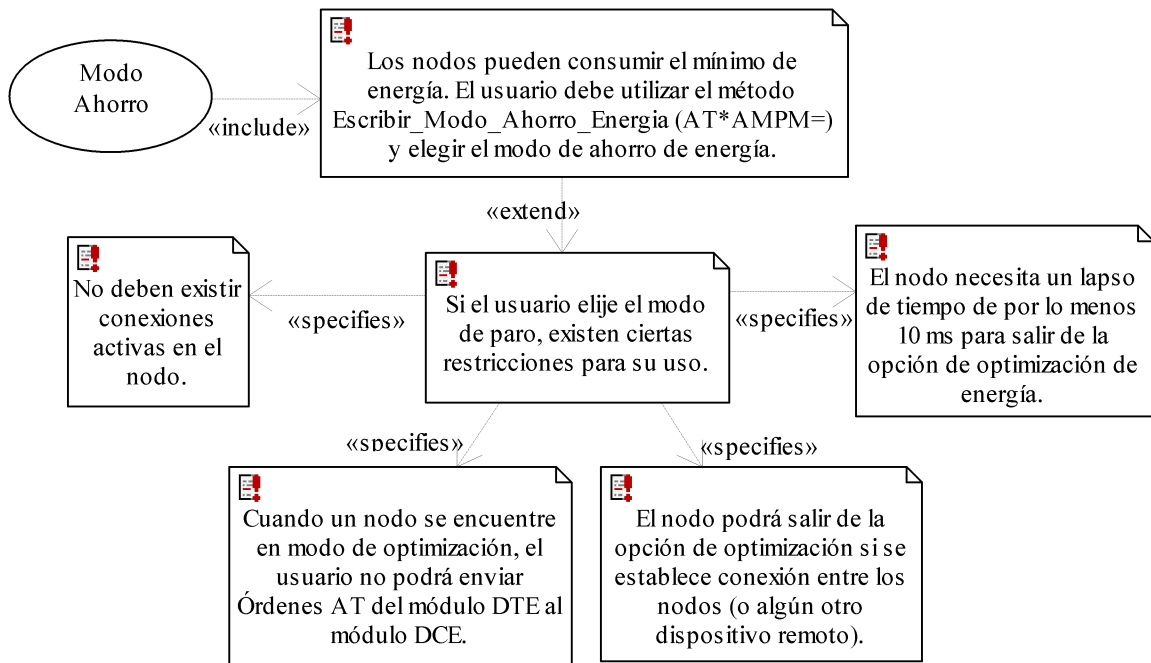


Figura 3.26. Diagrama de caso de uso para la generalización Modo Ahorro.

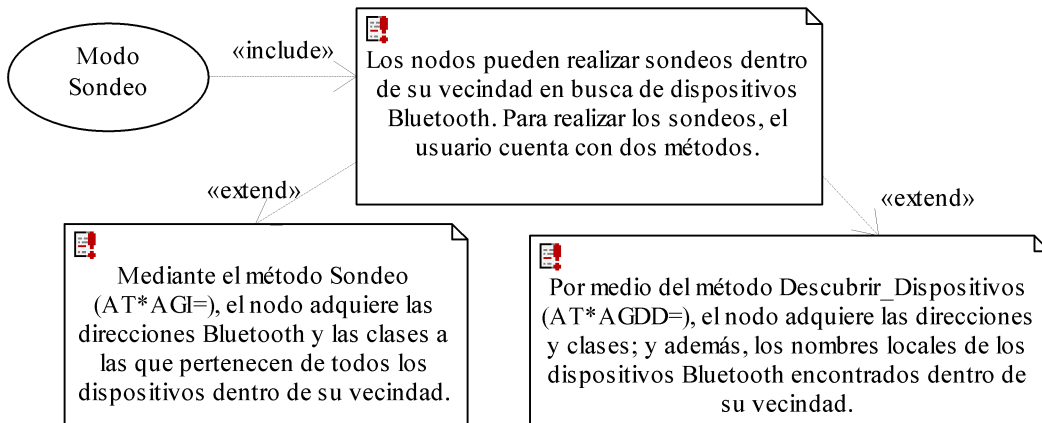


Figura 3.27. Diagrama de caso de uso para la generalización Modo Sondeo.

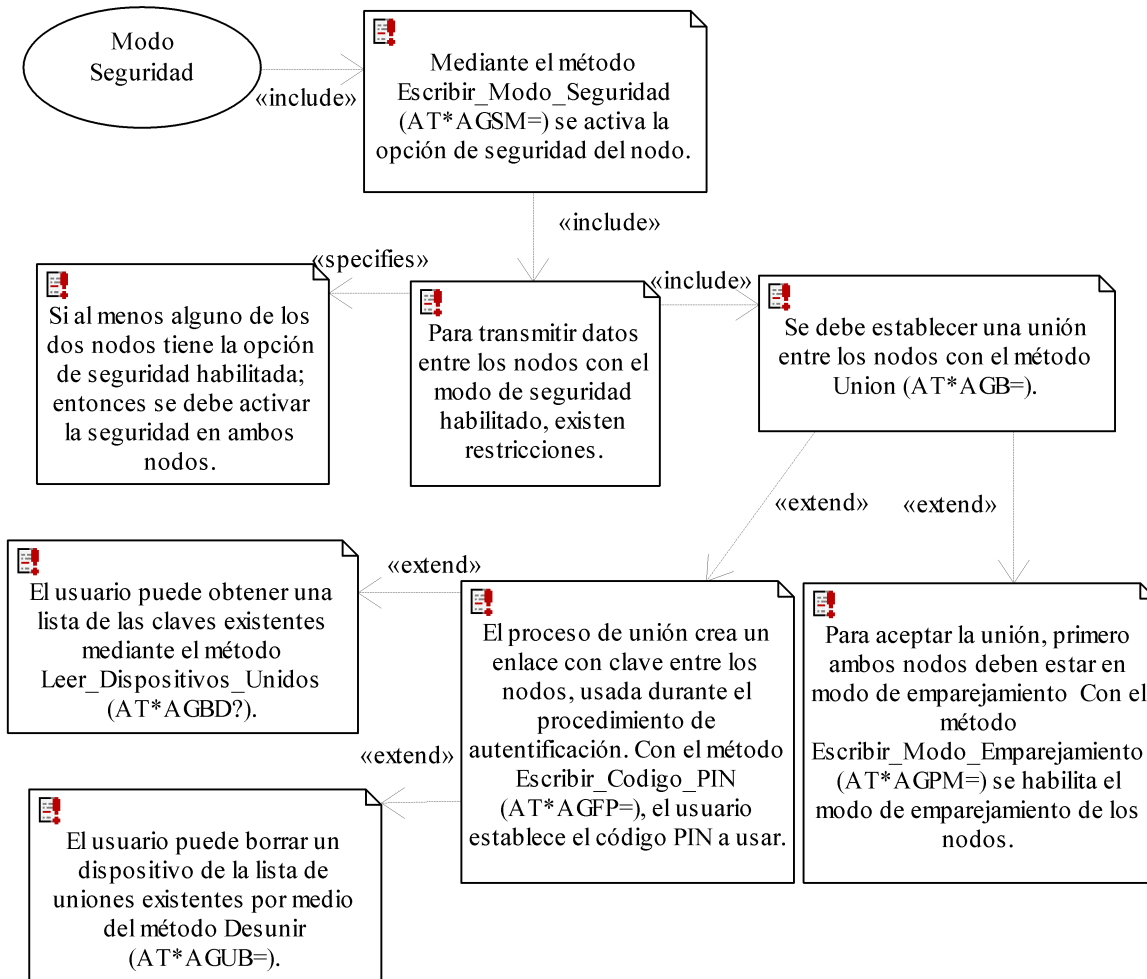


Figura 3.28. Diagrama de caso de uso para la generalización Modo Seguridad.

La Tabla 3.8 lista los métodos y órdenes empleados por el subsistema Servicios Suplementarios.

Tabla 3.8. Métodos y órdenes empleados por el subsistema Servicios Suplementarios.

Generalización	Método	Orden AT
Modo Ahorro	Escribir_Modo_Ahorro_Energia	AT*AMPM=
Modo Sondeo	Sondeo	AT*AGI=
	Descubrir_Dispositivos	AT*AGDD=
Modo Seguridad	Escribir_Modo_Seguridad	AT*AGSM
	Union	AT*AGB=
	Escribir_Modo_Emparejamiento	AT*AGPM=
	Escribir_Codigo_PIN	AT*AGFP=
	Leer_Dispositivos_Unidos	AT*AGBD?
	Desunir	AT*AGUB=

3.2.3.2. Modo de Datos

En el modo de datos el usuario puede enviar y recibir mensajes en formato ASCII del módulo DTE al módulo DCE; posteriormente, el módulo DCE envía el mensaje al nodo opuesto.

Por defecto los nodos se encuentran en modo de operación de Datos al ser iniciados; sin embargo, si el nodo se encuentra en modo de Órdenes AT, el usuario debe regresar al menú principal y utilizar el método Modo_Datos (AT*ADDM). La Figura 3.29 muestra los intercambios entre los modos de operación de los nodos. El modo de Datos se compone por el subsistema Gestión de Mensajes (Figura 3.30).

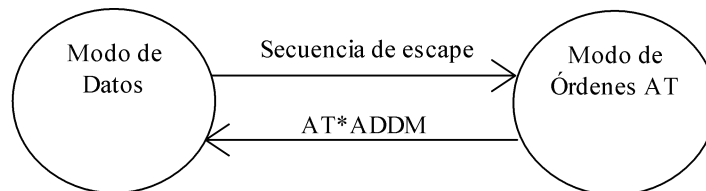


Figura 3.29. Intercambio entre modos de operación de los nodos.

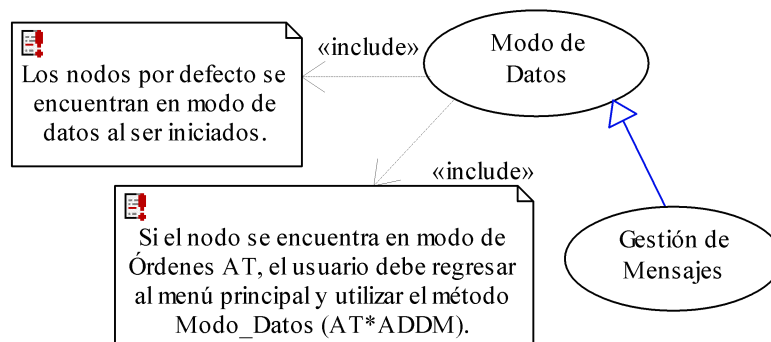


Figura 3.30. Diagrama de caso de uso para el Modo de Datos.

3.2.3.2.1. Subsistema Gestión de Mensajes

El subsistema Gestión de Mensajes permite al usuario enviar y recibir mensajes de texto entre los nodos. Las generalizaciones correspondientes al subsistema Gestión de Mensajes son:

- *Modo de Transmisión:* El usuario puede enviar mensajes de texto de un nodo a otro con el método Enviar_Mensaje (OK); sin embargo, existen las siguientes restricciones:
 - El usuario debe configurar previamente los nodos en modo de Órdenes AT (sección 2.2.3.1) y establecer el enlace entre ellos. Los dos nodos deben tener los mismos parámetros de Configuración RS232.
 - El nodo transmisor, para enviar un mensaje debe encontrarse en Modo Transmisión del subsistema Gestión de Mensajes; mientras que el nodo receptor debe encontrarse en Modo Recepción del subsistema Gestión de Mensajes.
 - El mensaje no debe ser mayor a 16 caracteres.
 - Solo se puede enviar un mensaje a la vez.
 - Cada nuevo mensaje enviado borra en pantalla el mensaje anterior.

El usuario debe seleccionar los caracteres a enviar a través del método Seleccionar (Select) y en caso de equivocarse en la escritura del mensaje, el usuario cuenta con dos opciones:

- Mediante el método Cancelar (Cancel): El usuario sale de la escritura de mensajes.
- Con el método Borrar (Erase): El usuario borra un caracter.
- *Modo de Recepción:* El usuario recibe los mensajes provenientes del nodo opuesto con las siguientes restricciones:
 - El usuario debe configurar previamente los nodos en modo de Órdenes AT (sección 2.2.3.1) y establecer el enlace entre ellos. Los dos nodos deben tener los mismos parámetros de Configuración RS232.
 - Para recibir el nodo receptor un mensaje debe encontrarse en Modo de Recepción del subsistema Gestión de Mensajes; mientras que el nodo transmisor debe encontrarse en Modo de Transmisión dentro del subsistema Gestión de Mensajes.
 - El nodo solo visualiza un mensaje entrante a la vez.
 - Por cada mensaje recibido se debe ingresar al Modo de Recepción del subsistema Gestión de Mensajes.

La Figura 3.31 muestra el diagrama de caso de uso para el subsistema Gestión de Mensajes. La Figura 3.32 muestra el diagrama de caso de uso para la generalización Modo de Recepción del subsistema Gestión de Mensajes. La Figura 3.33 muestra el diagrama de caso de uso para la generalización Modo de Transmisión del subsistema Gestión de Mensajes.

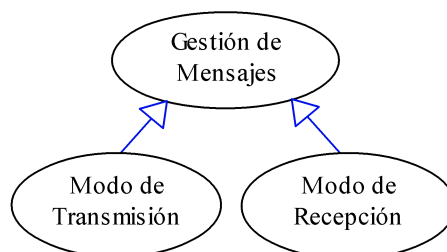


Figura 3.31. Diagrama de caso de uso para el subsistemas Gestión de Mensajes.

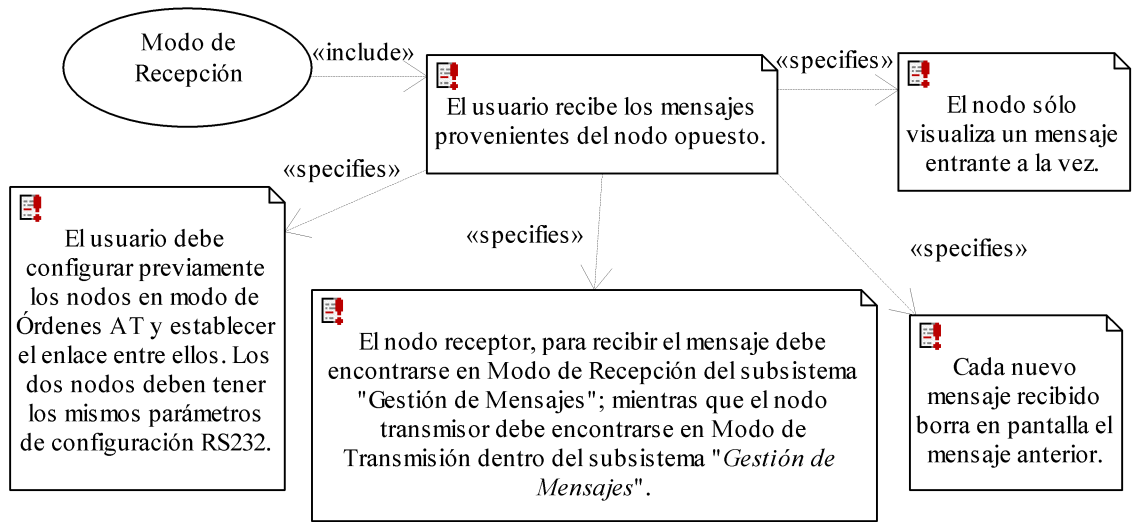


Figura 3.32. Diagrama de caso de uso para la generalización Modo de Recepción.

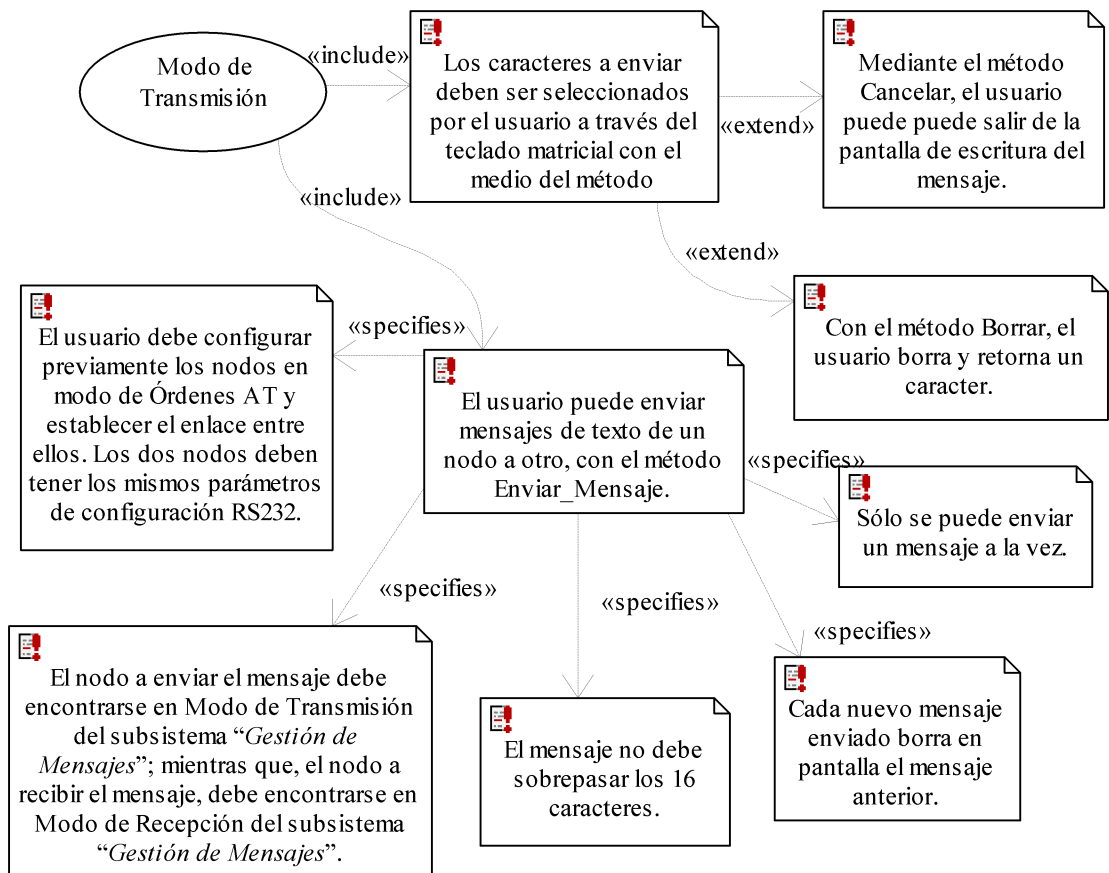


Figura 3.33. Diagrama de caso de uso para la generalización Modo de Transmisión.

Los métodos empleados por el subsistema Gestión de Mensajes se resumen en la Tabla 3.9.

Tabla 3.9. Métodos empleados por el subsistema Gestión de Mensajes.

Generalización	Método	Descripción
Modo de Transmisión	Enviar_Mensaje	Envía el mensaje a través del aire
	Seleccionar	Selecciona el caracter elegido por el usuario
	Borrar	Borra un carácter en pantalla
	Borrar_Mensaje	Borra el mensaje en pantalla
Modo de Recepción	Borrar_Mensaje	Borra el mensaje en pantalla

3.3. Diseño del Sistema

El propósito de la fase de diseño es que a partir de los requerimientos se examine la interacción entre el usuario y el sistema mediante la creación de diseños funcionales.

En esta fase se realizaron las siguientes actividades:

- Diseño del diagrama de estructura general para el sistema.
- Diseño de diagramas de estructura para cada subsistema.
- Diseño de diagramas de secuencia para los escenarios de mayor importancia.
- Diseño de diagramas de máquinas de estado para el análisis dinámico de los nodos.

3.3.1. Diagramas de estructura

La estructura interna del nodo se compone de 5 objetos como muestra la Figura 3.34: Los objetos Módulo DTE y Módulo DCE controlan las operaciones que realizan los actores (UI y Picored) sobre los componentes HW del mismo nombre; el objeto Modos de Operación se encarga del acceso a los modos de operación del nodo; y por último, los objetos Modo de Órdenes AT y Modo de Datos controlan el acceso a los subsistemas respectivos a cada modo de operación.

El diagrama de estructura general de la Figura 3.35 representa la funcionalidad estática de cada nodo DreamBlue, donde se puede observar la interacción que existe entre el nodo, actores, subsistemas, etc.

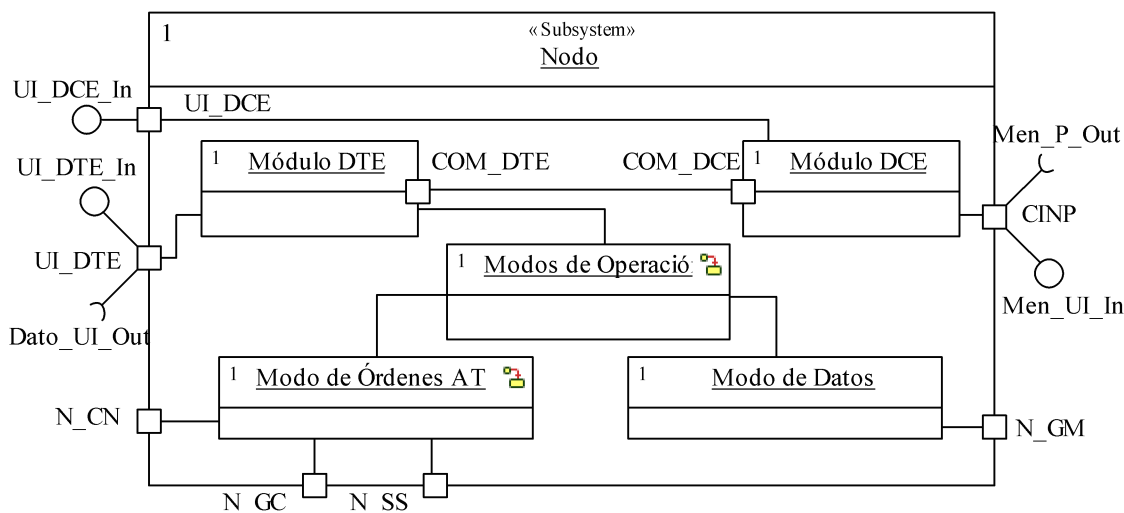


Figura 3.34. Diagrama de estructura del nodo.

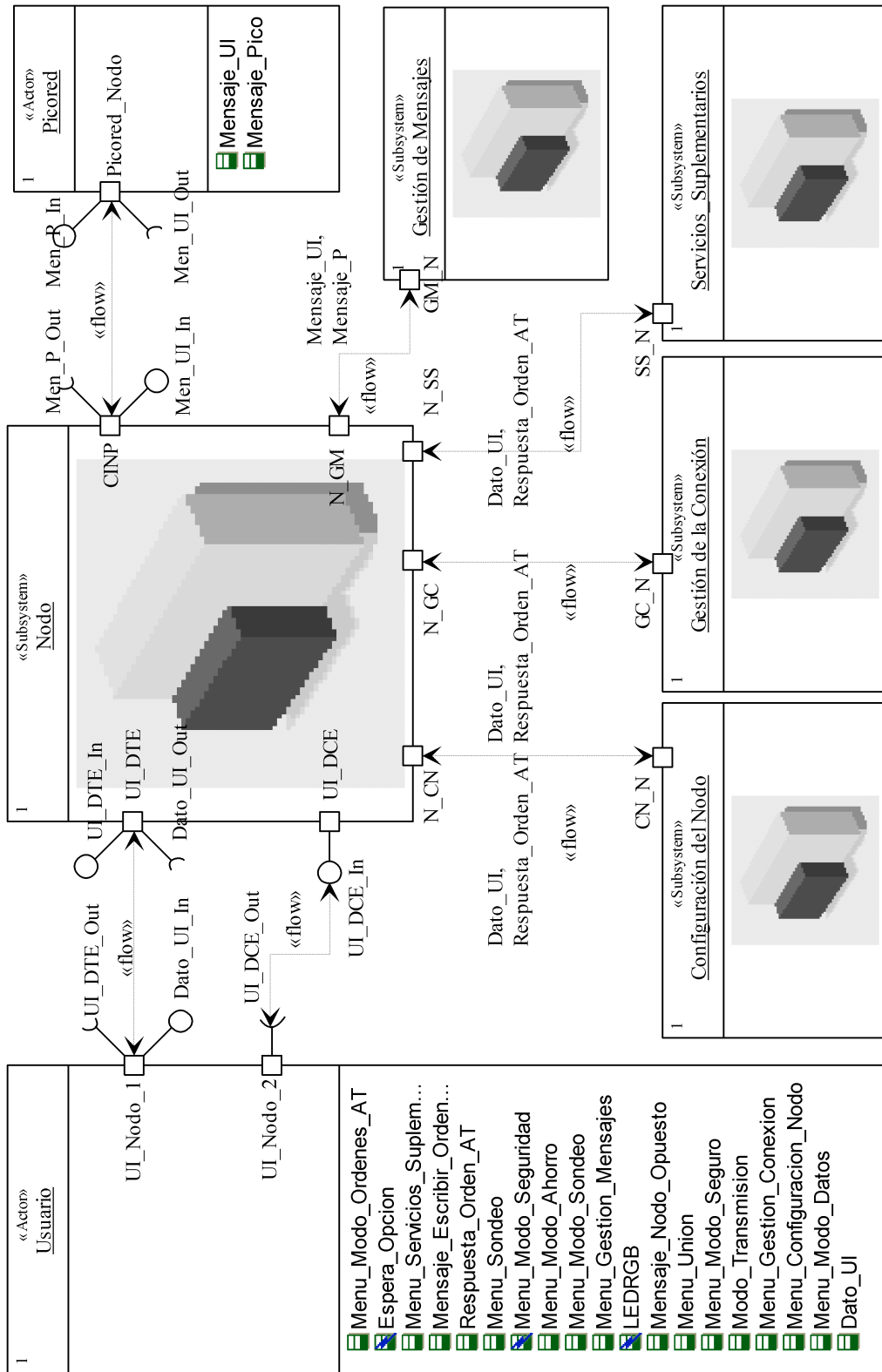


Figura 3.35. Diagrama de estructura general de cada nodo DreamBlue.

En la Figura 3.34 los puertos que interactúan con los actores del sistema son: UI_DTE, UI_DCE y CIMP (Secciones 3.2.1 y 3.2.2); los objetos Módulo DTE y Módulo DCE se comunican mediante los puertos internos COM_DTE y COM_DCE; mientras que los puertos N_CN, N_GC, N_SS y N_GM hacen referencia al intercambio de información entre los modos de operación y sus respectivos subsistemas. Los diagramas de estructura para los subsistemas Configuración del Nodo, Gestión de la Conexión, Servicios Suplementarios y Gestión de Mensajes se muestran en la Figura 3.36, Figura 3.37, Figura 3.38 y Figura 3.39 respectivamente. Cada uno de estos diagramas contiene un objeto de Control y se encarga de acceder a las generalizaciones dentro de cada subsistema.

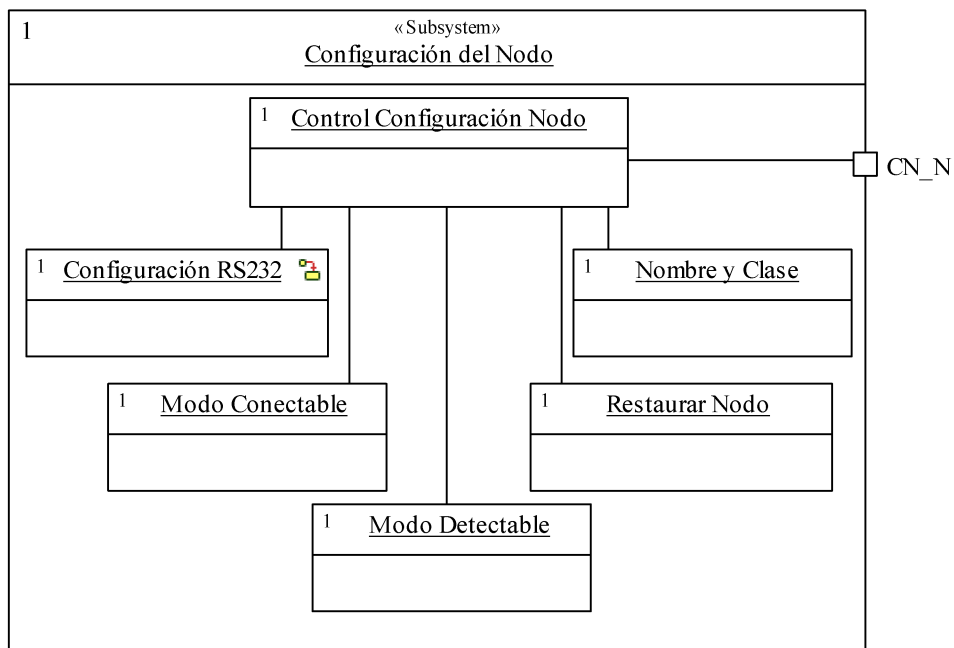


Figura 3.36. Diagrama de estructura del subsistema Configuración del Nodo.

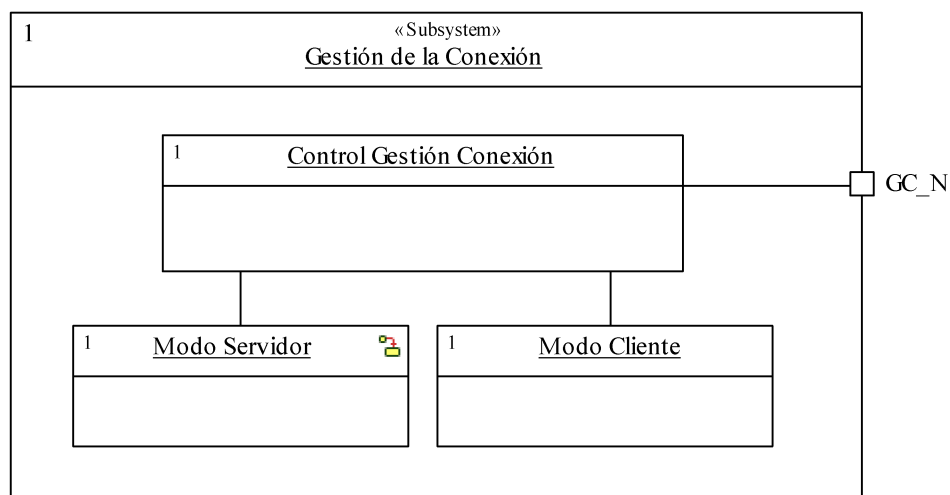


Figura 3.37. Diagrama de estructura del subsistema Gestión de la Conexión.

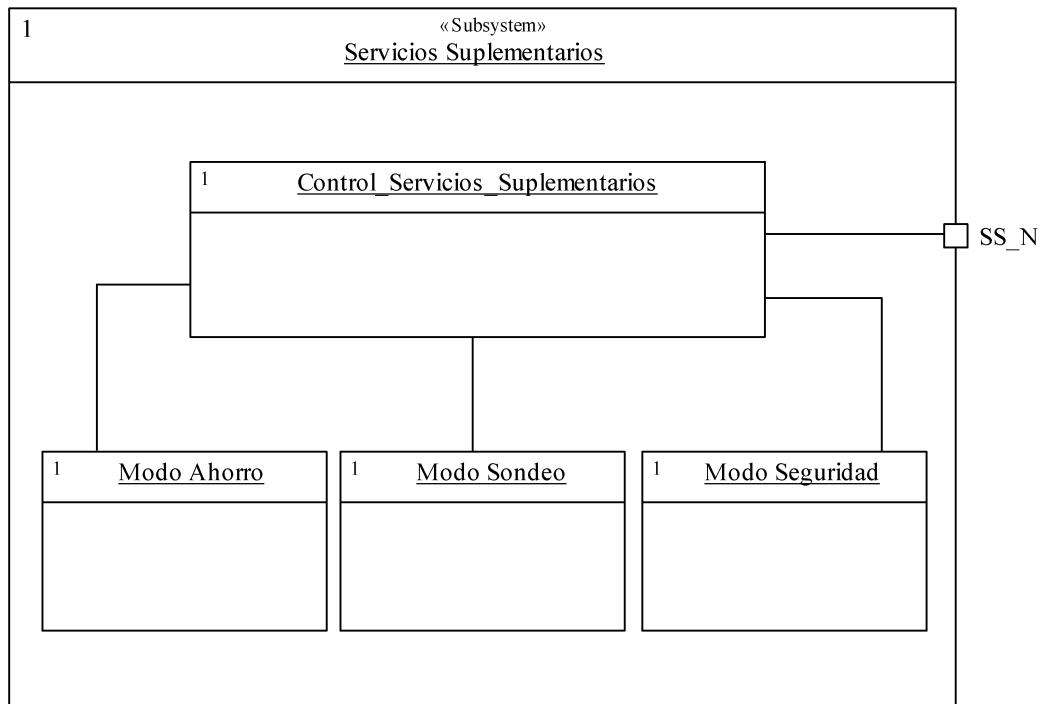


Figura 3.38. Diagrama de estructura del subsistema Servicios Suplementarios.

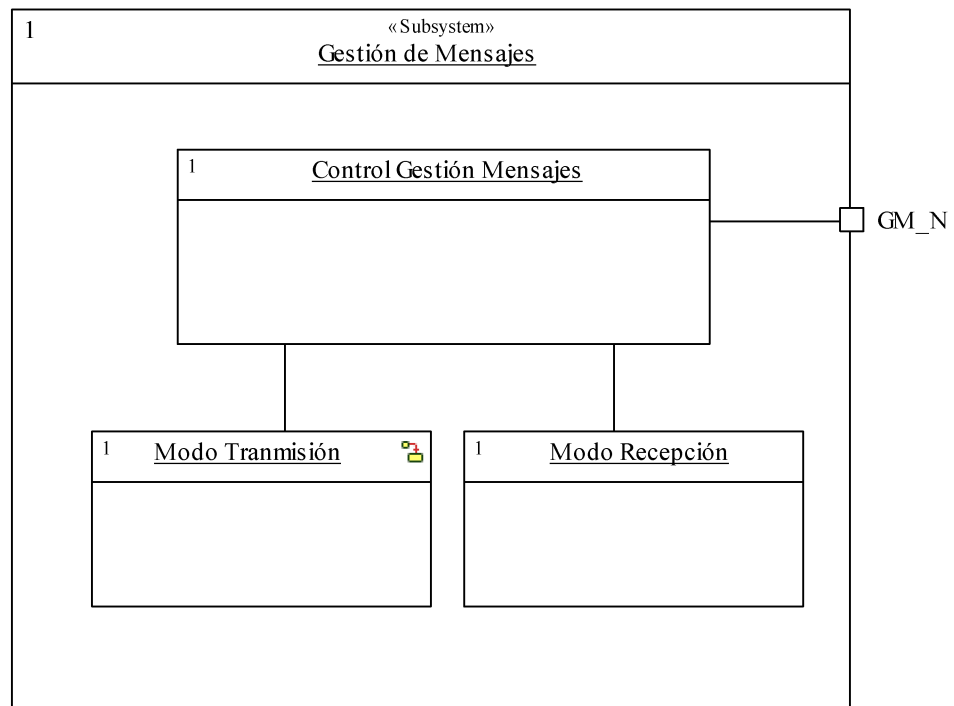


Figura 3.39. Diagrama de estructura del subsistema Gestión de Mensajes.

3.3.2. Diagramas de secuencia

A continuación se presentan los diagramas de actividades para los escenarios más relevantes donde el usuario interactúa con la funcionalidad del sistema DreamBlue.

La Figura 3.40 representa la secuencia de acceso al modo de operación por Órdenes AT (Escenario 1).

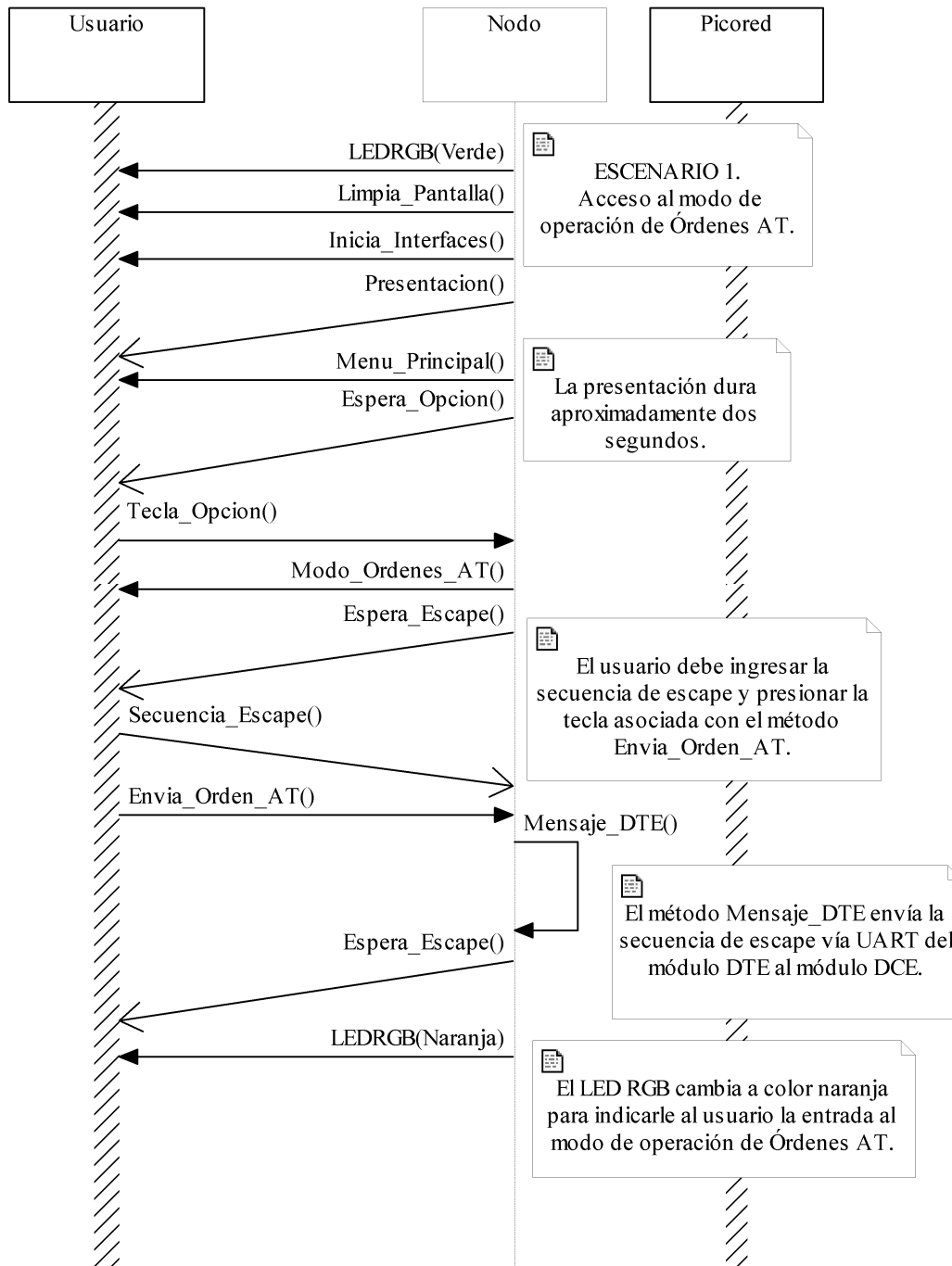


Figura 3.40. Diagrama de secuencia para el Escenario 1.

La Figura 3.41 muestra la secuencia de acceso al modo de operación de Datos (Escenario 2).

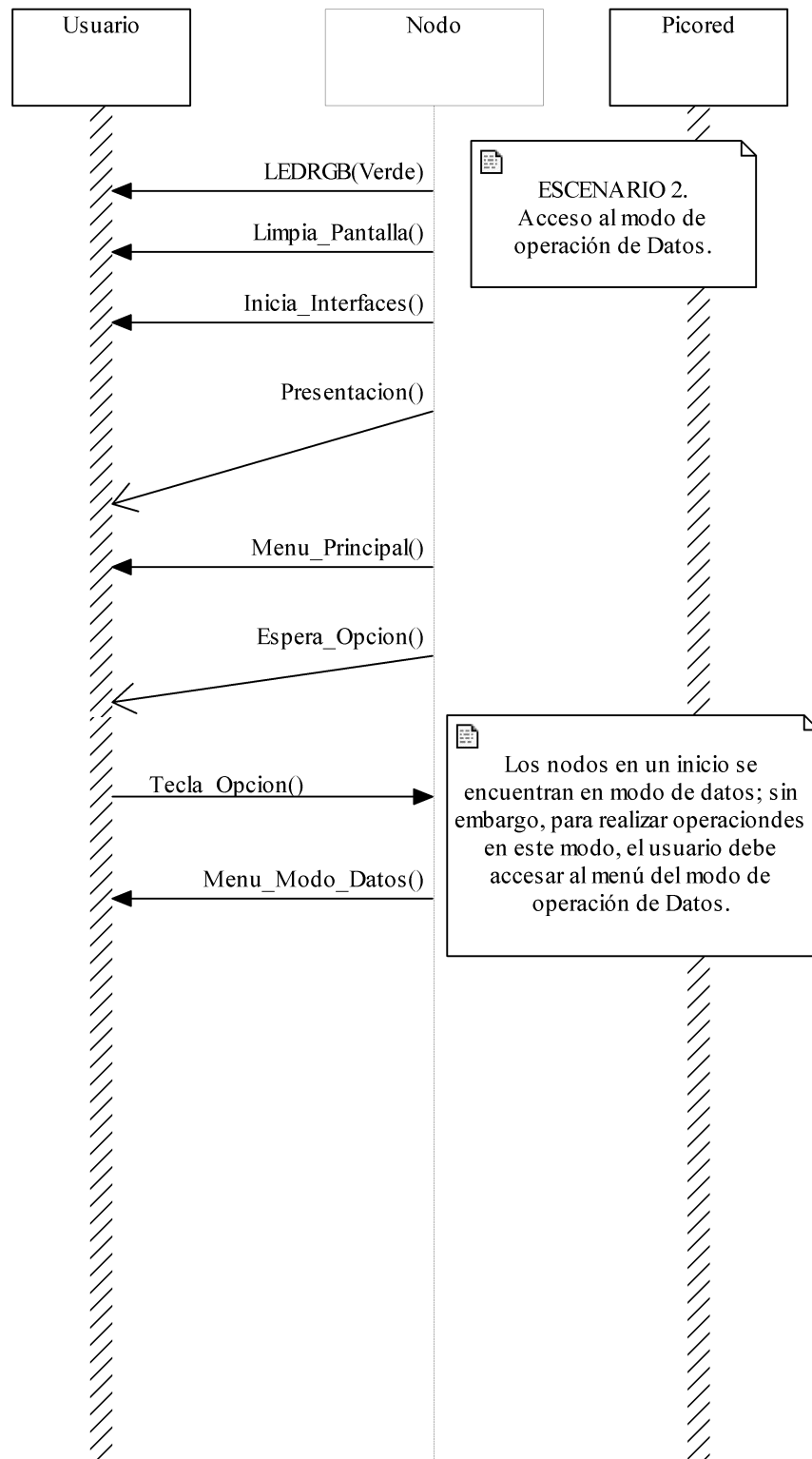


Figura 3.41. Diagrama de secuencia para el Escenario 2.

La Figura 3.42 representa la secuencia de acceso al subsistema Configuración del Nodo para establecer los parámetros de configuración RS232 sobre el nodo (Escenario 3).

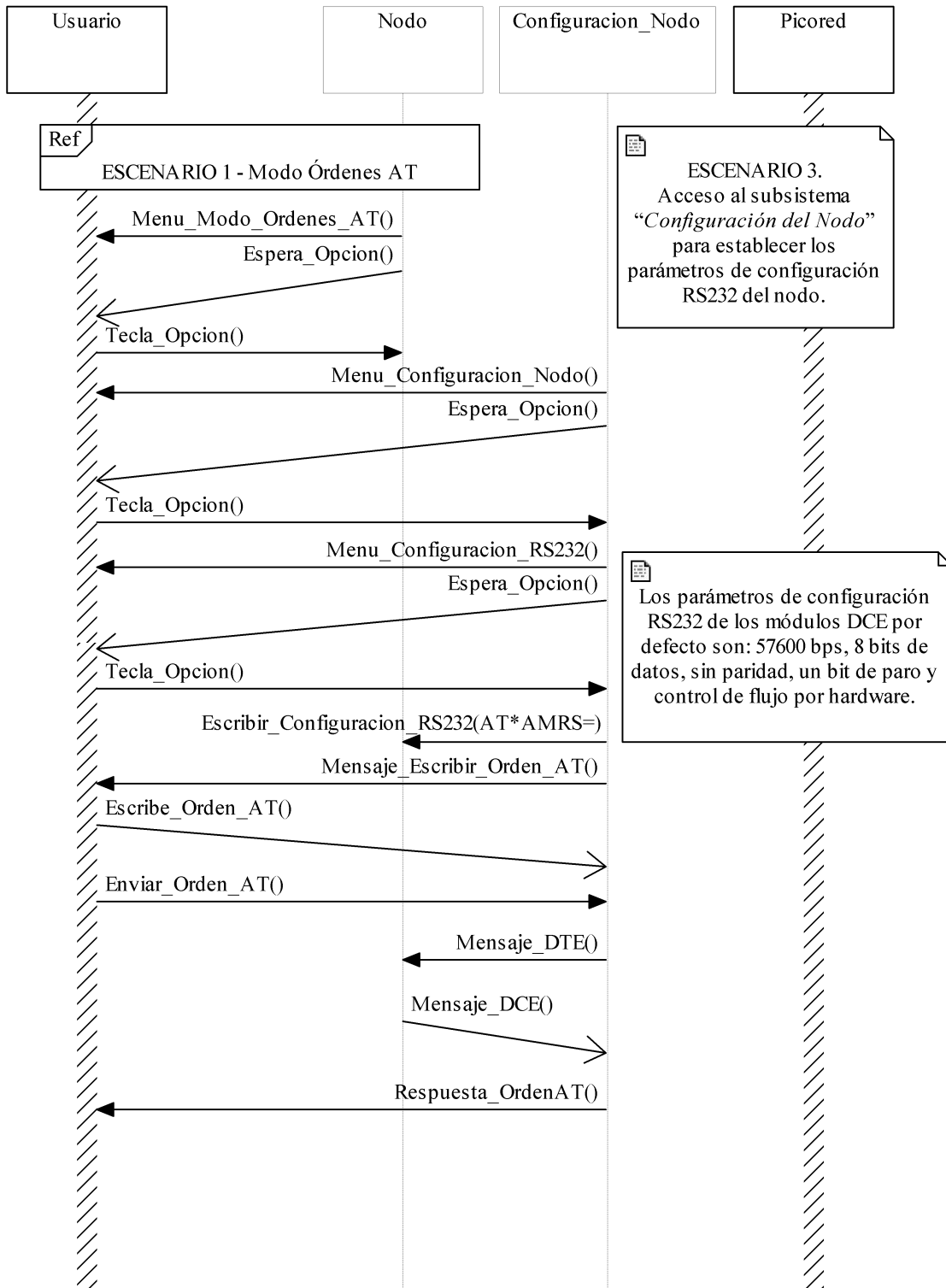


Figura 3.42. Diagrama de secuencia para el Escenario 3.

La Figura 3.43 representa la secuencia de acceso al subsistema Configuración del Nodo para establecer el tipo de modo detectable sobre el nodo (Escenario 4).

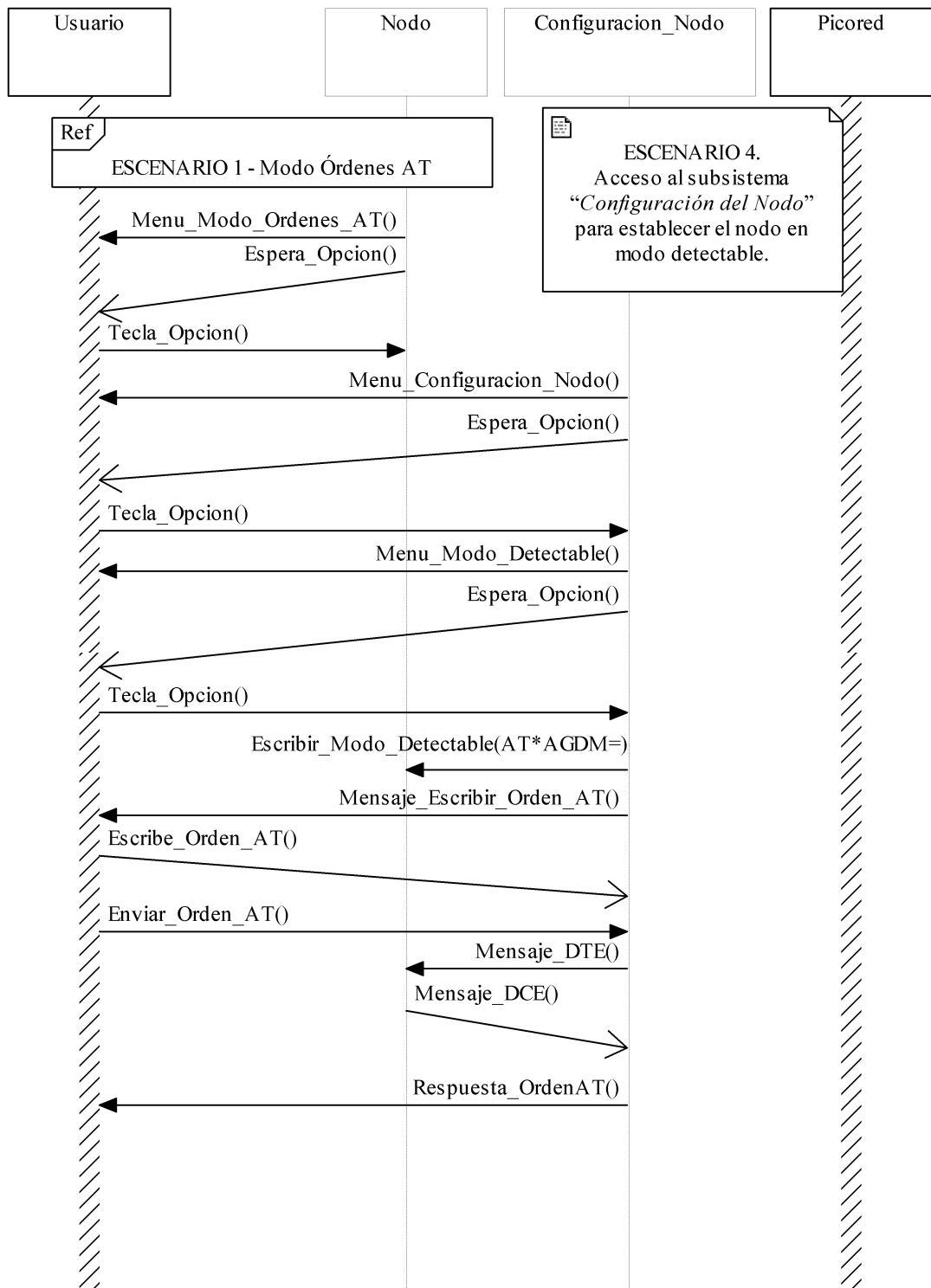


Figura 3.43. Diagrama de secuencia para el Escenario 4.

La Figura 3.44 representa la secuencia de acceso al subsistema Configuración del Nodo para establecer el tipo de modo conectable sobre el nodo (Escenario 5).

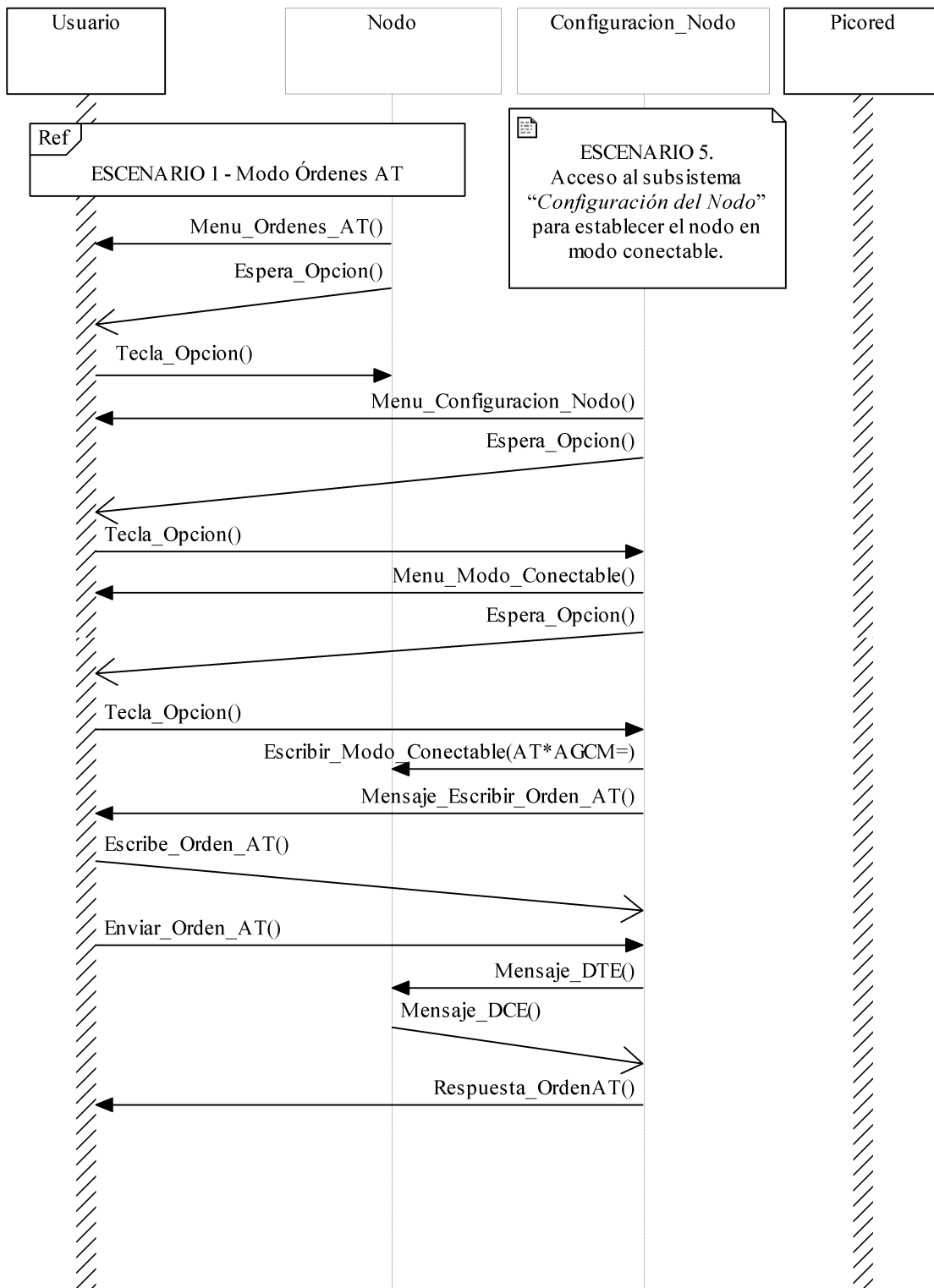


Figura 3.44. Diagrama de secuencia para el Escenario 5.

El diagrama de secuencia de la Figura 3.45, representa el acceso al subsistema Configuración del Nodo y establecer un nombre local al nodo (Escenario 6).

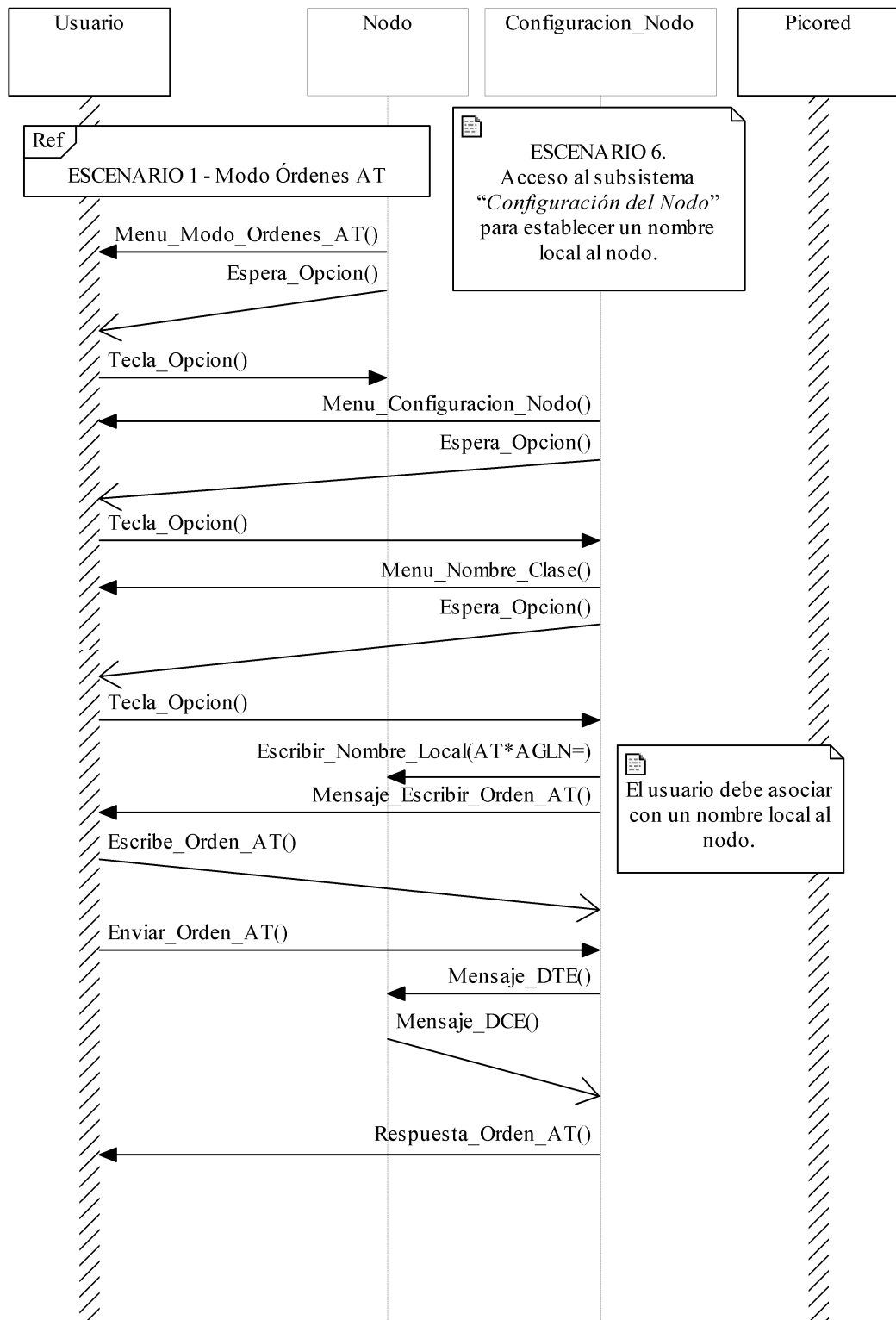


Figura 3.45. Diagrama de secuencia para el Escenario 6.

La Figura 3.46 muestra la secuencia de acceso al subsistema Configuración del Nodo para restaurar los parámetros del nodo a sus valores por defecto (Escenario 7).

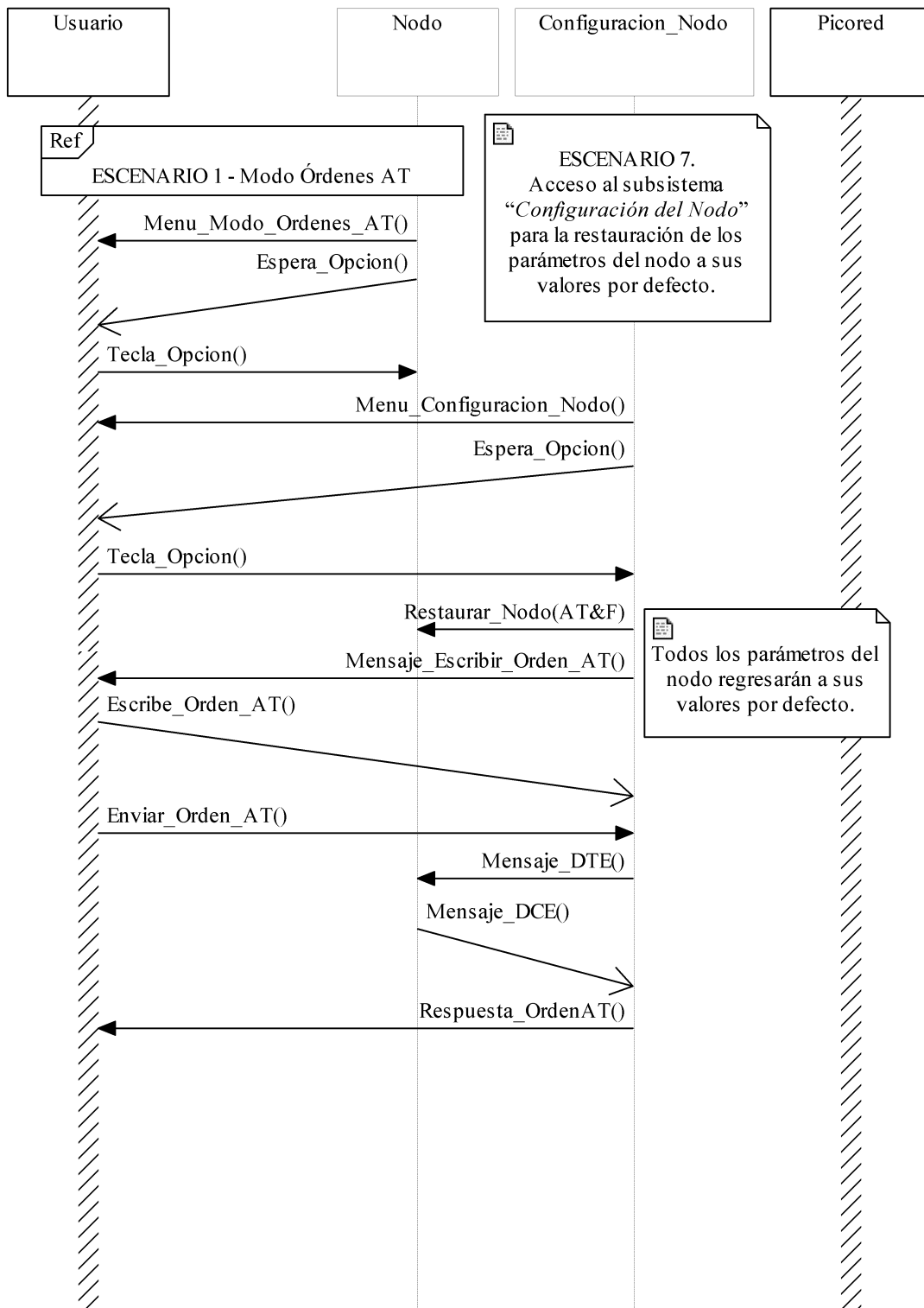


Figura 3.46. Diagrama de secuencia para el Escenario 7.

La Figura 3.47 muestra la secuencia de acceso al subsistema Gestión de la Conexión para escribir el perfil de puerto serial sobre el nodo en modo servidor (Escenario 8).

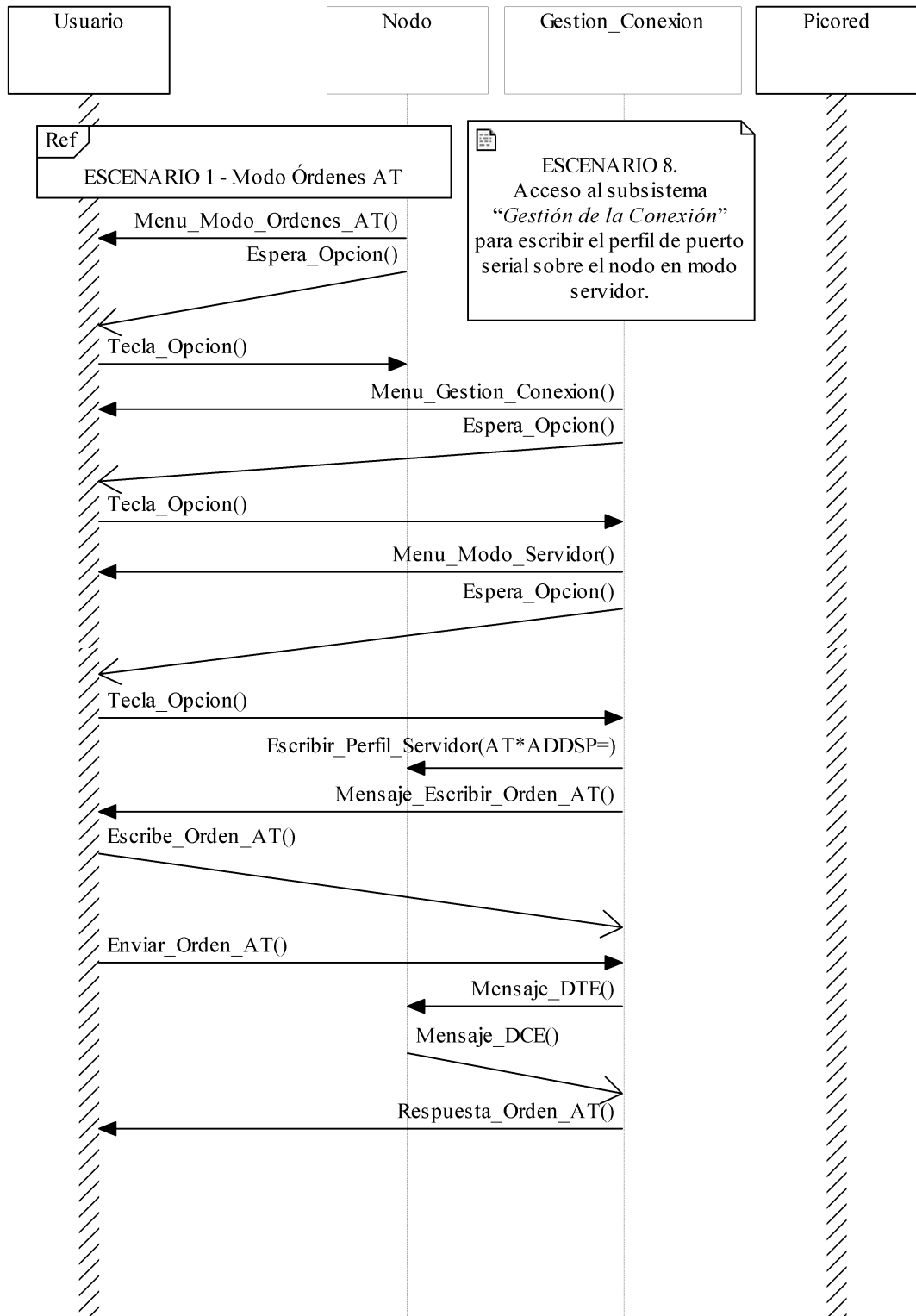


Figura 3.47. Diagrama de secuencia para el Escenario 8.

La Figura 3.48 muestra la secuencia de acceso al subsistema Gestión de la Conexión para escribir el soporte de enlaces multipunto en modo servidor (Escenario 9).

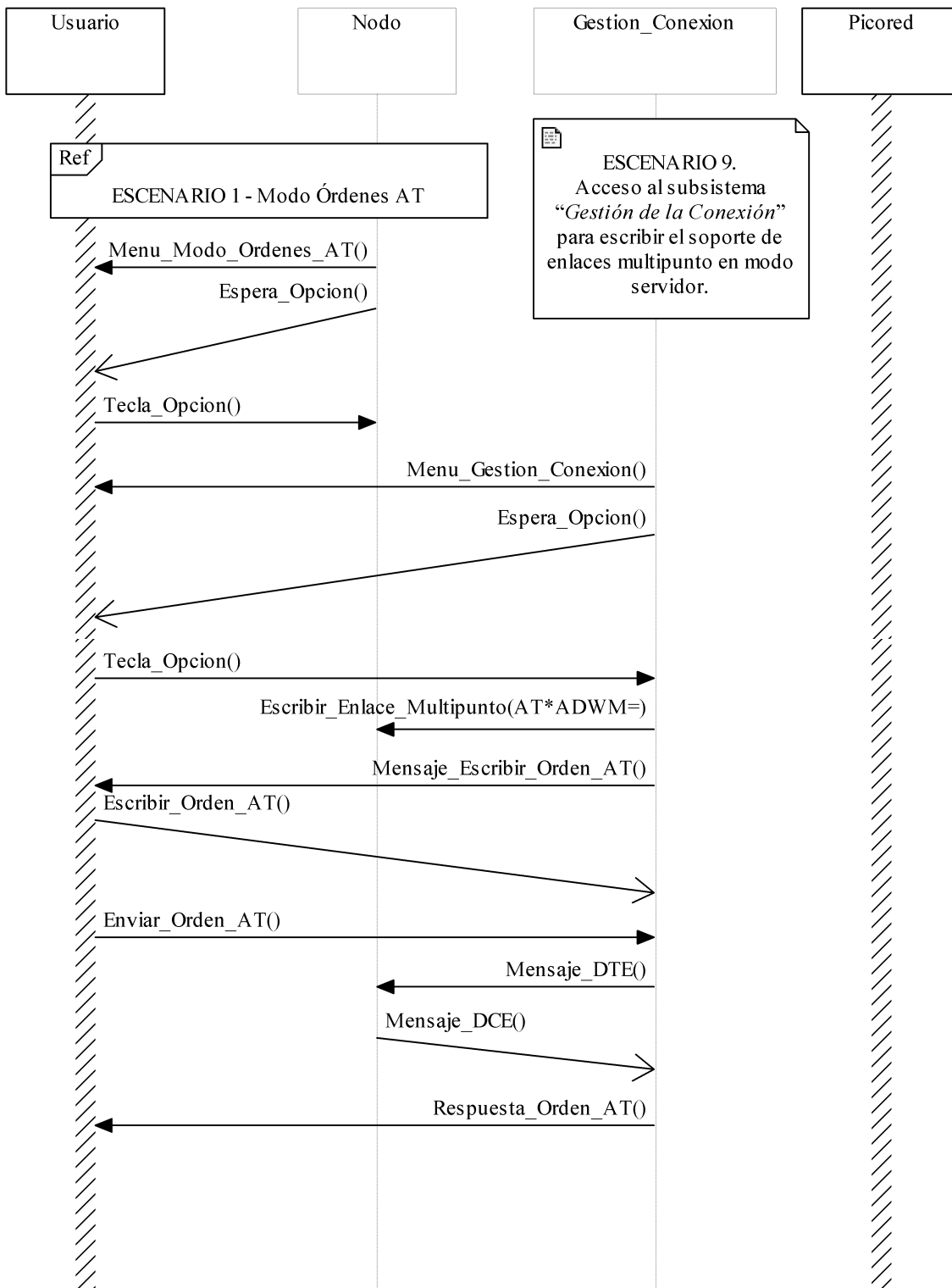


Figura 3.48. Diagrama de secuencia para el Escenario 9.

El diagrama de secuencia de la Figura 3.49, muestra el acceso al subsistema Gestión de la Conexión para escribir el tipo de intercambio del rol sobre el nodo en modo servidor (Escenario 10).

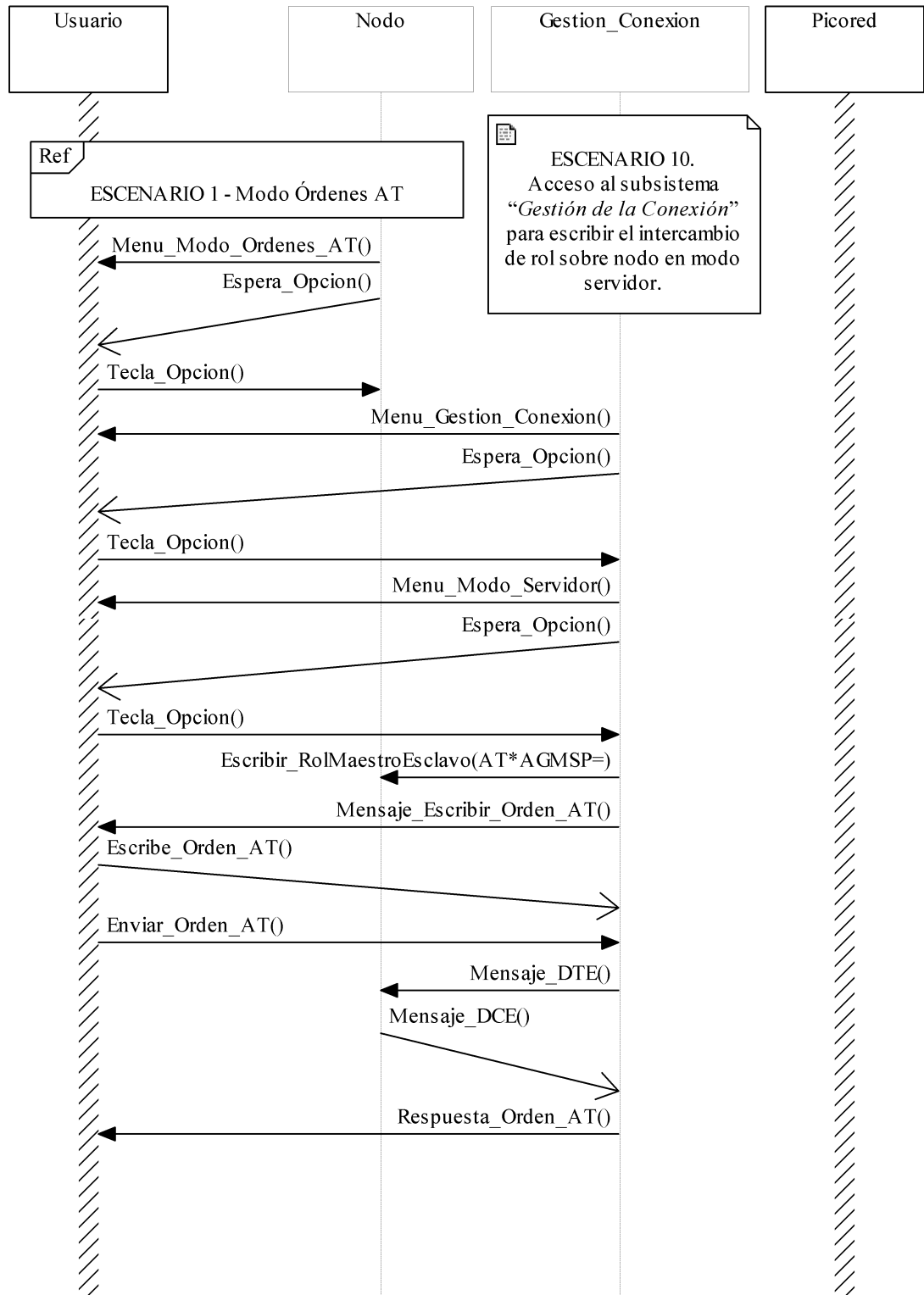


Figura 3.49. Diagrama de secuencia para el Escenario 10.

La Figura 3.50 muestra la secuencia de acceso al subsistema Gestión de la Conexión para la creación de una conexión serial multipunto con perfil de puerto serial en modo servidor (Escenario 11).

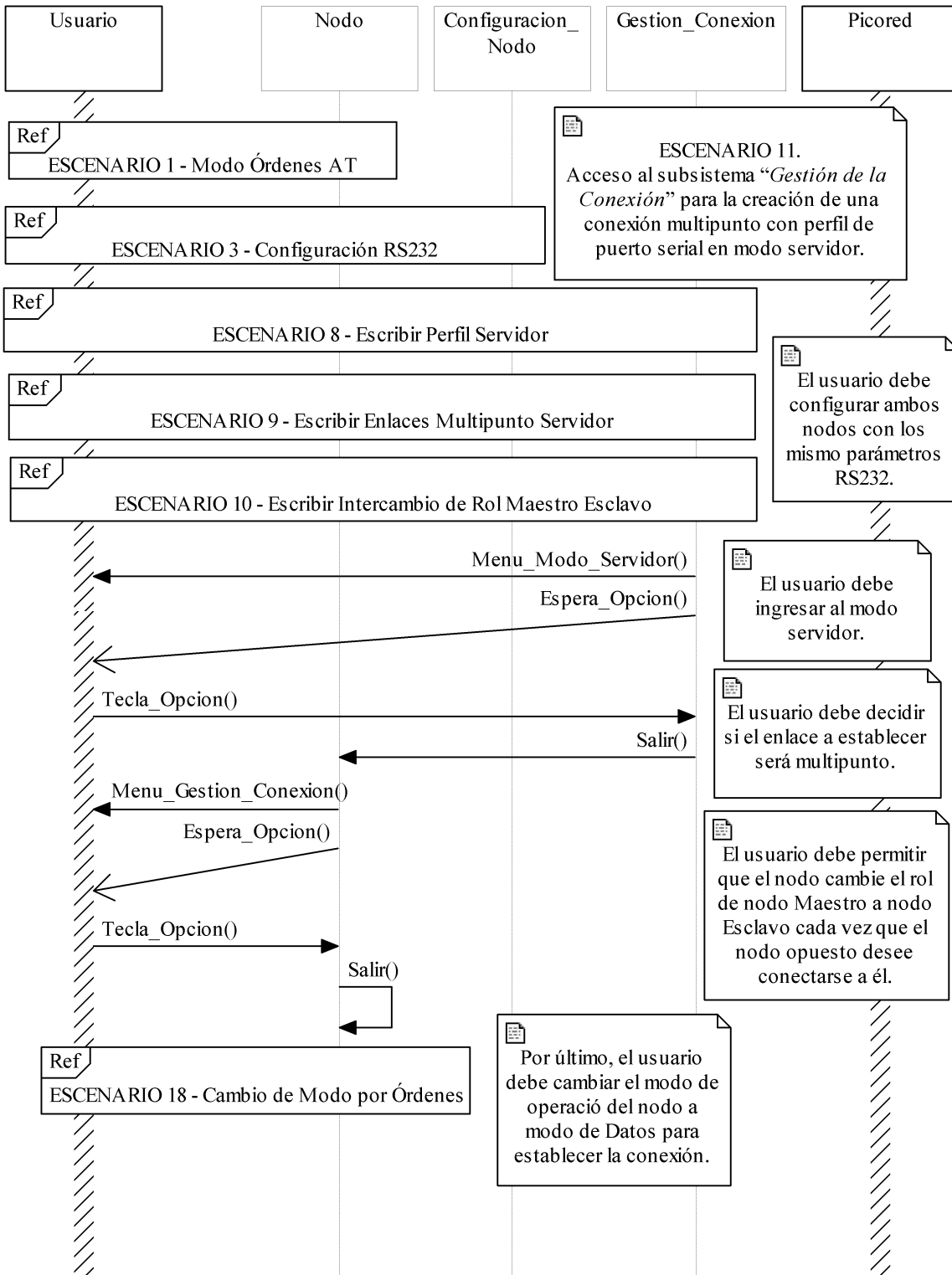


Figura 3.50. Diagrama de secuencia para el Escenario 11.

La Figura 3.51 muestra la secuencia de acceso al subsistema Servicios Suplementarios para escribir un tipo de modo de ahorro de energía sobre el nodo (Escenario 12).

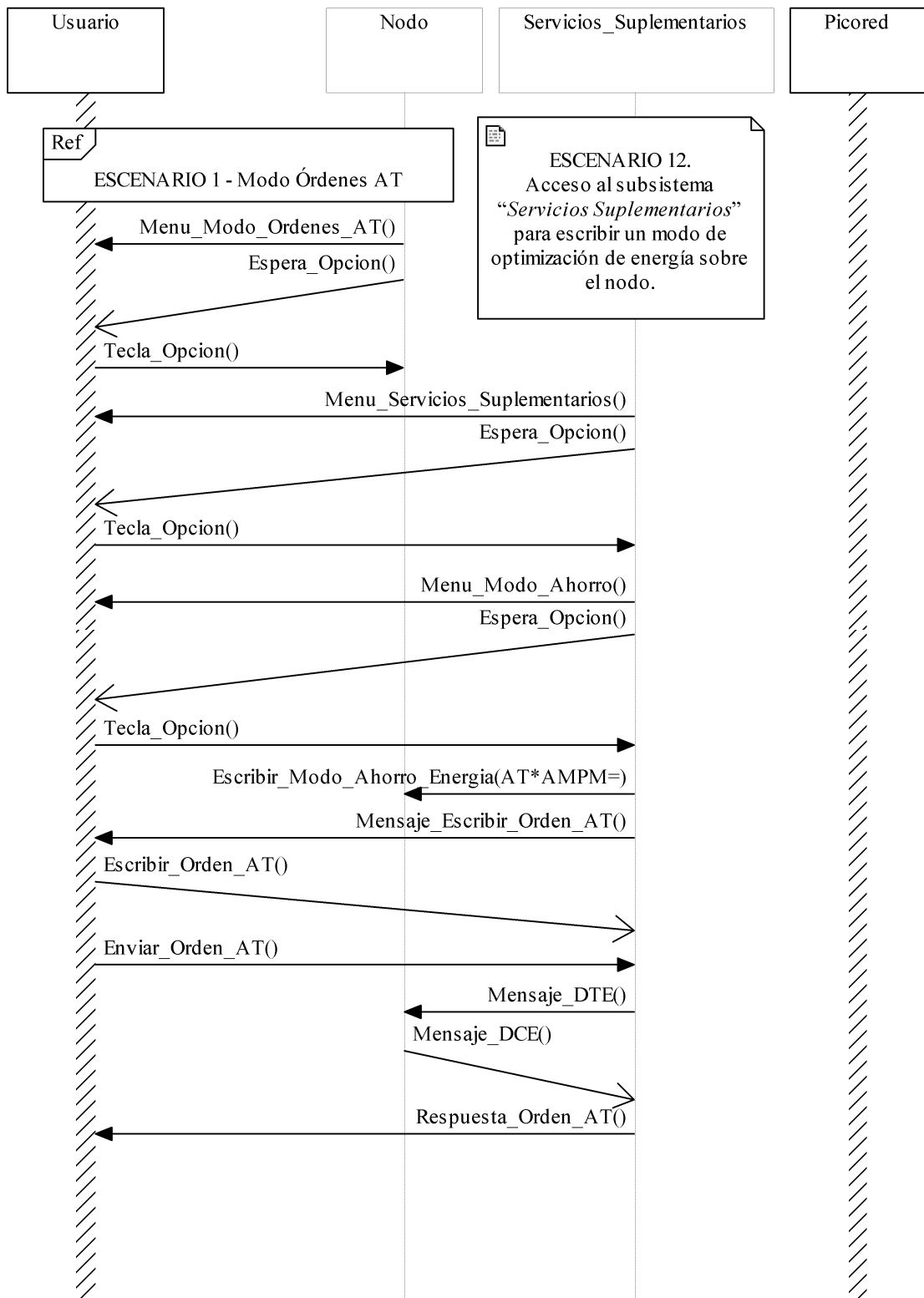


Figura 3.51. Diagrama de secuencia para el Escenario 12.

La Figura 3.52 muestra la secuencia de acceso al subsistema Servicios Suplementarios para descubrir dispositivos dentro de la vecindad del nodo (Escenario 13).

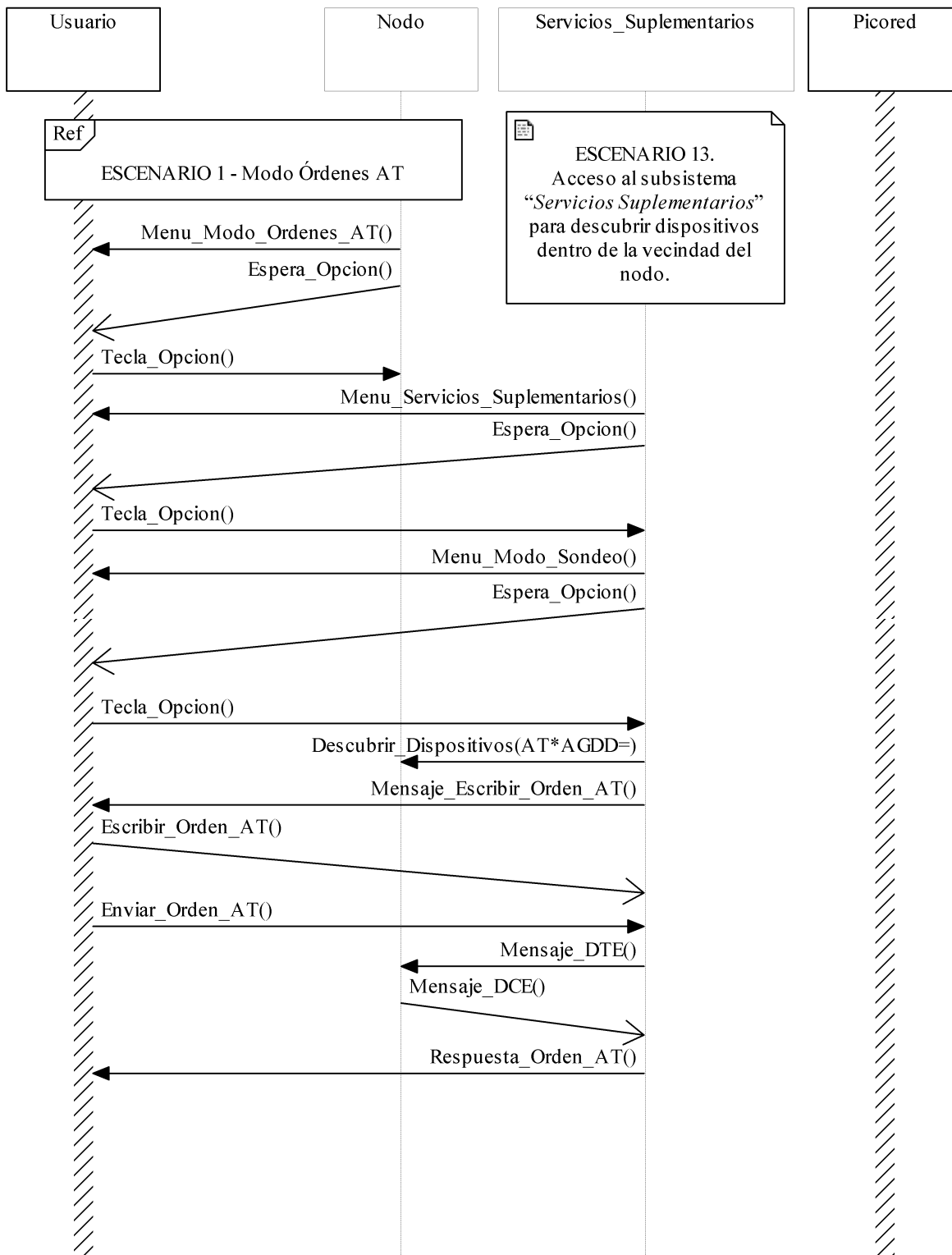


Figura 3.52. Diagrama de secuencia para el Escenario 13.

La Figura 3.53 muestra la secuencia al subsistema Servicios Suplementarios para establecer el tipo de modo de emparejamiento sobre el nodo (Escenario 14).

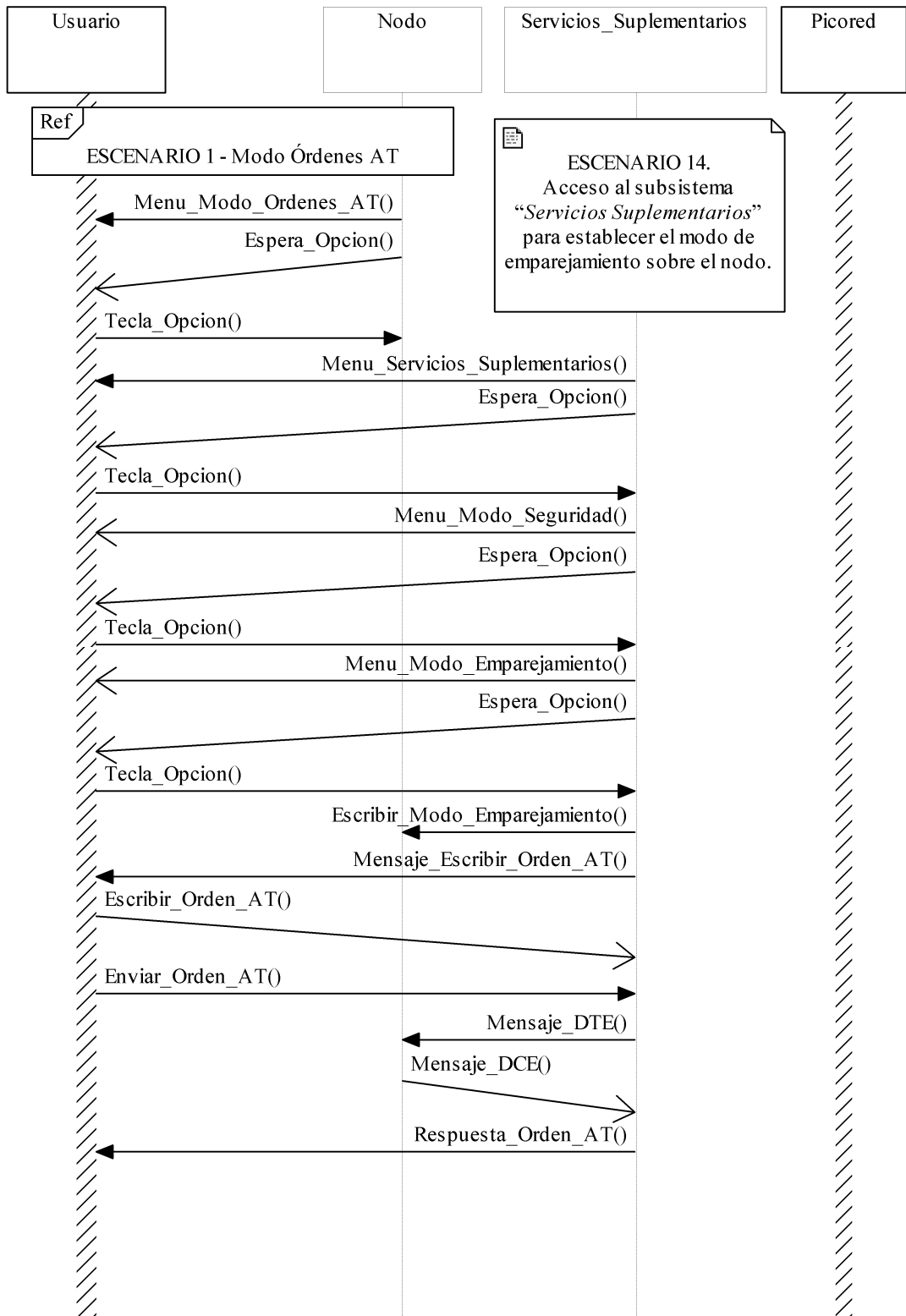


Figura 3.53. Diagrama de secuencia para el Escenario 14.

La Figura 3.54 muestra la secuencia de acceso al subsistema Servicios Suplementarios para escribir el código PIN sobre un nodo (Escenario 15).

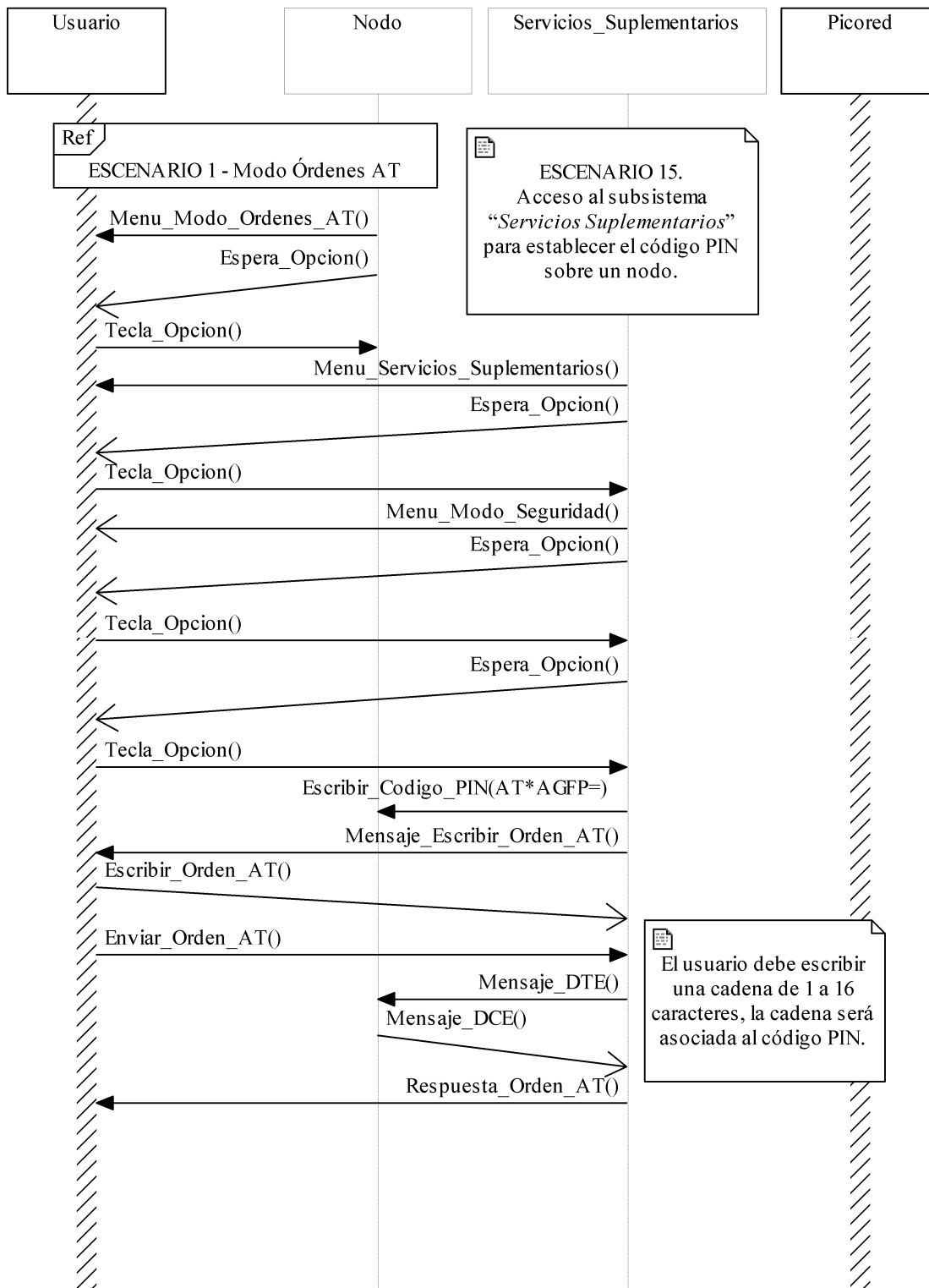


Figura 3.54. Diagrama de secuencia para el Escenario 15.

La Figura 3.56 muestra la secuencia de acceso al subsistema Servicios Suplementarios para escribir el tipo de modo de seguridad sobre el nodo (Escenario 17).

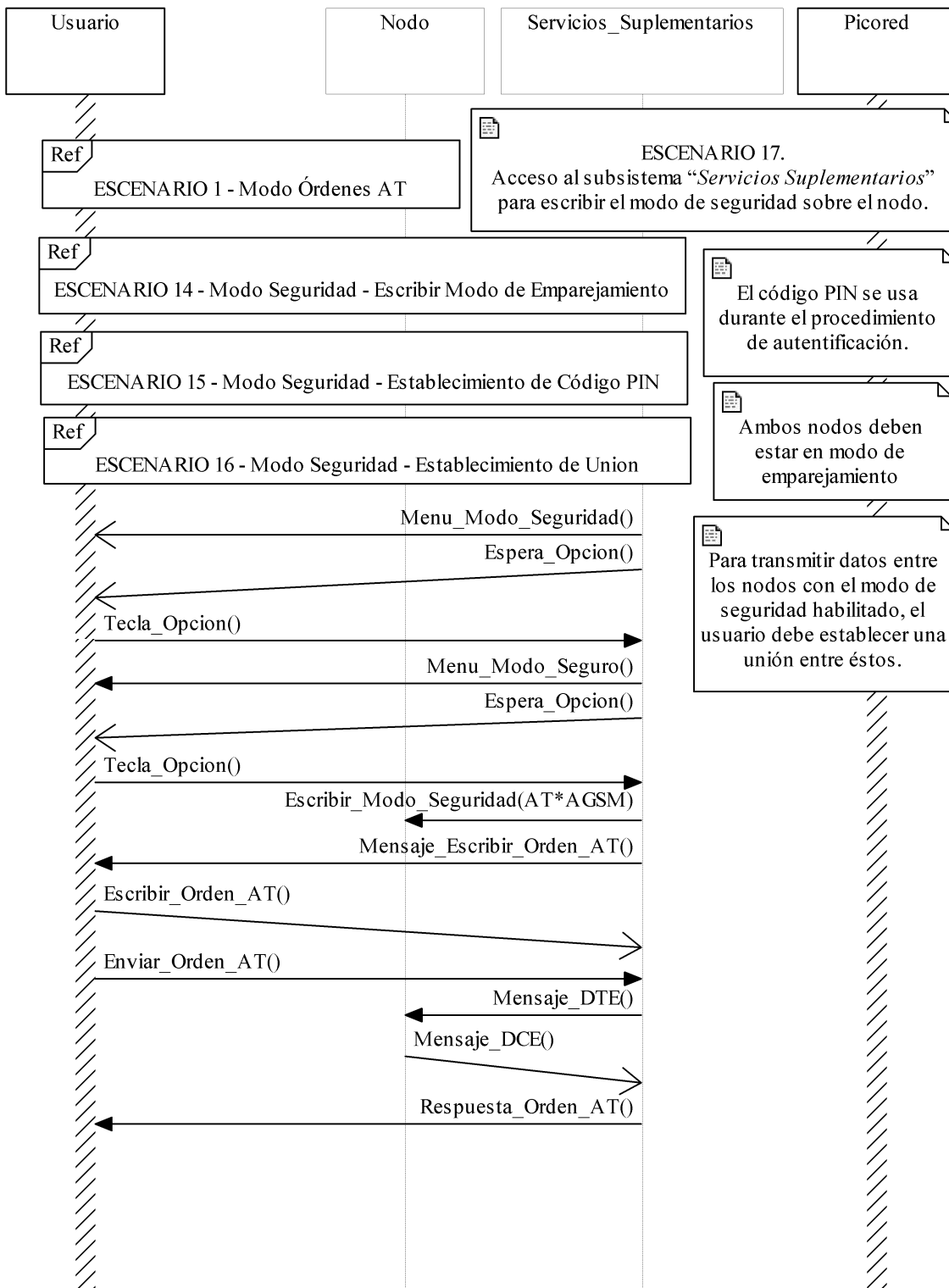


Figura 3.56. Diagrama de secuencia para el Escenario 17.

La Figura 3.57 muestra la secuencia para realizar el cambio del modo de operación de Órdenes AT al modo de Datos (Escenario 18).

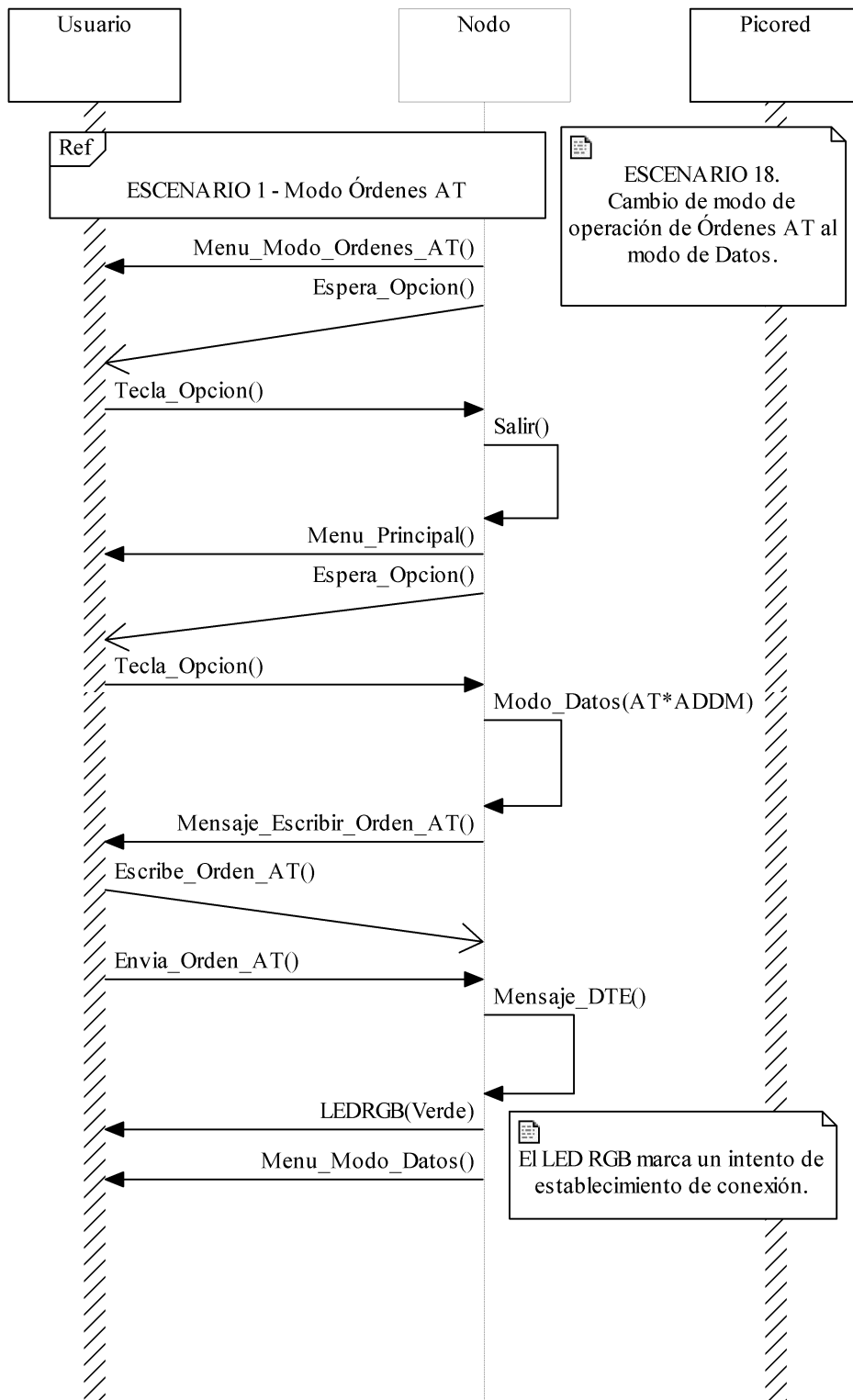


Figura 3.57. Diagrama de secuencia para el Escenario 18.

La Figura 3.58 muestra la secuencia para transmitir un mensaje de texto sobre la picored (Escenario 19).

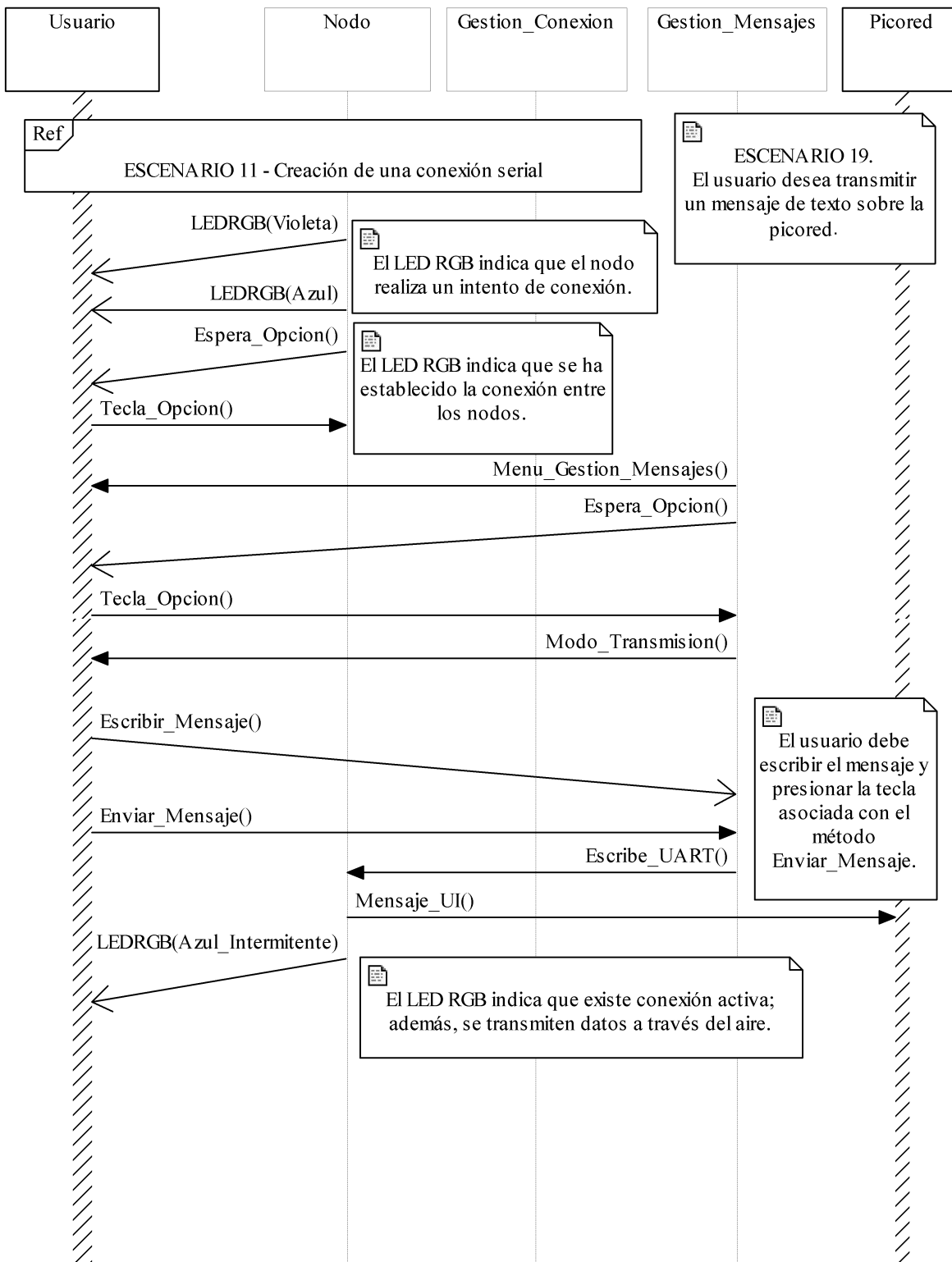


Figura 3.58. Diagrama de secuencia para el Escenario 19.

La Figura 3.59 muestra la secuencia para recibir un mensaje de texto sobre la picored (Escenario 20).

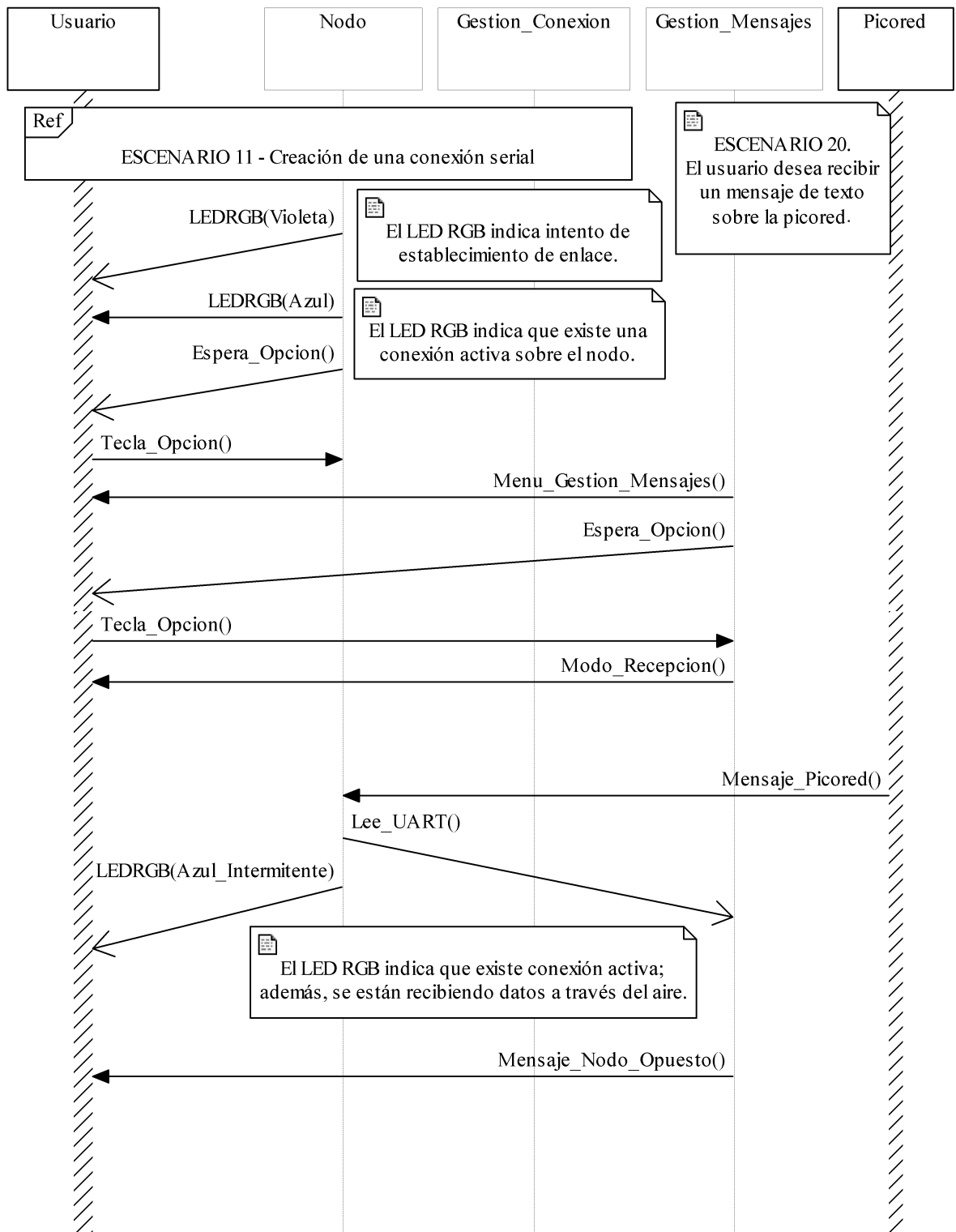


Figura 3.59. Diagrama de secuencia para el Escenario 20.

3.3.3. Diagramas de máquinas de estado

Los diagramas de máquinas de estado de la Figura 3.60, Figura 3.61, Figura 3.62, Figura 3.63 y Figura 3.64, ilustran las decisiones que el usuario debe realizar para elegir: un modo de operación, un subsistema, una generalización, o un método ubicados dentro de los diferentes menús de los nodos.

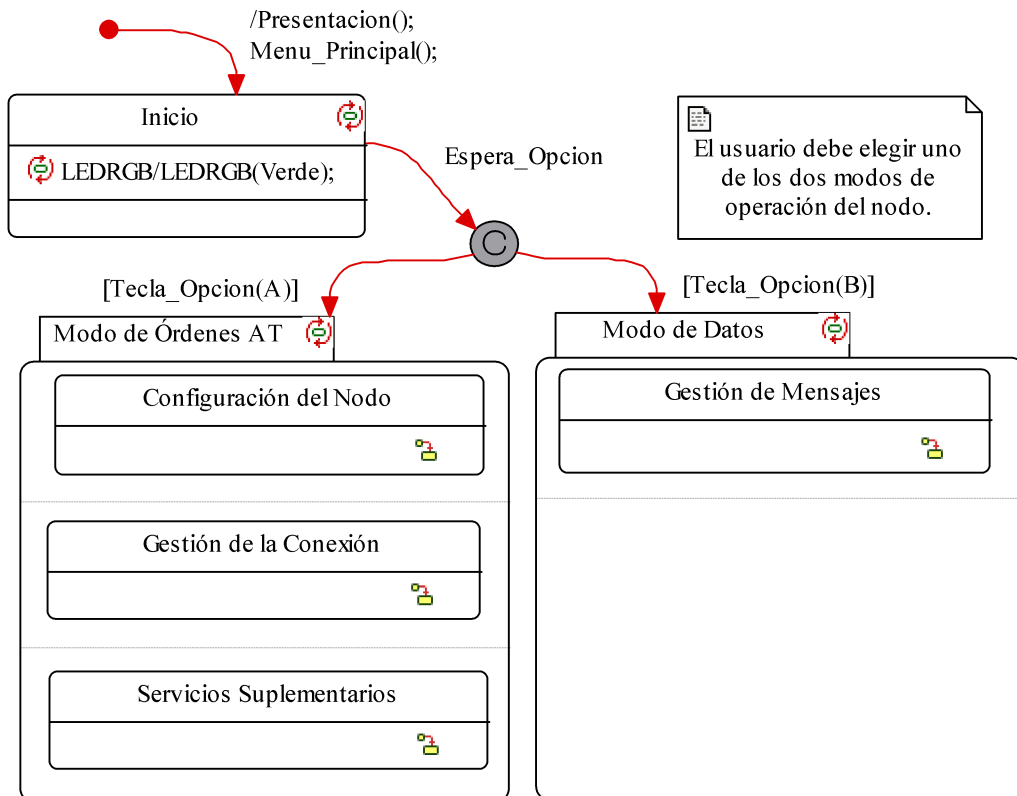


Figura 3.60. Máquina de estados para elegir modo de operación.

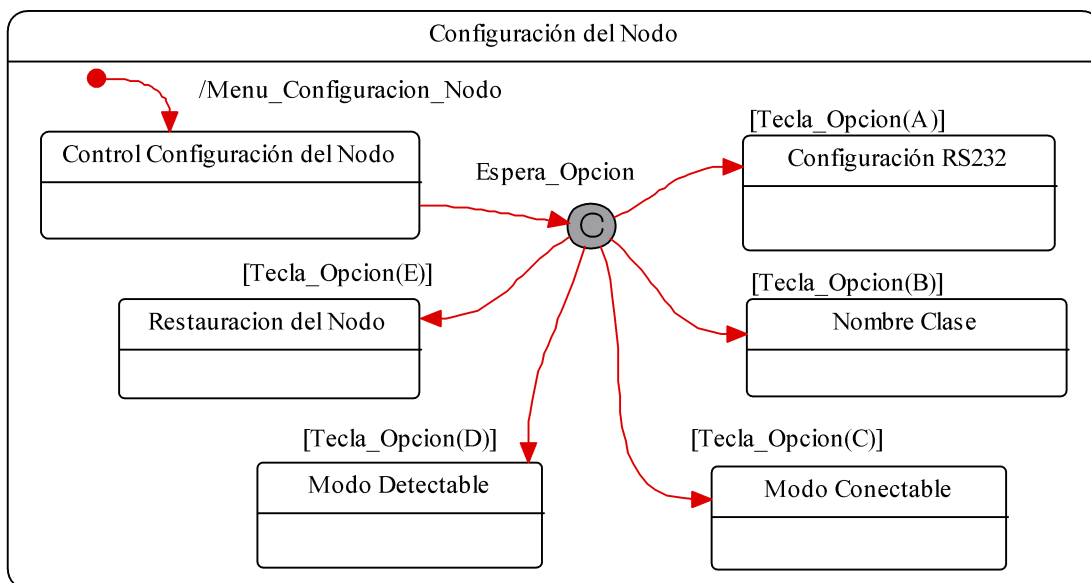


Figura 3.61. Máquina de estados dentro del subsistema Configuración del Nodo.

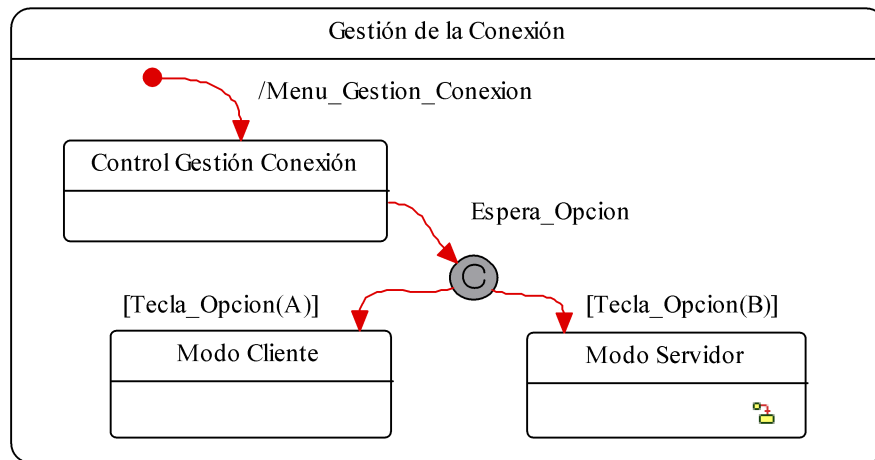


Figura 3.62. Máquina de estados para el establecimiento de una conexión serial.

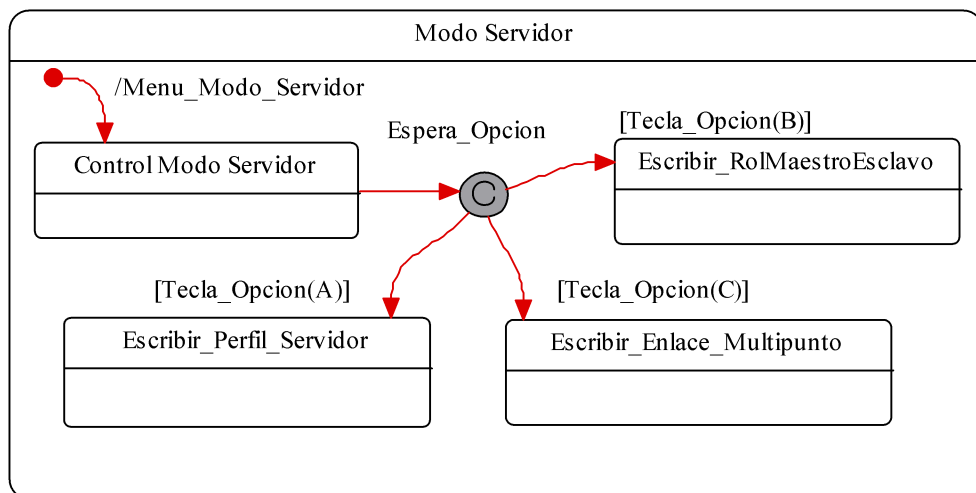


Figura 3.63. Máquina de estados dentro de la generalización Modo Servidor.

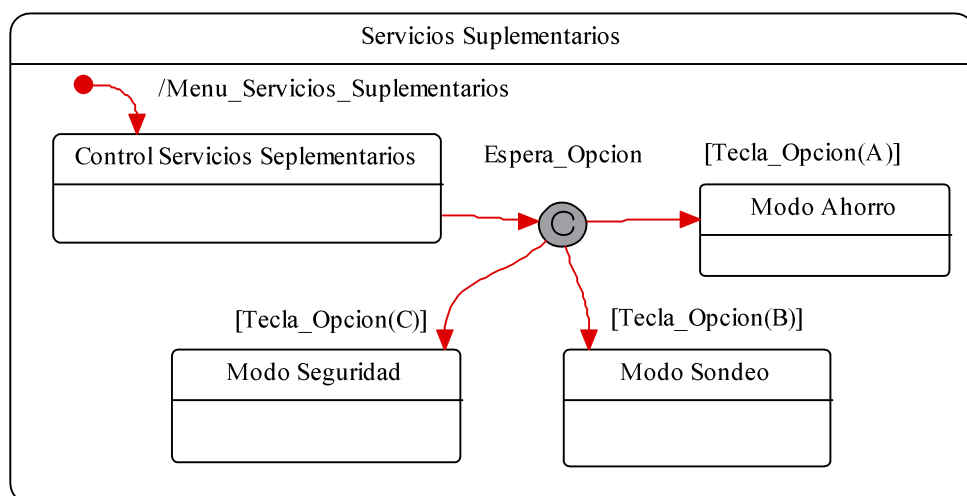


Figura 3.64. Máquina de estados dentro del subsistema Servicios Suplementarios.

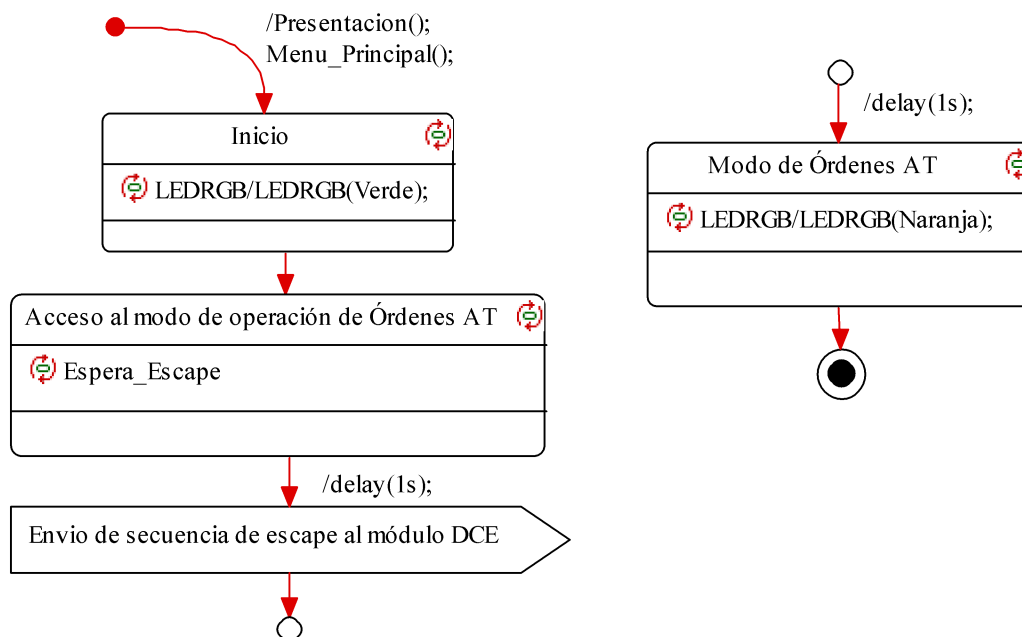


Figura 3.65. Máquina de estados para ingresar al modo de operación de Órdenes AT.

El diagrama de máquina de estados de la Figura 3.65 muestra la secuencia que realiza el nodo para ingresar al modo de operaciones de Órdenes AT. El diagrama de máquina de estados de la Figura 3.66 muestra los pasos que realiza el usuario sobre el nodo para ingresar al modo de Configuración RS232 y establecer los parámetros dentro del subsistema Configuración del Nodo. El diagrama de máquina de estados de la Figura 3.67 muestra los pasos que realiza el usuario sobre el nodo para el establecimiento de una conexión serial. El diagrama de máquina de estados de la Figura 3.68 muestra la secuencia para el envío de un mensaje a través de una conexión activa con perfil de puerto serial en modo servidor.

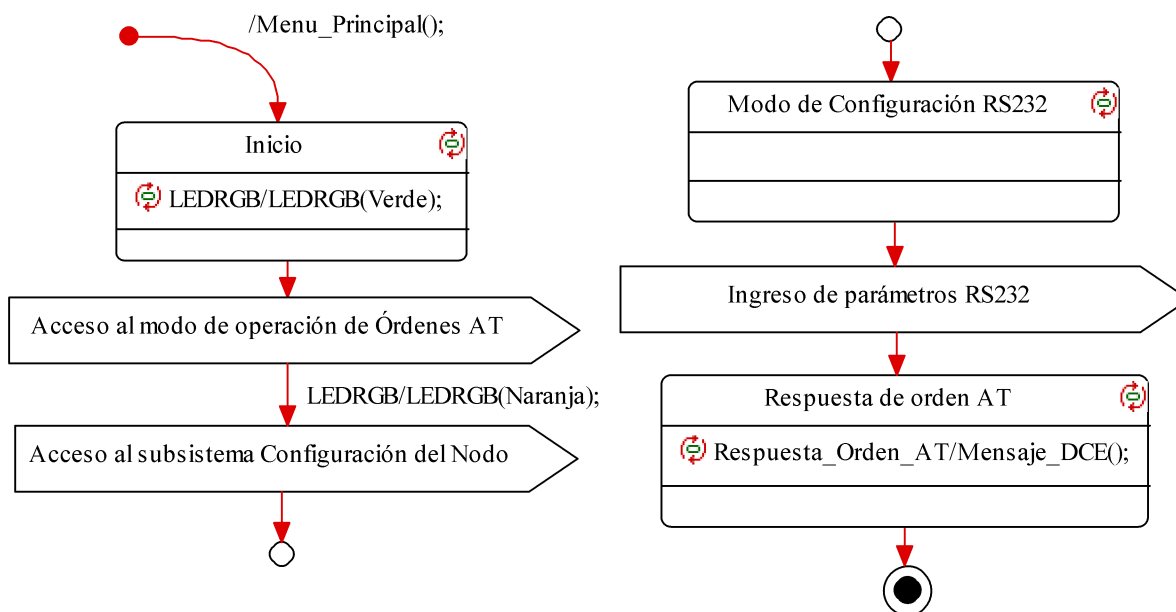


Figura 3.66. Máquina de estados para ingresar al modo de configuración RS232.

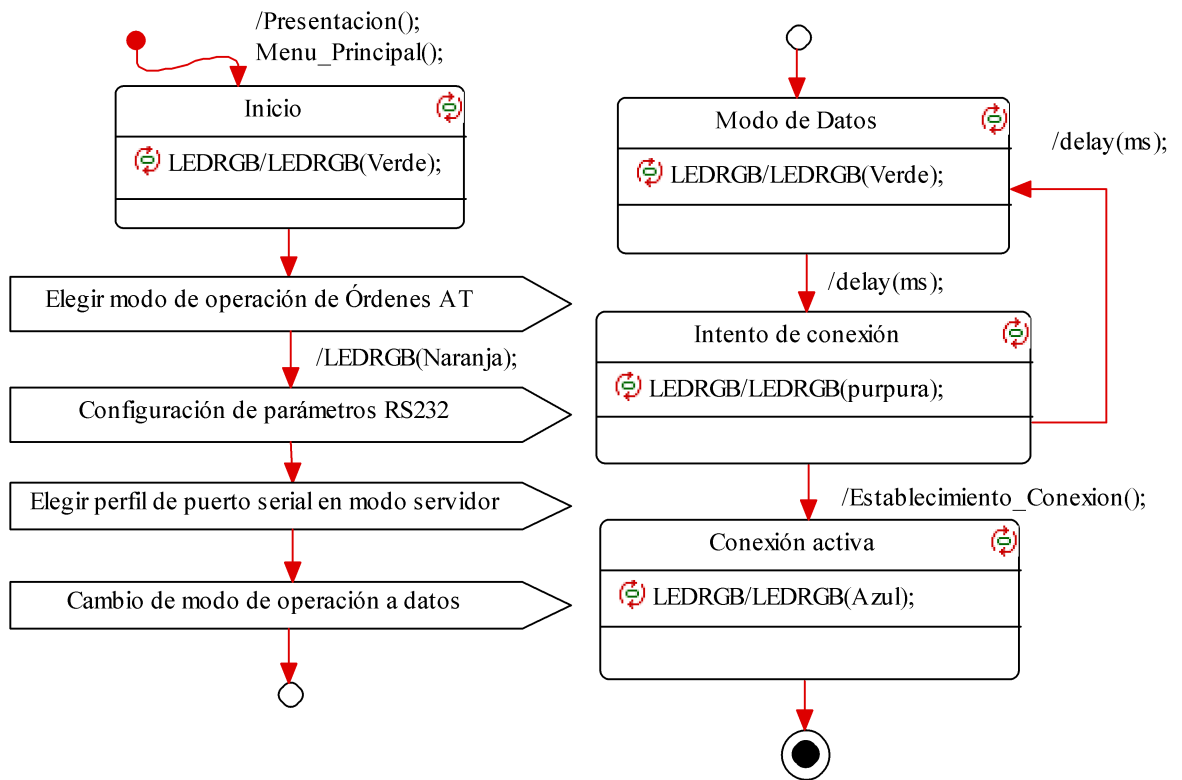


Figura 3.67. Máquina de estados de establecimiento de una conexión serial como Servidor.

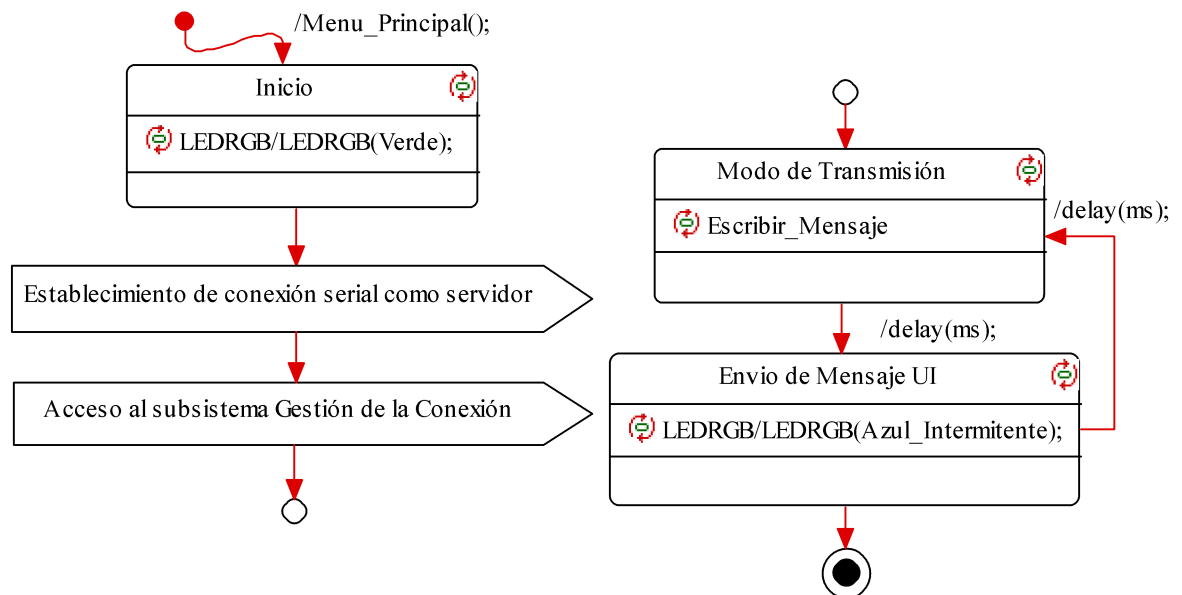


Figura 3.68. Máquina de estados para el envío de un mensaje.

3.4. Desarrollo del Sistema

El propósito de esta fase de la metodología²⁸ es la construcción y desarrollo de los componentes tanto HW como SW del sistema. Dentro de la fase de desarrollo se realizaron las siguientes actividades:

- Elección de los componentes HW para la construcción de los nodos.
- Construcción de las plataformas e interfaces de los módulos DTE y DCE.
- Desarrollo del SW para los módulos DTE.

3.4.1. Desarrollo electrónico

La elección de los componentes HW se realizó con base en los siguientes puntos:

- Los componentes HW deben cumplir con todos los requerimientos de la fase de especificaciones (Sección 3.2).
- Los componentes HW deben ser capaces de cubrir con todas las interfaces diseñadas para los nodos (Secciones 3.2.1 y 3.2.2).
- En el caso de la elección del MCU, éste debe contar con:
 - Al menos 2KB memoria de programa para efectuar todos los métodos diseñados.
 - Un temporizador (*timer*) encargado de realizar el efecto del cursor en pantalla²⁹.
 - Un módulo UART empotrado para establecer la comunicación serial entre el módulo DTE y el módulo DCE.
 - Un puerto de salida de 8 terminales para enviar los datos de usuario a la pantalla GLCD.
 - Un puerto de salida de 8 terminales encargado del control de la pantalla GLCD.
 - Un puerto de entrada de 4 terminales encargado de la captura de los datos de usuario provenientes del teclado matricial.
 - Un puerto de salida de 4 terminales encargado del “barrido” del teclado matricial.
 - Un puerto de salida de 2 terminales dedicadas a la gestión de la comunicación RS232.
 - Un módulo DSP para trabajos futuros sobre procesamiento de audio.
 - Un módulo CAN para trabajos futuros sobre el control y monitoreo de sensores.

3.4.1.1. Desarrollo del módulo DTE

Los componentes HW elegidos para el desarrollo del módulo DTE, cumplen con los requerimientos solicitados en la Sección 3.2.1. Las características principales de estos componentes HW se encuentran en el Anexo DTE. La Tabla 3.10 contiene los componentes HW que forman al módulo DTE.

²⁸ En la fase de desarrollo del sistema no se llevan a cabo los procesos: Desarrollo electromecánico y Desarrollo de otras tecnologías.

²⁹ El método asociado al cursor es opcional por lo que no fue contemplado en la fase de requerimientos.

Tabla 3.10. Componentes del módulo DTE.

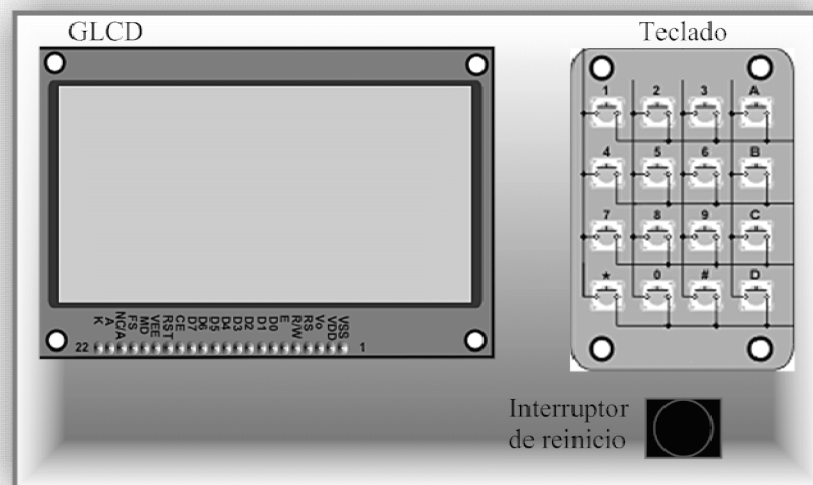
Componente HW	Interfaz	Fabricante
MCU dsPIC30F4013	-	Microchip
GLCD de 128x64 píxeles	UI_DTE	Sure electronics
Teclado matricial de 4x4 interruptores	UI_DTE	General
Interruptor de reinicio	UI_DTE	General
Transceptor RS232 MAX232	COM_DTE	Texas Instruments
Conector DB9 hembra	COM_DTE	General
Conector de alimentación de 2.1 mm	-	General

3.4.1.1.1. Desarrollo de la UI_DTE

Con el propósito de facilitar el desarrollo y uso de la UI_DTE, la pantalla GLCD y el teclado se colocan de forma contigua; el interruptor de reinicio se ubica por debajo del teclado y del MCU³⁰. La Figura 3.69 muestra los componentes de la UI_DTE.

3.4.1.1.2. Desarrollo de la COM_DTE

En el desarrollo de la COM_DTE, el transceptor MAX232 se coloca al lado del conector DB9 hembra con el propósito de reducir el costo en área sobre la plataforma de desarrollo del módulo DTE (Figura 3.70).

**Figura 3.69.** Componentes de la UI_DTE

³⁰ En la Figura 3.69 no se muestra la colocación del MCU debido a que no es parte de la UI_DTE.

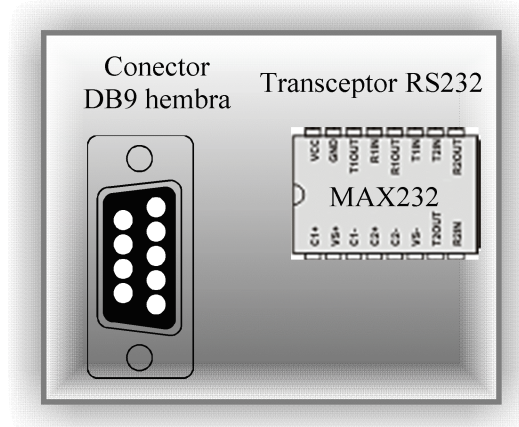


Figura 3.70. Componentes de la COM_DTE.

3.4.1.2. Desarrollo del módulo DCE

El módulo DCE se construye a partir de los componentes HW mostrados en la Tabla 3.11 y se cumplen con los requerimientos de la Sección 3.2.2. Las características principales de los componentes HW para el módulo DCE se encuentran en el Anexo DCE.

Tabla 3.11. Componentes del módulo DCE.

Componente HW	Interfaz	Fabricante
Adaptador de puerto serie OEMSPA13i	CINP	connectBlue
Adaptador de RS232 OEMII	COM_DCE e UI_DCE	connectBlue

3.4.1.2.1. Desarrollo de la CINP

En el desarrollo de la CINP se seleccionó el dispositivo OEMSPA13i (Figura 3.71) debido a que cumple con todos los componentes HW solicitados por la interfaz CINP. La Figura 3.72 muestra los principales componentes del dispositivo OEMSPA13i [URL2].

Tabla 3.12. Componentes del dispositivo OEMSPA13i.

Componente HW	Interfaz	Fabricante
Antena interna Mica 2.4Ghz ³¹	CINP	gigaAnt
Transceptor RS232 ADM3315	CINP	Analog Devices
Radio Bluetooth PBA 313 05	CINP	Infineon/Ericsson
Controlador de banda base PBM 990 80	CINP	Infineon/Ericsson

³¹ En el anexo módulo DCE se resumen las características de la antena interna.

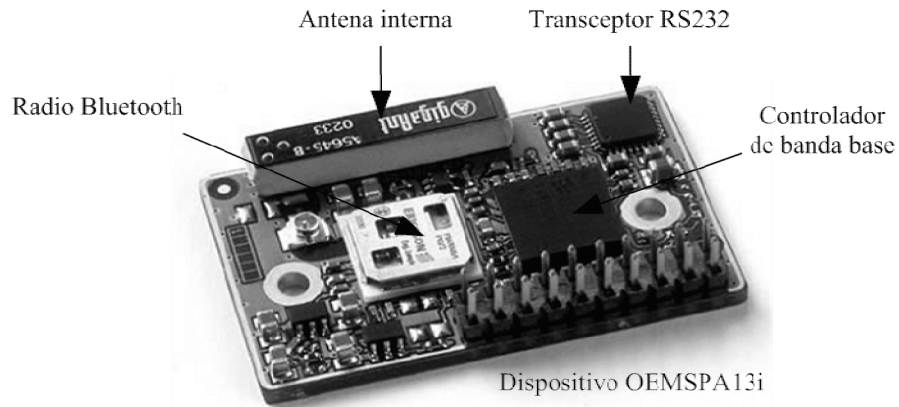


Figura 3.71. Componentes de la CINP [8].

3.4.1.2.2. Desarrollo de la COM_DCE e UI_DCE

En el desarrollo tanto de la COM_DCE y la UI_DCE, se eligió el adaptador de RS232 OEMII (Figura 3.72). El adaptador cumple con los componentes HW solicitados por la interfaces COM_DCE y UI_DCE. La Tabla 3.13 lista los componentes principales del adaptador de RS232 OEMII [URL2].

Tabla 3.13. Componentes del adaptador de RS232 OEMII.

Componente HW	Interfaz	Fabricante
Conector DB9 macho	COM_DCE	connectBlue
LED RGB	UI_DCE	connectBlue
Interruptor de restauración	UI_DCE	connectBlue
Conector de alimentación de 2.1 mm	-	connectBlue

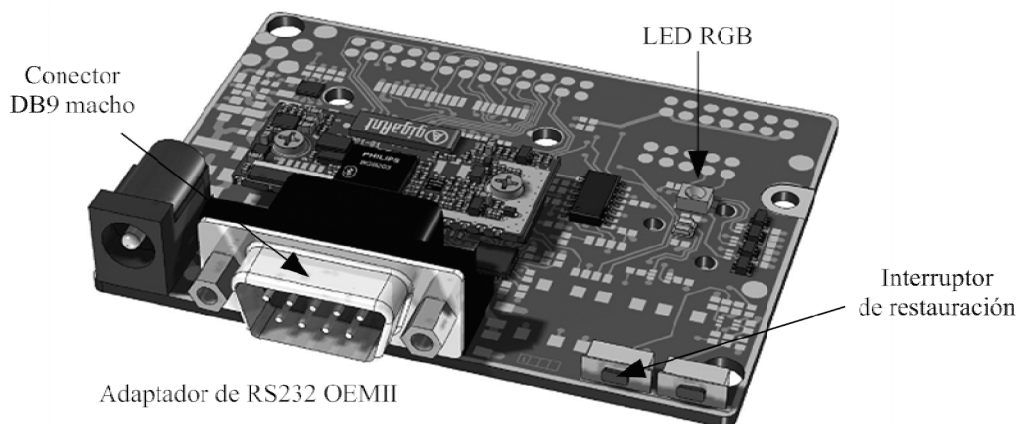


Figura 3.72. Componentes de la COM_DCE y de la UI_DCE [8].

3.4.2. Desarrollo del software

Para la construcción del SW del sistema DreamBlue se emplearon los componentes SW mostradas por la Tabla 3.14.

Tabla 3.14. Componentes SW empleados.

Componente SW	Función	Fabricante
Rhapsody	Entorno de diseño y desarrollo utilizado para el modelado del sistema DreamBlue	Telelogic IBM
mikroC dsPIC	Entorno de desarrollo utilizado para realizar el programa sobre el MCU dsPIC30F4013	Mikroelektronika
MASTER-PROG	Programa utilizado para la programación y actualización del firmware sobre los MCU dsPIC30F4013	Edutronica
GLCD Bitmap generator	Entorno de desarrollo para crear figuras monocromáticas	MikroElektronika

La Figura 3.73 muestra la interfaz del programa Rhapsody; la Figura 3.74 muestra el entorno de desarrollo para el programa mikroC dsPIC [URL6]; la Figura 3.75 muestra el entorno de desarrollo de figuras monocromáticas para el GLCD; y por último, la Figura 3.76 muestra la interfaz del programador MASTER-PROG, utilizado para la programación y actualización del firmware sobre los MCU dsPIC30F4013 de cada nodo.

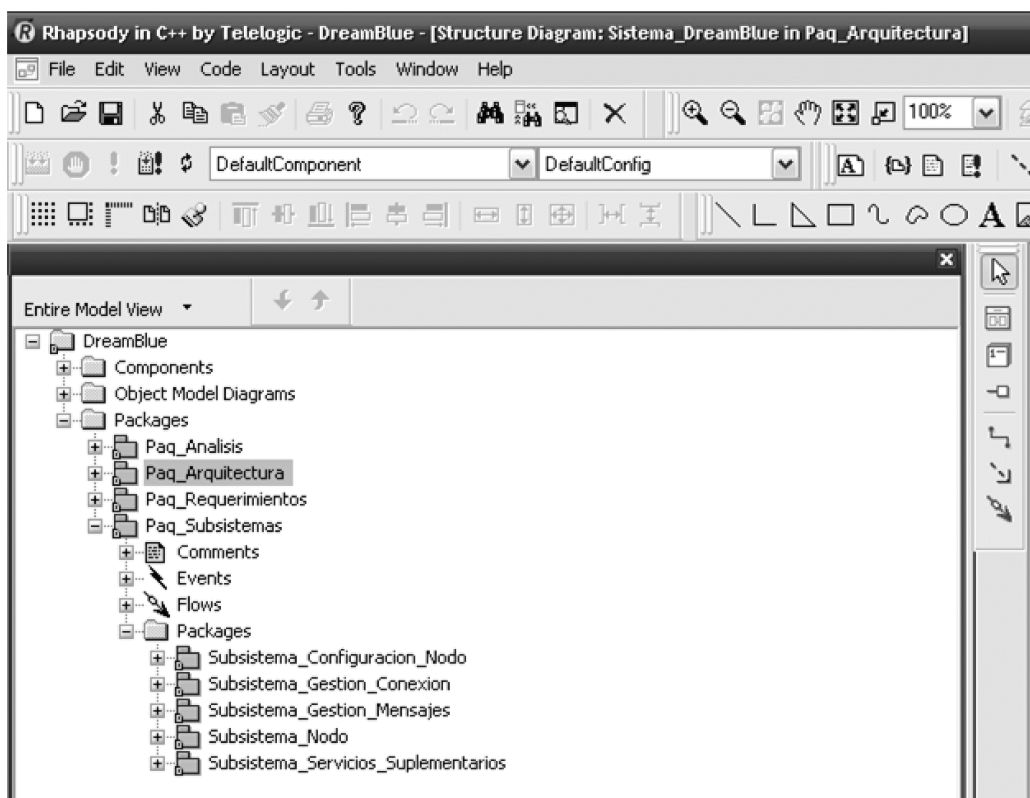


Figura 3.73. Interfaz del programa Rhapsody.

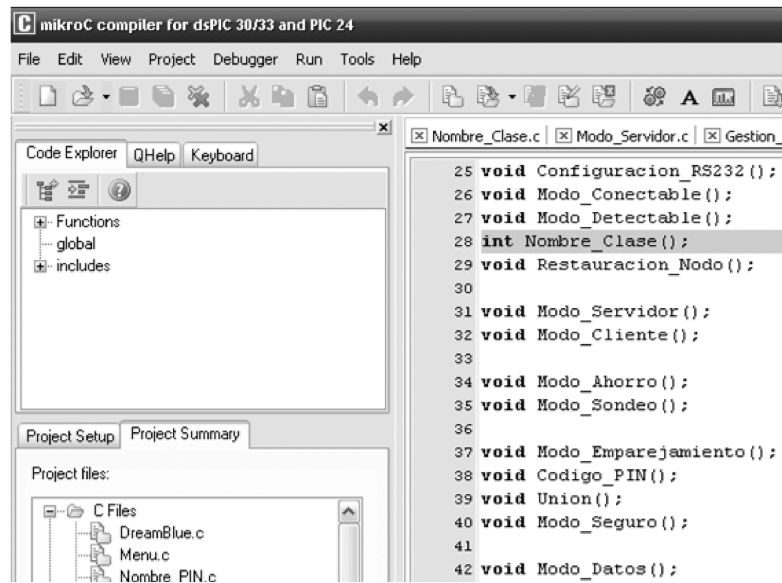


Figura 3.74. . Entorno de desarrollo mikroC dsPIC.

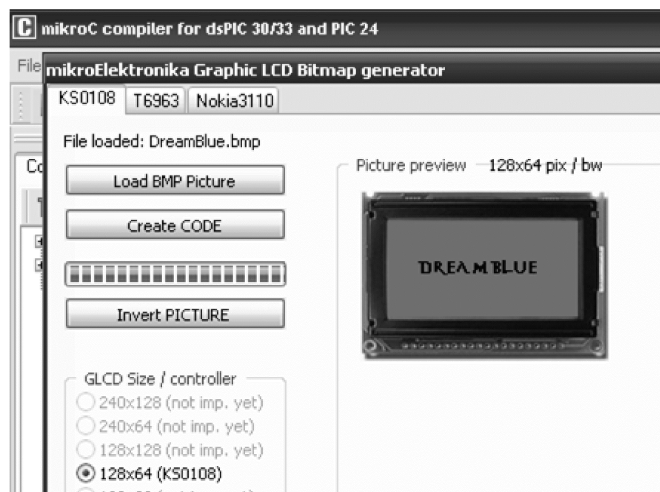


Figura 3.75. Entorno de diseño y para imágenes GLCD Bitmap generator.



Figura 3.76. Interfaz del programador MASTER-PROG.

3.5. Integración

El propósito de esta fase es la integración de los módulos DTE y DCE para unificar cada uno de los nodos del sistema DreamBlue. En la fase de integración se realizaron las siguientes actividades:

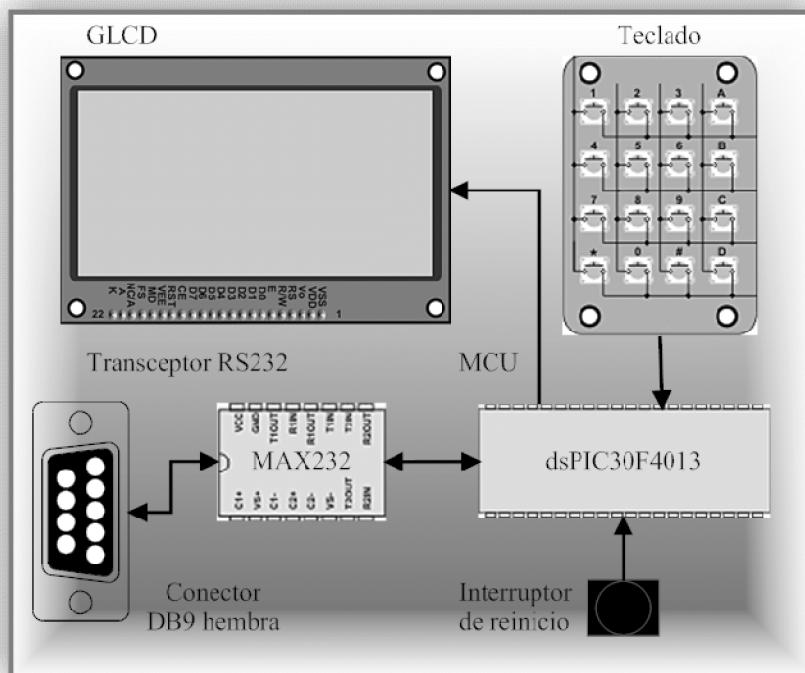
- Integración del módulo DTE.
- Integración del módulo DCE.
- Integración del nodo DreamBlue.

3.5.1. Integración del módulo DTE

En la integración del módulo DTE, el MCU dsPIC30F4013 se encarga de administrar las tareas sobre la UI_DTE y la COM_DTE, además de establecer la comunicación entre el módulo DTE y el módulo DCE. La Figura 3.77 muestra el diagrama de integración para el módulo DTE.

3.5.2. Integración del módulo DCE

El dispositivo OEMSPA13i se ensambla al adaptador de RS232 para formar el módulo DCE como se muestra el diagrama de la Figura 3.78.



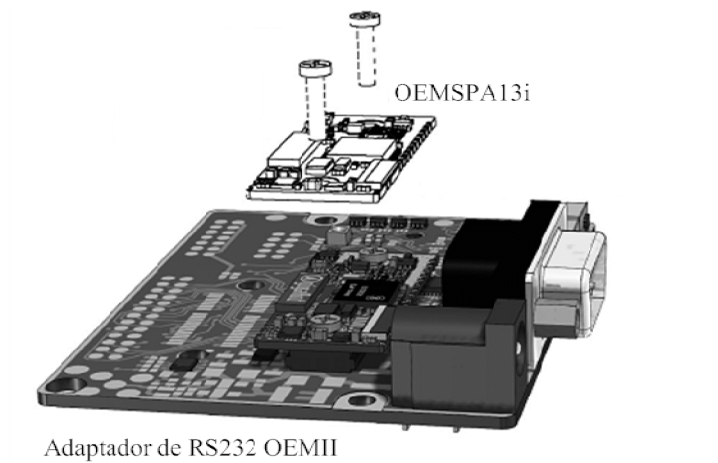


Figura 3.78. Diagrama de integración del módulo DCE [8].

3.5.3. Integración del nodo DreamBlue

Para formar cada uno de los nodos DreamBlue, el módulo DTE y el módulo DCE se acoplan por medio de sus interfaces COM_DTE y COM_DCE respectivamente. La Figura 3.79 representa el diagrama del prototipo final para cada uno de los nodos del sistema DreamBlue.

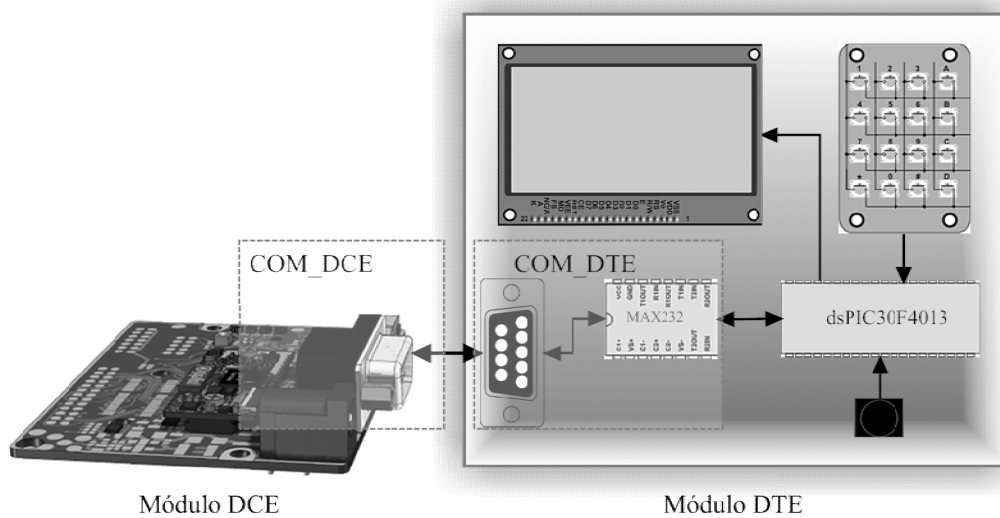


Figura 3.79. Diagrama de integración del nodo DreamBlue.

4. Resultados

Como resultado del trabajo de investigación se obtuvieron dos prototipos finales para los nodos del sistema DreamBlue (Figura 4.1), los cuales cumplen las especificaciones de requerimientos del sistema (Sección 3.2).

A continuación se describen las pruebas más importantes para la verificación de los nodos DreamBlue.

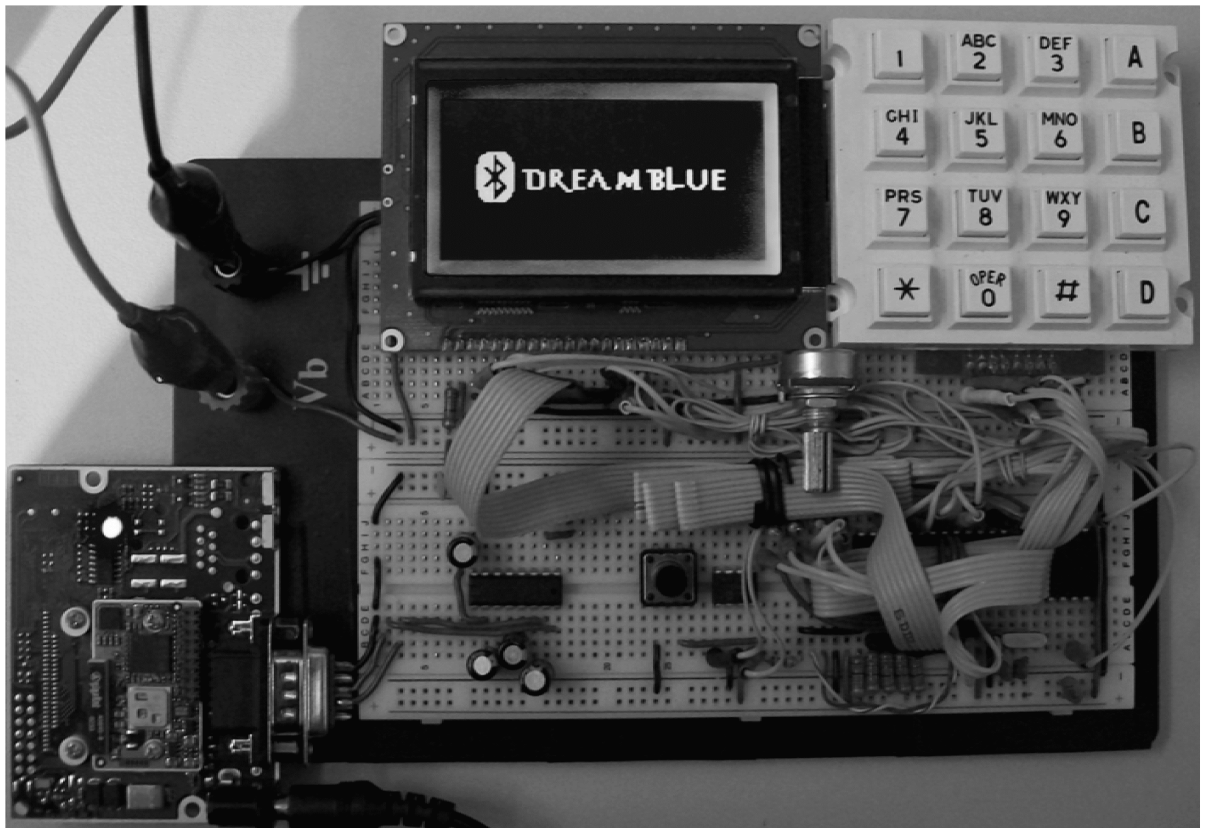


Figura 4.1. Prototipo final para los nodos del sistema DreamBlue.

4.1. Verificación del sistema DreamBlue

El propósito de esta fase de la metodología consiste en realizar pruebas de campo al sistema y verificar su correcto funcionamiento. Las actividades efectuadas sobre los nodos DreamBlue fueron:

- Acceso a los modos de operación.
- Pruebas de configuración a diferentes velocidades de transmisión.
- Asignación de nombres locales.
- Pruebas de sondeo de dispositivos.
- Configuración del enlace entre los nodos.
- Establecimiento de seguridad en la conexión.
- Establecimiento de una picored entre los nodos DreamBlue.
- Pruebas de transmisión de mensajes entre los nodos DreamBlue.
- Pruebas de cobertura.

Las pruebas efectuadas sobre los nodos siguieron los pasos descritos por los diagramas de secuencia obtenidos en la fase de diseño del sistema (Sección 3.3.2).

Para ingresar a los dos modos de operación de los nodos DreamBlue se implementó el menú mostrado en la Figura 4.2. Para realizar la prueba de ingreso al modo de operación de Órdenes AT se efectuaron los pasos descritos por el escenario 1. El menú del modo de operación de Órdenes AT se muestra en la Figura 4.3.



Figura 4.2. Menú principal de los modos de operación de los nodos DreamBlue.



Figura 4.3. Modo de operación de Órdenes AT.

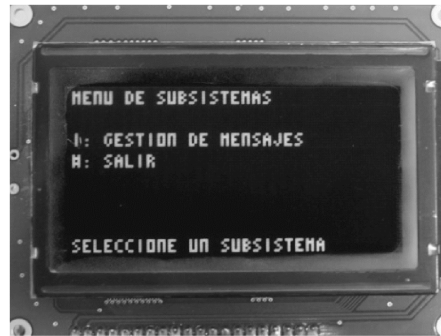
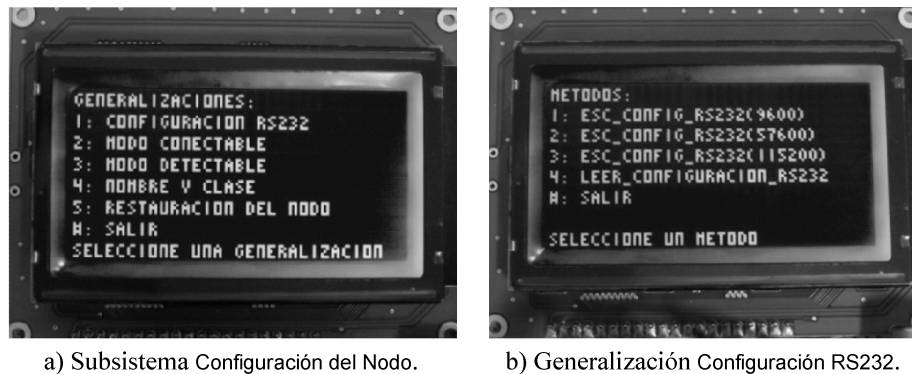


Figura 4.4. Modo de operación de Datos.

Para verificar el ingreso al modo de operación de Datos se probaron los pasos diseñados en el escenario 2 (Figura 4.4).

Para realizar las pruebas de configuración de la velocidad de transmisión sobre los nodos se siguieron los pasos descritos en el escenario 3 (Figura 4.5).

No se implementaron velocidades de transmisión superiores a 115 200 baudios debido a que el módulo DCE no soporta la configuración inmediata a dichas velocidades³². La versión final del sistema DreamBlue implementa tres diferentes velocidades de transmisión: 9 600, 57 600 y 115 200 baudios. La Figura 4.6 muestra la verificación del ajuste de velocidad de transmisión sobre un nodo DreamBlue.



a) Subsistema Configuración del Nodo.

b) Generalización Configuración RS232.

Figura 4.5. Velocidades de transmisión de los nodos.



Figura 4.6. Respuesta de lectura de la velocidad de transmisión.

³² Para ello es necesario inhabilitar el suministro de energía del módulo DCE.



Figura 4.7. Generalización Nombre y Clase.

La Figura 4.7 muestra el menú implementado para la elección de los métodos de lectura de la dirección Bluetooth del nodo, escritura y lectura de los nombres locales.

Debido a que en un inicio no existe distinción entre los nodos DreamBlue, a cada nodo se le asignó un nombre local para facilitar las pruebas de configuración inicial del enlace, de seguridad y de transmisión de mensajes entre los nodos DreamBlue. Los nombres asignados³³ son NODO_DREAMBLUE_A (Figura 4.8) y NODO_DREAMBLUE_B.

Después de asignar los nombres locales a cada nodo DreamBlue, se activó el modo detectable (escenario 5) sobre ambos nodos para realizar pruebas de sondeo de dispositivos Bluetooth (escenario 13). Las pruebas de sondeo fueron ejecutadas sobre el NODO_DREAMBLUE_A (Figura 4.9).

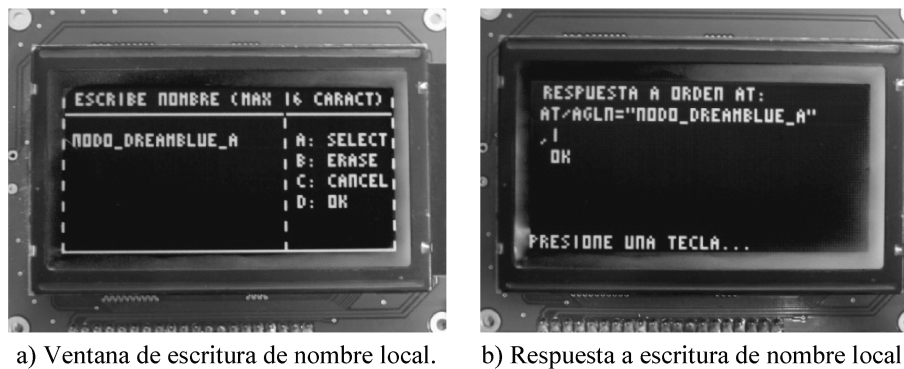


Figura 4.8. Escritura de nombre local al nodo.

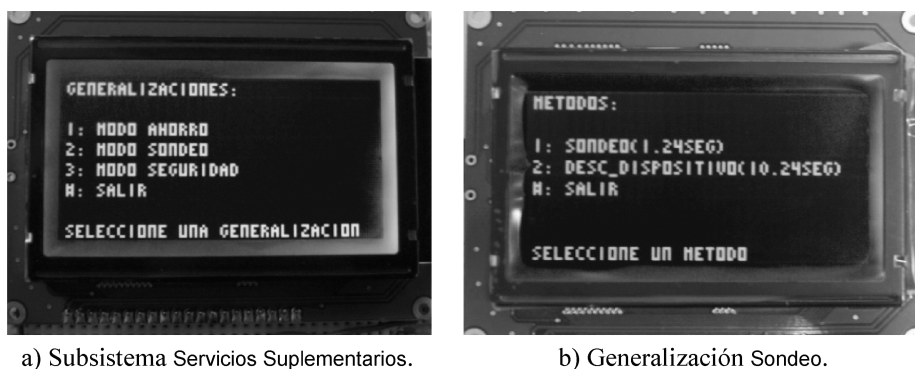
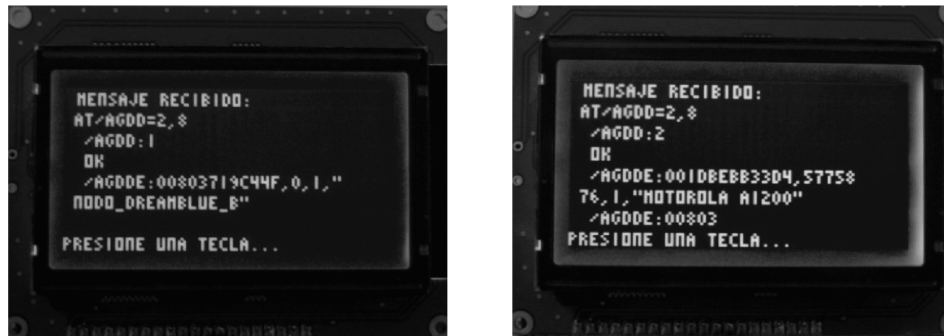


Figura 4.9. Sondeo de dispositivos.

³³ Ambos nombres se definieron en el escenario 6.



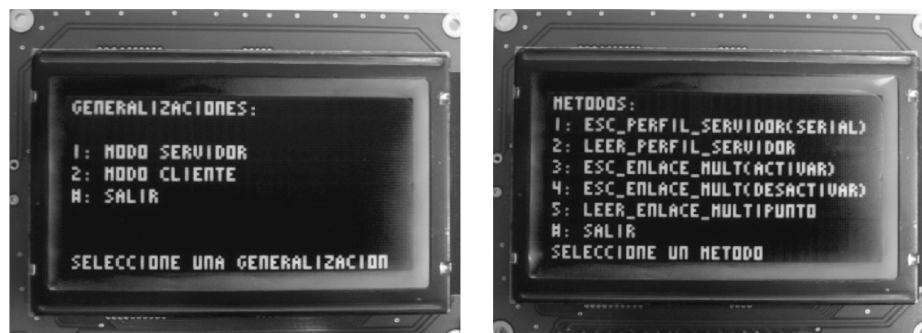
a) Mensaje recibido al sondeo.

b) Teléfono celular Motorola A1200 encontrado.

Figura 4.10. Prueba de descubrimiento de dispositivos Bluetooth.**Figura 4.11.** Sondeo ejecutado por el teléfono celular Motorola A1200.

Durante la prueba de sondeo y para cada dispositivo encontrado con el modo detectable activado se obtuvieron los parámetros: Número de dispositivos encontrados, Dirección Bluetooth, Clase de dispositivo, Índice de nombre válido y Nombre local (Figura 4.10). Además, se realizó una prueba de sondeo sobre un teléfono celular Motorola A1200 y se verificó la existencia de los nodos DreamBlue en su respectiva vecindad (Figura 4.11).

Para la prueba de configuración del enlace entre los nodos se estableció el perfil de puerto serie sobre ambos nodos DreamBlue en modo servidor (escenario 8) (Figura 4.12).



a) Subsistema Gestión de la Configuración.

b) Generalización modo servidor.

Figura 4.12. Configuración del enlace entre los nodos.



Figura 4.13. Generalización modo de seguridad.

Se implementó el menú de la Figura 4.13 para establecer el modo de seguridad en los nodos dentro del subsistema Servicios Suplementarios.

Para configurar el enlace entre los nodos DreamBlue con el soporte de seguridad, primero se activó el modo de emparejamiento sobre ambos nodos (escenario 14) (Figura 4.14). A continuación se asignó el código PIN utilizado en el proceso de autenticación (escenario 15) (Figura 4.15).

Una vez asignado el código PIN sobre ambos nodos DreamBlue, se estableció la unión entre éstos. Para establecer la unión se escribió la dirección Bluetooth del dispositivo a unir sobre cada nodo DreamBlue (escenario 17) (Figura 4.16).



a) Método modo emparejamiento.

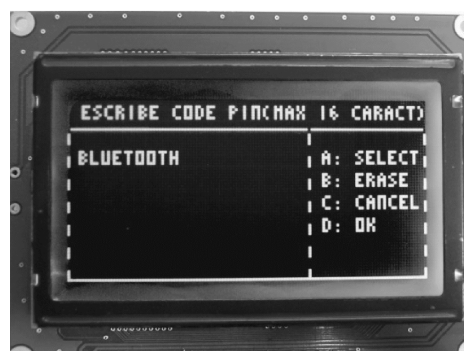


b) Activación del modo de emparejamiento.

Figura 4.14. Establecimiento del modo de emparejamiento.

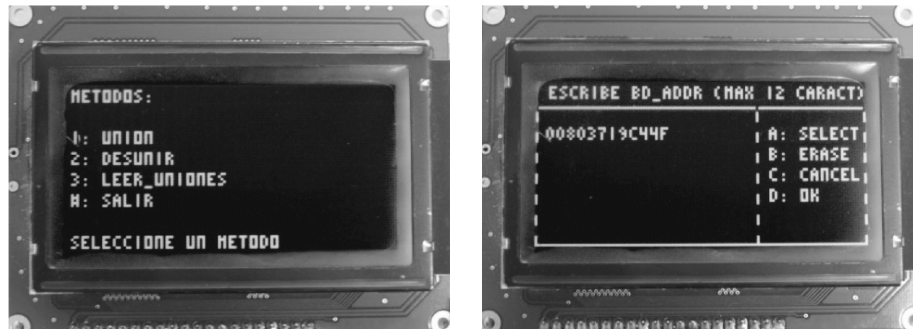


a) Método código PIN.



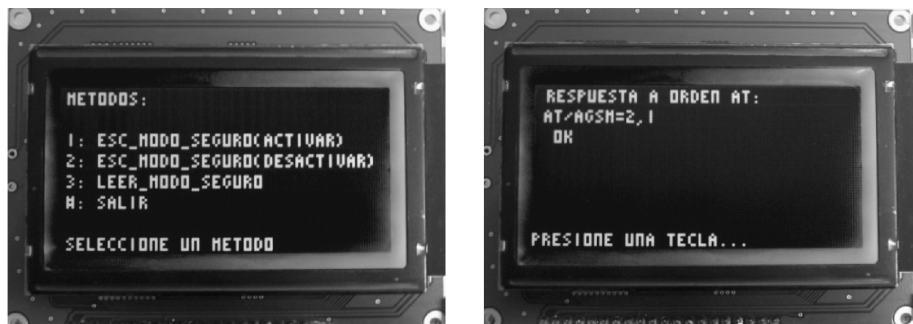
b) Escritura del código PIN.

Figura 4.15. Establecimiento del código PIN.



a) Método Unión.

b) Escritura de la dirección Bluetooth.

Figura 4.16. Establecimiento del proceso de unión.

a) Método modo seguro.

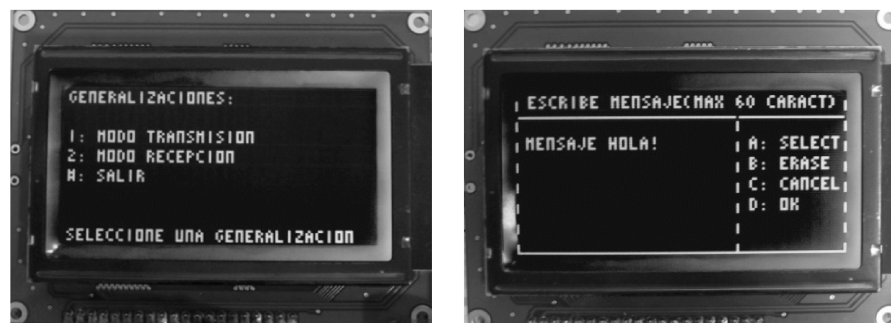
a) Respuesta de activación del modo seguro.

Figura 4.17. Establecimiento del modo seguro sobre los nodos.

Finalmente, se activó el modo seguro sobre ambos nodos DreamBlue (escenario 17) (Figura 4.17).

Una vez realizados los siguientes escenarios sobre cada nodo DreamBlue: Escenario 4 para activar el modo detectable, escenario 5 para activar el modo conectable, escenario 8 para establecer el perfil de puerto serie en modo servidor y escenario 18 para cambiar de modo de Órdenes AT al modo de operación de Datos; se realizó la prueba de envío de mensajes entre los nodos de la siguiente manera:

- Se ingresó al modo de transmisión sobre el nodo NODO_DREAMBLUE_A dentro del subsistema Gestión de Mensajes (escenario 19) (Figura 4.18).
- Se ingresó al modo de recepción sobre el nodo NODO_DREAMBLUE_B dentro del subsistema Gestión de Mensajes (escenario 20) (Figura 4.19).



a) Subsistema Gestión de Mensajes.

b) Escritura de mensaje de texto.

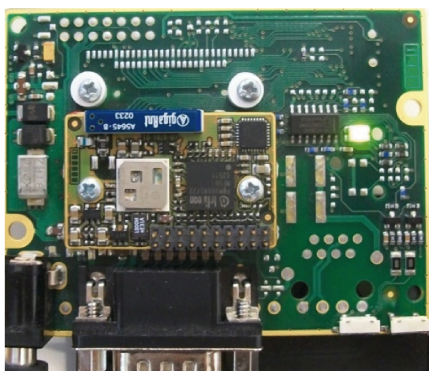
Figura 4.18. Escritura de un mensaje de texto.



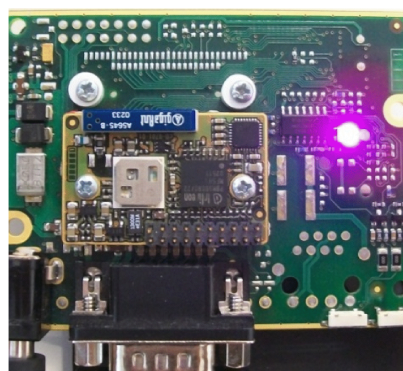
Figura 4.19. Lectura de un mensaje de texto.

Durante las pruebas descritas anteriormente se verificó en cada uno de los nodos DreamBlue el ingreso a los modos de operación y la ejecución del establecimiento del enlace mediante los LED RGB en cada nodo (Figura 4.20).

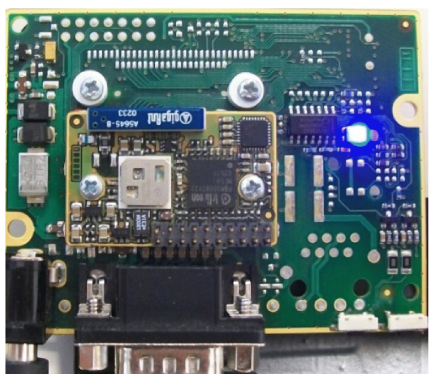
Para determinar la cobertura máxima del sistema DreamBlue se realizaron pruebas de alcance durante el establecimiento del enlace entre los nodos. Se observó que los nodos DreamBlue en espacio libre alcanzan un rango máximo de cobertura aproximado de 8 m; sin embargo, con paredes u otros objetos entre los dispositivos ya enlazados, el alcance máximo se reduce aproximadamente a 6 m. Si la conexión se pierde, se deben aproximar los nodos DreamBlue a una distancia no mayor a 3 m para asegurar que vuelvan a recuperar el enlace.



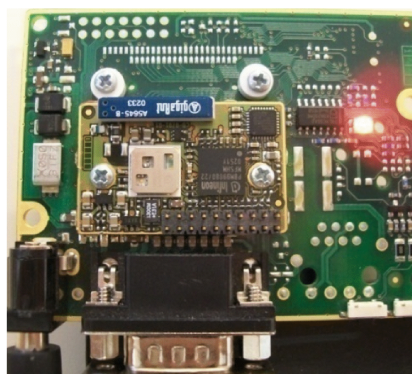
a) Modo de operación de Datos sin intento de establecimiento de enlace.



b) Modo de operación de Datos con intento de establecimiento de enlace.



c) Modo de operación de Datos con enlace establecido.



d) Modo de operación de Órdenes AT.

Figura 4.20. Estados de operación de los nodos DreamBlue.

4.2. Entrega y mantenimiento

La Tabla 4.1 lista las actualizaciones realizadas al sistema DreamBlue durante su ciclo de vida y que forman parte de la fase de entrega y mantenimiento de la metodología SPIES.

Tabla 4.1. Versiones de actualización del sistema DreamBlue.

Versión	Actualizaciones
0.1	Implementación del modo de operación de Órdenes AT y del modo de operación de Datos
0.2	Implementación del subsistema Configuración del Nodo
0.3	Implementación del subsistema Gestión de la Conexión
0.4	Implementación del subsistema Servicios Suplementario
0.5	Incorporación de métodos automáticos dentro de los subsistemas
0.6	Corrección de error de reinicio del sistema
0.7	Corrección de error en la Configuración RS232
0.8	Implementación del subsistema Gestión de Mensajes
0.9	Corrección de error en la captura de la orden AT*AMPM
1.0	Soporte de envío y recepción de mensajes de texto

La Tabla 4.2 lista los precios³⁴ correspondientes a las herramientas HW y SW en la elaboración de los nodos DreamBlue.

Tabla 4.2. Costos de las herramientas HW y SW para cada nodo DreamBlue³⁵.

HW	Precio (MX)	SW	Precio (MX)
dsPIC30F4013	\$ 130.00	Rhapsody (versión de evaluación)	\$ 0.00
Placa de pruebas	\$ 250.00	mikroC dsPIC (versión de evaluación)	\$ 0.00
Teclado matricial	\$ 99.00	GLCD Bitmap generator (versión de evaluación)	\$ 0.00
Pantalla GLCD	\$ 399.00	MASTER-PROG (incluido con el programador)	\$ 0.00
Programador MASTER-PROG	\$350.00	-	-
OEMSPA13i	\$ 645.19	-	-
Adaptador RS232 OEMII	\$ 1,654.34	-	-
Varios	\$ 100.00	-	-
SUBTOTAL	\$ 3527.53	SUBTOTAL	\$ 0.00
		TOTAL	\$ 3527.53

³⁴ Fecha de compra entre los meses de Febrero y Septiembre del año 2010

³⁵ La Tabla 4.2 no lista los precios de envío de las herramientas HW.

5. Conclusiones y trabajos futuros

El empleo de la metodología para sistemas empuotrados SPIES ha sido de gran ayuda en el diseño y desarrollo del sistema DreamBlue, sobre todo en el planteamiento de requerimientos iniciales, así como plantear los procedimientos para evaluar el correcto funcionamiento del sistema durante su ciclo de vida.

Con base en la bibliografía consultada, se logró comprender el funcionamiento del protocolo de comunicaciones Bluetooth. Como resultado se presenta el capítulo 2 del presente documento de tesis. Se ha implementado satisfactoriamente una picored inalámbrica Bluetooth entre los nodos programables.

El sistema DreamBlue puede considerarse un sistema abierto ya que cumple con los estándares del protocolo Bluetooth y es una herramienta destinada a la enseñanza que da pauta a nuevas aplicaciones en proyectos donde se requiera bajo consumo de energía. Los nodos del sistema DreamBlue pueden ser utilizados por alumnos y profesores que necesiten emplearlos para sus labores académicas y de investigación.

Los nodos DreamBlue obtenidos cumplen con el objetivo inicial de este proyecto de tesis, ya que son una herramienta de enseñanza para el protocolo de comunicaciones Bluetooth destinados a las siguientes materias: Comunicaciones Digitales, Redes de Computadoras I y Redes de computadoras II de la carrera de Ingeniería en Electrónica; Redes de Computadoras de la carrera de Ingeniería en Computación; y Sistemas de Comunicaciones I y Sistemas de Comunicaciones II de la carrera de Ingeniería en Mecatrónica.

Respecto a la metodología de desarrollo, queda pendiente la fase de mejora continua del sistema y se recomienda considerar las opiniones de los usuarios finales para integrarlas a la funcionalidad del sistema DreamBlue.

Con base en lo anterior se puede concluir que:

- Bluetooth ha comprobado su eficacia al conectar dispositivos en forma inalámbrica, a cortas distancias y bajo consumo de potencia; sin embargo, en sistemas donde el tiempo de respuesta sea un factor crítico la tecnología no es la ideal, debido a que presenta dos tipos de retardos asociados a la búsqueda de dispositivos en la vecindad y al establecimiento del enlace entre dispositivos Bluetooth.
- Los nodos DreamBlue construidos tienen un alcance limitado a 10 m, para ampliar el alcance, la especificación Bluetooth define un alcance máximo de 100 m para dispositi-

tivos con mayor potencia de transmisión; sin embargo, esta medida representa una mayor emisión de radiación y por consiguiente mayor consumo de corriente.

Se obtuvo un modelado con el objetivo de introducir diversas tecnologías inalámbricas que puedan emplearse en proyectos futuros en las carreras de Ingeniería en Electrónica, Ingeniería en Computación e Ingeniería en Mecatrónica. Se proponen los siguientes trabajos futuros:

- Diseñar e implementar los subsistemas: Gestión de Archivos y Gestión de Audio.
- Diseño y desarrollo de aplicaciones sobre teléfonos celulares para la decodificación de paquetes provenientes de los nodos DreamBlue.
- Emplear el modelado del sistema obtenido para su reutilización en el estudio y desarrollo de aplicaciones basadas en tecnologías inalámbricas como Wi-Fi o ZigBee.
- Desarrollo de una aplicación para el monitoreo de forma remota de variables eléctricas y mecánicas en sistemas de control.

Bibliografía

- [1] BGR. “*WLAN Interference to IEEE802.15.4*,” Zensys, Technical Report, 2007.
- [2] Bluetooth SIG. “Bluetooth Specification,” Bluetooth SIG Press, Version 4.0, 2009.
- [3] Bluetooth SIG. “*Bluetooth Technology Overview for Smart Meters and Home Area Network*,” Bluetooth SIG Press, 2010.
- [4] P. Bruce. “Real-Time UML Workshop for Embedded Systems,” Newnes, 2007.
- [5] C. Chamú. “Desarrollo de un Sistema Educativo para la Enseñanza del Protocolo de Comunicaciones CAN,” Tesis de Licenciatura, UTM, 2005.
- [6] CMMI Product Team. “*CMMI for Development (CMMI-DEV v1.2)*,” Software Engineering Institute, Carnegie Mellon University, 2006.
- [7] connectBlue AB. “*Bluetooth Serial Port Adapter AT Commands*,” connectBlue AB, Document Revision, 2010.
- [8] connectBlue AB. “*OEM Serial Port Adapter, Electrical & Mechanical Data Sheet*,” connectBlue AB, Data Sheet, 2003.
- [9] F. Eady. “Hands-On ZigBee: Implementing 802.15.4 with Microcontrollers,” Newnes, 2008.
- [10] J. Ganssle. “The Art of Designing Embedded Systems,” Newnes, Second Edition, 2008
- [11] J. Ganssle, T. Noergaard, F. Eady, L. Edwards. “Embedded Hardware,” Newnes, 2008.
- [12] I. Garcia, A. Herrera. “*Using the Software Process Improvement approach for Defining a Methodology for Embedded Systems Development using the CMMI-DEV v1.2*,” cit, pp.233-240, 2010 10th IEEE International Conference on Computer and Information Technology, 2010.
- [13] M. Gartrell, J. Kelly, S. Razgulin. “*MobileLeech: Sharing Internet Data Access on Mobile Devices using Bluetooth*,” Department of Computer Science, University of Colorado, Technical Report, 2008.
- [14] J. Hou, B. Chang, D. Cho, M. Gerla. “*Minimizing 802.11 Interference on Zigbee Medical Sensors*,” Department of Computer Science University of California, Los Angeles. Technical Report, 2009.

- [15] S. Huang, L. Rudolph. "Bluetooth Essentials for Programmers," Cambridge University Press, 2007.
- [16] H. Labiod, H. Afifi, C. De Santis. "Wi-Fi, Bluetooth, ZigBee and Wimax". Springer, 2007.
- [17] U. Lee, S. Jung, D. Cho, A. Chang, J. Choi, M. Gerla. "Bluetooth-based P2P Content Distribution to Mobile Users," Department of Computer Science University of California, Los Angeles, Technical Report, 2008.
- [18] E. Mendoza. "Implementación de un sistema de captura en redes 802.11 y BT," Tesis de Licenciatura, UTM, 2005.
- [19] Microchip. "dsPIC30F3014/4013 Data High-Performance, 16-Bit Digital Signal Controllers," Microchip Technology Inc., Data Sheet, 2008.
- [20] S. Mukhopadhyay, M. Huang. "Sensors Advancements in Modeling, Design Issues, Fabrication and Practical Applications," Springer, 2008.
- [21] R. Rojas. "Comunicación inalámbrica de sistemas IEEE 488 mediante una picored Bluetooth," Tesis de Licenciatura, UTM, 2005.
- [22] E. Rojas, C. Morales. "Monitor cardiaco móvil en teléfonos celulares para la detección de anomalías cardiacas," cit, pp.217-225, V Semana Nacional de Ingeniería Electrónica, Ocotlán Jalisco, 2009.
- [23] M. Samek. "Practical UML Statechart," Newnes, 2009.
- [24] F. Shahin. "ZigBee Wireless Networks and Transceivers," Newnes, 2008.
- [25] J. Singh, D. Picopoulos, C. Todd. "Evaluating the Adaptative Frequency Hopping Mechanism to Enable Bluetooth-WLAN Coexistence," University College London, Technical Report, 2003.
- [26] SURE Electronics Inc. "128x64 Graphic LCD Module Brief Introduction User's Guide," SURE Electronics Inc., Data Sheet, 2008.
- [27] R. Verdone, D. Dardari. G. Mazzini, A. Conti. "Wireless Sensors and Actuator Networks," AP, 2007.
- [28] W. Xiaofeng, Z. Qingqing, F. Mengfei. "SMS Receiving and Dispatching System Based on Embedded Linux," cccm, vol. 2, pp.475-479, 2008 ISECS International Colloquium on Computing, Communication, Control, and Management, 2008.

Sitios URL

- [URL1] <http://www.bluetooth.com/> "Página electrónica oficial sobre la tecnología inalámbrica Bluetooth," Noviembre 2010.
- [URL2] <http://www.connectblue.se/> "Página electrónica de la empresa connectBlue AB," Noviembre 2010.
- [URL3] <http://www.ibm.com/developerworks/downloads/r/rhapsodydeveloper/> "Página electrónica sobre la herramienta Rhapsody de la empresa IBM," Septiembre 2010.
- [URL4] <http://www.lammertbies.nl/comm/info/hayes-at-commands.html>. "Página electrónica sobre el origen de las órdenes Hayes AT," Marzo 2010.
- [URL5] <http://www.microchip.com/> "Página electrónica de la empresa Microchip Technology Incorporated," Noviembre 2010.

[URL6] <http://www.mikroe.com/> “Página electrónica de la empresa MikroElektronika,” Octubre 2010.

[URL7] <http://www.multicomp.com/> “Página electrónica de la empresa Multicomp,” Noviembre 2010.

A. Anexo DTE

En este Anexo se describen las características eléctricas y mecánicas de los componentes HW más importantes (pantalla GLCD, teclado matricial y dsPIC30F4013), que forman el módulo DTE.

A.1. Pantalla GLCD

La pantalla GLCD de la firma SURE electronics se compone de 20 terminales para administrar señales de control y datos (Figura A.1).

En la Tabla A.1 se muestra la asignación para cada terminal de la pantalla GLCD.

Tabla A.1. Asignación de terminales para la pantalla GLCD [26].

Terminal	Símbolo	Descripción
1	VSS	Masa
2	VDD	Alimentación
3	V0	Ajuste de contraste
4	RS(CS)	Instrucciones
5	RW(SID)	Escritura Lectura
6	E/(SCLK)	Señal de habilitación
7-14	DB0-DB7	Bus de datos
15	CH1	Selección de chip 1
16	CH2	Selección de chip 2
17	RESET	Sistema de restauración
18	VEE	Voltaje para el controlador LCD
19	LED_A	Luz de fondo Anodo
20	LED_K	Luz de fondo Catodo

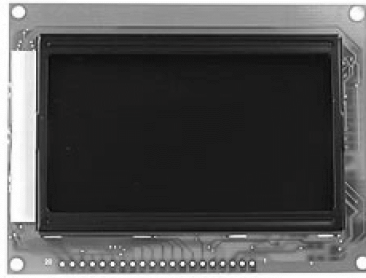


Figura A.1. Pantalla GLCD.

La Tabla A.2 lista las características eléctricas para la pantalla GLCD.

Tabla A.2. Características eléctricas de la pantalla GLCD [26].

Parámetro	Símbolo	Condición	Mín	Típico	Máx	Unidad
Voltaje de operación general	VDD	-	4.5	5.0	5.5	V
Corriente de operación general	IDD	-	-	2.02	4	mA
Voltaje de operación para el LCD	VDD V0	0°C	8.4	8.8	9.2	V
		25°C	8.1	8.5	8.9	V
		50°C	7.4	7.8	8.2	V
Voltaje de operación para la luz lateral	VF	-	-	4.2	4.6	V
Corriente de operación para la luz lateral	IF	VF=4.2	-	180	300	mA
Voltaje de entrada nivel 'H'	VIH	-	0.7VDD	-	VDD	V
Voltaje de entrada nivel 'L'	VIL	-	0	-	0.3VDD	V

La Figura A.2 muestra las dimensiones externas de la pantalla GLCD.

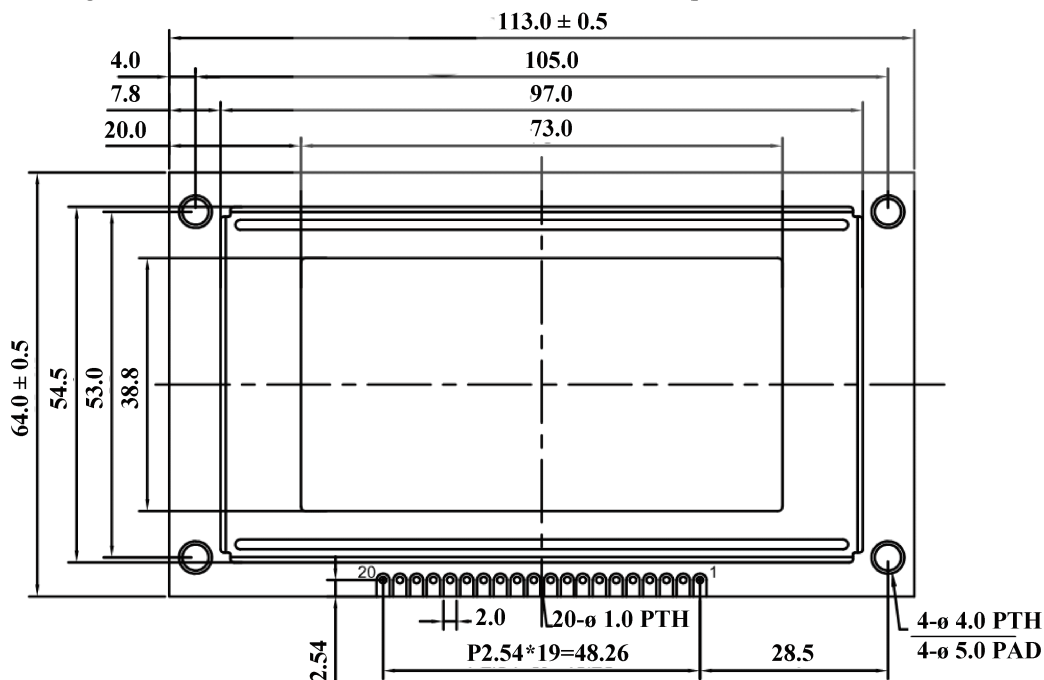


Figura A.2. Dimensiones externas de la pantalla GLCD [26].

La Tabla A.3 contiene las características mecánicas de la pantalla GLCD.

Tabla A.3. Características mecánicas de la pantalla GLCD.

Parámetro	Valor estándar	Unidad
Dimensión del módulo	113x65	mm
Vista de área	73x38.8	mm
Agujero de montaje	105x53	mm
Distancia entre puntos	0.52x0.52	mm

A.2. Teclado matricial

El teclado matricial de 4x4 interruptores de la firma microcomp tiene las siguientes especificaciones eléctricas (Figura A.3):

- Conectores estandarizados: 20 mA, 24 VCD.
- Resistencias de contacto: 200 Ω máximo.
- Vida: 1,000,000 ciclos por tecla.
- Temperatura de operación: -20°C a +60°C.
- Temperatura de almacenamiento: -45°C a +60°C.

La Tabla A.4 contiene la asignación de terminales y de símbolos para cada terminal del teclado matricial.

Tabla A.4. Asignación de terminales para teclado matricial [URL7].

Terminal	Símbolo	Descripción
1	ROW 2	Fila 2 del teclado
2	ROW 3	Fila 3 del teclado
3	COL 1	Columna 1 del teclado
4	ROW 4	Fila 4 del teclado
5	COL 2	Columna 2 del teclado
6	COL 3	Columna 3 del teclado
7	COL 4	Columna 4 del teclado
8	ROW 1	Fila 1 del teclado



Figura A.3. Teclado matricial de 4x4 interruptores.

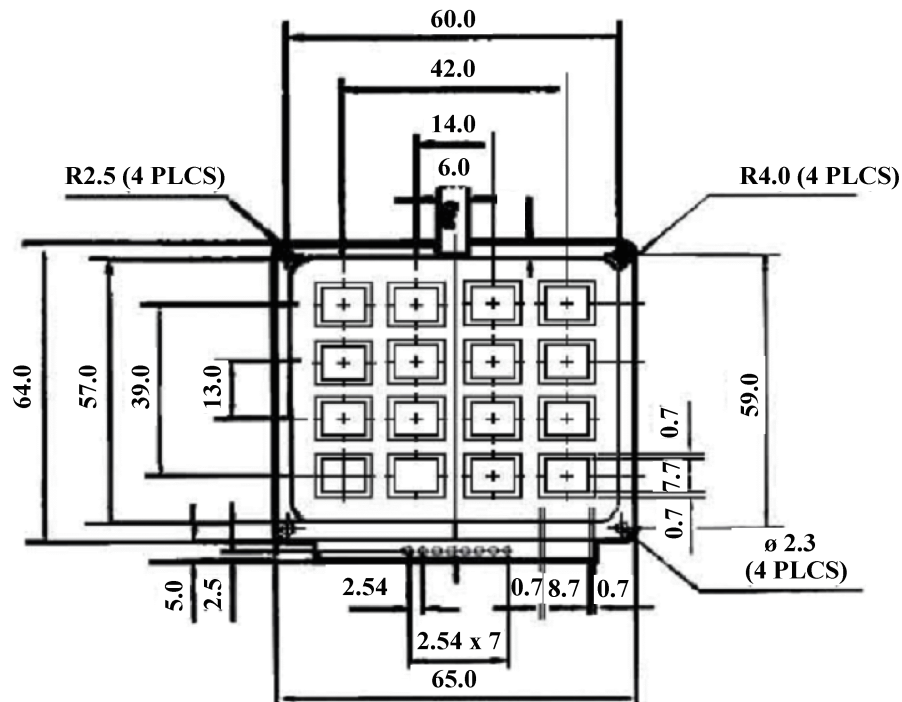


Figura A.4. Dimensiones externas del teclado matricial [URL7].

La Figura A.4 muestra las dimensiones externas para el teclado matricial de 4x4 interruptores proporcionado por la firma microcomp. La Tabla A.5 contiene las características mecánicas el teclado matricial.

Tabla A.5. Características mecánicas del teclado matricial [URL7].

Parámetro	Valor estándar	Unidad
Dimensión del módulo	65x64	mm
Vista de área	73x38.8	mm
Agujero de montaje	65x57	mm
Distancia entre interruptores	0.7	mm
Altura	8.9	mm

A.3. MCU dsPIC30F4013

Para la administración de los subsistemas y métodos de los nodos, se utilizó el MCU dsPIC30F4013 de la firma Microchip (Figura A.5) [URL5].

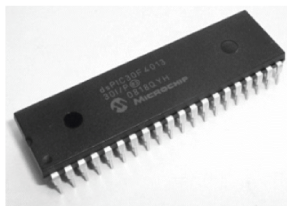


Figura A.5. MCU dsPIC30F4013 [19].

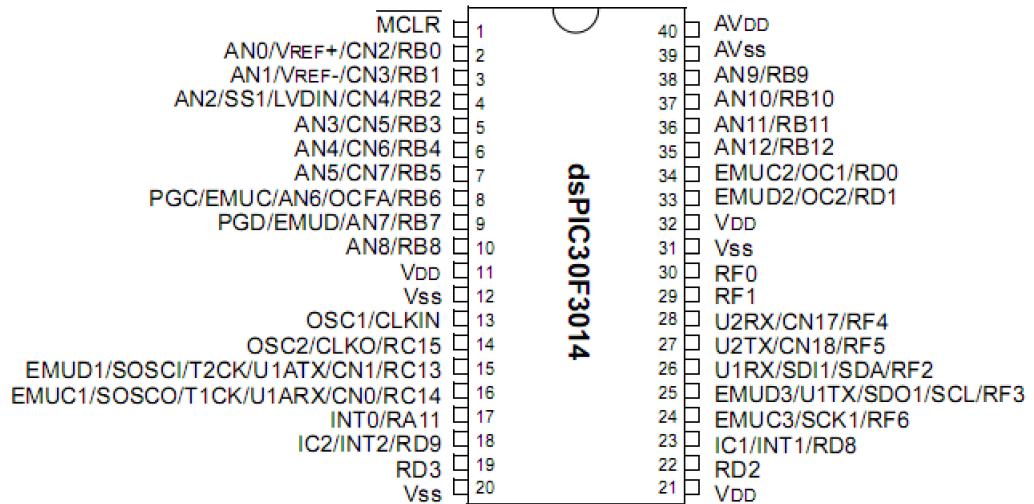


Figura A.6. Distribución de terminales para el MCU dsPIC30F4013 [19].

La Figura A.6 muestra la distribución de terminales para el MCU dsPIC30F4013 para un encapsulado tipo DIP [19].

La Tabla A.6 muestra las principales características del MCU dsPIC30F4013 empleado en la gestión de las funciones de los nodos [19].

Tabla A.6. Principales características del dsPIC30F4013.

Parámetro	Información
Memoria	Memoria de programa de 48 KB de capacidad.
	Hasta 16 K instrucciones.
	2048 bytes de memoria SRAM.
	1024 de memoria EEPROM.
Periféricos	16 registros de trabajo de 16 bits cada uno.
	5 temporizadores de 16 bits.
	4 módulos de captura de 16 bits.
	4 módulos comparadores o de salida PWM de 16 bits.
	Interfaz de conexión de datos DCI con soporte de CODEC de audio AC'97 e I ² S.
	2 Módulos UART.
	1 módulo SPI.
	1 módulo CAN.
1 módulo I ² C	
	Convertidor analógico digital de 12 bits.

La Tabla A.7 proporciona la función de las terminales utilizadas del MCU dsPIC30F4013 en la implementación de los nodos.

Tabla A.7. Asignación de terminales utilizadas en el MCU dsPIC30F4013.

Terminal	Símbolo	Función
1	/MCLR	Voltaje de programación de entrada. (<i>Master Clear</i>) y restauración del MCU
2	RB0	Línea 1 de datos para la pantalla GLCD y fila 2 del teclado matricial
3	RB1	Línea 2 de datos para la pantalla GLCD y columna 1 del teclado matricial
4	RB2	Línea 3 de datos para la pantalla GLCD y fila 3 del teclado matricial
5	RB3	Línea 4 de datos para la pantalla GLCD y columna 2 del teclado matricial
6	RB4	Línea 5 de datos para la pantalla GLCD y columna 3 del teclado matricial
7	RB5	Línea 6 de datos para la pantalla GLCD y fila 3 del teclado matricial
8	RB6	Línea 7 de datos para la pantalla GLCD y fila 4 del teclado matricial
9	RB7	Línea 8 de datos para la pantalla GLCD y fila 1 del teclado matricial
11	VDD	Alimentación del MCU
12	VSS	Masa
13	OSC1	Entrada 1 para oscilador de cristal
14	OSC2	Entrada 2 para oscilador de cristal
19	RD3	Orden para instrucción o datos para la pantalla GLCD
22	RD2	Habilitación para la pantalla GLCD
27	U2TX	Terminal de transmisión Tx de la UART 2 del MCU
28	U2RX	Terminal de recepción Rx de la UART 2 del MCU
33	RD1	Selección de chip 1
34	RD2	Selección de chip 2
39	AVSS	Restauración para la pantalla GLCD

B. Anexo DCE

En este Anexo se describen las características eléctricas y mecánicas del adaptador de RS232 OEMII y del dispositivo OEMSPA13i.

B.1. Adaptador de RS232 OEMII

El adaptador de RS232 OEMII consiste de:

- Un conector para acoplar los dispositivos Bluetooth de la firma connectBlue.
- Un conector DB9 macho para comunicar los productos de la firma connectBlue serie OEM, con dispositivos externos vía RS232.
- Un conector de alimentación de 2.1 mm.
- Un interruptor de restauración.
- Un interruptor de funciones.
- Un LED RGB para indicar el estado de los productos de la firma connectBlue serie OEM.

La Tabla B.1 muestra la asignación de terminales del conector DB9 del adaptador de RS232 OEM II.

Tabla B.1. Terminales del conector DB9 del módulo adaptador de RS232 OEM II [8].

Terminal	Nombre	Descripción
1	NC	No conectado (<i>Not Conenected</i>)
2	RD	Recepción de datos (<i>Receive Data</i>)
3	TD	Transmisión de datos (<i>Transmit Data</i>)
4	DTR	Terminal de datos lista (<i>Data Terminal Ready</i>)
5	GND	Tierra física (<i>Ground</i>)
6	DSR	Conjunto de datos listo (<i>Data Set Ready</i>)
7	RTS	Petición de envío (<i>Request To Send</i>)
8	CTS	Listo para enviar (<i>Clear To Send</i>)
9	NC	No conectado (<i>Not Conenected</i>)

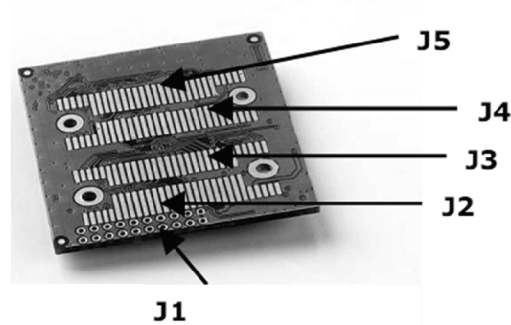


Figura B.1 Conectores J1, J2 y J3 del dispositivo OEMSPA13i ³⁶[8].

La Tabla B.2 contiene las características mecánicas del adaptador RS232 OEMII.

Tabla B.2. Características mecánicas del adaptador RS232 OEMII [8].

Parámetro	Valor	Unidad
Dimensión del módulo	75x58	mm
Área de ensamble	2x30	mm
Agujero de montaje	65x57	mm
Altura	8.9	mm

B.2. Dispositivo OEMSPA13i

El dispositivo OEMSPA13i empleado en el presente trabajo de tesis posee dos maneras para comunicarse con dispositivos externos [URL2]:

- Mediante el conector J1.
- Mediante las terminales de montaje superficial J2 y J3 (Figura B.1).

La Tabla B.3 describe la asignación y características para cada terminal de los conectores J1 y J2 del adaptador de RS232 OEMII [8].

Tabla B.3. Asignación de las terminales para los conectores J1 y J2 [8].

Terminal de J1 y J2	Símbolo	Nombre	Nivel	Tipo	Descripción
1, 2.	VSS	Tierra (<i>Ground</i>)	0V	-	Terminales conectados internamente
3, 4	VCC_5V	Alimentación (<i>Power</i>)	3.3V–6V	-	Terminales conectados internamente
5	RS232-CTS	Listo para enviar (<i>Clear To Send</i>)	RS232	Entrada	Control de flujo HW
6	RS232-TxD	Transmisión de datos (<i>Transmit Data</i>)	RS232	Salida	-
7	RS232-RTS	Peticion de envio (<i>Request To Send</i>)	RS232	Salida	Control de flujo HW

³⁶ El dispositivo OEMSPA13i no cuenta con los conectores J4 y J5.

8	RS232-RxD	Recepción de datos (<i>Receive Data</i>)	RS232	Entrada	-
9	RS232-DTR	Terminal de datos lista (<i>Data Terminal Ready</i>)	RS232	Salida	-
10	RS232-DSR	Conjunto de datos listo (<i>Data Set Ready</i>)	RS232	Entrada	También se puede usar para controlar el modo de ahorro de energía
11	RED/Mode	LED en color rojo (salida) y entrada en modo interfaz serie	CMOS	E/S	Señal multiplexada: RED: no válida hasta 500 ms después de encendido. Modo: durante el encendido selecciona un nivel lógico RS232 (alto) o UART (bajo).
12	Switch-0	Interruptor de función	CMOS	Entrada	Usado para conexiones externas
13	GREEN/ Switch-1	LED en color verde (salida) e interruptor de realmacenamiento (Restore)	CMOS	E/S	Señal multiplexada: GREEN: no válida hasta 500 ms después de encendido. Switch-1: si el nivel es bajo, las unidades regresan a un estado serie preestablecido
14	BLUE	LED en color azul (salida)	CMOS	Salida	-
15	UART1-CTS	Listo para enviar (<i>Clear To Send</i>)	CMOS	Entrada	Control de flujo HW
16	UART1-TxD	Transmisión de datos (<i>Transmit Data</i>)	CMOS	Salida	-
17	UART1-RTS	Petición de envío (<i>Request To Send</i>)	CMOS	Salida	Control de flujo HW
18	UART1-RxDI	Recepción de datos (<i>Receive Data</i>)	CMOS	Entrada	-
19	UART1-DTR	Terminal de datos lista (<i>Data Terminal Ready</i>)	CMOS	Salida	-
20	UART1-DSR	Conjunto de datos listo (<i>Data Set Ready</i>)	CMOS	Entrada	Listo para recibir o transmitir datos.

La Figura B.2 muestra el diagrama a bloques de los componentes del dispositivo OEMS-PA13i.

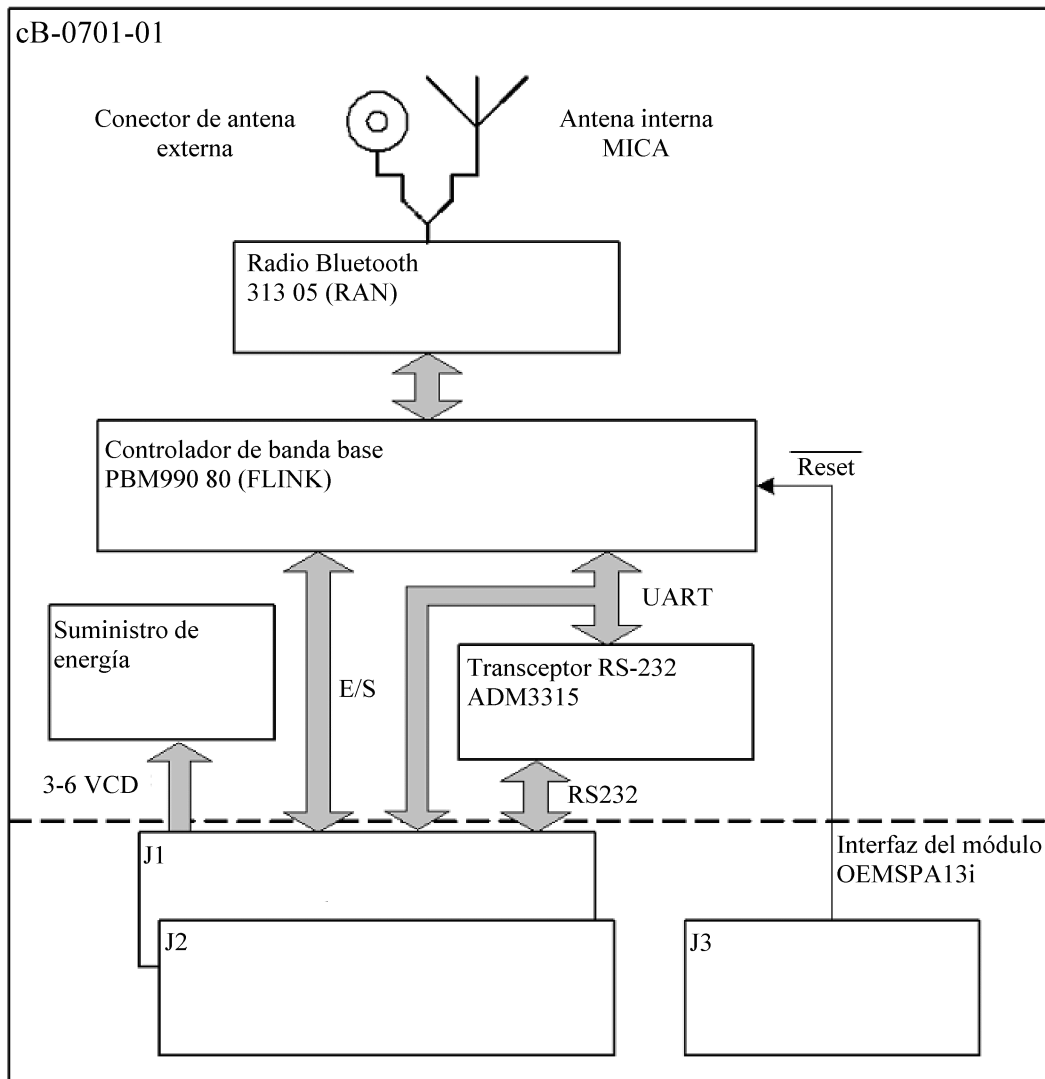


Figura B.2. Componentes del dispositivo OEMSPA13i [8].

Las características principales de la antena interna, integrada en el dispositivo OEMSPA13i se muestran en la Tabla B.4.

Tabla B.4. Características de la antena interna del dispositivo OEMSPA13i.

Parámetro	Información
Nombre	Mica 2.4 GHz
Fabricante	gigaAnt
Tipo Antena	parche (<i>patch</i>)
Polarización	Lineal
Ganancia	+2.7 dBi
Dimensiones (LxAnxAI)	20.5 x 3.6 x 3.35 mm
Conector	Montaje superficial

El dispositivo OEMSPA13i tiene una fuente de alimentación lineal, la cual hace que la corriente sea constante si el voltaje de alimentación cambia.

El consumo de corriente depende del estado de operación, de la velocidad de transferencia de datos y de la carga en las terminales de la interfaz RS232. La Tabla B.5 muestra el consumo de corriente para el dispositivo OEMSPA13i.

Tabla B.5. Consumo de corriente del dispositivo OEMSPA13i [8].

Parámetro	Modo		Valores		Unidad	
			≤ 57.6	> 57.6		
Velocidad de transferencia ³⁷	-				kBits/s	
I_{CC}	No conectado	Promedio	25	60	mA	
		Pico	85	120	mA	
	No conectado, modo de paro habilitado	No conectado	Promedio	0.6	0.7	mA
		Conectado	Promedio	1.1	1.5	mA
	Conectado	Conectado	Promedio	2.3	2.5	mA
		Detenido (<i>Idle</i>)	Promedio	35	70	mA
	Conectado		Pico	100	150	mA
		Transmitiendo	Promedio	50	85	mA
	Pico		100	150	mA	
	$I_{RS232 \text{ Salida}}$ ³⁸	RS232	Promedio	7	7	mA
$R_{RS232 \text{ Carga}}$ ³⁹	RS232	Típico	20	20	K Ω	

³⁷ La frecuencia del reloj interno se incrementa cuando la velocidad de transferencia es mayor a 57.6 Kbps.

³⁸ Las señales RS232 TxD, RTS y DTR con una carga de 3 K Ω , no incluidas en I_{CC} , son dependientes de la carga.

³⁹ Carga interna en cada terminal de entrada RS232.

C. Anexo órdenes AT

En este anexo se describe el uso de las órdenes AT⁴⁰ utilizadas en el sistema DreamBlue [9].

C.1. Configuración del Nodo

En esta sección se describen las órdenes AT empleadas por los métodos del subsistema Configuración del Nodo (Sección 3.2.3.1.1).

Tabla C.1. Método Escribir_Configuracion_RS232 (AT*ARMS=).

Orden AT	Método	Descripción
AT*ARMS=<baud_rate>, <No_bits>, <bits_paro>,<paridad>, <control_flujo>, <cambio_despues_OK>, <guardar><CR>	Escribir_Configuracion_RS232	Aplica una nueva configuración RS232 al nodo
Parámetros solicitados	Tipo	Descripción
<baud_rate>	Enumerador	Valores estándares: 1: 300 2: 1 200 3: 2 400 4: 4 800 5: 9 600 6: 19 200 7: 38 400 8: 57 600 (por defecto) 9: 115 200
<bits_datos>	Enumerador	1: 8 bits (por defecto) 2: 7 bits

⁴⁰ Las siguientes tablas hacen referencia a las órdenes AT utilizadas especificadas en el documento [7].

		3: 6 bits 4: 5 bits
<bits_paro>	Enumerador	1: 1 bits (por defecto) 2: 2 bits
<paridad>	Enumerador	1: Sin paridad (por defecto) 2: Paridad par 3: Paridad impar
<control_flujo>	Enumerador	1: CTS/RTS usado para control de flujo (por defecto) 2: CTS/RTS no usado
<cambio_despues_OK>	Enumerador	1: La configuración RS232 del nodo no cambia hasta el siguiente ciclo de alimentación. 2: La configuración RS232 del nodo no cambia después de la respuesta de confirmación "OK"
<guardar>	Enumerador	1: La configuración del modo conectable solo permanece durante el periodo de alimentación actual del dispositivo OEMSPA13i 2: La configuración del modo conectable permanece durante todos los periodos de alimentación del dispositivo OEMSPA13i
	Respuesta	Descripción
	<CR><LF>OK<CR><LF>	Respuesta de éxito
	<CR><LF>ERROR<CR><LF>	Respuesta de error

Tabla C.2. Método Lectura_Configuracion_RS232 (AT*ARMS?).

Orden AT	Método	Descripción
AT*ARMS?<CR>	Leer_Configuracion_RS232	Lee la configuración RS232 actual sobre el nodo
	Respuesta	Descripción
	<CR><LF>AT*ARMS=<baud_rate>,<No_bits>,<bits_paro>,<paridad>,<control_flujo><CR><LF>OK<CR><LF>	Respuesta de éxito
	<CR><LF>ERROR<CR><LF>	Respuesta de error

Tabla C.3. Método Escritura_Modo_Conectable (AT*AGCM=).

Orden AT	Método	Descripción
AT*AGCM=<modo_conectividad>,<guardar><CR>	Escritura_Modo_Conectable	Escribe el modo de conectividad sobre el nodo
Parámetros solicitados	Tipo	Descripción

<modo_conectividad>	Enumerador	1: Modo conectable desactivado 2: Modo conectable activado (por defecto)
<guardar>	Enumerador	Véase Tabla C.1
Respuesta		Descripción
Véase Tabla C.1		-

Tabla C.4. Método Lectura_Modo_Conectable (AT*AGCM?).

Orden AT	Método	Descripción
AT*AGCM?<CR>	Lectura_Modo_Conectable	Lee el modo de conectividad actual sobre el nodo
Respuesta		Descripción
<CR><LF>*AGCM:<modo_conectividad><CR><LF>OK<CR><LF>		Respuesta de éxito
<CR><LF>ERROR<CR><LF>		Respuesta de error

Tabla C.5. Método Escritura_Modo_Detectable (AT*AGDM=).

Orden AT	Método	Descripción
AT*AGDM=<modo_detectable>,<guardar><CR>	Escritura_Modo_Detectable	Esta orden escribe el modo detectable sobre el nodo
Parámetros solicitados		Tipo
<modo_detectable>	Enumerador	1: Modo detectable desactivado 2: Modo detectable limitado activado 3: Modo detectable general activado (por defecto)
<guardar>	Enumerador	Véase Tabla C.1
Respuesta		Descripción
Véase Tabla C.1		-

Tabla C.6. Método Lectura_Modo_Detectable (AT*AGDM?).

Orden AT	Método	Descripción
AT*AGDM?<CR>	Lectura_Modo_Detectable	Lee el modo detectable actual sobre el nodo
Respuesta		Descripción
<CR><LF>*AGDM:<modo_detectable><CR><LF>OK<CR><LF>		Respuesta de éxito
<CR><LF>ERROR<CR><LF>		Respuesta de error

Tabla C.7. Método Escritura_Nodo_Local (AT*AGLN=).

Orden AT	Método	Descripción
----------	--------	-------------

AT*AGLN=<nombre_dispositivo>,<guardar><CR>	Escritura_Nombre_Local	Escribe el nombre local del nodo
Parámetros solicitados	Tipo	Descripción
<nombre_dispositivo>	Cadena	Nombre local actual del nodo (máximo 16 caracteres)
<guardar>	Enumerador	Véase Tabla C.1
Respuesta		Descripción
Véase Tabla C.1		-

Tabla C.8. Método Leer_Nombre_Local (AT*AGLN?).

Orden AT	Método	Descripción
AT*AGLN?<CR>	Leer_Nombre_Local	Lee el nombre local actual del nodo
Respuesta		Descripción
<CR><LF>*AGLN:<nombre_dispositivo><CR><LF>OK<CR><LF>		Respuesta de éxito
<CR><LF>ERROR<CR><LF>		Respuesta de error

Tabla C.9. Método Escritura_COD_Local (AT*AGLC=).

Orden AT	Método	Descripción
AT*AGLC=<cod>,<guardar><CR>	Escritura_COD_Local	Escribe la clase de dispositivo sobre el nodo
Parámetros solicitados	Tipo	Descripción
<cod>	Entero	<p>El parámetro ha sido dividido en tres segmentos: Clase de servicio, Dispositivo de clase mayor y Dispositivo de clase menor</p> <p>Clase de servicio (bits 13-23):</p> <p>Bit 16: Posicionamiento (identificación de locación)</p> <p>Bit 17: Red (LAN, Ad hoc, etc.)</p> <p>Bit 18: (Impresión, bocinas, etc.)</p> <p>Bit 19: Captura (escáner, micrófono, etc.)</p> <p>Bit 20: Transferencia de objetos (v-folder, v-inbox, etc.)</p> <p>Bit 21: Audio (bocina, micrófono, servicio manos libres, etc.)</p> <p>Bit 22: Telefonía (teléfono inalámbrico, manos libres, modem, etc.)</p> <p>Bit 23: Información (servidor WEB, servidor WAP, etc.)</p> <p>Dispositivos de clase mayor (bits 8-12):</p> <p>00000: Miscelánea</p>

		00001: Computadora (escritorio, portátil, PDA, etc.) 00010: Teléfono (celular, alámbrico, modem, etc.) 00011: Punto de acceso LAN 00100: Audio/Video (bocina, estéreo, video, VCR, etc.) 00101: Periférico (ratón, teclado, etc.) 00110: Imagen (impresora, escáner, cámara, etc.) 11111: Sin categoría
<guardar>	Enumerador	Véase Tabla C.1
	Respuesta	Descripción
Véase Tabla C.1		-

Tabla C.10. Método Leer_COD_Local (AT*AGLC?).

Orden AT	Método	Descripción
AT*AGLC?<CR>	Leer_COD_Local	Lee la clase de dispositivo actual del nodo
	Respuesta	Descripción
	<CR><LF>*AGLC:<cod><CR><LF>OK<CR><LF>	Respuesta de éxito
	<CR><LF>ERROR<CR><LF>	Respuesta de error

Tabla C.11. Método Leer_BD_ADDR (AT*AILBA?).

Orden AT	Método	Descripción
AT*AILBA?<CR>	Leer_BD_ADDR	Lee la dirección Bluetooth del nodo
	Respuesta	Descripción
	<CR><LF>*AILBA:<bd_addr><CR><LF>OK<CR><LF>	Respuesta de éxito
	<CR><LF>ERROR<CR><LF>	Respuesta de error
	Parámetros obtenidos	Tipo
	<bd_addr>	Bd_Addr
		Valor
		Dirección Bluetooth local del nodo

Tabla C.12. Método Restauracion_Nodo (AT&F).

Orden AT	Método	Descripción
AT&F<CR>	Restauracion_Nodo	Restaura todas las configuraciones a sus valores de fábrica, si se han almacenado configuraciones en la memoria del dispositivo OEMSPA13i
	Respuesta	Descripción
Véase Tabla C.1		-

C.2. Gestión de la Conexión

A continuación se describen las órdenes AT empleadas por los métodos utilizados por el subsistema Gestión de la Conexión (Sección 3.2.3.1.2).

Tabla C.13. Método Escribir_Perfil_Servidor (AT*ADDSP=).

Orden AT	Método	Descripción
AT*ADDSP=<rol_y_perfil>,<guardar><CR>	Escribir_Perfil_Servidor	Escribe el perfil del nodo en modo servidor y se activa cuando el nodo cambia a modo de operación de Datos
Parámetros solicitados	Tipo	Valor
<rol_y_perfil>	Enumerador	0: Rol Dispositivo B (DevB), en perfil de puerto serie (por defecto) 1: Rol de pasarela, perfil de red dial-up. 2: Rol de punto de acceso, perfil de acceso LAN 255: Sin perfil
<guardar>	Enumerador	Véase Tabla C.1
Respuesta	Descripción	
Véase Tabla C.1	-	

Tabla C.14. Método Escribir_Enlace_Multipunto (AT*ADWM=).

Orden AT	Método	Descripción
AT*ADWM=<habilitar>,<auto_seguimiento>,<guardar><CR>	Escribir_Enlace_Multipunto	Escribe el modo de enlace multipunto sobre el nodo
Parámetros solicitados	Tipo	Valor
<activar>	Enumerador	0: Enlace multipunto desactivado (valor por defecto) 1: Enlace multipunto activado
<auto_seguimiento>	Enumerador	0: Los datos recibidos de un dispositivo conectado al nodo, serán dirigidos solamente al huésped (por defecto) 1: Valor destinado para uso futuro
<guardar>	Enumerador	Véase Tabla C.1
Respuesta	Descripción	
Véase Tabla C.1	-	

Tabla C.15. Método Leer_Enlace_Multipunto (AT*ADWM?).

Orden AT	Método	Descripción
AT*ADWM?<CR>	Leer_Enlace_Multipunto	Lee el modo de enlace multipunto que existe sobre el nodo
Respuesta		Descripción
<CR><LF>*ADWM:<habilitar>,<auto_seguimiento>,<guardar><CR><LF><OK><CR><LF>		Respuesta de éxito
<CR><LF>ERROR<CR><LF>		Respuesta de error

Tabla C.16. Método Escribir_RolMaestroEsclavo (AT*AGMSP=).

Orden AT	Método	Descripción
AT*AGMSP=<rol>,<guardar><CR>	Escribir_RolMaestroEsclavo	Permite que el nodo actúe como nodo Maestro o como nodo Esclavo
Parámetros solicitados	Tipo	Valor
<rol>	Enumerador	0: Siempre el dispositivo a conectarse se convierte en nodo Maestro 1: Siempre permite al dispositivo a conectarse, seleccionar el rol entre nodo Maestro o nodo Esclavo (por defecto)
<guardar>	Enumerador	Véase Tabla C.1
Respuesta		Descripción
Véase Tabla C.1		-

Tabla C.17. Método Conectar_A_Servicio_Serial (AT*ADCP=).

Orden AT	Método	Descripción
AT*ADCP=<bd_addr>,<rol_y_perfil>,<canal_servidor_rfcomm>,<maestro><CR>	Escribir_RolMaestroEsclavo	Se conecta a un servicio serial habilitado sobre algún dispositivo remoto
Parámetros solicitados	Tipo	Valor
<bd_addr>	Bd_Addr	Dirección Bluetooth del dispositivo a conectarse
<rol_y_perfil>	Enumerador	0: Rol Dispositivo B (DevB), en perfil de puerto serie 1: Rol de pasarela (Gateway), perfil de red Dial-Up 2: Rol de punto de acceso LAN, perfil de acceso LAN
<canal_servidor_rfcomm>	Enumerador	Número del canal RFCOMM del servicio al que se desea conectar 0: Se encuentra el primer canal disponible de forma automática

		1-30: Canales del servicio RFCOMM 31-255: Valores inválidos
<maestro>	Enumerador	0: El dispositivo remoto elige ser nodo Maestro o nodo Esclavo 1: El dispositivo siempre se convertirá en nodo Maestro en nuevas conexiones
Respuesta		Descripción
Véase Tabla C.1		-

Tabla C.18. Método Escribir_Perfil_Cliente (AT*ADDCP=).

Orden AT	Método	Descripción
AT*ADDCP=<rol_y_perfil>,<guardar><CR>	Escribir_Perfil_Cliente	Escribe el perfil del nodo en modo cliente y se activa cuando el nodo cambia a modo de operación de Datos
Parámetros obtenidos	Tipo	Valor
<rol_y_perfil>	Enumerador	0: Rol DevB, perfil de puerto serie (por defecto) 1: Rol de pasarela, perfil de red dial-up 2: Rol de punto de acceso, perfil de acceso LAN 255: Sin perfil
<guardar>	Enumerador	Véase Tabla C.1
Respuesta		Descripción
Véase Tabla C.1		-

C.3. Servicios Suplementarios

A continuación se describen las órdenes AT empleadas por los métodos utilizados por el subsistema Servicios Suplementarios (Sección 3.2.3.1.3).

Tabla C.19. Método Escribir_Modo_Ahorro (AT*AMPM=).

Orden AT	Método	Descripción
AT*AMPM=<modo>,<guardar><CR>	Escribir_Modo_Ahorro	Escribe el modo de ahorro de energía sobre el nodo
Parámetros solicitados	Tipo	Valor
<modo>	Enumerador	1: Modo siempre conectado. El nodo en los modos de bajo consumo o en modo de paro 2: Modo de bajo consumo. El nodo entrará en modo de bajo consumo cuando sea posible (por defecto) 3: Modo de paro. El nodo entrará en modo de paro cuando sea posible (modo no soportado por los nodos Dream-Blue)

<guardar>	Enumerador	Véase Tabla C.1
Respuesta		Descripción
Véase Tabla C.1		-

Tabla C.20. Método Leer_Modo_Ahorro (AT*AMPM?).

Orden AT	Método	Descripción
AT*AMPM?<CR>	Leer_Modo_Ahorro	Lee el modo de ahorro que existe sobre el nodo
Respuesta		Descripción
<CR><LF>*AMPM:<modo><CR><LF>OK<CR><LF>		Respuesta de éxito
<CR><LF>ERROR<CR><LF>		Respuesta de error

Tabla C.21. Método Sondeo (AT*AGI=).

Orden AT	Método	Descripción
AT*AGI=<tipo_sondeo>, <longitud_sondeo>, <max_No_dispositivos><CR>	Sondeo	Se lleva a cabo un procedimiento para encontrar dispositivos con el modo descubrible activado dentro de la vecindad del nodo
Parámetros solicitados	Tipo	Valor
<tipo_sondeo>	Enumerador	1: Sondeo limitado 2: Sondeo general (por defecto)
<longitud_sondeo>	Enumerador	Tiempo máximo del sondeo: Rango: 1-48 Tiempo: <longitud_sondeo>* 1.28 seg. Rango en segundos: 1.28-61.44
Respuesta		Descripción
<CR><LF>*AGI:<bd_addr>,<cod><OK><CR><LF>		Respuesta por cada dispositivo encontrado
<CR><LF>OK<CR><LF>		Respuesta de éxito
<CR><LF>ERROR<CR><LF>		Respuesta de error

Tabla C.22. Método Descubrir_Dispositivos (AT*AGDD=).

Orden AT	Método	Descripción
AT*AGDD=<tipo_sondeo>, <longitud_sondeo><CR>	Descubrir_Dispositivos	Procedimiento de sondeo para encontrar dispositivos
Parámetros	Tipo	Valor
<tipo_sondeo>	Enumerador	1: Sondeo limitado 2: Sondeo general (por defecto)
<longitud_sondeo>	Enumerador	Tiempo máximo del sondeo:

	Rango: 8-48 Tiempo: <longitud_sondeo>* 1.28 seg. Rango en segundos: 10.24-61.44
Respuesta	Descripción
<CR><LF>*AGDD:<no_dispositivos><OK><CR><LF>	Respuesta de éxito
*AGDDE:<bd_addr>,<cod>,<nombre_valido>,<nombre_dispositivo><OK><CR><LF>	Respuesta por cada dispositivo encontrado
<CR><LF>ERROR<CR><LF>	Respuesta de error

Tabla C.23. Método Escribir_Modo_Emparejamiento (AT*AGPM=).

Orden AT	Método	Descripción
AT*AGPM=<modo>,<guardar><CR>	Escribir_Modo_Emparejamiento	Escribe el modo de emparejamiento del nodo
Parámetros solicitados	Tipo	Valor
<modo>	Enumerador	1: Modo de emparejamiento desactivado 2: Modo de emparejamiento activado (por defecto)
<guardar>	Enumerador	Véase Tabla C.1
Respuesta	Descripción	
Véase Tabla C.1	-	

Tabla C.24. Método Leer_Modo_Emparejamiento (AT*AGPM?).

Orden AT	Método	Descripción
AT*AGPM?<CR>	Leer_Modo_Emparejamiento	Lee el modo de emparejamiento sobre el nodo
Respuesta	Descripción	
<CR><LF>*AGPM:<modo><CR><LF>OK<CR><LF>	Respuesta de éxito	
<CR><LF>ERROR<CR><LF>	Respuesta de error	

Tabla C.25. Método Escribir_Codigo_PIN (AT*AGFP=).

Orden AT	Método	Descripción
AT*AGFP=<codigo_pin>,<guardar><CR>	Escribir_Codigo_PIN	Escribe el código PIN utilizado por el nodo durante el proceso de unión
Parámetros solicitados	Tipo	Valor
<codigo_pin>	Cadena	El código PIN es una cadena de 1 a 16 caracteres en formato ASCII
<guardar>	Enumerador	Véase Tabla C.1

Respuesta	Descripción
Véase Tabla C.1	-

Tabla C.26. Método Leer_Codigo_PIN (AT*AGFP?).

Orden AT	Método	Descripción
AT*AGFP?<CR>	Leer_Codigo_PIN	Lee el código PIN utilizado por el nodo durante el procedimiento de unión
Respuesta	Descripción	
<CR><LF>*AGFP:<codigo_pin><CR><LF>OK<CR><LF>	Respuesta de éxito	
<CR><LF>ERROR<CR><LF>	Respuesta de error	

Tabla C.27. Método Union (AT*AGPM=).

Orden AT	Método	Descripción
AT*AGB=<bd_addr><CR>	Union	Se lleva a cabo un procedimiento de unión con otro dispositivo Bluetooth
Parámetros solicitados	Tipo	Valor
<bd_addr>	Bd_Addr	Dirección Bluetooth del dispositivo con el que se desea establecer la unión
Respuesta	Descripción	
<CR><LF>OK<CR><LF>	Respuesta de éxito	
<CR><LF>ERROR<CR><LF>	Respuesta de error	

Tabla C.28. Método Desunir (AT*AGPM?).

Orden AT	Método	Descripción
AT*AGUB=<bd_addr><CR>	Desunir	Los dispositivos Bluetooth previamente unidos con el nodo son eliminados de la lista de uniones
Respuesta	Descripción	
<CR><LF>OK<CR><LF>	Respuesta de éxito	
<CR><LF>ERROR<CR><LF>	Respuesta de error	

Tabla C.29. Método Leer_Uniones (AT*AGBD?).

Orden AT	Método	Descripción
AT*AGBD?<CR>	Leer_Uniones	Lista los dispositivos unidos al nodo
Respuesta	Descripción	
<CR><LF>*AGBD:<no_dispositivos><CR><LF>OK<CR><LF>	Respuesta de éxito	
*AGBDE:<bd_addr>,<nombre_valido>,<nombre_dispositivo><CR><LF>	Respuesta por cada dispositivo unido al nodo	
<CR><LF>ERROR<CR><LF>	Respuesta de error	

Tabla C.30. Método Escribir_Modo_Seguridad (AT*AGSM=).

Orden AT	Método	Descripción
AT*AGSM=<modo>, <guardar><CR>	Escribir_Modo_Seguridad	Escribe el modo de seguridad sobre el nodo
Parámetros solicitados	Tipo	Valor
<modo>	Enumerador	1: Modo de seguridad desactivado (por defecto) 2: Modo de seguridad activado (autenticación y encriptación activados)
<guardar>	Enumerador	Véase Tabla C.1
Respuesta	Descripción	
Véase Tabla C.1	-	

Tabla C.31. Método Leer_Modo_Seguridad (AT*AGSM?).

Orden AT	Método	Descripción
AT*AGSM?<CR>	Leer_Modo_Seguridad	Lee el modo de seguridad que existe sobre el nodo
Respuesta	Descripción	
<CR><LF>*AGSM:<modo><CR><LF>OK<CR><LF>	Respuesta de éxito	
<CR><LF>ERROR<CR><LF>	Respuesta de error	

D. Manual de usuario

En este anexo se describe la forma de utilizar los nodos del sistema DreamBlue.

D.1. Operaciones básicas sobre los nodos DreamBlue

D.1.1. Suministro de energía a los nodos DreamBlue

Para suministrar energía a los nodos DreamBlue, el usuario debe seguir el diagrama de flujo de la Figura D.1.

La Figura D.2 muestra cómo suministrar energía al módulo DTE (5 VDC) y cómo conectar el cable de energía del módulo DCE; así mismo se pueden observar los puertos de integración para unir el módulo DCE al módulo DTE.

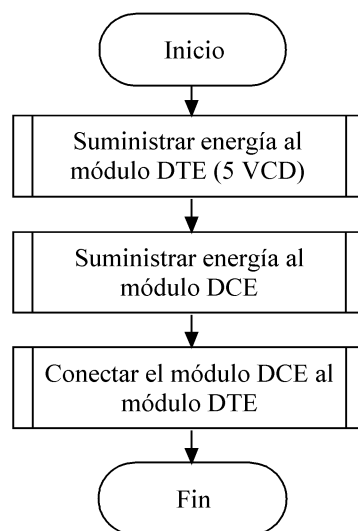


Figura D.1. Suministro de energía de los nodos.

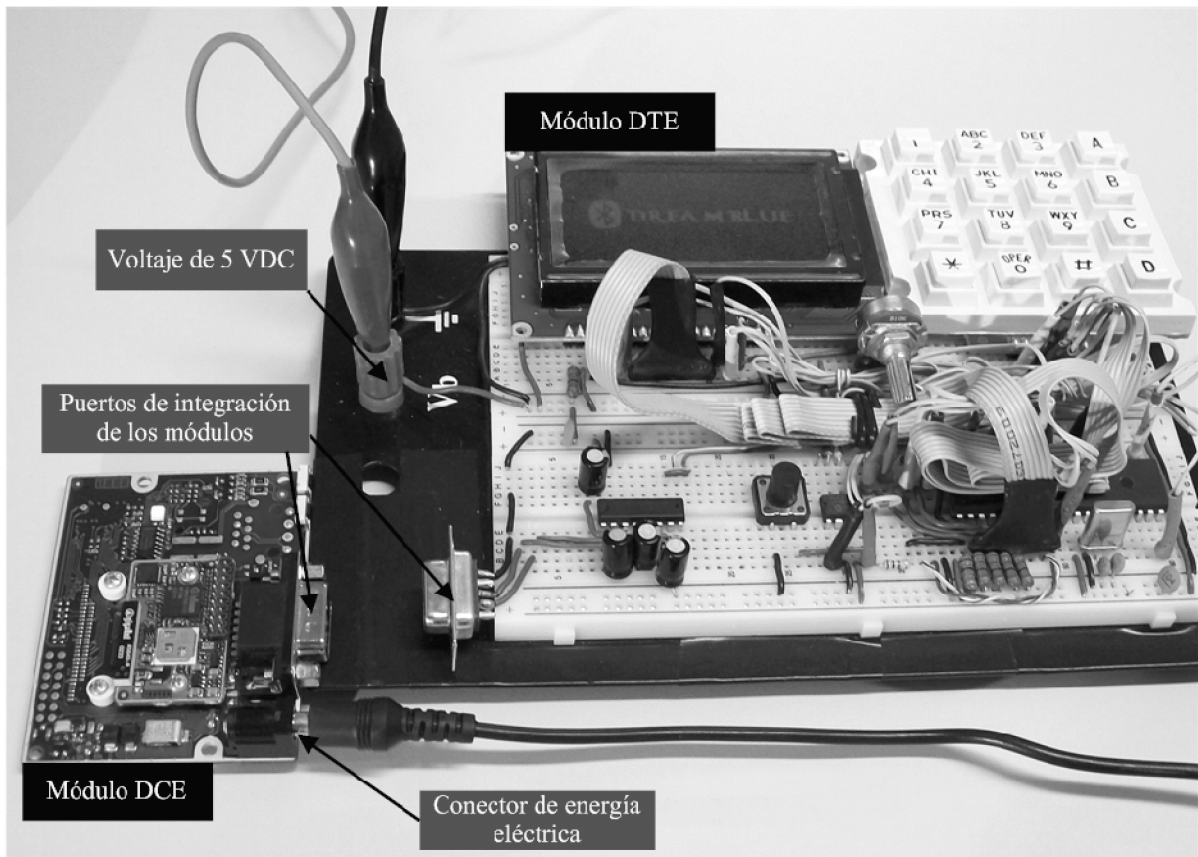


Figura D.2. Alimentación de los módulos del nodo.

D.1.2. Operaciones sobre el teclado

Para seleccionar las opciones mostradas en los diferentes menús del sistema, el usuario debe pulsar la tecla dependiendo del número indicado por la opción (Figura D.3).



Figura D.3. Selección de opciones.

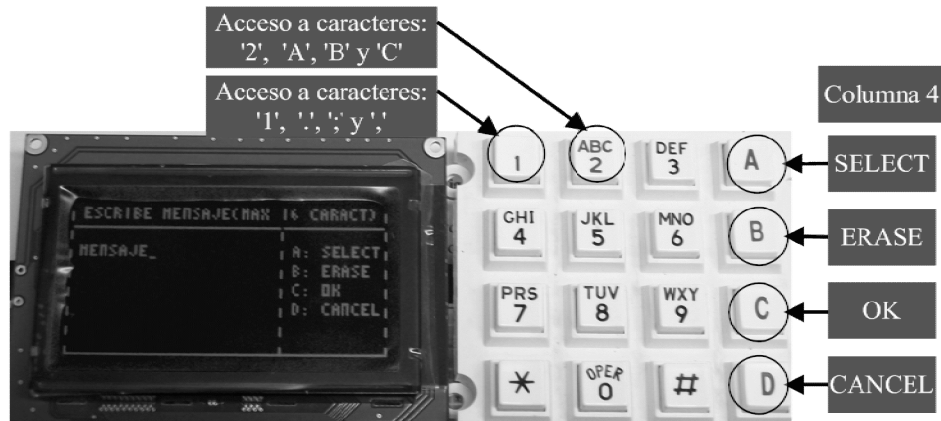


Figura D.4. Acceso y selección de escritura de caracteres.

Existen algunos métodos con los que el usuario necesite escribir caracteres de manera consecutiva. Las etiquetas sobre cada tecla⁴¹ indican los diversos caracteres a los que puede acceder el usuario, pulsando varias veces la tecla donde se encuentre el carácter deseado; enseguida, con la opción SELECT (A) se elige el carácter e incrementará el cursor en una posición. En caso de que el usuario decida borrar un carácter anterior sobre la cadena, éste puede retroceder un espacio con la opción ERASE (B). Una vez completada la secuencia de caracteres deseada, el usuario debe elegir la opción OK (C) para finalizar la escritura de la cadena. Para cancelar la operación de escritura de caracteres, el usuario debe elegir la opción CANCEL. Cabe señalar que el usuario no debe confundir el acceso a los caracteres ‘A’, ‘B’, ‘C’ y ‘D’ con las teclas de la columna 4 (Figura D.4).

D.1.3. Configuración de la velocidad de transmisión

El diagrama de flujo de la Figura D.5 muestra la manera de configurar la velocidad de transferencia (el usuario debe establecer la misma velocidad en ambos nodos DreamBlue).

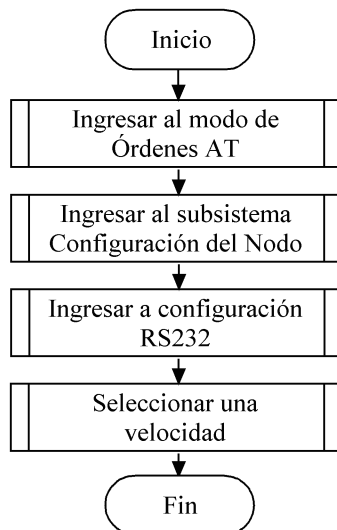


Figura D.5. Configuración de la velocidad.

⁴¹ Las teclas 1, *, 0 y #, no muestran la secuencia de caracteres a las que el usuario puede acceder.

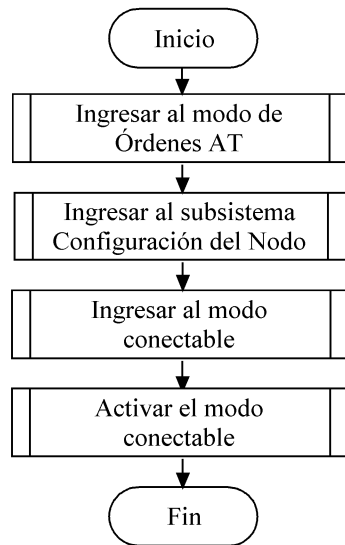


Figura D.6. Activación del modo conectable.

D.1.4. Activación del modo conectable

El diagrama de flujo de la Figura D.6 describe cómo activar el modo conectable sobre los nodos (el usuario debe realizar este paso en ambos nodos DreamBlue).

D.1.5. Activación del modo detectable

El diagrama de flujo de la Figura D.7 describe cómo activar el modo detectable sobre los nodos (el usuario debe realizar este paso en ambos nodos DreamBlue).

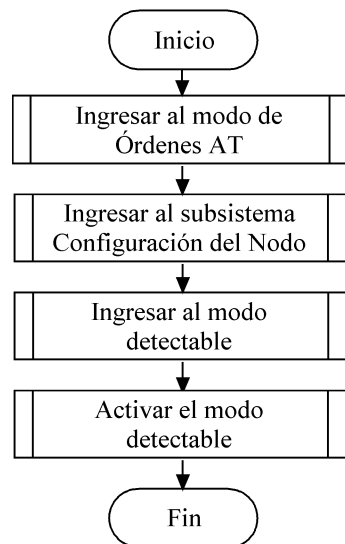


Figura D.7. Activación del modo detectable.

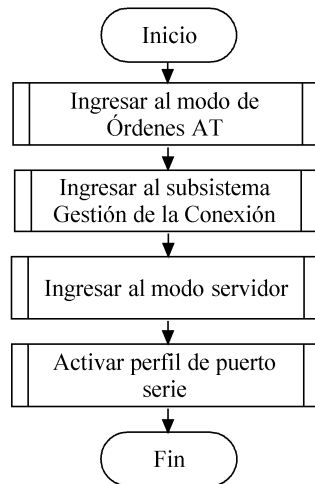


Figura D.8. Configuración de enlace.

D.1.6. Configuración del enlace

El diagrama de flujo de la Figura D.8 describe cómo configurar los nodos DreamBlue para establecer un enlace con perfil de puerto serie sobre en modo servidor (el usuario debe realizar este paso en ambos nodos DreamBlue).

D.1.7. Establecimiento de seguridad entre los nodos DreamBlue

Para establecer la seguridad del enlace entre los nodos DreamBlue, el usuario debe seguir los pasos descritos a continuación sobre ambos nodos.

D.1.7.1 Activación del modo de emparejamiento

El diagrama de flujo de la Figura D.9 muestra como activar el modo de emparejamiento sobre los nodos DreamBlue (el usuario debe activar el modo de emparejamiento sobre ambos nodos DreamBlue).

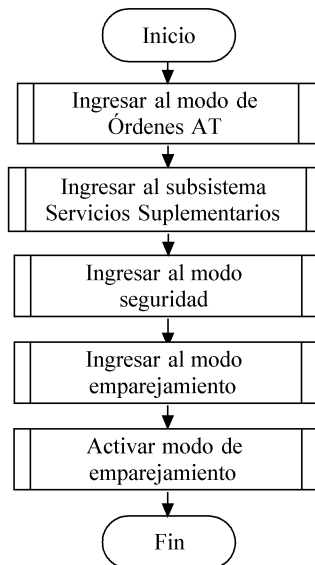


Figura D.9. Activación del modo de emparejamiento.

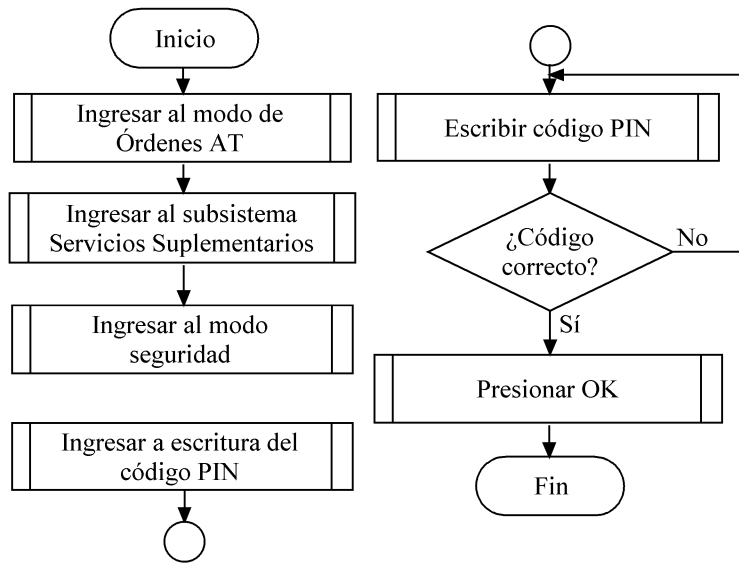


Figura D.10. Escritura del código PIN.

D.1.7.2 Escritura del código PIN

El diagrama de flujo de la Figura D.10 muestra el ingreso al modo de escritura del código PIN sobre los nodos DreamBlue (el usuario debe establecer el mismo código en ambos nodos DreamBlue).

D.1.7.3 Unión entre los nodos DreamBlue

El diagrama de flujo de la Figura D.11 muestra el ingreso al modo unión para escribir la dirección Bluetooth del nodo a unir (para que los nodos DreamBlue puedan unirse correctamente, ambos necesitan tener el modo de emparejamiento activado y el mismo código PIN para el proceso de autenticación).

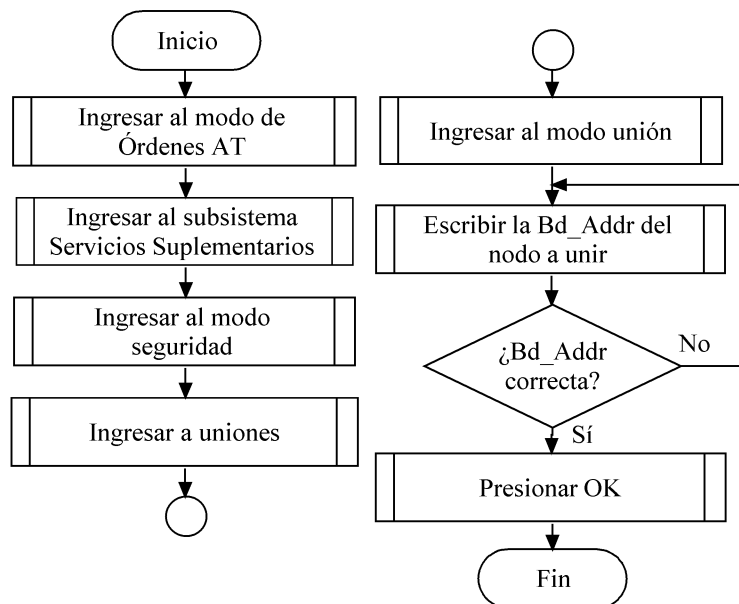


Figura D.11. Escritura de la Bd_Addr para el proceso de unión.

D.1.7.4 Activación del modo seguro

El diagrama de flujo de la Figura D.12 muestra el proceso de activación del modo seguro sobre ambos nodos DreamBlue (el usuario debe activar en ambos nodos el modo seguro).

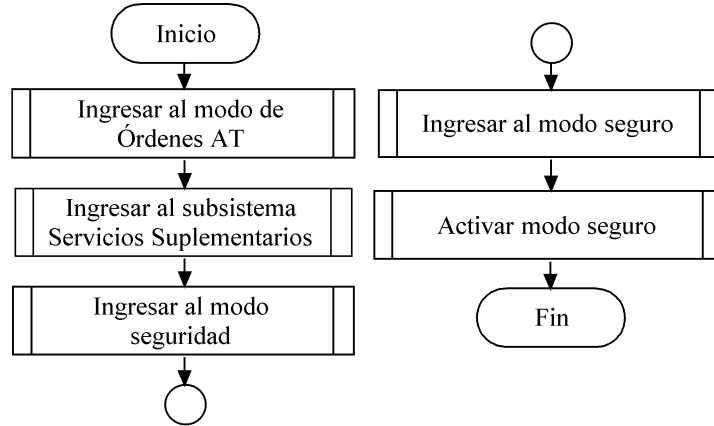


Figura D.12. Activación del modo seguro.

D.1.8. Sondeo de dispositivos

El diagrama de flujo de la Figura D.13 muestra una forma de realizar un sondeo de dispositivos Bluetooth en la vecindad sobre cualquier nodo DreamBlue (el usuario deberá esperar aproximadamente 10 segundos después de elegir el método para obtener una lectura correcta de los dispositivos Bluetooth encontrados en la vecindad).

D.1.9. Envío de mensaje entre los nodos DreamBlue

Una vez realizados los pasos anteriores descritos en este manual (no es necesario realizar el proceso de seguridad ni el sondeo de dispositivos sobre los nodos), los nodos pueden transmitirse mensajes de texto siguiendo los pasos descritos por el diagrama de flujo de la Figura D.14.

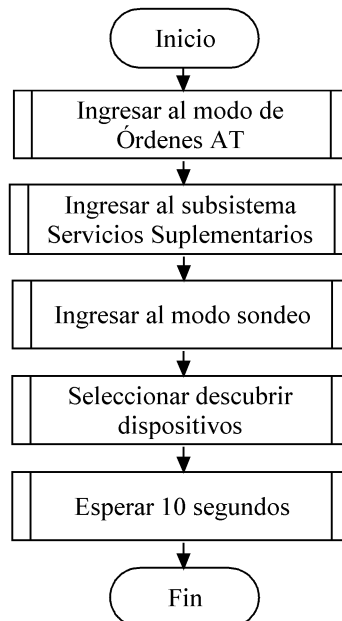


Figura D.13. Sondeo de dispositivos.

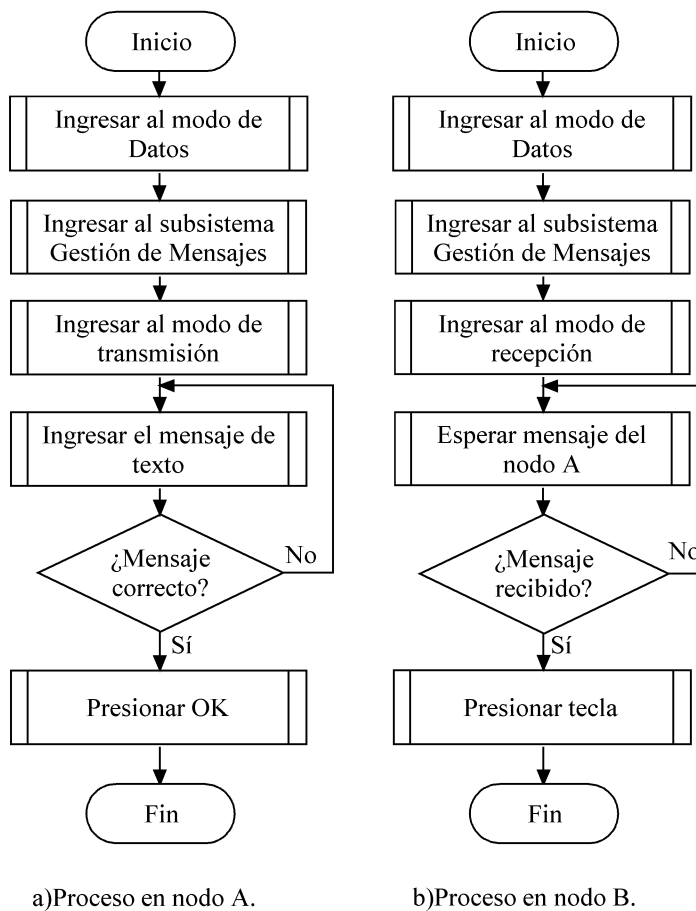


Figura D.14. Envío de mensajes de texto entre los nodos DreamBlue.

D.2. Solución de errores

Existen dos tipos de errores que pueden ocurrir sobre los nodos DreamBlue:

- *Error interno del nodo*: Este tipo de error suele ocurrir cuando el módulo DTE retiene datos en la memoria temporal y rompe la sincronización con el módulo DCE, ocasionando errores sobre los mensajes de respuesta a las órdenes AT (Figura D.15). Para solucionar el tipo de error interno, el usuario debe restaurar el módulo DTE (Figura D.17).
- *Error de escritura en orden AT*: Este tipo de error ocurre cuando se ha enviado de forma incorrecta una orden AT al módulo DCE (Figura D.16). Para solucionar el error de escritura sobre la orden AT es suficiente con intentar nuevamente el ingreso de la orden AT; sin embargo, se recomienda al usuario restaurar el módulo DTE y regresar al sitio donde se encontraba antes de ocurrir el error (el módulo DCE seguirá en modo de Órdenes AT).



Figura D.15. Error interno del nodo DreamBlue.



Figura D.16. Error de escritura en orden AT.

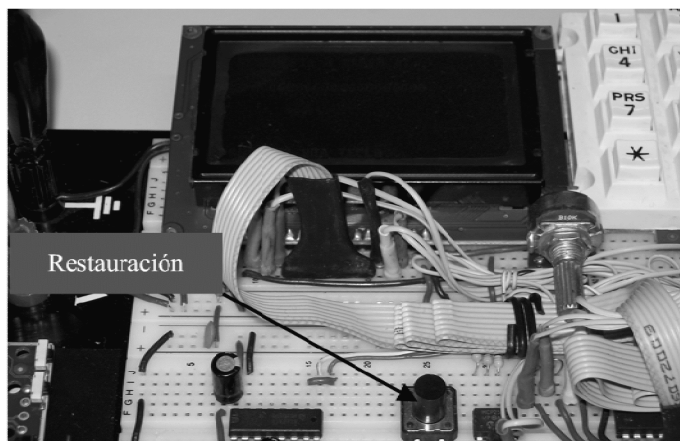


Figura D.17. Restauración del módulo DTE.

