

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Proyecto Fin de Carrera

Diseño, implementación y validación de un simulador de servicios de datos en DECT



AUTOR: Esteban Sánchez Sánchez
DIRECTOR: Juan José Alcaraz Espín

Mayo / 2005



Autor	Esteban Sánchez Sánchez
E-mail del Autor	est_san2@yahoo.com
Director(es)	Juan José Alcaraz
E-mail del Director	juan.alcaraz@upct.es
Codirector(es)	
Título del PFC	Diseño, Implementación y validación de un simulador de servicios de datos en DECT
Descriptor(es)	
<p>Resúmen:</p> <p>El proyecto que se presenta se ha centrado en el estudio, diseño, implementación y validación de un simulador de los protocolos DECT orientados a la integración de los servicios de datos con los servicios tradicionales de voz sobre DECT.</p> <p>Para este trabajo se ha realizado un análisis detallado de las últimas versiones de los documentos que conforman el estándar, donde se definen los nuevos procedimientos y características requeridos por los servicios de datos.</p> <p>La herramienta software ha sido implementada en C++, apoyándose en la librería de simulación por eventos discretos OMNET++. La filosofía de orientación a objetos de C++ y OMNET++ permiten que el simulador sea escalable y portable.</p> <p>Gracias a la cantidad de parámetros configurables, es posible simular un gran número de características y procedimientos, por lo que esta herramienta es adecuada tanto para la investigación centrada en la evaluación de prestaciones como para aplicaciones didácticas.</p>	
Titulación	Ingeniería Superior de Telecomunicación
Intensificación	Telemática
Departamento	Tecnologías de la Información y las Comunicaciones
Fecha de Presentación	Mayo 2005

Agradecimientos:

En primer lugar quisiera agradecer a mi director de proyecto Juan José Alcaraz por confiar en mí para el desarrollo de este proyecto. Quisiera dedicar este proyecto a toda mi familia y en especial a mis padres Esteban y Carmen, a mi hermana Carmen y a mis abuelos Antonio, Juana, Juan y Carmen por su confianza, su comprensión y su apoyo incondicional.

Quisiera agradecer también el apoyo de mi novia María, y de mis amigos Jorge, Kiko, Pablo N., D.Damián, Carlos N., Carlos A. Neng., Sergio, Paco, F. Mula, Juan Carlos, Laura Z. Por último quisiera agradecer a todos los médicos y enfermeros que cuidaron de mí en el hospital de Virgen de la Caridad sin los cuales hoy no estaría aquí.

ÍNDICE DE CAPÍTULOS

CAPÍTULO 1 INTRODUCCIÓN	21
1.1 ANTECEDENTES	21
1.1.1 DECT FRENTE A VOWLAN	21
1.2 OBJETIVOS DEL PROYECTO	23
1.3 CONTENIDO DEL DOCUMENTO	24
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA DECT	27
2.1 INTRODUCCIÓN	27
2.2 INFRAESTRUCTURA DE UN SISTEMA DECT	27
2.2.1 LA PARTE PORTÁTIL (PP)	27
2.2.2 LA PARTE FIJA (FP)	28
2.3 EL ESTÁNDAR BASE DECT	28
2.4 PERFILES DE SERVICIOS DE DATOS (DSP)	29
2.4.1 OBJETIVO DEL LOS DSPS	30
2.4.2 DESCRIPCIÓN DE LOS SERVICIOS	30
2.4.2.1 Tipos de servicios	30
2.5 LA CAPA FÍSICA (PHL)	32
2.5.1 ESTRUCTURA DE LA TDMA	32
2.5.2 FORMATOS FÍSICOS DE LOS PAQUETES	33
2.6 LA CAPA DE ACCESO AL MEDIO (MAC)	34
2.6.1 FORMATO DE LOS PAQUETES MAC	34
2.6.1.1 Formato de la cabecera (campo A)	34
2.6.1.2 Formato del campo de datos (campo B)	35
2.6.2 PORTADORAS LÓGICAS MAC (BEARERS)	37
2.6.3 PORTADORAS DE TRÁFICO	38
2.6.4 SERVICIOS ORIENTADOS A LA CONEXIÓN	38
2.6.5 CONEXIONES SIMÉTRICAS Y ASIMÉTRICAS	38
2.6.5.1 Tipos de servicios y reglas de las conexiones simétricas	38
2.6.5.2 Tipos de servicios y reglas de las conexiones asimétricas	39
2.6.5.3 Procedimiento para la retransmisión de datos	39
2.7 INTRODUCCIÓN A LA CAPA DLC	42
2.7.1 SERVICIOS Y ENTIDADES DEL PLANO DE USUARIO (PLANO-U)	42
2.7.2 SERVICIO FRAME RELAY MEJORADO LU10 (EFREL)	43
2.7.2.1 Formato de las tramas FU10a y FU10c	44
2.7.3 CLASES DE TRANSMISIÓN DEL PARA EL SERVICIO LU10	46
2.7.3.1 Procedimientos de la clase 2	46
2.7.3.2 Procedimientos de la clase 3	48
2.8 SERVICIO DE RADIO PAQUETES (DPRS)	49
2.8.1 OBJETIVOS DEL SERVICIO	50
2.8.2 REQUISITOS Y PROCEDIMIENTOS DEL SISTEMA DECT	50
2.8.2.1 Requisitos de la capa PHL	50
2.8.2.2 Requisitos de la capa MAC	50
2.8.2.3 Requisitos de la capa DLC	51
2.8.3 PROCEDIMIENTOS DE LA CAPA DLC	51
2.8.3.1 Servicio Frame Relay mejorado LU10 (EFREL)	51
2.8.3.2 Tamaño de la ventana	51
2.8.3.3 Procedimientos de la clase 2 de transmisión	52
2.8.4 SERVICIO FRAME RELAY (FREL)	53

2.8.5 CONVENIONES GENERALES DE INTERACTUACIÓN PARA FREL	54
2.8.6 CONVENIONES DE INTERACTUACIÓN CON IP	54
CAPÍTULO 3: ESCENARIO DE REFERENCIA	55
3.1. INTRODUCCIÓN	55
3.2. DESCRIPCIÓN DEL ESCENARIO	55
3.3. ARQUITECTURA DEL PROTOCOLO	57
3.4. UNIDADES DE DATOS DEL PROTOCOLO	58
CAPÍTULO 4: DESCRIPCIÓN DEL SIMULADOR	61
4.1 INTRODUCCIÓN	61
4.2 IMPLEMENTACIÓN DE LA PILA DE PROTOCOLOS	61
4.2.1 IMPLEMENTACIÓN DE LA CAPA PHL	62
4.2.1.1 Implementación de la trama TDMA	62
4.2.2 IMPLEMENTACIÓN DE LA CAPA MAC	63
4.2.2.1 Diseño de la conexión asimétrica	63
4.2.2.2 Procedimiento de intercambio de conexiones	68
4.2.2.3 Servicios implementados	72
4.2.3 IMPLEMENTACIÓN DE LA CAPA DLC	73
4.2.3.1 Implementación del servicio LU10	73
4.2.3.2 Implementación de la clase 2	73
4.2.4 IMPLEMENTACIÓN DEL TRÁFICO TCP/IP	75
4.2.5 MODELO DE ERRORES DEL CANAL	76
4.2.5.1 Cadena de Markov de dos estados	77
4.2.5.2 Formato de protección de un solo campo	78
4.2.5.3 Formato de protección de subcampos.	79
4.3 DESCRIPCIÓN DE LOS MÓDULOS	79
4.3.1 MÓDULO PRINCIPAL “TCPDECT”	80
4.3.2 MÓDULO “DECTNET”	80
4.4 IMPLEMENTACIÓN DE LOS MENSAJES	84
4.4.1 IMPLEMENTACIÓN DE LAS PRIMITIVAS	84
4.4.2 COMUNICACIÓN ENTRE DLC Y MAC	85
4.4.2.1 Primitivas MAC à DLC	85
4.4.2.2 Primitivas DLC à MAC	85
4.4.3 COMUNICACIÓN ENTRE IWU Y DLC	85
4.4.3.1 Primitivas IWU à DLC	85
4.4.3.2 Primitivas DLC à IWU	86
4.4.4 IMPLEMENTACIÓN DE LOS PAQUETES FÍSICOS QUE VIAJAN A TRAVÉS DE LA RED	86
4.4.5 DESCRIPCIÓN DE LA COMUNICACIÓN ENTRE LOS MÓDULOS DEL SIMULADOR	86
4.4.5.1 Transmisión de un paquete IP	86
4.4.5.2 Transmisión un reconocimiento para el servidor	92
CAPÍTULO 5: MANEJO DEL SIMULADOR	97
5.1. INSTALACIÓN DE OMNET++	97
5.2. OPCIONES DE COMPILACIÓN	97
5.3. CONFIGURACIÓN DE UNA SIMULACIÓN	97
5.3.1. PARÁMETROS DE ENTRADA TCP	98
5.3.2. PARÁMETROS DE ENTRADA DEL SISTEMA DECT	98
5.4. EJECUCIÓN DE UNA SIMULACIÓN	100

5.4.1. PARÁMETROS DE SALIDA PARA TCP	101
5.4.2. PARÁMETROS DE SALIDA DEL SISTEMA DECT	102
5.5. FICHEROS DE REGISTRO DE LOS MÓDULOS	103
5.5.1. FICHERO “SERVER.LOG”	103
5.5.1.1. Ejemplo del fichero “server.log”	104
5.5.2. FICHERO “APPLICATION.LOG”	104
5.5.2.1. Ejemplo del fichero “application.log”	104
5.5.3. FICHEROS “IWU_APPLICATION.LOG” E “IWU_ROUTER.LOG”	105
5.5.3.1. Ejemplo del fichero “iwu_router.log”	105
5.5.3.2. Ejemplo del fichero “iwu_application.log”	105
5.5.4. FICHEROS “DLCTRXDWN.LOG” Y “DLCTRXUP.LOG”	106
5.5.4.1. Ejemplos del fichero “DlcTrxDwn.log”	106
5.5.5. FICHEROS “DLCRXDWN.LOG” Y “DLCRXUP.LOG”	108
5.5.5.1. Ejemplos del fichero “DlcRxDwn.log”	108
5.5.6. FICHEROS “MACTRXDWN.LOG” Y “MACTRXUP.LOG”	110
5.5.6.1. Ejemplos del fichero “MacTrxDwn.log”	111
5.5.7. FICHEROS “MACRXDWN.LOG” Y “MACRXUP.LOG”	112
5.5.7.1. Ejemplos del fichero “MacRxDwn.log”	112
5.5.8. FICHERO “CHANNEL.LOG”	113
5.5.8.1. Ejemplos del fichero “Channel.log”	114
5.5.9. FICHEROS “DOWNLINK.LOG” Y “UPLINK.LOG”	114
5.5.9.1. Ejemplo del fichero “Downlink.log”	115

CAPÍTULO 6: EVALUACIÓN DE PRESTACIONES MEDIANTE SIMULACIÓN **119**

6.1. INTRODUCCIÓN:	119
6.2. SERVICIOS ASIMÉTRICOS PARA SIMULAR:	119
6.3. CÁLCULO TEÓRICO DEL THROUGHPUT	120
6.4. VENTANA DE RECEPCIÓN DE LA APLICACIÓN	123
6.5. TAMAÑO DEL BUFFER DE LA FT	123
6.6. SELECCIÓN DEL TIEMPO DE VIDA DE LAS SDUS	124
6.7. EXPERIMENTOS PARA UN CANAL SIN ERRORES	124
6.7.1. EXPERIMENTO 1	124
6.7.2. EXPERIMENTO 2	126
6.7.3. EXPERIMENTO 3	127
6.7.4. EXPERIMENTO 4	128
6.7.5. EXPERIMENTO 5	129
6.7.6. EXPERIMENTO 6	130
6.7.7. EXPERIMENTO 7	131
6.7.8. EXPERIMENTO 8	132
6.7.9. EXPERIMENTO 9	133
6.7.10. EXPERIMENTO 10	134
6.8. EXPERIMENTOS PARA UN CANAL CON ERRORES	137
6.8.1. EXPERIMENTO 11	137
6.8.2. EXPERIMENTO 12	139
6.8.3. EXPERIMENTO 13	141
6.8.4. EXPERIMENTO 14	141
6.8.5. EXPERIMENTO 15	142

CAPÍTULO 7: CONCLUSIONES Y LÍNEAS FUTURAS **143**

7.1 INTRODUCCIÓN	143
7.2 CONCLUSIONES	143

7.3 LÍNEAS FUTURAS DEL PROYECTO	144
7.4 FUTURO DE DECT	144
7.5 OTROS PROYECTOS RELACIONADOS CON DECT	145

ÍNDICE DE FIGURAS

Figura 2. 1 Entidades funcionales	27
Figura 2. 2 Pila de protocolo del sistema DECT	29
Figura 2. 3 Formato de slot completo	32
Figura 2. 4 Formato de slot doble	33
Figura 2. 5 Paquete básico P32	33
Figura 2. 6 Paquete P80	34
Figura 2. 7 Campos de la cabecera	35
Figura 2. 8 Formato desprotegido	35
Figura 2. 9 Protección por múltiples subcampos (IP)	35
Figura 2. 10 Protección de un solo campo	36
Figura 2. 11 Ejemplo de configuración de portadoras con 1 portadora doble simplex y 1 portadora doble dúplex.....	37
Figura 2. 12 Relación entre LLME, entidad DLC y conexión MAC	43
Figura 2. 13 Trama Fu10a	45
Figura 2. 14 Trama FU10c	45
Figura 2. 15 Trama para el mensaje de sincronización	47
Figura 2. 16 Configuración plano-U	49
Figura 2. 17 Configuración Plano-C	49
Figura 2. 18 Configuración para el servicio FREL	53
Figura 3. 1 Escenario de referencia	55
Figura 3. 2 Escenario de referencia para nuestro simulador	56
Figura 3. 3 Pila de protocolos para IP	57
Figura 3. 4 Pila de protocolos para nuestro escenario	58
Figura 3. 5 Unidades de datos del protocolo para cada capa y sus relaciones (LU10, IP)	59
Figura 3. 6 Unidades de datos del protocolo para cada capa y sus relaciones (LU10, IPQ)	59
Figura 4. 1 Asociación entre las entidades DLC y las conexiones MAC.....	64
Figura 4. 2 Configuración de las portadoras con una portadora “n” y con el enlace downlink usando la conexión de mayor capacidad (Modelo 1).....	65
Figura 4. 3 Configuración de las portadoras con dos portadoras “n” y con el enlace downlink usando la conexión de mayor capacidad (Modelo 2).....	67
Figura 4. 4 Reasociación de las entidades DLC a las conexiones MAC.....	69
Figura 4. 5 Configuración de las portadoras con una portadora “n” y con el enlace uplink usando la conexión de mayor capacidad (Modelo 1)	70
Figura 4. 6 Configuración de las portadoras con dos portadoras “n” y con el enlace uplink usando la conexión de mayor capacidad (Modelo 2)	71
Figura 4. 7 Cadena de Markov de 2 estados.....	77
Figura 4. 8 Esquema que muestra la relación de jerarquía entre los módulos del simulador	79
Figura 4. 9 Contenido del módulo “tcpdect”	80
Figura 4. 10 Contenido del módulo “dectnet”	81
Figura 4. 11 Contenido del módulo “fixedpart” y “fixedtermination”	81
Figura 4. 12 Módulo “portablepart”.....	82
Figura 4. 13 Contenido del módulo “portabletermination”	84
Figura 4. 14 Comunicación entre las entidades	84
Figura 4. 15 Procedimiento de transmisión (entre el servidor y la parte fija)	89
Figura 4. 16 Procedimiento de transmisión (entre la parte fija y la parte portátil)	90
Figura 4. 17 Procedimiento de transmisión (entre la terminación portátil y la aplicación)	91
Figura 4. 18 Procedimiento de transmisión (entre la aplicación y la terminación portátil)	94

Figura 4. 19 Procedimiento de transmisión (entre la parte portátil y la parte fija)	95
Figura 4.20 Procedimiento de transmisión (entre la terminación fija y el servidor)	96
Figura 5. 1 Entorno gráfico (tcpdect)	100
Figura 5. 2 Entorno gráfico (tcpdect.dectnet)	101
Figura 5. 3 Ejemplo de toma de medidas.....	102
Figura 5. 4 Ejemplo del fichero “server.log”.....	104
Figura 5. 5 Ejemplo del fichero “application.log”.....	105
Figura 5. 6 Ejemplo del fichero “iwu_roter.log”	105
Figura 5. 7 Ejemplo del fichero “iwu_application.log”	106
Figura 5. 8 Ejemplo de llegada de una primitiva con una SDU (“DlctrxDwn.log”)	106
Figura 5. 9 Ejemplo de inicio de trama TDMA (“DlcTrxDwn.log”)	107
Figura 5.10 Ejemplo de petición de datos para transmitir en la TDMA (“DlcTrxDwn.log”)	107
Figura 5.11 Ejemplo de recepción de una ACK para el nivel de enlace (“DlcTrxDwn.log”).....	108
Figura 5. 12 Ejemplo de recepción de fragmentos de la capa MAC (“DlcRxDwn.log”)	109
Figura 5. 13 Ejemplo de detección de pérdida de PDUs (“DlcRxDwn.log”)	109
Figura 5. 14 Ejemplo de recepción de PDUs perdidas(“DlcRxDwn.log”)	110
Figura 5. 15 Ejemplo de transmisión de paquetes MAC (“MacTrxDwn.log”).....	111
Figura 5. 16 Ejemplo de fallo en la recepción de los ACKs para MAC (“MacTxDwn.log”)	111
Figura 5. 17 Ejemplo de recepción de ACKs para las portadoras a nivel MAC (“MacTrxDwn.log”)	112
Figura 5. 18 Ejemplo de recepción de paquetes a nivel MAC (“MacRxDwn.log”).....	113
Figura 5. 19 Ejemplo de recepción de paquetes a nivel MAC (MacRxDwn.log”)	113
Figura 5. 20 Ejemplo de cambio del estado del canal	114
Figura 5. 21 Ejemplo del canal en estado malo.....	114
Figura 5. 22 transmisión de un paquete IP del servidor a la parte portátil	117
Figura 7. 1 Comparación de las ventas entre los terminales DECT y los terminales Wi-Fi.....	145

ÍNDICE DE TABLAS

Tabla 2. 1 Formato de los paquetes para el modo de protección de múltiples subcampos.....	36
Tabla 2. 2 Formato de los paquetes para el modo de protección de un solo campo ...	36
Tabla 2. 3 Acción realizada por el transmisor	41
Tabla 2. 4 Acción del receptor	41
Tabla 2. 5 Acción a realizar por el transmisor	42
Tabla 2. 6 Estructura de las PDUS para LU10.....	44
Tabla 2. 7 Tamaño de la ventana.....	52
Tabla 4. 1 Probabilidad de que haya errores en la cabecera o en el campo de datos.	78
Tabla 6. 1 Throughput teórico para la conexión de menor capacidad	121
Tabla 6. 2 Throughput teórico para la conexión de mayor capacidad (modelo 1).....	122
Tabla 6. 3 Throughput teórico para la conexión de mayor capacidad (modelo 2).....	122
Tabla 6. 4 Experimentos para un canal sin errores	124
Tabla 6. 5 Resultados del experimento 1	125
Tabla 6. 6 Resultados del experimento 2.....	126
Tabla 6. 7 Resultados del experimento 3.....	127
Tabla 6. 8 resultados del experimento 4	128
Tabla 6. 9 Resultados del experimento 5.....	129
Tabla 6. 10 Resultados del experimento 6	130
Tabla 6. 11 resultados del experimento 7	131
Tabla 6. 12 Resultados del experimento 8.....	132
Tabla 6. 13 Experimentos para un canal con errores.....	137
Tabla 6. 14 Resultados del experimento 11	138
Tabla 6. 15 Resultados del experimento 12	139
Tabla 6. 16 Resultados del experimento 13	141
Tabla 6. 17 Resultados del experimento 14	141
Tabla 6. 18 Resultados del experimento 15	142

ÍNDICE DE GRÁFICAS

Gráfica 6. 1 Evolución del throughput	133
Gráfica 6. 2 Evolución del retardo extremo a extremo.....	134
Gráfica 6. 3 Comparativa del throughput (2 niveles de modulación)	134
Gráfica 6. 4 Comparativa del retardo (2 niveles de modulación)	135
Gráfica 6. 5 Comparativa del throughput (4 niveles de modulación)	135
Gráfica 6. 6 Comparativa del retardo (4 niveles de modulación)	136
Gráfica 6. 7 Comparativa del throughput (8 niveles de modulación)	136
Gráfica 6. 8 Comparativa del retardo (8 niveles de modulación)	136

DEFINICIONES DE TÉRMINOS

Campo A: cabecera de un paquete MAC del sistema DECT.

Campo B: campo de datos de un paquete MAC del sistema DECT.

Canal físico: canal que se crea por la transmisión en un determinado slot sobre un canal RF a lo largo de sucesivas tramas TDMA.

Canal RF: rango de frecuencias usado para transmitir una portadora RF.

Capa Física (PHL): capa primera de la pila de protocolo DECT.

Celda: dominio al que da servicio un sistema con una FP.

Central de control de la parte fija (CCFP): componente de la FP que se encarga de implementar los procedimientos de la pila de protocolo del sistema DECT.

Clase 2: conjunto de procedimientos para la transmisión de paquetes a nivel de enlace con el servicio LU10.

Clase 3: conjunto de procedimientos para la transmisión de paquetes a nivel de enlace con el servicio LU10.

Clúster: grupo de una o más celdas en el cual existe la posibilidad de handover.

Conexión MAC: asociación entre las entidades de Control de Multi-Portadora MAC (MBC) de los dos extremos de la comunicación (fuente y destino).

Conexión asimétrica: conexión MAC donde el número de portadoras "k" en un sentido es mayor o igual que el número de portadoras "m"+"n".

Conexión simétrica: conexión MAC donde el número de portadoras de tráfico en ambas direcciones de comunicación es el mismo.

DECT: tecnología de acceso radio para comunicaciones inalámbricas.

DLC: capa de control del enlace que pertenece al segundo nivel de la pila de protocolos del estándar DECT.

Double Slot: uno de los 12 slots de una trama TDMA que es usado para soportar un canal físico.

Enlace de datos DLC: asociación entre dos entidades de la capa DLC.

Enlace de bajada (Downlink): canal por el que viajan los datos desde la terminación fija a la terminación portátil.

Enlace de subida (Uplink): canal por el que viajan los datos desde la terminación móvil a la terminación fija..

Entidad de dirección de la capa inferior (LLME): entidad de dirección que abarca un número de capas inferiores y que es usada para describir todos los procesos y

actividades de control que no siguen el modelo tradicional de comunicación entre capas.

FU10a: formato de los paquetes DLC (PDUs) para la dirección principal de una conexión asimétrica.

FU10b: formato de los paquetes DLC (PDUs) para ambas direcciones de comunicación en una conexión simétrica.

FU10c: formato de los paquetes DLC (PDUs) para las portadoras “n” de una conexión asimétrica.

Full slot: uno de los 24 slots de una trama TDMA que es usado para soportar un canal físico.

Half slot: uno de los 48 slots de una trama TDMA que es usada para soportar un canal físico.

IP_detección_error: servicio de la capa MAC que proporciona una detección de errores con protección de múltiples subcampos.

IP_corrección_error: servicio de la capa MAC que proporciona una corrección de errores con protección de múltiples subcampos.

IPQ_detección_error: servicio de la capa MAC que proporciona una detección de errores con protección de un solo campo.

IPQ_corrección_error: servicio de la capa MAC que proporciona una corrección de errores con protección de un solo campo.

LU10: servicio de la capa DLC de un sistema DECT que proporciona un transporte fiable para los paquetes de las capas superiores.

MAC: capa de acceso al medio del segundo nivel de la pila de protocolos del estándar DECT.

Modo orientado a la conexión (C/O): modo de transmisión que transfiere los datos desde un origen a uno o más destinos usando un protocolo basado en tres fases: establecimiento de la conexión, transferencia de los datos y liberación de la conexión.

Modo no orientado a la conexión (C/L): modo de transmisión en el que se transfiere directamente desde la fuente al destino sin que exista ninguna fase de establecimiento y posterior liberación de la conexión.

Msub: formato de protección de los paquetes MAC que divide la información del campo de datos en subcampos cada uno con su propio CRC.

NWK: cuarto nivel de la pila de protocolos del estándar DECT.

Parte fija (FP): agrupación física que contiene todos los elementos fijos entre la red local y la interfaz aérea DECT.

Parte fija de Radio (RFP): estación base que proporciona la cobertura a los terminales móviles de un sistema DECT haciendo uso de una o más antenas.

Parte portátil (PP): agrupación física que contiene todos los elementos portátiles necesarios para proporcionar un teleservicio al usuario.

Perfil de acceso genérico (GAP): documento añadido al estándar base DECT que asegura la interoperabilidad entre FPs y PPs de diferentes fabricantes.

Perfil de acceso para paquetes vía radio (DPRS): documento añadido al perfil del estándar base DECT donde se definen los requisitos para proporcionar un servicio de transporte fiable de paquetes de datos.

PHL: primer nivel de la capa de protocolos del estándar DECT.

Plano-C: plano de control en la pila de protocolo de DECT.

Plano-U: plano de usuario en la pila de protocolo DECT.

Portadora RF: frecuencia central ocupada por una transmisión DECT.

Portadora “k”: portadora en la dirección principal en una conexión asimétrica.

Portadora “m”: portadora en la dirección inversa en una conexión asimétrica.

Portadora “n”: portadora con la información de control para las portadoras “k” de la dirección principal en una conexión asimétrica.

Portadora MAC: elemento creado por la capa MAC para proporcionar un servicio usando un canal físico.

Portadora MAC simplex: portadora que opera en una dirección en un canal físico.

Portadora MAC simplex doble: pareja de dos portadoras sencillas operando en la misma dirección usando dos canales físicos.

Portadora MAC dúplex: pareja de dos portadoras sencillas operando en direcciones opuestas usando dos canales físicos.

Primitivas: mensajes de comunicación entre las capas adyacentes del protocolo.

Red DECT: red que usa la interfaz aérea DECT para conectar una red local a una o más aplicaciones portátiles.

Fragmentación: proceso de dividir una SDU en una o más PDUs para entregárselas a una capa inferior del protocolo.

Servicio de portadora: Tipo de servicio de Telecomunicaciones que proporciona una determinada capacidad para la transmisión de señales entre las interfaces de los usuarios de una red.

Slot: ranura temporal con una determinada duración.

Ssub: formato de protección de los paquetes MAC que emplea un único CRC.

Telecomunicación: cualquier transmisión, emisión y/o recepción de información usando sistemas electromagnéticos.

Teleservicio: tipo de servicio de Telecomunicaciones que proporciona la capacidad completa, incluyendo las funciones de los equipos terminales, para llevar a cabo la comunicación a distancia entre usuarios haciendo uso de una serie de reglas establecidas de ante mano por un protocolo.

Terminación fija: grupo de funciones que implementan todos los procesos y procedimientos DECT en la parte fija de la interfaz aérea DECT.

Terminación portátil: grupo de funciones que implementan los servicios y procesos DECT en la parte portátil de la interfaz aérea DECT.

Trama TDMA: multiplexación por división en el tiempo de 10 ms de duración que contiene 24 Full slots.

Unidad de interacción (IWU): unidad usada para la interacción de subredes con diferentes protocolos.

VoIP: estándar para el transporte de tráfico de voz sobre el protocolo IP.

VoWLAN: tecnología que emplea las redes LAN inalámbricas con tecnología Wi-Fi (WLAN) para proporcionar servicios voz sobre IP.

WLAN: red de área local inalámbrica.

Wi-Fi: conjunto de estándares para redes inalámbricas basado en las especificaciones IEEE 802.11.

ABREVIACIONES

ARQ	Repetición Automática de Petición.
ASAP	Perfil de Acceso Específico para una Aplicación
CI	Interfaz para la coexistencia.
C/L	Modo no Orientado a la Conexión.
C/O	Modo Orientado a la Conexión.
CCFP	Central de Control de la Parte Fija.
CI	Interfaz Común (estándar).
CRC	Comprobación de Redundancia Cíclica.
DECT	Telecomunicaciones inalámbricas mejoradas digitales.
DLC	Control del Enlace de Datos.
DNW	Red DECT.
DSP	Perfil para el Servicio de Datos
DPRS	Servicio de paquetes vía radio.
ECTEL	Asociación de Fabricantes de Sistemas de Telecomunicación.
EFREL	Frame Relay Mejorado.
ETSI	Instituto Europeo de Normas de Telecomunicaciones.
FP	Parte Fija.
FREL	Servicio Frame Relay.
FT	Terminación Fija.
GAP	Perfil de Acceso Genérico.
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos
GBN	Go-Back-N.
IP	Protocolo de Internet.
IPv4	Protocolo de Internet Versión 4.
IPv6	Protocolo de Internet Versión 6.

ISDN	Red Digital de Servicios Integrados.
IWF	Funciones de Interactuación.
IWU	Unidad de Interactuación.
LCE	Entidad de Control del Enlace.
LRMS	Servicio de Mensajes de Baja Velocidad.
IDBS	Servicio de Datos de Portadora Isosíncrona.
MAC	Capa de Control de Acceso al Medio.
MBC	Control Multi-Portadora.
NACK	Reconocimiento Negativo.
PA	Aplicación Portátil.
PAP	Perfil de Acceso Público.
PHL	Capa Física.
Plano-C	Plano de Control.
Plano-U	Plano de Usuario.
PP	Parte Portátil.
PT	Terminación Portátil.
RF	Frecuencia de Radio.
RFP	Parte Fija de Radio.
RSP	Protocolo de Repetición Selectiva
SAP	Punto de Acceso al Servicio.
SDU	Unidad de Datos del Servicio.
TDD	Duplexación por División en el Tiempo.
TDMA	Acceso Múltiple por División en el Tiempo.
WLAN	Red de Área Local Inalámbrica.

Capítulo 1

Introducción

1.1 Antecedentes

Actualmente se estima que alrededor del 80% de las personas que se desplazan por su lugar de trabajo pueden necesitar potencialmente emplear terminales móviles para comunicaciones de voz. El uso cada vez mayor de las redes LAN inalámbricas de datos basadas en la tecnología Wi-Fi (WLANs) dentro de los entornos de trabajo ha dado la idea a ciertos fabricantes que implementan los estándares Wi-Fi de incorporar terminales móviles que usen la tecnología Wi-Fi para proporcionar servicios de voz sobre IP (VoIP) en la red de datos LAN (VoWLAN). Todo esto puede llevar a pensar que los sistemas DECT usados para proporcionar tradicionalmente servicios de voz usando centralitas privadas inalámbricas están condenados a desaparecer. Los fabricantes de los sistemas DECT han reaccionado ante esta situación y han decidido incorporar nuevas características para proporcionar servicios de datos sobre DECT. Podemos por tanto identificar 2 tendencias enfrentadas para proporcionar soluciones integradas de voz y datos en un entorno inalámbrico:

- Incorporación de nuevas características a los estándares Wi-Fi (pensados para transportar en un principio tráfico de datos) para que puedan soportar tráfico de voz.
- Incorporación de nuevas características al estándar DECT (pensada en un principio para soportar tráfico de voz) para transportar paquetes de datos.

1.1.1 DECT frente a VoWLAN

Los sistemas DECT y las redes de datos inalámbricas (WLAN) se presentan como dos opciones para la integración de los servicios de voz y datos. A continuación se muestra una comparación entre ambas soluciones para proporcionar servicios integrados de voz y datos:

- **Ancho de Banda:**

Desde el punto de vista del transporte de paquetes de datos, la tecnología DECT no puede comparar sus velocidades con las proporcionadas por Wi-Fi, aunque la incorporación de nuevas características al estándar DECT como el uso de los nuevos niveles de modulación y nuevos formatos de los paquetes permite obtener una velocidad de hasta 2Mbps en una portadora en frecuencia.

- **Interoperabilidad:**

Los estándares Wi-Fi (IEEE 802.11) fueron pensados en un principio para transportar tráfico de voz. Entre los estándares Wi-Fi, los estándares 802.11b y 802.11g no soportan aspectos tales como la seguridad, la calidad de servicio para la voz o el handover de las llamadas. Por este motivo los fabricantes que desean implementar estas características para proporcionar un servicio de voz tienen que implementar soluciones propietarias, dificultando la interoperación entre equipos de diferentes fabricantes. Por otro lado el estándar DECT

incorpora un perfil conocido como perfil de acceso genérico (GAP) que asegura la compatibilidad total entre equipos de diferentes fabricantes.

- **Interferencias:**

La banda de los 2,4 Ghz empleada por 802.11b y 802.11g, es usada por otras redes inalámbricas (como por ejemplo las redes con Bluetooth), por radio aficionados e incluso por los hornos microondas. Por este motivo el tráfico de voz está expuesto a múltiples fuentes de interferencia lo que puede provocar una degradación de la calidad de servicio. DECT por su parte opera en una frecuencia protegida (1880-1900 Mhz) por lo que no sufre interferencias procedentes de otras tecnologías que operan en la misma banda.

- **Handover de llamadas entres estaciones base:**

Mientras que los sistemas de telefonía DECT soportan el handover entre estaciones base en un tiempo no superior a los 10 mseg y de manera completamente transparente para el usuario, esta característica no está estandarizada para 802.11. En la tecnología Wi-Fi el cambio de estación base provoca una interrupción temporal del tráfico de paquetes de aproximadamente 50 mseg. En las redes Wi-Fi la interrupción del tráfico antes de volver a establecer la conexión con la nueva estación base no resulta muy perjudicial para el tráfico de datos pero no así para el tráfico de voz. Además el nuevo estándar 802.11i que incorpora características de seguridad hará que el tiempo que dura esta interrupción aumente a aproximadamente 70 mseg (tiempo de handover que se considera totalmente inadecuado para las comunicaciones de voz) lo que provocará que el usuario escuche una interrupción en la conversación. Este problema del tiempo de retardo está siendo solucionado por muchos fabricantes de tecnología Wi-Fi empleando soluciones propietarias.

- **Cobertura de radio para los usuarios de voz:**

Las aplicaciones WLAN se usan normalmente en los espacios de trabajo y no se suelen emplear en lugares tales como pasillos, baños, escaleras, etc. Esto no se puede afirmar para el caso de los usuarios de servicios de voz. La solución para el caso de Wi-Fi es emplear más puntos de acceso. El problema que puede plantear es la situación de los puntos de acceso ya que si los puntos están muy cercanos pueden interferirse entre sí, mientras que si los puntos se alejan demasiado, puede haber problemas de cobertura y de handover.

- **Capacidad de los puntos de acceso para usuarios de voz:**

Los puntos de acceso para WLAN soportan en la práctica hasta 3 conversaciones de voz mientras que los puntos de acceso de los sistemas DECT pueden soportar entre 4 y 8 conversaciones simultáneas.

- **Seguridad:**

Los mecanismos de protección usados por los terminales Wi-Fi como el protocolo de encriptación inalámbrico (WEP) y la autenticación con la dirección MAC son insuficientes. En la actualidad el organismo encargado de desarrollar ellos estándares de Wi-Fi (la IEEE) ha desarrollado el estándar 802.11i para mejorar la seguridad. No obstante ya se ha comentado antes dicho estándar introduce un retardo al realizar el cambio de estación que no

resulta adecuado para las conversaciones de voz. El estándar DECT por su parte incluye un mecanismo de autenticación e identificación de 128 bits y un procedimiento de encriptación que usa claves de 64 bits para el cifrado. Estos mecanismos resultan suficientes para una adecuada protección de la información.

- **Funcionamiento de los terminales:**

Los terminales Wi-Fi permiten 4 horas de conversación y 60 horas en espera, mientras que los terminales DECT permiten 12 horas de conversación y más de 100 horas en espera.

A la vista de todas las características anteriores, se puede llegar a la conclusión de que un sistema DECT puede ser una buena solución para la integración de los servicios de voz y datos en aquellos entornos en los que no se necesitan anchos de banda excesivos y donde se da importancia a la calidad del tráfico de voz. Así pues DECT resulta un estándar flexible que permite definir una amplia variedad de conexiones para proporcionar servicios de transporte de datos así como servicios multimedia.

1.2 Objetivos del proyecto

Los principales fabricantes de los sistemas DECT que se encargan de la elaboración de los documentos que conforman el estándar, han introducido recientemente modificaciones con la intención de conseguir una mejor prestación para los servicios de datos. Entre las modificaciones se pueden destacar las mejoras introducidas a nivel físico y a nivel de la capa de acceso como son el uso de nuevos niveles de modulación y nuevos formatos para los paquetes . A raíz de las mejoras introducidas en las capas inferiores del protocolo, a día de hoy han sido definidos nuevos servicios de datos a nivel de enlace, muchos de los cuales se encuentran bajo estudio. Con la definición de estos nuevos servicios de datos un gran número de nuevos documentos técnicos y perfiles están siendo generados para sustituir a los antiguos [3] o para aportar unas guías para la implementación práctica de los mismos. La documentación es muy extensa y ofrece múltiples soluciones para la integración de los servicios de datos, por lo que se ha considerado interesante realizar un proyecto de desarrollo en el que se acometa el diseño e implementación práctica de una de estas soluciones usando los nuevos procedimientos y características definidas en el estándar DECT. Los objetivos del proyecto son:

- 1) Analizar los documentos del estándar para proponer un escenario de referencia en donde se empleen las soluciones de DECT para la integración de los servicios de datos junto con los servicios tradicionales de voz.
- 2) Sintetizar los nuevos procedimientos y características de DECT que hacen falta para la implementación del escenario de referencia. Habrá que dar a conocer cuales son estos nuevos procedimientos y como se pueden usar.
- 3) Realizar una implementación práctica de las nuevas características y procedimientos definidos por el estándar DECT mediante el diseño y construcción de una herramienta software que simule el escenario de referencia. Las principales características del software a desarrollar son la escalabilidad, la flexibilidad y la portabilidad.

- 4) La integración de los nuevos servicios para datos proporcionados por DECT con los protocolos TCP e IP.
- 5) Realizar una evaluación de prestaciones de TCP/IP sobre los nuevos servicios de datos DECT mediante simulación.
- 6) Preparar el simulador como una herramienta docente para mostrar en detalle la interacción de los protocolos de un sistema DECT con los protocolos TCP e IP.

Para la construcción del simulador ha sido necesario realizar un análisis detallado de los documentos del estándar base DECT. A la hora de construir el simulador, se ha empleado para su programación el lenguaje C++ [25] junto con el entorno gráfico de desarrollo proporcionado por el software OMNET++[23] de libre distribución. Por último se ha buscado proporcionar una visión general y descriptiva de las nuevas características introducidas en el estándar DECT, debido a que actualmente, aparte de los estándares, la documentación al respecto y las publicaciones relacionadas con el tema son muy escasas.

1.3 Contenido del documento

Este documento se divide en 6 capítulos junto con el presente, en los cuales se trata de dar una visión del trabajo realizado para la realización del proyecto. Como complemento a los capítulos se incluye un índice con las definiciones y términos empleados en el documento. Se recomienda al lector que si ya dispone de nociones sobre el sistema DECT y sus últimas modificaciones, empiece la lectura del proyecto a partir del capítulo 3. Así mismo si el lector no está familiarizado con el uso de OMNET++, se recomienda que para documentarse acuda a la página del autor de OMNET++ [23], antes de proceder a la lectura del capítulo 4. Con objeto de facilitar la comprensión del documento se proporciona a continuación una breve descripción acerca de los capítulos que componen este documento:

Capítulo 2: “Características del sistema DECT”

En este capítulo se proporciona la información necesaria acerca del funcionamiento desde el punto de vista del plano de usuario cada una de las capas modificadas (PHL, MAC y DLC) de la pila de protocolos del sistema DECT y se incluye una introducción a perfil DPRS [15]. El objeto de estas descripciones es comprender el trabajo desarrollado en los capítulos dedicados a la implementación del simulador. No es el objeto del presente documento el incluir una descripción detallada de cada una de las capas, por lo que únicamente se han descrito las características y servicios que se consideran necesarios para la comprensión del presente documento, obviando aquellas características que se consideran lo suficientemente documentadas por publicaciones anteriores. No obstante y dado que mucho de las características y servicios aquí mencionados son de reciente modificación, no se han encontrado libros o publicaciones que se refieran a ellas (exceptuando los que se refieren a especificaciones obsoletas).

Capítulo 3: “Escenario de Referencia”

El objetivo de este capítulo es introducir el entorno para el que se va a construir el simulador, describiendo aspectos tales como: la configuración física, la arquitectura del protocolo y la relación entre la PDUs para cada uno de los niveles de la pila del protocolo.

Capítulo 4: “Descripción del simulador”

En este capítulo se describen los modelos empleados para implementar las capas y los procedimientos del protocolo. Con este objetivo se incluye una descripción de los módulos que componen el simulador así como de las funciones implementadas en cada uno de ellos.

Capítulo 5: “Manejo del simulador”

En este capítulo se proporciona la información necesaria para poder realizar simulaciones con el modelo que se ha construido. Se describen los parámetros que se pueden configurar del modelo y los parámetros que se pueden medir. También se proporciona información sobre el formato de los ficheros de salida de cada módulo que se emplean para validar el funcionamiento de los módulos.

Capítulo 6: “Evaluación de prestaciones mediante simulación”

El objetivo de este capítulo es describir los experimentos realizados para evaluar el rendimiento de los servicios que se han implementado en el simulador.

Capítulo 7: “Conclusiones y líneas futuras”

En este capítulo se incluye una visión general de los resultados a lo largo del desarrollo del simulador y se marcan las posibles directrices para simulaciones futuras.

A lo largo de este documento se hará uso de una gran cantidad de términos técnicos y abreviaciones, por lo que con objeto de facilitar su comprensión se evitará su uso siempre que sea posible. No obstante se proporciona un índice con las definiciones de los términos técnicos más empleados y otro índice para las abreviaciones.

Capítulo 2

Características del sistema DECT

2.1 Introducción

El estándar europeo DECT (Telecomunicaciones inalámbricas mejoradas digitales) define una tecnología de acceso radio para comunicaciones inalámbricas. Conceptualmente, da lugar a sistemas de comunicaciones de baja potencia sin hilos full-dúplex similares a los "celulares" que son ya ampliamente conocidos, estando la principal diferencia en que DECT está optimizado para coberturas locales o restringidas con alta densidad de tráfico. Esta característica lo hace ideal para su uso en entornos residenciales, empresariales y públicos. Al tratarse de sistemas de baja potencia las áreas de cobertura no superan normalmente los 200 metros, si bien es posible aumentar la cobertura empleando antenas especiales. Su equivalente en EE.UU. es PWT (Personal Wireless Telecommunications).

2.2 Infraestructura de un sistema DECT

En este punto se describen los elementos funcionales que configuran el sistema DECT. La parte inalámbrica de un sistema DECT está compuesto por dos extremos que son la parte fija (FP) y la parte portátil (PP) que están conectadas por un enlace vía radio. Estas dos partes se pueden ver más en detalle descomponiéndolas en partes más pequeñas como se muestra en la figura 2.1:

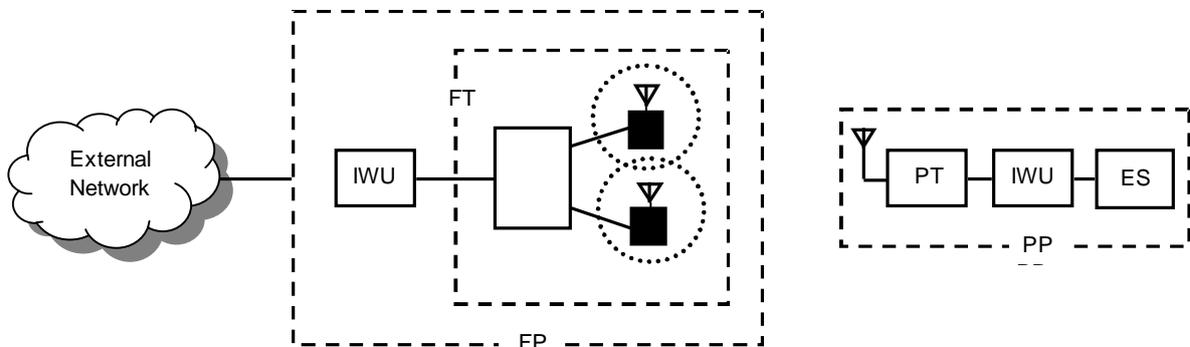


Figura 2. 1 Entidades funcionales

Los sistemas DECT se pueden conectar a redes globales como la red telefónica conmutada (RTC), la red digital de servicios integrados (RDSI), la red celular GSM o Internet. A continuación se explican más detalladamente todas y cada una de estos elementos funcionales que forman las partes de un sistema DECT.

2.2.1 La parte portátil (PP)

Esta componente del sistema se corresponde con el terminal móvil que está constituido por dos elementos:

1. La terminación portátil (PT), que proporciona todas las funcionalidades de acceso.

2. La aplicación portátil o sistema final (ES), proporciona el resto de funciones de la parte de la aplicación. Estas funciones por ejemplo los servicios proporcionados por un terminal.

Entre estos dos componentes debe existir una IWU (unidad de interacción) que se encarga de convertir los mensajes y señales usados en el sistema DECT en un formato compresible por la aplicación

2.2.2 La parte fija (FP)

Los elementos de una parte fija son:

1. Una o varias terminaciones fijas (FT) que proporcionan las funcionalidades de acceso del sistema. Cada terminación puede contener a su vez una o varias estaciones base (RFPs) que proporcionan el acceso vía radio. Cada estación base puede tener una o varias antenas.
2. Una red local que puede ser un conmutador, una PABX o una red LAN que se encarga de encaminar los datos entre los terminales y que puede actuar como IWU (unidad de interacción) para conectar la red a una red diferente global (RTC, RDSI, GSM,...) convirtiendo los mensajes y señales usados en la interfaz aérea en un formato compresible para la nueva red y viceversa. La red local debe contener una base de datos con información sobre la localización de los terminales para poder encaminar la información.

2.3 El estándar base DECT

A finales de 1987, ECTEL, la asociación de fabricantes de sistemas de telecomunicaciones de la unión europea comenzó a trabajar en una especificación común para la telefonía digital inalámbrica en el mercado europeo. De esta forma se crearon una serie de grupos de trabajo que asentaron los principios de DECT. En 1988 nace el instituto de estándares europeos (ETSI [1]). En 1989 la ETSI crea a partir del comité técnico de equipos de radio y sistemas un subcomité técnico en el que incorpora los grupos de trabajo de ECTEL de forma que se consigue que los fabricantes de equipos, los operadores y los organismos reguladores trabajen juntos en la definición y evolución de los estándares DECT.

En 1992 aparece la primera edición completa del estándar base DECT. Estas primeras especificaciones únicamente aclaraban la manera en que debe hacerse el acceso vía radio pero no consideraban características de interoperabilidad entre equipos de distintos fabricantes. Por este motivo a petición de la comisión europea a mediados de 1996 aparece la segunda edición de los estándares base DECT a los que se añade el concepto de GAP [9] (perfil de acceso genérico) para asegurar la interoperabilidad de los equipos para aplicaciones de voz de diferentes fabricantes.

De esta forma los principales objetivos del estándar se pueden resumir en tres requisitos fundamentales que deben ser de obligado cumplimiento a la hora de fabricar los equipos:

- Las estándar bases están obligados a proporcionar una interfaz para la coexistencia (CI) de forma que todas las aplicaciones puedan compartir el espectro sin interferirse mutuamente.

- Una oportunidad para la interoperación entre equipos de diferentes fabricantes.
- Especificar la interoperación con ciertas aplicaciones, lo que se consigue a través de la adición de estándares denominados perfiles de aplicación.

En la figura 2.2 se muestra el modelo de capas definidas en el estándar base para el sistema DECT:

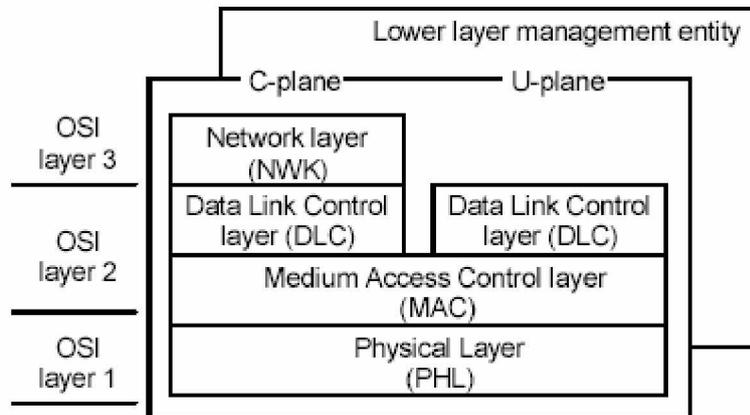


Figura 2. 2 Pila de protocolo del sistema DECT (extraída de ETSI EN 300 175-1 v1.7.1)

El modelo de capas de DECT distingue dos planos de operación en la capa DLC:

- El plano de control (plano-C): contiene todo los procedimientos del control del protocolo interno de un sistema DECT.
- El plano de usuario (plano-U): contiene todos los procedimientos que actúan directamente sobre los paquetes con los datos de usuario (control de usuario).

En nuestro simulador únicamente serán implementados los procedimientos de las capas que realizan alguna función en el plano de usuario, porque son los que actúan directamente sobre los datos. Los procedimientos internos de control de la pila de protocolos del sistema DECT no serán implementados, puesto que no son necesarios. Como la capa de red (NWK) únicamente realiza funciones de control, sus procedimientos no serán implementados.

Como se ha comentado previamente, el perfil GAP [9] fue uno de los primeros perfiles de aplicación y a partir de entonces se han ido añadiendo numerosos perfiles al estándar base para definir el empleo de DECT en diferentes aplicaciones.

2.4 Perfiles de servicios de datos (DSP)

Para dar una primera idea del uso de los sistemas DECT con aplicaciones de datos, se puede consultar el informe técnico "Descripción general del servicio de datos" [18], publicado por la ETSI en el año 2001. Los perfiles de servicios de datos (DSPs) de DECT hacen uso de los potentes mecanismos para la transmisión de datos definidos en los documentos que componen el estándar base DECT. Gracias a estos mecanismos, los DSPs en comparación con la transmisión de datos por modems (usando un canal de voz) proporcionan elevados "throughputs", bajas tasas de error de bit, mayor fiabilidad, uso mejorado del espectro y menor consumo de potencia.

Además la compatibilidad de los DSPs con el perfil GAP [9] asegura que los sistemas de voz y datos integrados pueden ser implementados eficientemente.

2.4.1 Objetivo del los DSPs

El objetivo de los DSPs de DECT es hacer posible la interoperación entre las FPs y las PPs para dar soporte a los tradicionales servicios de voz DECT junto con los nuevos servicios de datos. Para conseguirlo los DSPs definen de una manera clara y sin ambigüedades los procedimientos del protocolo, los mensajes y los elementos de información necesarios para proporcionar un servicio específico deseado. Se definen un número específico de servicios suficientes para cubrir una gran variedad de aplicaciones. En particular, los de los DSPs se pueden resumir en los siguientes puntos:

- Proporcionar niveles de funcionalidad apropiados en cada área de aplicación.
- Dar servicio al mayor número posible de aplicaciones usando el menor número de DSPs posibles.
- Separar en todo lo posible el desarrollo y evolución de cada parte de los perfiles correspondiente a un servicio o aplicación del resto de partes. Se pretende así conseguir que el desarrollo de los estándares responda de la manera más rápida a la evolución del mercado. sin que existan restricciones tecnológicas surgidas de la dependencia entre dichas partes.
- Permitir la implementación de múltiples servicios (incluyendo voz y datos) en el mismo equipamiento sin ningún tipo de conflicto o confusión entre ellos.

2.4.2 Descripción de los servicios

Los DSPs de DECT están organizados en un número de estándares separados, cada uno de los cuales define los aspectos necesarios para proporcionar un tipo o nivel de servicio. El resultado es un árbol familiar de estándares relacionados. En el siguiente apartado se hace una descripción resumida de cada perfil.

2.4.2.1 Tipos de servicios

En los últimos apartados se mencionan los perfiles implementados para proporcionar los servicios de datos. Estos perfiles son:

- **Servicio de paquetes vía radio (DPRS)** [15]. En este perfil se definen los requisitos para la transmisión de paquetes vía radio entre terminales portátiles y fijos. Se usa para los servicios: V.24, Ethernet, PP o IP interworking.
- **Servicio de mensajes de baja velocidad (LRMS)** [16]. En este servicio proporciona la transmisión de mensajes cortos e información de difusión.
- **Servicio de portadora de datos isosíncrona (IDBS)** [17]. Proporciona una conexión isosíncrona y transparente de flujos de datos síncronos optimizada para aplicaciones que requieren un flujo continuo como es el caso de las aplicaciones de video telefonía o video conferencia.

Por su importancia en la implementación del simulador, hay que destacar el perfil que define los requisitos generales de DECT para proporcionar servicios de transmisión de paquetes de datos vía radio. Dicho perfil se conoce con el nombre de DPRS. El perfil DPRS surge de la evolución natural del antiguo perfil (descatalogado por la ETSI) para adaptarse a las nuevas modificaciones introducidas en DECT con objeto de proporcionar los servicios demandados en la actualidad. El perfil DPRS se convierte en la base para otros perfiles que se emplean para aplicaciones específicas y que se conocen con el nombre de perfiles de acceso específico para aplicaciones (ASAPs). Entre los perfiles de acceso específico se pueden citar:

- **DPRS ASAP V.24**, que define un perfil de acceso específico para aplicaciones específicas de datos en empresas, pequeñas oficinas y hogares residenciales/privados. Para ello combina una selección de las interacciones con V.24 definidas en los anexos de DPRS para los servicios de datos.
- **DPRS ASAP Ethernet**, define el perfil de acceso específico para aplicaciones específicas de datos en empresas y residencias privadas. Para ello combina una selección de las interacciones con Ethernet definidas en los anexos de DPRS para los servicios de datos.
- **DPRS ASAP DMAP**, define un perfil dirigido principalmente para las residencias privadas combinando una selección de los servicios de datos DECT extraídos de DPRS con los servicios de voz ofrecidos por el perfil GAP con el objetivo de proporcionar un perfil de acceso multimedia.

Todos estos perfiles de acceso específico pueden consultarse en [15]. El acceso a redes IP está también contemplado en los anexos del perfil DPRS donde se proporciona una solución para la integración de los servicios de un sistema DECT con los servicios proporcionados por aplicaciones que emplean en sus capas superiores la pila de protocolos TCP/IP. En esta solución el sistema DECT se puede ver como un sistema de acceso que proporciona un mecanismo fiable para el transporte de datos de las capas superiores por la interfaz radio. A su vez el sistema DECT puede seguir proporcionando el servicio tradicional de voz por lo que esta solución se convierte en una solución para la integración de los servicios de voz y datos dentro de un mismo escenario. En el presente proyecto se usará esta solución de integración de servicios entre protocolos de diferentes sistemas para configurar el escenario de referencia tal como se verá en el capítulo 3.

Como se ha comentado antes todos estos documentos basan su contenido en el perfil DPRS que su vez surge de las modificaciones introducidas recientemente en los documentos del estándar base para poder proporcionar un mejor servicio de datos. Fuera de los documentos que conforman el estándar base, no se ha encontrado información sobre las modificaciones introducidas y sobre el perfil DPRS en ningún documento o libro consultado para la realización del proyecto. Por estos motivos se ha considerado de vital importancia el reflejar en este capítulo para cada capa del protocolo que sea necesario las modificaciones introducidas así como aquellos aspectos que se consideran básicos para la comprensión del diseño y funcionamiento del simulador. La información descrita en los siguientes apartados será referida cuando se describa el simulador. Además se incluye un apartado dedicado a los requisitos del perfil DPRS ya que se considera la base para el desarrollo de toda la documentación técnica consultada y será necesario referirse a él para explicar los aspectos relacionados con la configuración del escenario de simulación.

2.5 La capa Física (PHL)

La capa física divide el espectro radioeléctrico en canales físicos. Esta división tiene lugar en dos dimensiones: la frecuencia y el tiempo. El sistema usado es el protocolo de Acceso por División en el Tiempo (TDMA) sobre múltiples portadoras de frecuencia.

Normalmente se proporcionan 10 portadoras en frecuencia dentro del ancho de banda del sistema DECT (El ancho de banda típico asignado a DECT va de los 1880 Mhz a los 1900 Mhz). El estándar [5] ofrece además la posibilidad de ampliar el ancho de banda para cubrir usos futuros. Cada una de las portadoras en frecuencia de la estructura TDMA define una trama de 10 mseg dividida en 24 slots, donde cada slot puede ser usado para transmitir un paquete físico de datos. Cada uno de los paquetes transmitidos contiene un campo para la sincronización, junto con un campo para la información de control, la información de servicio y para el control de errores. Además el espectro radioeléctrico se divide espacialmente en celdas, donde los mismos canales físicos pueden ser re-usados en diferentes localizaciones geográficas.

2.5.1 Estructura de la TDMA

La estructura de la TDMA se repite en tramas de 11520 símbolos cada 10 mseg resultando una velocidad máxima de 1152 Ksímbolos/s. En función del tamaño del paquete físico, variará el formato de los slots de la trama TDMA y con ello la estructura de la trama. Los formatos definidos en el estándar son:

- Formato de mitad de slot.
- Formato de slot completo.
- Formato de slot doble.

En el actual documento únicamente se describirán los formatos que han sido considerados en el diseño del simulador, para poder referirse a ellos posteriormente. Como el tamaño en bits de los paquetes variará en función del nivel de modulación seleccionado, todos los tamaños de los campos relacionados con los paquetes se expresarán por defecto en símbolos, pudiendo usar el término de bits cuando ambos términos sean equivalentes (como ocurre en el caso de usar una modulación de 2 niveles). En los formatos que aparecen en las siguientes figuras el tamaño se expresa en bits porque se considera que se emplean dos niveles de modulación.

Formato de slot completo: En cada slot se transmite un paquete físico de formato P32, explicado en el apartado 2.5.2.

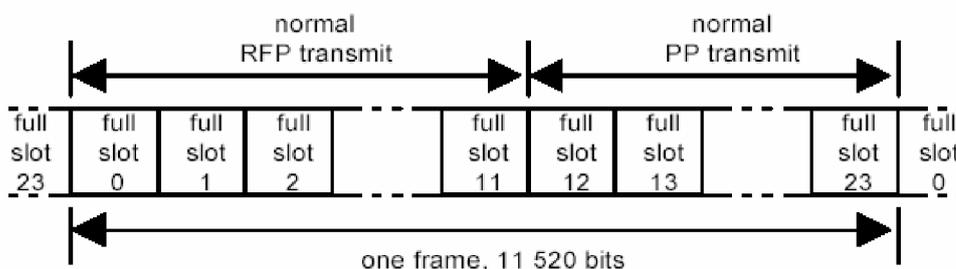


Figura 2. 3 Formato de slot completo (extraída de ETSI EN 300 175-1 v1.7.1)

Formato de slot doble: El tamaño de cada slot es igual al de dos slots completos. En cada slot doble se transmite un paquete físico de formato P80, explicado en el apartado 2.5.2.

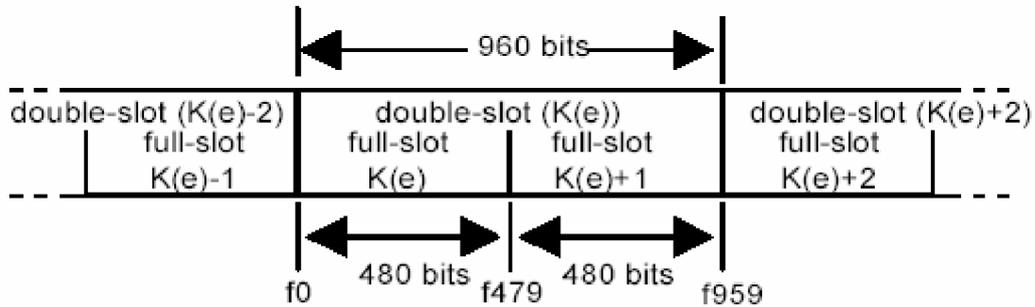


Figura 2. 4 Formato de slot doble (extraída de ETSI EN 300 175-1 v1.7.1)

2.5.2 Formatos físicos de los paquetes

Los formatos de los paquetes físicos que son usados para transmitir los datos dentro de los dominios de la frecuencia, tiempo y espacio son:

- Paquete corto P00
- Paquete básico P32
- Paquete de baja capacidad P08j
- Paquete de alta capacidad P80

En el actual documento únicamente se mostrarán los formatos para los paquetes más usuales empleados por las conexiones para transportar datos, los cuales son P32 y P80.

Paquete básico P32. Consta de 420 o 424 símbolos y es usado por las conexiones para transportar datos.

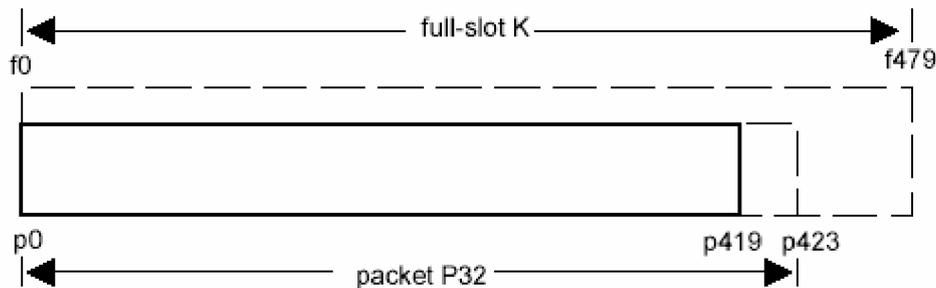


Figura 2. 5 Paquete básico P32 (extraída de ETSI EN 300 175-1 v1.7.1)

Como cada paquete ocupa un único slot, el número de slots que formarán la trama TDMA será de 24.

Paquete de gran capacidad P80. Consta de 900 o 904 símbolos y emplea el formato de doble slot por lo que el número de slots que forman la trama TDMA será de 12.

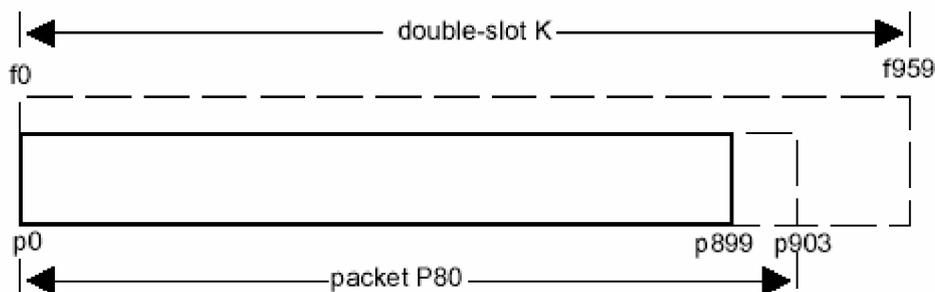


Figura 2. 6 Paquete P80 (extraída de ETSI EN 300 175-1 v1.7.1)

En función del nivel de modulación empleado variará el número de bits por símbolo. Los niveles de modulación que se van a considerar para la implementación de nuestro simulador son: 2, 4 y 8. La implementación de los procedimientos de modulación no ha sido necesaria en nuestro simulador ya que los valores necesarios han sido tomados de las tablas del estándar donde se proporcionaban directamente los resultados que nos interesan.

2.6 La capa de acceso al medio (MAC)

La capa MAC realiza dos funciones principales. En primer lugar se encarga de la selección de los canales físicos así como del establecimiento y liberación de conexiones sobre estos canales. La segunda función incluye la multiplexación y demultiplexación de la información de control, junto con la información de las capas superiores y la información de control de errores, dentro de paquetes físicos que serán transmitidos en los slots. Estas funciones son usadas para proporcionar tres tipos de servicios independientes:

- Servicios de difusión.
- Servicios orientados a la conexión (C/O).
- Servicios no orientados a la conexión (C/L).

En este documento se describirán únicamente los servicios de transmisión de datos orientados a la conexión.

2.6.1 Formato de los paquetes MAC

En este apartado se describirá la estructura física del paquete MAC, así como la disposición de la información en el campo de datos del paquete que depende del formato de slot empleado, del nivel de modulación y del modo de protección de información seleccionados. Si no se dice lo contrario, la información que aparezca reflejada en tablas e ilustraciones se corresponderá con el formato de slot full 32 que es el implementado para el simulador.

2.6.1.1 Formato de la cabecera (campo A)

La cabecera contiene principalmente información de control del protocolo. Esta información está protegida por un CRC de 16 símbolos. En la figura 2.7 extraída del estándar [6] se muestra la disposición de la información en la cabecera.



figura 2. 7 Campos de la cabecera (extraída de ETSI EN 300 175-3 v1.8)

2.6.1.2 Formato del campo de datos (campo B)

El campo B es el campo de datos del paquete y presenta 4 formatos distintos en función del modo de protección de la información.

Formato desprotegido: carece de cualquier tipo de protección para el campo de datos, por lo que todos los símbolos de este campo su pueden usar enteramente para almacenar información útil para el receptor. Tiene el gran inconveniente de que la protección de estos datos es nula.

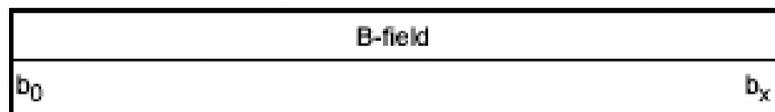


Figura 2. 8 Formato desprotegido (extraída de ETSI EN 300 175-3 v1.8)

Formato de protección con múltiples subcampos (IP): El campo de datos se divide en una serie de subcampos (B_x) de datos de longitud fija cada uno protegido por un CRC (RA) de 16 bits. La cantidad de información útil a introducir se ve reducida por el uso de CRCs, pero a cambio este formato ofrece un gran nivel de protección.

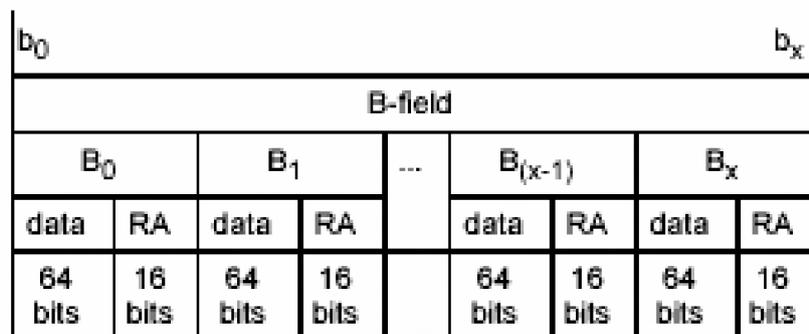


Figura 2. 9 Protección por múltiples subcampos (IP) (extraída de ETSI EN 300 175-3 v1.8)

Formato de protección con un solo campo (IPQ): aparece incluido por primera vez en la V1.7.1 del estándar [6] del año 2002. Este formato usa un campo con información útil protegido por un único CRC (RB0) de 16 ó 32 bits (en función del nivel de modulación) con la idea de aumentar la información útil que se puede introducir en el campo B.

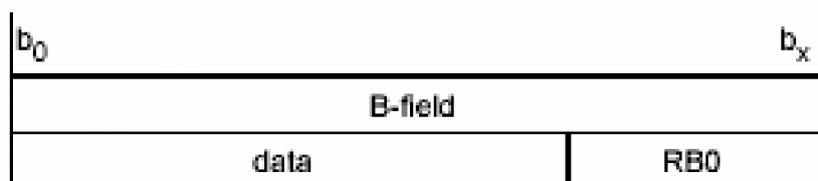


Figura 2. 10 Protección de un solo campo (extraída de ETSI EN 300 175-3 v1.8)

Formato codificado y protegido: al igual que el formato de protección de un solo campo aparece incluido también por primera vez en la V1.7.1 del estándar [6] del año 2002. Este formato usa el principio del turbo código el cual permite introducir mayor cantidad de información útil en el campo de datos. El funcionamiento de dicho principio se encuentra fuera del alcance del presente documento. Si el lector quiere obtener más información sobre este principio puede encontrarla en el anexo I que aparece por primera vez en la V1.7.1 del estándar [6].

El uso de nuevos niveles de modulación a partir de la V1.7.1 del estándar [6] ha permitido aumentar el número de bits útiles por símbolo de información útil que será incluido en el campo de datos. Así pues podemos hablar de distintos tamaños para los campos de datos para cada uno de los 4 formatos. Estos tamaños están recogidos para cada formato en una tabla (una para cada formato de slot) incluida en el estándar [6] partir de su versión 1.7.1. En el presente documento únicamente se mostrarán las dos tablas para el formato de slot full 32 que han sido consultadas para la implementación del simulador y que se corresponden con los formatos de protección de múltiples subcampos y el formato de protección de un solo subcampo respectivamente. Aclarar que según el estándar, para los niveles 2 4 y 8 que se van a usar en nuestro simulador, el nivel de modulación de la cabecera se mantiene siempre en dos niveles para ambos formatos, independientemente del nivel de modulación del campo de datos.

Nivel de modulación	Tamaño de la cabecera(bits)	Tamaño del campo de datos(bits)	Número de Subcampos
2	104	320	4
4	104	640	8
8	104	960	12

Tabla 2. 1 Formato de los paquetes para el modo deprotección de múltiples subcampos

Vemos como en la tabla 2.1 a medida que aumenta el nivel de modulación aumenta el número de bits por símbolo en el campo de datos y con ello el número de subcampos.

Nivel de modulación	Tamaño de la cabecera(bits)	Tamaño del campo de datos(bits)	Tamaño del CRC(bits)
2	104	304	16
4	104	608	32
8	104	928	32

Tabla 2. 2 Formato de los paquetes para el modo de protección de un solo campo

Para el formato de protección de un solo campo (tabla 2.2) a medida que aumenta el número de niveles de modulación aumenta también el número de bits por símbolo en

el campo de datos, siendo el número de bits del campo de CRC el mismo (32 bits) para los niveles 4 y 8. Como se ha comentado en el apartado 2.5.2, serán los valores de estas tablas los que se usarán a la hora de construir los paquetes de datos de MAC en nuestro simulador.

2.6.2 Portadoras lógicas MAC (Bearers)

La capa MAC crea portadoras sobre las que transmite los paquetes físicos. No hay que confundir el concepto de portadoras MAC con el de portadoras en frecuencia. Las portadoras MAC son elementos creados por las entidades de control de la capa MAC de acuerdo con un determinado servicio para transportar los paquetes. De ahora en adelante se usará el término de portadora para referirse a una portadora creada por la capa MAC para proporcionar un servicio sobre un slot de una TDMA, hablando de portadoras en frecuencia cuando sea necesario diferenciarlas. Hay definidos cuatro tipos de portadoras MAC:

1. **La portadora simplex:** que se define como la capacidad de transporte obtenida cogiendo una ranura temporal (slot) de cada trama DECT (10 msec). Cada portadora puede ir sólo en un sentido de la comunicación y puede ser descendente (de la parte fija a la parte portátil) o ascendente (de la parte portátil a la parte fija). Hay dos tipos de portadoras simples:
 - Las portadoras cortas que sólo contienen la cabecera (campo A).
 - Las portadoras largas que contienen la cabecera (campo A) y el campo de datos (campo B).
2. **La portadora dúplex:** que está formada por dos portadoras simplex que transmiten en sentidos opuestos. Estas portadoras usarán la misma portadora de frecuencia y estarán separadas 12 ranuras temporales (la mitad de una trama DECT)
3. **La portadora doble simplex:** que está formada por dos portadoras simplex operando en la misma dirección usando la misma portadora de frecuencia y separadas 12 ranuras temporales.
4. **La portadora doble dúplex:** que está formada por dos portadoras dúplex asociada a la misma conexión MAC.

En la figura 2.11 se muestra un ejemplo en el que aparece una portadora doble dúplex (slots 10 y 22) y una portadora doble simplex (slots 5 y 17) que son el tipo de portadoras lógicas se han usado en nuestro simulador para diseñar las conexiones asimétricas.

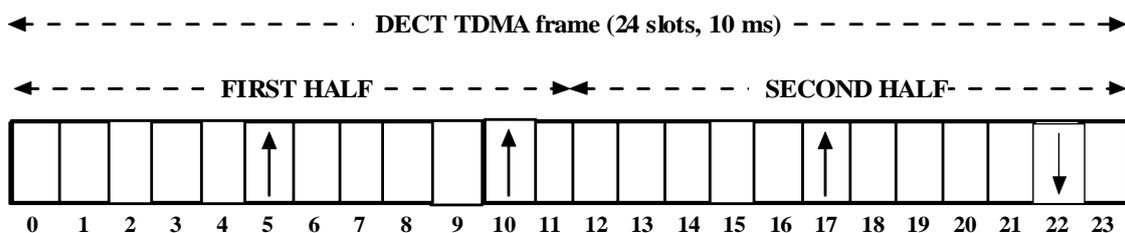


Figura 2. 11 Ejemplo de configuración de portadoras con 1 portadora doble simplex y 1 portadora doble dúplex

2.6.3 Portadoras de tráfico

Puede ser una portadora dúplex, doble simplex o doble dúplex que contiene los datos proporcionados por las capas superiores. Cada portadora de tráfico es creada y controlada por una entidad de la capa MAC

2.6.4 Servicios orientados a la conexión

Cada entidad de la capa MAC crea una conexión MAC, y proporciona un servicio independiente a la capa superior (DLC). Cada conexión MAC puede usar los servicios de una o más portadoras. Una conexión de una sola portadora usará sólo una portadora dúplex mientras que una conexión multiportadora usará una portadora dúplex más una o varias portadoras dúplex o simplex.

2.6.5 Conexiones simétricas y asimétricas

Los servicios orientados a la conexión se dividen en las siguientes categorías:

- **Conexiones simétricas:** tienen el mismo número de portadoras simplex en ambas direcciones de transmisión.
- **Conexiones asimétricas:** tienen un número diferente de portadoras con tráfico de datos en cada dirección de transmisión. Normalmente hay una o dos portadoras en la dirección opuesta por lo que el ancho de banda es menor que en la dirección principal.
- **Conexiones multiportadora:** son conexiones asimétricas o asimétricas que emplean más de una portadora lógica para una dirección de la comunicación.

En todas las conexiones la capa DLC proporciona al nivel MAC el número óptimo de portadoras que debe establecer así como el mínimo número necesario. Cuando se ha establecido la conexión, el nivel MAC informa a la capa DLC del número de portadoras que han sido establecidas. Este número tiene que ser al menos igual al número mínimo indicado por la DLC para que la conexión sea correcta.

2.6.5.1 Tipos de servicios y reglas de las conexiones simétricas

Se distinguen 5 tipos de servicios en función de la protección del canal I y del caudal:

- **Tipo 1:** IN_retardo_mínimo: protección limitada de errores, mínimo retardo y caudal fijo.
- **Tipo 2:** IN_retardo_normal: protección limitada de errores, retardo normal y caudal fijo.
- **Tipo 3:** IP_detección_error: detección de errores y caudal fijo.
- **Tipo 4:** IP_corrección_error: corrección de errores y caudal variable.
- **Tipo 5:** IP_codificado_protegido: emplea principio del turbocódigo

Cuando el servicio 3 usa el formato de protección de un subcampo del campo B se llama IPQ_detección_error. En el caso del servicio 4 se llama IPQ_corrección_error.

Los parámetros más importantes a configurar para estos servicios se pueden consultar en las tablas del estándar [6].

2.6.5.2 Tipos de servicios y reglas de las conexiones asimétricas

Se distinguen 4 tipos de servicios en función de la protección del canal de información y del caudal:

- **Tipo 6:** IN_retardo_normal: protección limitada de errores, retardo normal y caudal fijo.
- **Tipo 7:** IP_detección_error: detección de errores y caudal fijo.
- **Tipo 8:** IP_corrección_error: corrección de errores y caudal variable.
- **Tipo 9:** IP_codificado_protegido: emplea principio del turbocódigo.

Cuando el servicio 7 usa el formato de protección de un subcampo del campo de datos (campo B) se llama IPQ_detección_error. En el caso del servicio 8 se llama IPQ_corrección_error.

Las reglas para una conexión asimétrica son:

1. Las portadoras simplex se asignan siempre en pares separadas 12 ranuras temporales (portadoras dobles simplex).
2. Hay k portadoras simplex en la dirección principal y $m+n$ portadoras simplex en la dirección opuesta y se debe cumplir que $k \geq m+n \geq 1$.
3. Las portadoras n son llamadas portadoras especiales y son usadas para transportar reconocimientos (por ejemplo los ACKs para el control de flujo).
4. Las portadoras m tienen el mismo formato que las portadoras k .

2.6.5.3 Procedimiento para la retransmisión de datos

En este apartado se describirá el procedimiento de retransmisión automática que emplea la capa MAC para el caso del servicio de corrección de errores. Dicho procedimiento se conoce con el nombre de MOD-2. El procedimiento MOD-2 usa dos bits de reconocimiento llamados Q2 y BCK para el mecanismo de control de flujo y retransmisión de los paquetes de datos.

Uso del bit Q2:

Cuando el extremo receptor recibe un paquete con la cabecera correcta pone el bit Q2 a "1". Si por el contrario hay un fallo en la cabecera el bit Q2 se pone a "0". Cuando el receptor envíe un paquete al extremo transmisor incluirá este bit Q2. De esta forma si el transmisor recibe un paquete con el bit Q2 a "0" sabrá que el paquete llegó con errores al receptor. En el caso de que el bit Q2 valga "1" sabrá que al receptor le llegó el paquete con éxito. La actuación del transmisor una vez que tiene el bit Q2 variará en función del servicio a proporcionar por la capa MAC. Para el caso de que el servicio sea de corrección de errores:

1. Si Q2 vale "0" se realizará una retransmisión automática del paquete.
2. Si Q2 vale "1" se mirarán los subcampos B del paquete para el caso de conexiones asimétricas.

El esquema de retransmisión MOD-2 de un paquete será detenido para cualquier paquete que exceda su tiempo máximo de vida (expresado en tramas TDMA). La detención de este esquema tiene como consecuencia un salto en el transmisor, cuyo procedimiento se explicará más adelante.

Uso del bit BCK:

Este bit es usado por el receptor para informar al transmisor del siguiente número de secuencia del paquete que se espera el cual se encuentra en la cabecera del paquete recibido por el receptor. El valor a asignar a BCK puede variar en función del procedimiento de salto.

Situación de los bits Q2 y BCK:

La situación de estos bits, varía en función del tipo de portadora lógica.

- Para el caso de las portadoras dúplex estos bits están situados en la cabecera (campo A) del paquete de cada portadora en la dirección inversa.
- En el caso de las portadoras dobles simplex los bits Q2 y BCK para cada portadora lógica simplex van incluidos en un subcampo de datos de una de las portadoras en la dirección inversa.

Así pues para el caso de las conexiones asimétricas, las portadoras "n", contienen en al menos uno de sus subcampos de datos la información de control de Q2 y BCK para cada una de las portadoras "k" en la dirección principal, resultando un total de "2k" bits por subcampo.

Procedimiento de Salto:

Se entiende que existe un salto en el transmisor cuando se asigna un número de secuencia a un paquete que no es el que se esperaba según el procedimiento normal. Veamos como se desarrolla este procedimiento en cada uno de los extremos de la comunicación.

A) Receptor:

El receptor deberá detectar este salto y actuar en consecuencia. Así pues el receptor dispone de las siguientes variables de estado:

- **LAST-BCK:** "número de secuencia del paquete que fue incluido en el último BCK para indicar el siguiente paquete que se esperaba."
- **THIS-PKT:** "número de secuencia del paquete que acaba de recibir el receptor en la actual ráfaga."
- **LAST-PKT:** "número de secuencia del paquete que recibió el receptor en la última ráfaga."

El receptor después de recibir un paquete evaluará las variables de estado según la tabla 2.3 para saber como actuó el transmisor. A continuación comprueba los CRC de los campos de datos y actúa como se indica en la tabla 2.4.

	THIS-PKT=LAST-PKT	THIS-PKT=LAST-BCK	How transmitter acted
(a)	Yes	Yes	Retransmit
(b)	No	Yes	normal advance
(c)	Yes	No	unnecessary retransmit
(d)	No	No	Jump
(e)	Unknown	Yes	Either retransmit or advance
(f)	Unknown	No	Either retransmit or jump
(g)	Unknown	Unknown	Unknown

NOTE: The states (e), (f) happens when LAST-PKT is unknown and the state (g) when THIS-PKT is unknown.

Tabla 2. 3 Acción realizada por el transmisor (extraída de ETSI EN 300 175-3 v1.8)

Table 78 state	Rb-CRC result	how Receiver shall act
(g)	Not considered	NEXT-BCK = LAST-BCK
(a), (b), (e)	Rb-CRCs passed	NEXT-BCK = LAST-BCK + 1 (MOD 2)
(a), (b), (e)	Rb-CRCs failed	NEXT-BCK = LAST-BCK
(c)	Rb-CRCs passed	NEXT-BCK = LAST-BCK; RX still requires the same I_P packet
(c)	Rb-CRCs failed	NEXT-BCK = LAST-BCK; RX still requires the same I_P packet
(d), (f)	Rb-CRCs passed	NEXT-BCK = LAST-BCK
(d), (f)	Rb-CRCs failed	NEXT-BCK = LAST-BCK + 1 (MOD 2)

NOTE: In cases (e) and (f), the receiver can not be sure what was the action in the TX side. In these cases the packet should be delivered to the DLC layer.

Tabla 2. 4 Acción del receptor (extraída de ETSI EN 300 175-3 v1.8)

En el procedimiento normal cuando un paquete llega con la cabecera correcta al receptor, entonces éste comprueba mirando su número de secuencia si se trata del paquete que se esperaba. Si es el que se esperaba entonces se comprueban los CRCs de los campos de datos. Si todos los CRC son correctos, entonces se indica en BCK el número de secuencia del siguiente paquete que se espera, y se envía en la primera portadora que vaya en dirección al transmisor. Si por el contrario hay algún fallo en el campo de datos entonces se sigue pidiendo el mismo paquete.

Cuando el transmisor salta se deberá a que el paquete ha alcanzado su máximo tiempo de vida por lo que será descartado y se transmitirá un nuevo paquete con un nuevo número de secuencia. Este salto provocará la situación (d) siendo detectada por el receptor el cual actuará como muestra las filas 6 ó 7 de la tabla 2.4 en función de que el CRC sea correcto o no respectivamente.

B) Transmisor:

Se define un variable LTIP que almacena el número de secuencia del último paquete transmitido. Las dos variables de estado en el transmisor son:

- **ACKN:** “cuando vale 1 reconoce que el receptor recibió el paquete con la cabecera correcta”.
- **LAST-BCK:** “número de secuencia del último paquete solicitado por el receptor.”

Para las portadoras dúplex un paquete es reconocido cuando la cabecera es correcta y el bit Q2 vale 1. Para las portadoras simplex un paquete es reconocido cuando la cabecera es correcta y el bit Q1 vale 1. Cuando el transmisor recibe un paquete del receptor actúa de acuerdo con la tabla 2.5.

	ACKN	LTI _p = LAST-BCK	How transmitter shall act
(a)	Yes	No	normal advance (or retransmit)
(b)	Yes	Yes	retransmit or jump
(c)	No	No	retransmit or jump
(d)	No	Yes	retransmit or jump

Tabla 2. 5 Acción a realizar por el transmisor (extraída de ETSI EN 300 175-3 v1.8)

2.7 Introducción a la capa DLC

La capa de enlace se encarga de proporcionar enlaces de datos fiables para las capas superiores. Los esfuerzos realizados por MAC consiguen eliminar muchas de las imperfecciones de las transmisiones vía radio, por lo que la capa DLC fue diseñada para trabajar de manera complementaria a la capa inferior MAC para conseguir unos niveles altos de fiabilidad. El plano de usuario (plano-U) de DLC ofrece una serie de servicios a los paquetes de datos (SDUs) que recibe de las capas superiores. Estos servicios se nombran colectivamente LUX, con los miembros individuales de la familia que son nombrados LU1, LU2, LU3, etc. Cada uno de estos servicios funcionará totalmente de manera independiente, y se podrá acceder a ellos a través de su punto de acceso correspondiente. Dependiendo de los requisitos del servicio de las capas más altas las transmisiones de datos proporcionadas por DLC pueden utilizar una o múltiples conexiones MAC.

2.7.1 Servicios y entidades del plano de usuario (plano-U)

En el presente proyecto nos centraremos en el plano de usuario que es el que interesa a la hora de implementar el simulador como se ha comentado en el apartado 2.3. Los servicios que la capa de enlace ofrece a las capas superiores son todos opcionales en el sentido ya que cada servicio corresponde a unas necesidades particulares. Cuando las capas superiores solicitan un determinado servicio a la capa de enlace, se crea una entidad que se encarga de proporcionar el servicio. Cada entidad que proporciona un servicio en el plano-U se divide a su vez en dos niveles:

- 1) La entidad del nivel superior (LUX). Contendrá todas las funciones que dependen del servicio, y por lo tanto definirá la mayoría de los procedimientos. En función del tipo de servicio solicitado la entidad puede ser del tipo LU1, LU2, LU3, etc.
- 2) La entidad del nivel inferior (FB). Se ocupa del almacenamiento y la fragmentación de las tramas del plano de usuario (unidades de datos de protocolo LUX) de/hacia la capa MAC.

En la capa DLC existe una entidad de gerencia del enlace (LLME) que se encarga de los procedimientos de gerencia de las entidades para proporcionar un tipo de servicio. Las capas superiores realizarán la negociación de los servicios aplicables del plano de usuario durante la fase de establecimiento de la conexión. Las demandas convenidas del servicio serán entregadas directamente a la capa DLC a través de la entidad de gerencia del enlace (LLME). La LLME instalará entonces el servicio requerido creando una entidad DLC que asociará a una conexión de la capa MAC. Una vez asociadas ambas entidades (DLC y MAC), se encarga de controlar el encaminamiento de tramas del plano-U de/hacia las conexiones MAC disponibles y controlar el inicio, modificación, y cierre de dichas conexiones. En la figura 2.12 se muestra la relación entre la entidad de manejo de gerencia del nivel de enlace (LLME) y una entidad DLC (nivel superior e inferior) asociada a una conexión MAC.

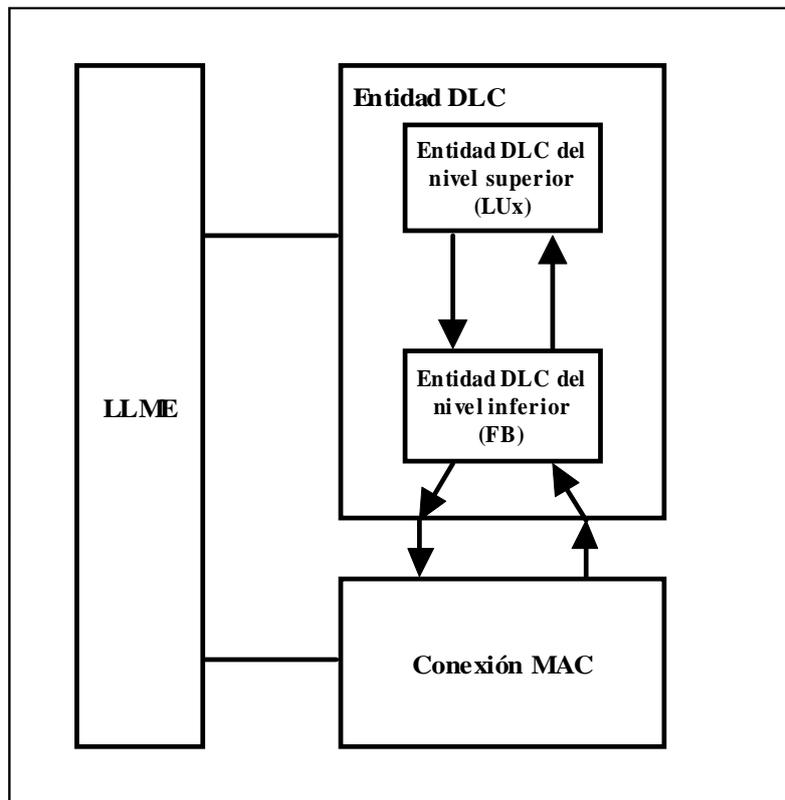


Figura 2. 12 Relación entre LLME, entidad DLC y conexión MAC

2.7.2 Servicio Frame Relay mejorado LU10 (EFREL)

Nos centraremos en el servicio LU10 del plano de usuario que resulta más adecuado para su integración en nuestro simulador con los servicios proporcionados por las capas superiores TCP/IP. Por tanto se va a seleccionar LU10 como la entidad del nivel superior. En este servicio los paquetes IP entregados por las capas superiores son adaptados a un formato de paquete comprensible por el sistema DECT para poder ser transmitidos por la interfaz aérea. Una vez convertidos al formato adecuado, el servicio proporciona transmisión fiable de dichos paquetes con un alto grado de corrección de errores y una baja complejidad. Para ello LU10 usa un algoritmo de ventana deslizante junto con un procedimiento de retransmisiones automáticas proporcionando a las capas superiores (en el caso de nuestro simulador a TCP/IP) un intercambio fiable de datos dentro del plano de usuario de la capa de enlace del sistema DECT.

El nuevo servicio LU10 es una mejora del servicio anterior de frame relay conocido como servicio LU2. Nos centraremos ahora en describir los servicios que LU10 proporciona a las capas superiores. A continuación se muestran resumidos los servicios ofrecidos por LU10 y que serán implementados en nuestro simulador:

1. Transmisiones fiables de paquetes de las capas superiores (SDUs) punto a punto. En nuestro simulador los paquetes de las capas superiores son paquetes IP. Los nuevos procedimientos que emplea LU10 en ambos extremos de la comunicación para proporcionar el transporte fiable de los paquetes aparecen recogidos en el estándar DECT bajo el nombre de procedimientos de clase 2 y de clase 3 y se encuentran implementados en el nivel superior de la entidad DLC.
2. Fragmentación en el extremo transmisor de los paquetes IP de las capas superiores en fragmentos que sean adecuados (PDUs) para poder ser transmitidos por la interfaz aérea del sistema DECT. El tipo de formato a emplear para las PDUs depende de la clase (2 ó 3) seleccionada en la entidad del nivel superior para proporcionar el transporte fiable de los datos de los paquetes de las capas superiores. En función del tipo de formato usado se decidirá la entidad del nivel inferior (FB) que se asociará al nivel superior. Las combinaciones disponibles para el servicio LU10 se resumen en la tabla 2.6 extraída de [7].

Transmission class	PDU structure
Class 2 unidirectional	FU10a/FU10c
Class 2 bi-directional	FU10b
Class 3 unidirectional	FU10a/FU10c
Class 3 bi-directional	FU10b

Tabla 2. 6 Estructura de las PDUs para LU10 (extraída de ETSI EN 300 175-4 v1.8)

El formato FU10b no será empleado en nuestro simulador, ya que se van a usar las clases unidireccionales, por lo que si se desea conocer más sobre este formato se recomienda acudir al estándar [7].

3. Reensamblado en el extremo receptor de los fragmentos de la pila de protocolo del sistema DECT (PDUs) para formar los paquetes IP que serán entregados a las capas superiores.
4. Capacidad de introducir en una PDU datos de SDUs diferentes.

2.7.2.1 Formato de las tramas FU10a y FU10c

Como se ha comentado en el apartado 2.7.1, las entidades de nivel inferior (FB) asociadas a la entidad de nivel superior (LUx) se encargarán de almacenar estas tramas. Los formatos FU10a y FU10c proporcionan delimitación de la trama y envío de números de secuencia para el control de flujo y el control de errores. Está diseñada para ofrecer alta eficiencia en enlaces unidireccionales. Las fragmentos (PDUs) con formato FU10a (figura 2.13) son usadas para transportar la información útil de las capas superiores (SDUs) en la dirección principal.

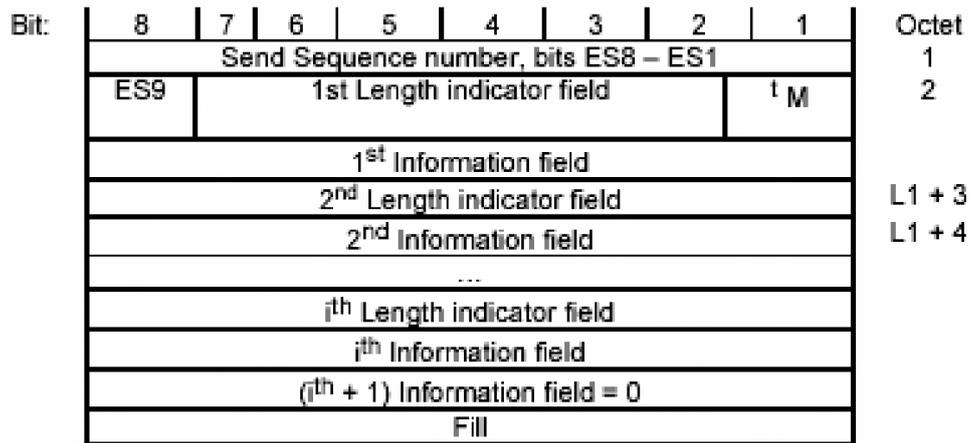


Figura 2. 13 Trama Fu10a (extraída de ETSI EN 300 175-4 v1.8)

Constan de un campo para indicar el número de secuencia de la PDU y de una serie de campos de información cada uno con su indicador de longitud. Sobre el uso de los indicadores de longitud hay que mencionar lo siguiente:

- Para el caso de que el paquete de la capa superior a introducir en la trama tenga un tamaño igual al tamaño máximo permitido para la trama, se empleará únicamente el primer campo para indicar la longitud.
- En el caso de que la SDU a introducir necesite de más de una trama (y por lo tanto haya que fragmentar la SDU), para la última trama si la información restante es menor que el máximo tamaño permitido de trama, entonces se dividirá esta información en subcampos de longitud fija cada uno con su indicador, siendo el último subcampo siempre de longitud menor o igual que el resto. Es en esta situación (cuando sobra espacio en la trama) cuando se puede optar por introducir información de una SDU diferente (característica exclusiva del servicio LU10).

Cabe mencionar que el campo de un bit que aparece a la derecha del indicador de longitud se denomina campo M y es usado para indicar si la trama contiene el final de una SDU o sólo parte de ella, así como para indicar si contiene partes de otras SDUs.

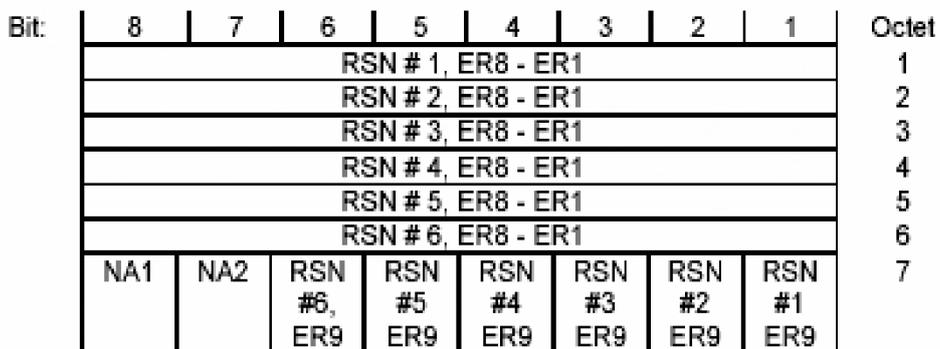


Figura 2. 14 Trama FU10c (extraída de ETSI EN 300 175-4 v1.8)

Las tramas del tipo FU10c (figura 2.14) se usan en las conexiones asimétricas para transportar la información de control en la dirección inversa. Contiene una lista de los números de secuencia recibidos en la dirección principal. Para saber el significado de

los bits N1 y N2, consultar el apartado 2.7.3.1 dedicado a los procedimientos de clase 2. El término FU10N se refiere al tamaño de los fragmentos (PDUs de DLC) y depende del formato de slot, tipo de servicio solicitado a la capa MAC y del nivel de modulación empleado. En la tabla 12.2.1.1 del estándar [7] se muestran todas las longitudes definidas. Para calcular el tamaño de la información útil que contiene la PDU, se deben restar a los valores de la tabla dos bytes. Estos dos bytes se corresponden con la información de control que indica el número de secuencia de la PDU y el primer indicador de longitud.

2.7.3 Clases de transmisión del para el servicio LU10

Como se ha comentado en el apartado 2.7.1 una clase de transmisión engloba los procedimientos empleados por un determinado servicio de la entidad del nivel superior (LUx) para proporcionar un transporte fiable de los paquetes extremo a extremo. Múltiples instancias de una entidad DLC del plano de usuario pueden existir entre un FT y un PT. Estas instancias operan de manera independiente por lo que los procedimientos de las clases que se detallan a continuación están definidos para una única instancia del tipo LU10 y son genéricos.

2.7.3.1 Procedimientos de la clase 2:

Esta clase se caracteriza por las demandas de caudal variable con un retardo máximo controlado. Cada trama entregada por las capas superiores es retransmitida hasta que es reconocida por el otro extremo o expira el tiempo de vida asignado a esa trama en cuyo caso es descartada. La retransmisión a nivel DLC usará un procedimiento de retransmisión automática (ARQ).

Procedimiento definido para el transmisor:

La entidad transmisora añade los números de secuencia a todas las PDUs para ser entregadas a la capa MAC. Esta entidad mantendrá una ventana cuyo tamaño estará definido por el número de secuencia de la trama a enviar y el número de secuencia de la última trama confirmada. El tamaño máximo de la ventana podrá ser negociado en la fase de establecimiento. Así mismo un tamaño más pequeño podrá ser adoptado por el transmisor en cualquier momento de la transmisión. Siempre que el tamaño de la ventana del transmisor alcance el tamaño máximo permitido el lado que envía comenzará la retransmisión de todas las PDUs que no hayan expirado, empezando con la PDU más vieja no reconocida. Esta retransmisión automática será parada siempre que se reciba trama que reconozca una o más tramas de la ventana del transmisor. La recepción de un NACK provocará la retransmisión de las PDUs indicadas usando el protocolo de retransmisión adoptado (selectivo o Go_Back_N).

En el momento en que un paquete de las capas superiores es entregado a la capa DLC, se le asocia un temporizador. El tiempo de vida de dicho paquete será igual a $T(R)$ tramas TDMA. Cada vez que el temporizador alcance el valor $T(R)$, el paquete será descartado y no será retransmitido. La ventana de transmisión no se desplazará debido a la expiración de los fragmentos (PDUs) que pertenecen al paquete descartado. Si una trama contiene partes de más de una SDU, el tiempo de vida corresponde al tiempo de vida de la última SDU incluida en la trama. El valor del tiempo de vida será definido en la fase de establecimiento y no podrá ser modificado. Si no se ha establecido un valor en la fase de establecimiento el tiempo de vida se considera igual a infinitas tramas TDMA.

Cuando una o más tramas expiran, entonces la última PDU será reemplazada por un mensaje de sincronización que contiene el número de secuencia de esta PDU. En el caso de que expiren todas las PDUs de la ventana se reenviará el mensaje de sincronización. Entonces la ventana de transmisión podrá coger nuevas PDUs. Cuando el transmisor recibe la confirmación del mensaje de sincronización sabe que el receptor ha resincronizado su ventana de recepción y puede desplazar la ventana para transmitir nuevas PDUs. El formato del mensaje de sincronización se muestra en la figura 2.15 extraída del estándar [7].

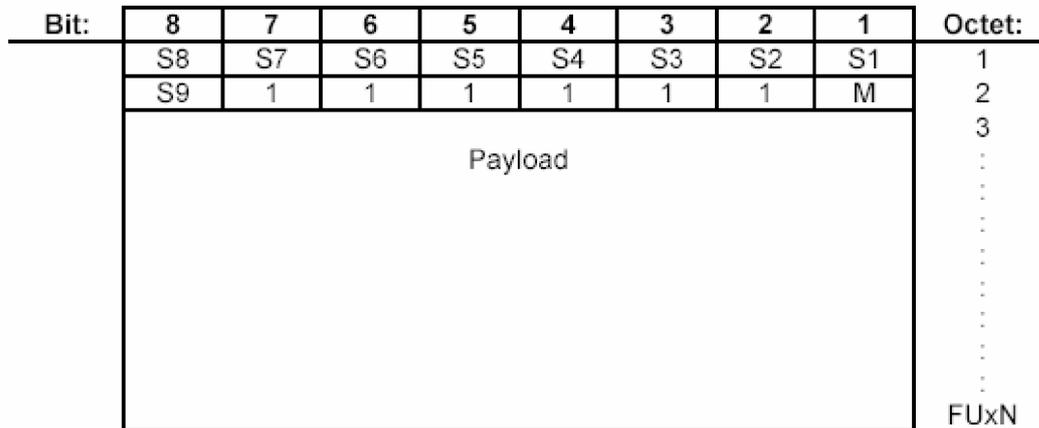


Figura 2. 15 Trama para el mensaje de sincronización (extraída de ETSI EN 300 175-4 v1.8)

Procedimiento del receptor:

La entidad receptora podrá aceptar los paquetes de datos de la capa MAC en cualquier orden. Los paquetes marcados como desconocidos o que contengan errores serán descartados. Los paquetes de la capa MAC que contengan fragmentos (PDUs) válidos serán procesados en el orden de llegada. Las PDUs cuyo número de secuencia se encuentre fuera de la ventana de recepción serán descartadas. Sólo los mensajes de sincronización serán siempre aceptados, incluso si su número de secuencia se encuentra fuera de la ventana de recepción. Cuando se reciban una serie de una o más PDUs que no contienen errores y cuyos números de secuencia son consecutivos serán confirmadas. Los datos serán entregados a las capas superiores.

El mensaje de sincronización será reconocido usando un tipo de reconocimiento especial para indicar al transmisor que puede actualizar su ventana. El mensaje de reconocimiento de sincronización tiene dos formatos dependiendo del tipo de conexión. Para conexiones simétricas el formato es el mismo que el del mensaje de sincronización y devuelve el número de sincronización de ese mensaje. Para las conexiones asimétricas manda un mensaje ACK/NACK. El formato para este mensaje se muestra en la figura 2.14. El código NA1, NA2 que indica que se trata de un mensaje de respuesta a un mensaje de sincronismo es N1=0 y N2=0. El resto de códigos para NA1 y NA2 se usan para indicar el número de reconocimientos del mensaje (como máximo 6). Los códigos se pueden consultar en [7].

En esta clase se define un parámetro denominado L(R) que se refiere al tiempo que permanecerán las tramas fuera de secuencia en el buffer antes de empezar a enviar NACKs para pedir el reenvío de los paquetes perdidos. Su valor dependerá del tipo de servicio solicitado a la capa MAC.

Cuando se usa el servicio de corrección de errores (apartado 4.2.2.3, capítulo 4) en modo protegido:

$$L(R) = \text{tiempo de vida del paquete MAC (expresado en tramas TDMA)} + 1.$$

Cuando se usa el servicio de detección de errores (apartado 4.2.2.3, capítulo 4) en modo protegido:

$$L(R) = 1 \text{ trama TDMA.}$$

Para el caso de la recepción de PDUs que se encuentran dentro de la ventana de recepción pero que contienen errores o están fuera de la secuencia, si después de almacenarlos $L(R)$ tramas permanecen fuera de secuencia, se mandará un mensaje NACK para indicar la retransmisión de los paquetes perdidos. Mientras permanezcan fuera de secuencia se seguirán mandando NACKs cada $L(R)$ tramas. Las PDU que se encuentran fuera de secuencia dejarán de serlo y serán entregadas a las capas superiores en el momento que lleguen las PDUs perdidas. A la hora de recuperar tramas perdidas o erróneas en el caso de usar el protocolo Go-back-N el mensaje NACK llevará el número de secuencia de la trama más pequeña perdida. Si se usa el protocolo de repetición selectiva llevará múltiples números de secuencia, uno por cada PDU perdida. Si durante el tiempo que están almacenadas en el buffer las PDUs fuera de secuencia llegan nuevas PDUs duplicadas con el mismo número de secuencia serán descartadas si la PDU original es correcta. Si la trama duplicada ha llegado correctamente también se puede emplear para corregir errores de la trama original.

2.7.3.2 Procedimientos de la clase 3:

Este servicio se caracteriza por las demandas de caudal fijo. Emplea la corrección de errores basada en la retransmisión, aunque se mantiene el caudal incluso cuando hay errores. Sólo se puede usar el modo de operación de detección de errores del canal I de la capa MAC. El nivel de corrección es definido por los atributos del servicio, fijando un retardo nominal y el ancho de banda restante disponible para la capa MAC. En el caso de que una trama no pueda ser entregada de forma correcta dentro de los límites del servicio, ambos extremos del enlace borrarán esta trama. En el transmisor, cualquier trama que supera el tiempo de vida es descartada sin notificarlo al otro extremo y no volverá a ser retransmitida. En este caso el receptor generará una trama para reemplazarla y señalará la interrupción. A continuación el receptor procederá como si hubiera recibido la trama correctamente, incluyendo el envío de reconocimientos positivos cuando sea necesario.

Procedimiento del transmisor:

El procedimiento es idéntico al de la clase 2 pero no emplea los mensajes de sincronización para informar al otro extremo de que el tiempo de vida de los paquetes ha expirado y deben ser descartados. Al igual que en el caso anterior el valor del parámetro $T(R)$ se definirá en la fase de establecimiento y no podrá ser cambiado. En el caso que no se establezca este valor durante el establecimiento de la conexión tomará por defecto un valor de 15 tramas TDMA (no infinito como en el caso de la clase 2). Tanto el transmisor como el receptor encargado de enviar los reconocimientos conocen de antemano el valor del $T(R)$ de manera que cuando se cumpla este tiempo el emisor descartará los paquetes asociados a ese fragmento y el receptor enviará un falso reconocimiento para estos paquetes. Cuando el transmisor recibe este reconocimiento falso actualiza su ventana.

Procedimiento de receptor:

El procedimiento de recepción es similar al de la clase 2 pero en este caso no se usa el parámetro L(R) y los paquetes fuera de secuencia almacenados en el buffer no serán descartados (como ocurría en la clase 2) cuando expire su tiempo vida (T(R)) si no que serán entregados a las capas superiores indicándoles que ha habido pérdida de paquetes. La entidad receptora actuará entonces como si esas tramas no estuvieran fuera de secuencia y se enviará un reconocimiento falso al otro extremo indicando que las tramas se recibieron correctamente.

2.8 Servicio de radio paquetes (DPRS)

El perfil DPRS [15] define la manera de implementar un servicio de paquetes de datos sobre la interfaz DECT. DPRS es la base de todas las aplicaciones de datos basadas en DECT y se puede usar con otros perfiles o para interactuar con otro tipo de redes. En la figura 2.16 y la figura 2.17 extraídas de [15] se muestra la configuración de capas para el plano de usuario y para el plano de control respectivamente:

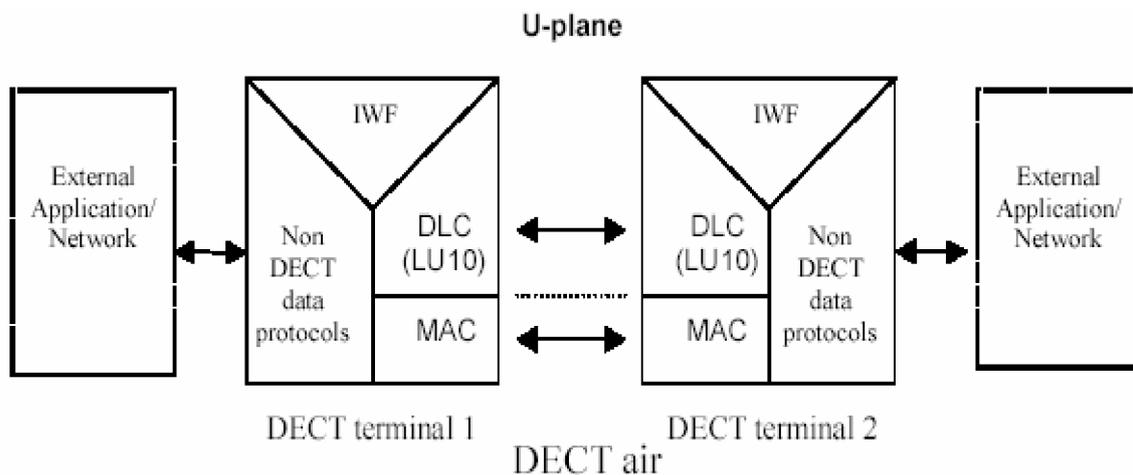


Figura 2. 16Configuración plano-U (extraída de ETSI EN 301 649 v1.4)

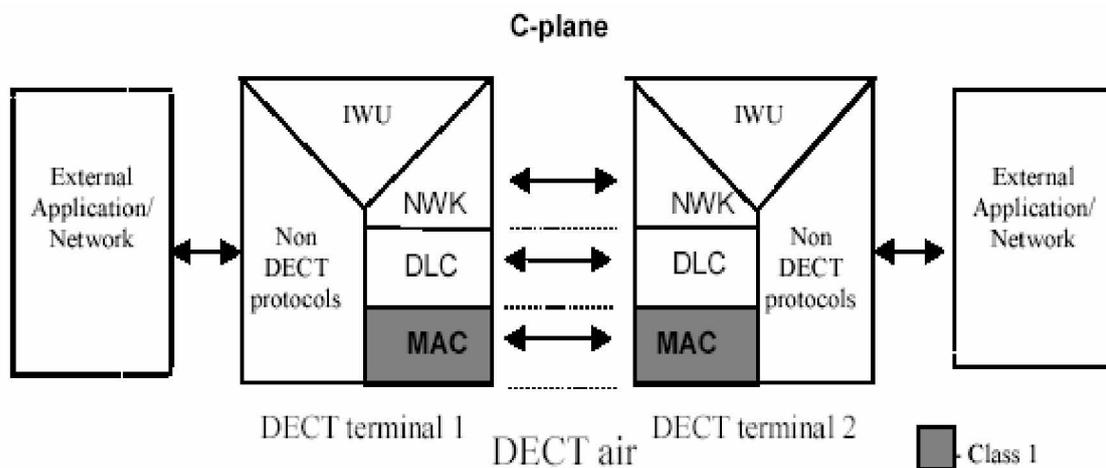


Figura 2. 17Configuración Plano-C (extraída de ETSI EN 301 649 v1.4)

El terminal DECT puede ser una PP o una FP. La capa IWF del plano-U determina las interacciones entre dicho plano y la aplicación. El tipo de aplicación determina el servicio a usar. Para el caso de aplicaciones que requieren interactuar con otro tipo de redes se definen dos servicios adicionales:

- Frame Relay.
- Flujo de caracteres.

En este documento nos centraremos en el servicio frame relay que es el que nos interesa a la hora de plantear nuestro simulador.

2.8.1 Objetivos del servicio

Los objetivos de funcionamiento y de servicio que persigue DPRS se pueden consultar en detalle en las tablas 1 y 2 incluidas en [15]. A continuación se enumerarán los objetivos de funcionamiento y servicio que se consideran de importancia para la realización del simulador. Los objetivos de funcionamiento que se pueden destacar son:

- El tamaño mínimo de SDU soportado por el servicio frame relay será de 1528 bytes.
- La máxima velocidad sostenible por slot debe ser de 115,2 Kbps.
- La máxima velocidad sostenible en una dirección de comunicación debe ser de 2Mbps. Esta velocidad se consigue con 8 niveles de modulación y portadoras de tráfico en 23 de los 224 slots de la trama TDMA.

En cuanto a los objetivos del servicio tenemos que destacar el requisito de usar ARQ para que exista una protección de los datos que viajan entre la PP y la FP y viceversa.

2.8.2 Requisitos y procedimientos del sistema DECT

Como corresponde a los perfiles, DPRS selecciona una serie de opciones disponibles de las capas PHL, MAC, DLC y las impone como requisitos imprescindibles para conseguir los objetivos de servicio y funcionamiento. Para cada uno de estos requisitos se define un procedimiento. A continuación se mostrará una selección los requisitos de las capas MAC y DLC que son necesarios para la implementación de un simulador del control de flujo asimétrico del plano-U a nivel de enlace. Para una descripción completa de estos requisitos y sus procedimientos consultar la última revisión del estándar [15].

2.8.2.1 Requisitos de la capa PHL

La capa física deberá cumplir todos los requisitos relacionados con la interfaz radio y deberá soportar las combinaciones de esquemas de modulación que se incluyen en [15]. Los esquemas de modulación se corresponden con una modulación de 2, 4 y 8 niveles.

2.8.2.2 Requisitos de la capa MAC

Los requisitos para la capa MAC se pueden resumir en la tabla 6 incluida en [15]. Los requisitos necesarios para la implementación del simulador serán los que hacen referencia al formato de los paquetes. Los procedimientos a usar están descritos a lo largo del apartado 2.6 dedicado al funcionamiento de la capa MAC.

2.8.2.3 Requisitos de la capa DLC

Los requisitos de la capa DLC se resumen en la tabla 8 incluida en el perfil DPRS [15]. Los procedimientos para los requisitos se han descrito a fondo en el apartado dedicado a la capa DLC. A continuación como recordatorio se muestra un resumen (Entre corchetes se muestra la referencia con la que se identifica dicho servicio dentro del perfil DPRS):

- **Servicio LU10 (Frame relay mejorado) [DPRS-D.1]:** se trata de un servicio Frame Relay mejorado que se proporciona a través del punto de acceso LU10. Este servicio actúa sobre el campo de datos de los paquetes de las capas superiores (SDUs). Estos paquetes serán fragmentados en PDUs y transmitidos empleando los procedimientos definidos por la clase correspondiente. LU10 proporciona mecanismos para el transporte fiable de SDUs genéricas y para definir y preservar la longitud de cada SDU.
- **FU10a [DPRS-D.2]:** proporciona una estructura para la trama y funciones de almacenamiento (buffering) para transmitir o recibir de la capa MAC los datos del plano de usuario cumpliendo los requisitos de mínimo retardo entre capas. Este tipo de trama es usada para la dirección principal de transmisión en enlaces unidireccionales.
- **FU10c [DPRS-D.4]:** Este tipo es usado para transportar los reconocimientos positivos o negativos en las conexiones. La trama contiene una lista de los números de secuencia recibidos por el extremo de la conexión.

2.8.3 Procedimientos de la capa DLC

En estos apartados se hacen una serie de comentarios y matizaciones sobre los procedimientos definidos para cada capa.

2.8.3.1 Servicio Frame Relay mejorado LU10 (EFREL)

Los procedimientos a seguir serán como los descritos en el apartado correspondiente del capítulo DLC. En estos procedimientos se resaltarán las siguientes aclaraciones:

1. Los paquetes de las capas superiores (SDUs) serán fragmentados en fragmentos de longitud fija (PDUs) que dependerán del formato seleccionado para las tramas.
2. Se usará el modo protegido de detección de errores. El modo protegido de corrección de errores podrá ser negociado por la red y sustituido por el modo de detección.
3. Se usará la clase de transmisión 2 de la capa DLC.
4. Se usará el protocolo de repetición selectiva.

2.8.3.2 Tamaño de la ventana

El tamaño de la ventana será negociado por la capa DLC y tendrá un valor entre 1 y 256. El valor por defecto del tamaño de la ventana será 32 y será empleado en caso de que no exista o falle la negociación. Cualquier dispositivo DPRS como mínimo uno de los siguientes valores (tabla 2.7) para el tamaño de la ventana extraídos de [15]:

Minimum mandatory supported window size	Maximum Data rate supported (at the air i/f)
32	Up to 96 kbit/s
64	> 96 kbit/s to 200 kbit/s
128	> 200 kbit/s

Tabla 2. 7 Tamaño de la ventana (extraída de ETSI EN 301 649 v1.4)

Cualquier terminal portátil aceptará en la negociación un valor comprendido entre el valor por defecto (32) y su valor correspondiente en la tabla. Para nuestro simulador el tamaño a emplear por la ventana será el definido por defecto, que es 32.

2.8.3.3 Procedimientos de la clase 2 de transmisión

Los procedimientos a seguir serán como los descritos en el apartado 2.7 incluido en este capítulo dedicado a la capa DLC: Únicamente se mostrarán las aclaraciones que se incluyen en el perfil DRPS sobre estos procedimientos en ambos extremos de la comunicación.

Procedimientos del transmisor:

En estos procedimientos se resaltarán las siguientes aclaraciones:

1. Si se emplea un servicio orientado a la conexión, el número de secuencia será iniciado a 0 al principio de la conexión MAC, y este valor será usado para la primera trama transmitida. Los números de las siguientes tramas serán consecutivos durante el tiempo de vida de la conexión MAC.
2. En el lado del transmisor un paquete de datos (SDU) completo será recibido de la entidad superior a través de la primitiva DL_U_DATA-req. La SDU será fragmentada en un número entero de fragmentos (PDUs). El último fragmento será completado con información de relleno si es necesario. Se usará un campo para indicar la longitud de la información contenida en cada PDU.
3. Varios fragmentos (PDUs) pueden enviarse de una vez a la capa MAC a través de la primitiva MAC_CO_DATA-req como respuesta a cada primitiva MAC_CO_DTR-ind. El número de PDUs será menor o igual al número máximo especificado en la primitiva MAC_CO_DTR-ind.

Procedimientos del receptor:

En estos procedimientos se resaltarán las siguientes aclaraciones:

1. El número de RN será 0 al inicio.
2. Varios fragmentos (PDUs) procedentes de la interfaz radio pueden ser recibidas por la capa MAC con la primitiva MAC_CO_DATA-ind. El extremo receptor reordenará los fragmentos usando los números de secuencia. A continuación buscará los extremos de los paquetes de las capas superiores (SDUs) y cuando los identifique unirá los datos de los diferentes fragmentos y formará el paquete (SDU) que entregará a las capas superiores
3. Un paquete (SDU) completo se pasará a la unidad de interacción (IWU) usando la primitiva DL_U_DATA-ind cuando se satisfagan las siguientes 2 condiciones:

- Se han recibido todos los fragmentos (PDUs) que forman un paquete para la capa superior (SDU).
- Se han identificado los extremos de un paquete para la capa superior (SDU).

2.8.4 Servicio Frame Relay (FREL)

El servicio frame relay puede ser usado para transportar los siguientes protocolos:

1. Ethernet [10].
2. Token Ring [11].
3. IP versión 4 o superior [12].
4. Protocolo punto a punto (PPP) [13].

En la figura 2.18 extraída de [15] se muestra la configuración típica para proporcionar un servicio FREL orientado o no a la conexión:

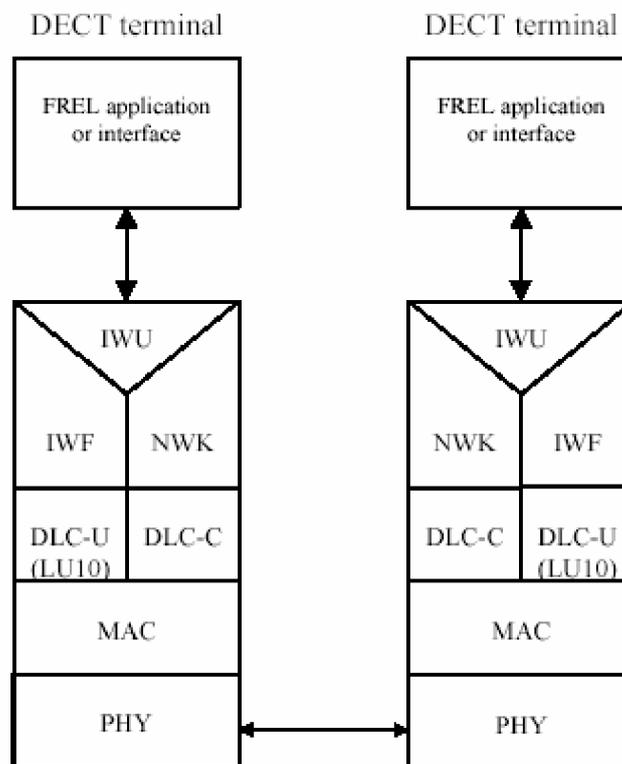


Figura 2. 18 Configuración para el servicio FREL (extraída de ETSI EN 301 649 v1.4)

Independientemente del protocolo de la red que se transporta a través del servicio FREL se definen una serie de acuerdos generales sobre el uso de este servicio. Así mismo se definen también convenciones para cada uno de los protocolos soportados. En este documento nos centraremos en los acuerdos de interacción con el protocolos de las capas superiores (TCP/IP).

2.8.5 Convenciones generales de interacción para FREL

En este apartado se definen una serie de acuerdos generales que deberán ser considerados a la hora de implementar el servicio Frame Relay (FREL). Los aspectos más relevantes de estas convenciones se pueden resumir en los siguientes apartados:

1. El servicio Frame Relay usará instancias de entidades que implementan el servicio LU10 (Frame Relay mejorado) de la capa DLC.
2. Todos los paquetes de datos de las capas superiores entregados al nivel de enlace (SDUs) serán siempre transmitidos como fragmentos (PDUs) de la capa DLC de DECT.
3. Todos los equipos serán capaces de manejar paquetes de las capas superiores (SDUs) de al menos 1528 bytes.

2.8.6 Convenciones de interacción con IP

En este apartado se especifican las convenciones para la interacción con el protocolo de Internet IP versión 4 o superior. Ejemplos de posibles implementaciones de la red externa de transporte y las funciones de la unidad de interacción (IWU) para cada una de ellas se muestran en la tabla B.2 incluida en [15]. En cuanto a los acuerdos para la interacción conviene destacar los siguientes:

1. Los paquetes IP de las capas superiores serán entregados al nivel de enlace del sistema DECT dentro de una sola SDU.
2. La SDU contiene la cabecera IP seguida de los datos.
3. Como la longitud de las SDUs entregadas por las capas superiores a las instancias de las entidades que implementan el servicio LU10 puede ser arbitraria no existen requisitos en cuanto a la información de relleno.

Capítulo 3

Escenario de Referencia

3.1. Introducción

El escenario que se desea simular representa la descarga de un archivos desde un servidor de Internet a un terminal móvil que implementa un cliente de descarga de archivos que emplea el protocolo TCP/IP. El terminal móvil forma parte de un sistema DECT que le proporciona las características de movilidad y actúa como protocolo de transporte a través de la interfaz aérea. En dicho escenario tiene lugar una integración de los servicios ofrecidos por los protocolos TCP/IIP con los servicios ofrecidos por la pila de protocolos del sistema DECT. Como se ha comentado en el apartado 2.4.2.1 del capítulo anterior las recomendaciones para este tipo de escenarios se encuentran recogidas en el anexo B del perfil DPRS [15]. Este tipo de escenario para la transferencia de datos IP se muestra en la figura 3.1 extraída de [20].

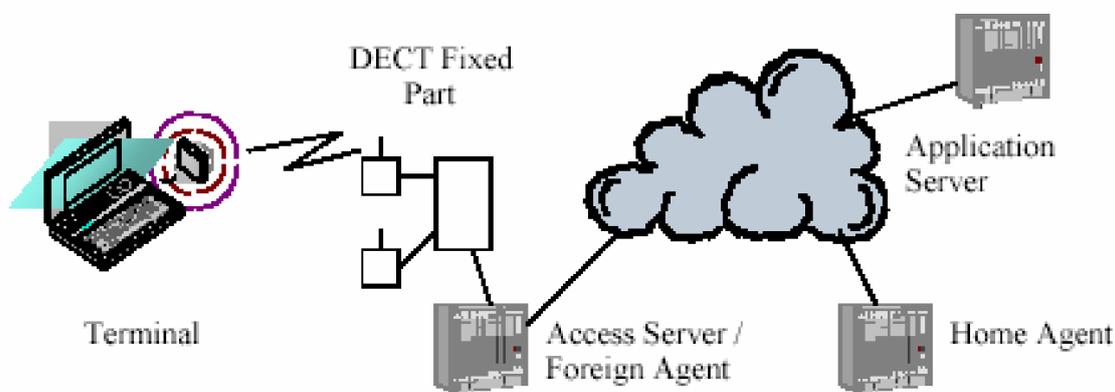


Figura 3. 1 Escenario de referencia (extraída de ETSI TR 102 010 v1.1.1)

A la hora de construir nuestro simulador se ha considerado únicamente el plano de usuario, por lo que los elementos relacionados con el plano de control y señalización como es el caso del "Home Agent" no han sido considerados.

3.2. Descripción del escenario

La configuración seleccionada en la figura 3.2 representa un sistema DECT perteneciente a una empresa, en la cual sus usuarios tienen acceso a un servidor público o privado que puede estar situado en otra localización (otra ciudad por ejemplo) y al que se conectan a través de una red de conmutación de paquetes IP (como Internet). Estos usuarios pueden usar el sistema DECT en el entorno privado de la manera tradicional para comunicaciones de voz, o bien pueden hacer peticiones de datos al servidor. En el sistema podrán convivir por lo tanto dos tipos de terminales:

- Terminales móviles que implementan el servicio tradicional de voz proporcionado por DECT.

- Terminales que implementan los servicios de voz y datos. Estos terminales pueden usarse para acceder servicios proporcionados por las redes IP, tales como el correo electrónico, consulta de páginas web, etc....

Este escenario se presenta por tanto como una posible solución para la integración de los servicios de voz y datos dentro de un mismo entorno de trabajo. Para la realización del presente proyecto se ha considerado un único terminal móvil del segundo tipo que accede a la interfaz aérea DECT. Este terminal realiza peticiones a un servidor de Internet a través de una red IP.

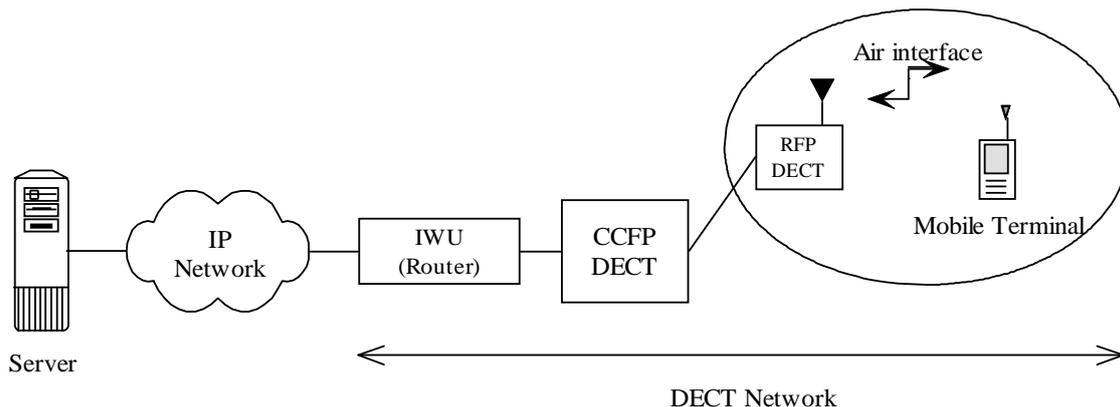


Figura 3. 2 Escenario de referencia para nuestro simulador

Dentro de la red IP viajarán dos tipos de paquetes IP:

- Paquetes con datos del archivo que está siendo descargado del servidor.
- Paquetes que contienen los reconocimientos para el control de flujo de los paquetes en la dirección de descarga.

Como los paquetes viajan en dos direcciones de comunicación deben existir dos enlaces entre el servidor y la terminación portátil:

- a) Enlace downlink (Servidor à Terminal). En esta dirección viajan los paquetes con los datos del archivo del servidor.
- b) Enlace uplink (Servidor ß Terminal). En esta dirección viajan los reconocimientos para los paquetes con datos del servidor.

Por tanto la red IP se encargará de transportar los paquetes IP en ambas direcciones los cuales serán entregados a la IWU o al servidor con una determinada cadencia en función de la velocidad del enlace seleccionada. En cuanto al sistema DECT consta de dos partes. La parte fija del sistema comprende:

- Una RFP, a través de la cual el terminal móvil tendrá acceso a la interfaz aérea.
- La CCFP, que se encargará de las funciones de control de las capas superiores de la pila de protocolos del sistema DECT.

- La IWU, que realiza en la red IP el papel de un router de acuerdo con lo especificado en [15] y se encarga de hacer las conversiones de formatos entre ambas redes permitiendo al terminal móvil el acceso a la red IP externa.

La parte portátil del sistema DECT está formada por el terminal móvil a través del cual se van a hacer las peticiones al servidor. A través de la interfaz aérea viajan paquetes DECT. Como TCP necesita un control de flujo bidireccional, será necesario emplear dos enlaces DLC. El diseño de la conexión entre la FP y la PP para estos dos enlaces será descrito en detalle en el apartado 4.2.2.1 del capítulo 4.

3.3. Arquitectura del protocolo

Para la construcción de la arquitectura del protocolo a emplear en nuestro simulador nos hemos basado en las pilas de protocolo que aparecen en la figura 2.23 y la del apartado dedicado al perfil DPRS. En esta figura aparece la pila de protocolo de una manera general independientemente de del nivel de aplicación. Para la elaboración de la pila de protocolos del sistema completo se ha consultado la pila de protocolo que aparece en la figura 3.3 y que muestra la configuración particular para el caso de que la aplicación del terminal móvil sea una aplicación que soporte el protocolo IP. En dicha figura se consideran tanto el plano de usuario como el plano de control y por este motivo aparece la capa NWK. Sin embargo tal y como se mencionó en el apartado 2.3 del capítulo 2 (donde se hablaba de las capas del protocolo de un sistema DECT), para nuestro simulador únicamente necesitamos implementar el plano de usuario por lo que como la capa NWK únicamente realiza funciones de control, no será considerada.

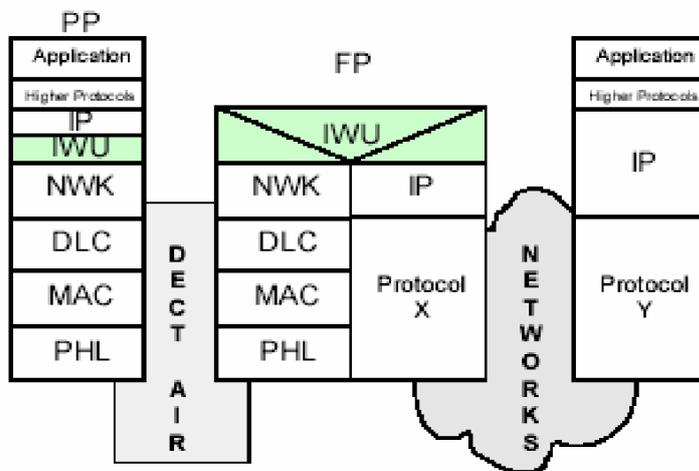


Figura 3. 3 Pila de protocolos para IP (extraída de ETSI TR 102 010 v1.1.1)

Considerando que la aplicación usa TCP en las capas superiores y que la red por la que viajan los paquetes IP es una red de conmutación de paquetes como Internet, la configuración (correspondiente al plano de usuario) completa del protocolo para nuestro escenario particular queda como se muestra en la figura 3.4.

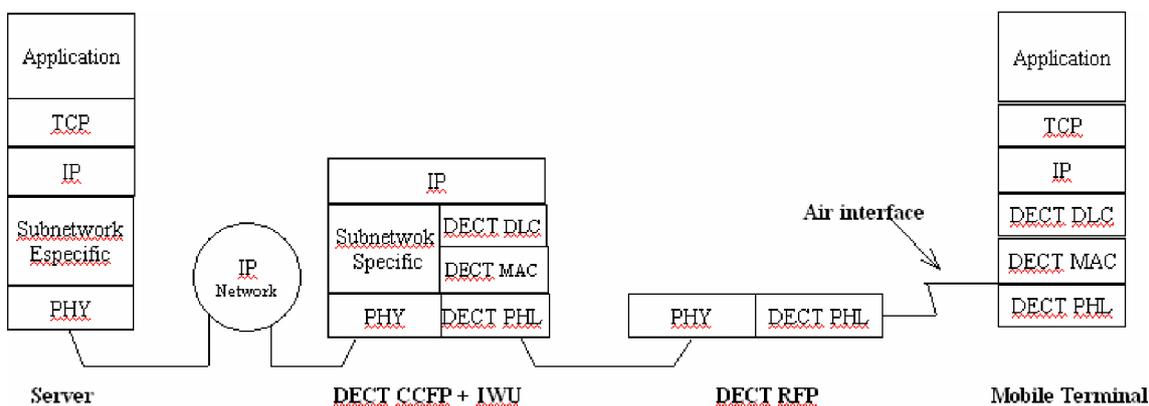


Figura 3. 4 Pila de protocolos para nuestro escenario

Como se puede apreciar en la figura 3.4, la pila básica de protocolos del sistema DECT se integra a la perfección con los niveles de la pila de protocolos TCP/IP, proporcionando a los niveles superiores un servicios de transmisión fiable de los paquetes IP por una interfaz aérea. Gracias a esta configuración en la que el terminal móvil implementa la pila de protocolos TCP/IP, el usuario del terminal podrá disfrutar de las ventajas de DECT junto con la amplia gama de servicios que ofrece el uso de los protocolos TCP/IP. Dentro del estándar otras configuraciones aparecen descritas para el caso de que la aplicación del terminal portátil no implemente el protocolo IP. Estas configuraciones no son adecuadas para el escenario que se pretende simular, pero si el lector desea consultarlas puede referirse al documento original donde aparece mencionada [19].

3.4. Unidades de datos del protocolo

Las unidades de datos de las diferentes capas que conforman el protocolo se muestran en la figura 3.5 y en la figura 3.6. Para su elaboración se han hecho las siguientes consideraciones:

- Se ha asumido que se usa TCP como protocolo de transporte. El servicio LU10 (apartado 2.7.2) es usado para fragmentar los paquetes IP entregados por las capas superiores al nivel de enlace (SDUs) en fragmentos (PDUs), con una cabecera de 2 bytes que se usa para contener los números de secuencia y la longitud del campo de datos (Payload). A diferencia del antiguo servicio LU2, LU10 permite que una unidad de datos contenga información de tramas DLC (SDUs) diferentes. Esta característica en el apartado 2.7.2. Por su parte el tamaño del campo de datos (Payload) dependerá del nivel de modulación seleccionado. En la tabla 12.2.1.1 incluida en [7] se pueden consultar los diferentes tamaños definidos por el estándar para cada formato de paquete.
- A nivel de la capa MAC el formato del paquete varía según el formato de protección seleccionado. En la figura 3.5 se muestra el formato de protección de múltiples campos (lp) que se pueden consultar en el apartado 2.1.1 extraído de [6]. El número de subcampos que componen el campo B dependerá del nivel de modulación seleccionado. En la tabla 2.1 extraída de [6] se muestran el número de subcampos para cada nivel de modulación. En la figura 3.6 el formato mostrado es el de protección de un solo campo (lpq). Al igual que para el anterior formato sus características pueden ser consultadas en el apartado 2.1.1 del presente documento extraído de [6]. El tamaño del campo B así como del subcampo CRC dependerá del nivel de modulación seleccionado. En la tabla 2.2 extraída de [6] se pueden consultar estos valores.

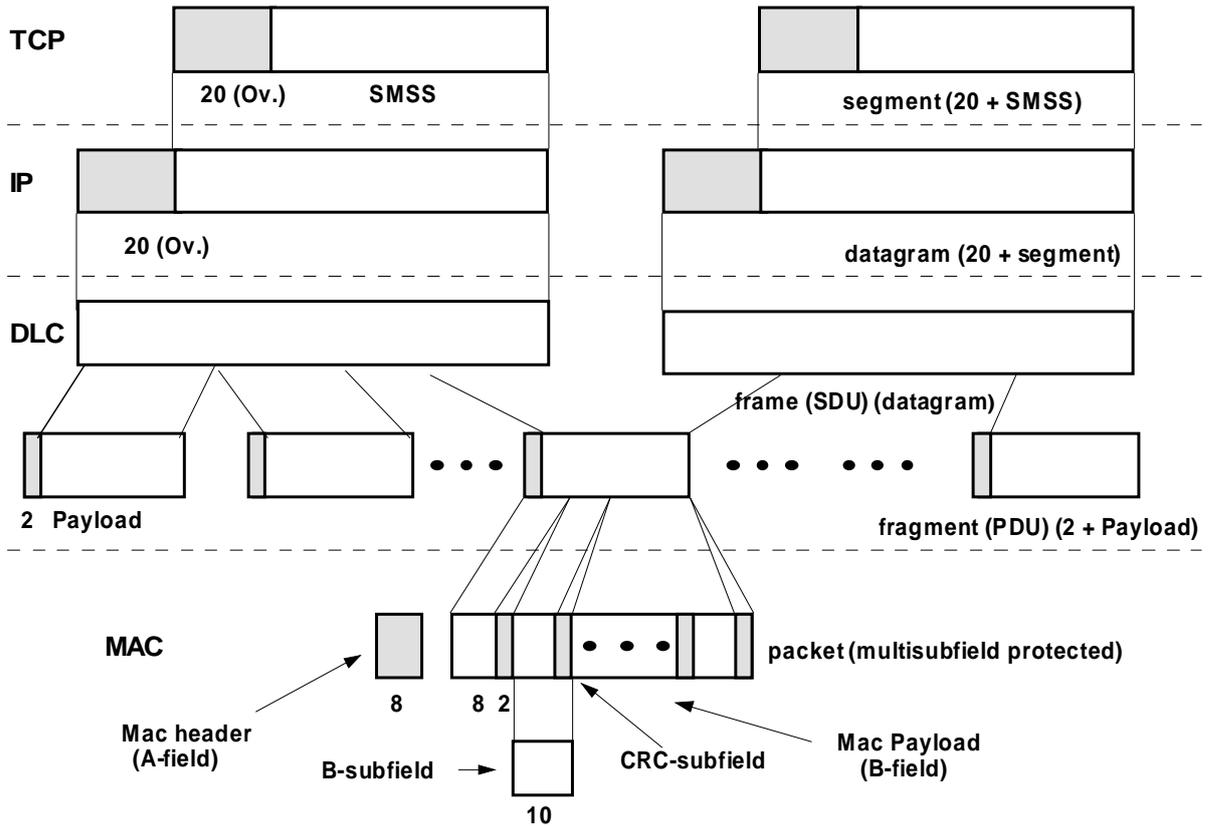


Figura 3. 5 Unidades de datos del protocolo para cada capa y sus relaciones (LU10,Ip)

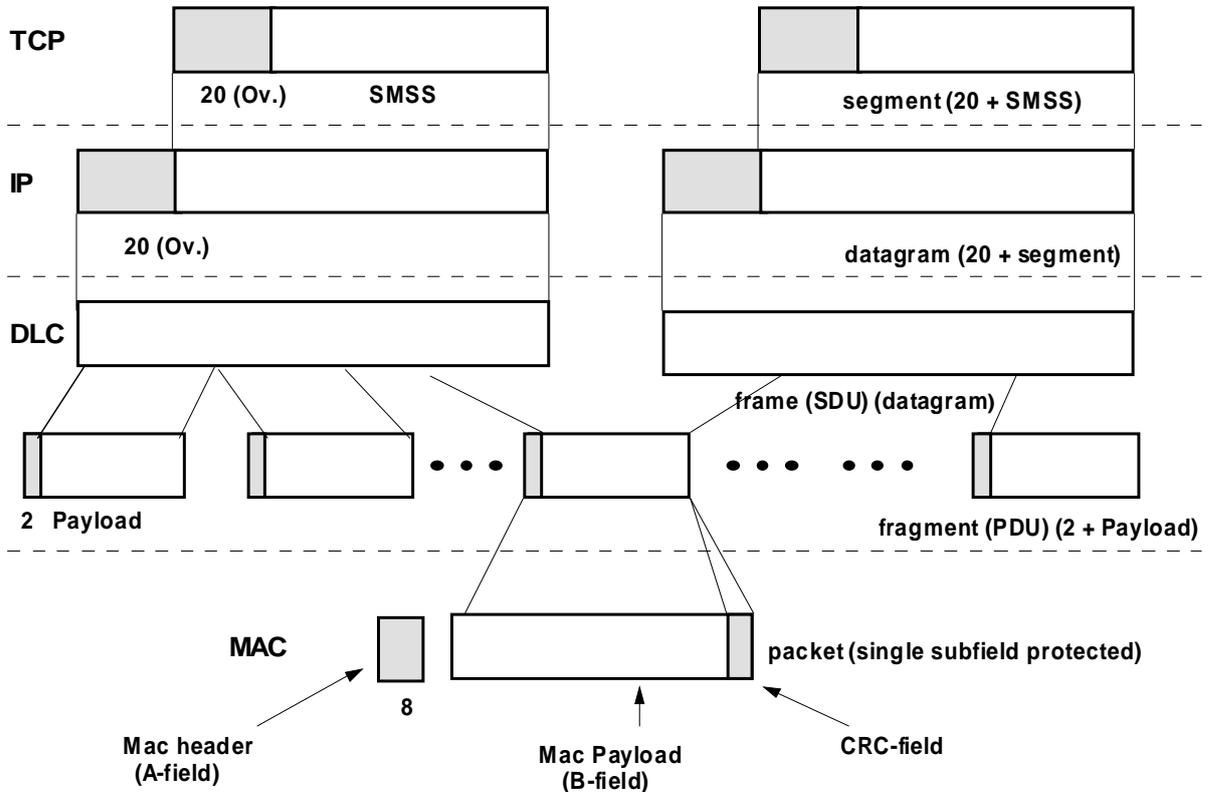


Figura 3. 6 Unidades de datos del protocolo para cada capa y sus relaciones (LU10, Ipq)

Capítulo 4

Descripción del simulador

4.1 Introducción

En este capítulo se van a describir los pasos que se han seguido para la construcción de un simulador que nos permita simular el sistema del escenario de referencia descrito en el capítulo 3. Para la creación del simulador ha sido necesario decidir el tipo de lenguaje de programación o herramienta que se va a emplear. Para tomar esta decisión se han hecho las siguientes consideraciones sobre el sistema que se pretende simular:

1. Los elementos que se deben modelar en el sistema son la pila de protocolos empleada, el tráfico que circula por la red y los módulos que componen la red.
2. Como se desea que el reloj de simulación avance únicamente cuando se produce un evento en el sistema (evitando así el consumo innecesario de tiempo de CPU), se necesitará emplear una simulación mediante eventos discretos.

Teniendo en cuenta estas consideraciones se ha decidido emplear como herramienta el software de libre distribución OMNET++. Se trata de un simulador de eventos discretos modular y orientado objetos que está implementado en C++ y que proporciona un API de simulación que será usada para la construcción de nuestro simulador. La explicación de las características y el uso del lenguaje de OMNET++ se consideran fuera del objetivo del proyecto por lo que se recomienda al lector que si desea conocer a fondo esta herramienta se descargue el manual online [23]. Las librerías de OMNET++ proporcionan métodos que se encargan de la planificación de eventos, de la creación de mensajes, del paso de mensajes entre módulos y de la actualización del tiempo de simulación. Nuestro trabajo ha consistido en decidir los módulos que se van a usar, la distribución jerárquica de los módulos, usar las herramientas que proporciona OMNET++ para la creación de los mensajes que van a circular entre módulos y apoyarse en las librerías de OMNET++ para la programación en C++ de los procedimientos de los niveles de la pila de protocolo que describen el comportamiento de los módulos que forman la red.

En los siguientes apartados se describirán las consideraciones que se han hecho y las decisiones que se han tomado para programar en C++ el comportamiento los niveles PHL, DLC y MAC de la red. También se incluye una descripción de los módulos que se han creado para modelar los dispositivos que componen la red. Por último se incluye una descripción del paso de mensajes entre módulos que representa un ejemplo de la circulación de tráfico por la red.

4.2 Implementación de la pila de protocolos

A la hora de implementar la pila de protocolos que se muestra en la figura 3.4 del capítulo 3 se hicieron las siguientes consideraciones:

- Para el servidor únicamente se contemplarán los aspectos relacionados con el formato de los paquetes y con el control de flujo a nivel de transporte.

- La implementación de la configuración de la pila de los protocolos que se puedan usar en Internet no es necesaria para el simulador.
- Los procedimientos de los niveles PHL, MAC y DLC de la capa serán implementados sin tener en cuenta los aspectos relacionados con el establecimiento de las conexiones, puesto que se encuentran relacionados con el plano de control, el cual no es objeto de nuestro simulador.

En los siguientes apartados se muestran las consideraciones, alternativas y decisiones que se han tomado para implementar los procedimientos de la pila de protocolos del sistema DECT de la PP y la FP y para los procedimientos TCP/IP entre el servidor y la aplicación.

4.2.1 Implementación de la capa PHL

Como se ha comentado en el apartado 2.5 del capítulo 2, la PHL se encarga de proporcionar el acceso a la interfaz física mediante el uso de portadoras en el dominio de la frecuencia y del sistema de tramas TDMA en el dominio del tiempo. La implementación de un simulador que use diferentes portadoras en frecuencia no es necesaria puesto que se supone que el terminal móvil trabaja siempre en una misma frecuencia. Así pues únicamente se ha implementado el procedimiento de acceso múltiple por división en el tiempo (TDMA) dentro de una portadora en frecuencia.

4.2.1.1 Implementación de la trama TDMA

Para implementar el procedimiento de acceso múltiple por división en el tiempo se ha usado una función dentro del módulo "timer" del simulador. Como los módulos MAC necesitan saber el slot de la trama para transmitir se ha implementado también un procedimiento para la generación automática de slots dentro de los módulos MAC (tanto transmisor como receptor).

Cada 10 mseg el módulo "timer" manda un mensaje al resto de módulos que forman el sistema, indicando el principio de una TDMA. Para el caso de los módulos MAC cuando reciben un mensaje de inicio de trama TDMA, dicho módulo inicia a su vez un procedimiento para la generación de slots. Este procedimiento consiste en mandar auto mensajes cada uno de los cuales marca el principio de un nuevo slot. El tiempo entre dos auto mensajes de inicio de slot tendrá que ser igual al tiempo de duración del slot. Este tiempo puede ser en principio fácilmente calculado dividiendo la duración de la trama TDMA (10 mseg) entre el número de slots por trama. El número de slots por trama varía en función del formato de slot considerado. Existen tres formatos posibles:

- Half slot
- Double slot
- Full slot

En nuestro simulador se ha decidido emplear el formato "full slot" por lo que la duración estimada para cada slot será de:

$$time_slot = TDMA_time / num_slots = 10 / 24 = 0.00041666\text{seg}$$

Al no tratarse de una cantidad exacta no se puede almacenar en una variable constante. Para solventar este problema se despreciarán las cifras a partir de la primera cifra del periodo siendo el valor final 0.416 mseg. Esta falta de precisión se verá traducida en una deriva temporal en cada slot del orden de 0.666 μ seg. En consecuencia la desviación temporal para la trama TDMA será de:

$$\text{Deriva temporal} = 0.666 * 24 = 0.0159984 \text{ mseg}$$

Esta desviación temporal no aumentará con cada nueva trama puesto que el mensaje del módulo "timer" que marca el principio de una nueva trama (y por tanto el final de la anterior) se produce en el tiempo de 10 mseg, por lo que la desviación temporal será corregida con cada nueva trama TDMA.

4.2.2 Implementación de la capa MAC

En los siguientes apartados se van a mencionar los servicios que se han seleccionado de la capa MAC para ser implementados en el simulador. Todos estos servicios aparecen en el apartado 2.6 del capítulo 2.

4.2.2.1 Diseño de la conexión asimétrica

Dentro de este apartado se van a describir las conexiones que se han diseñado para cada uno de los enlaces de comunicación. Las conexiones asimétricas permiten tener un número diferente de portadoras MAC con tráfico para el enlace en la dirección de descarga ("downlink") y para el enlace en la dirección de subida ("uplink"). Los principios definidos en el estándar MAC [6] para el uso de conexiones asimétricas se recogen en el apartado 2.1.1.1 del capítulo 2. Las consideraciones que se han seguido para el diseño de las conexiones para los enlaces han sido las siguientes:

- Se ha definido una conexión completamente asimétrica como aquella en la que el número de portadoras de tráfico en el sentido inverso de la comunicación (portadoras "m") es 0.
- Se ha empleado una conexión completamente asimétrica ($m = 0$) para cada uno de los enlaces
- Como el tráfico en cada enlace es diferente cada una de las 2 conexiones completamente asimétricas proporcionan diferentes anchos de banda. El ancho de banda dependerá del número de portadoras de tráfico en la dirección principal (portadoras "k").
- Por defecto el enlace de descarga ("downlink") que tiene la dirección de la estación base al terminal portátil, usará la conexión asimétrica que proporciona un mayor ancho de banda.
- El enlace de subida ("uplink") que tiene la dirección del terminal portátil a la estación base, usará la conexión asimétrica con menor capacidad.

En la figura 4.1 se muestra la asociación de las dos conexiones a las entidades de la capa de enlace para proporcionar los servicios a la capa superior TCP/IP. Para simplificar se ha eliminado en la figura la presencia de la entidad de gerencia del nivel de enlace (LLME). Si se desea conocer la función de la LLME acudir al apartado 2.7.1 del capítulo 2.

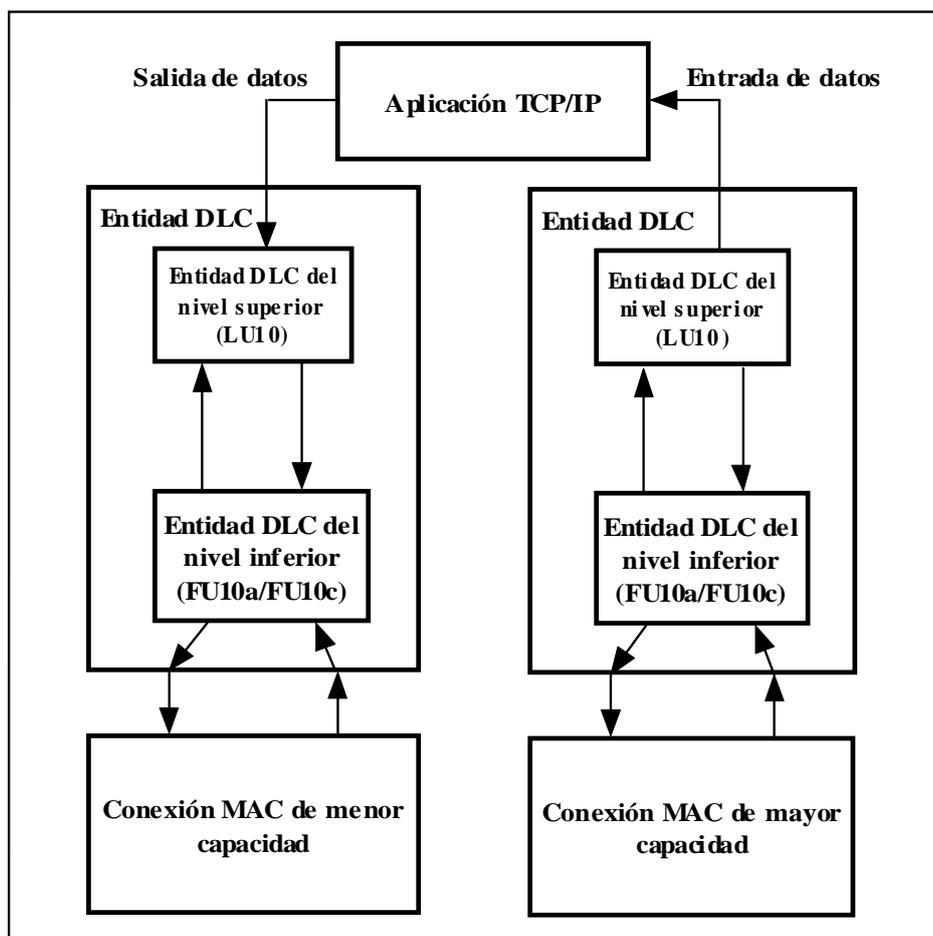


Figura 4. 1 Asociación entre las entidades DLC y las conexiones MAC

Como se ha comentado el ancho de banda de cada conexión depende del número de portadoras de tráfico usadas en la dirección principal de comunicación (portadoras “k”). Para cada una de las conexiones asimétricas existe una portadora “n” en la dirección opuesta que contiene la información para el control de flujo de las portadoras en la dirección principal. Hay que aclarar que tal como se impone en el estándar MAC [6] el formato de protección empleado para los paquetes que viajan en las portadoras “n” será siempre el formato de protección de múltiples subcampos independientemente del formato de las demás portadoras de la conexión. El objetivo de esta imposición es conseguir un mayor nivel de protección para la información que viaja en los subcampos. Esta información como se ha comentado anteriormente es de dos tipos:

- En el caso de usa el modo de corrección de errores, tendremos la información de control MAC para cada una de las portadoras en la dirección principal (portadoras “k”) que será introducida como mínimo en un subcampo. Como cada portadora necesita dos bits (Q2 y BCK) para el control de errores, el número de bits a rellenar en cada subcampo será de 2k.
- Independientemente del modo de protección a nivel MAC, se tendrá que usar como mínimo un subcampo de la portadora “n” para transportar la información de control para DLC. Esta información de control incluye los reconocimientos para el extremo transmisor. El formato de los reconocimientos para el caso

particular de usar la clase 2 a nivel DLC se puede ver en la figura 2.14 del capítulo 2 extraída de [7].

Para conseguir el mayor nivel de protección mencionado anteriormente, se replicarán los subcampos con los dos tipos de información dentro de la portadora “n”. La decisión de cuántos subcampos emplear para replicar cada tipo de información no se encuentra especificada en el estándar y será decisión del diseñador el decidir que tipo de información es la más importante (y por tanto la que debe usar un mayor numero de subcampos para ser replicada). Dado que el número de subcampos de los paquetes de la portadora “n” es siempre par se ha optó por dividir el campo de datos en dos grupos con el mismo número de subcampos replicando en cada grupo un tipo de información. Usando esta configuración se dio la misma importancia a los dos tipos de información.

La posición de las portadoras de ambas conexiones dentro de la trama TDMA se muestra en la figura 4.2. De los tipos de portadoras lógicas definidas en el apartado 2.6.2 del capítulo 2, se han usado:

- 9 portadoras lógicas dobles simples
- 2 portadoras lógicas dúplex.

Estas portadoras lógicas se vana repartir ahora entre las dos conexiones empleadas:

1. La conexión para el enlace “downlink” consta de 19 portadoras en la dirección principal (portadoras “k”) y 1 portadora con la información para el control de flujo de los datos que viajan en la dirección principal (portadora “n”). Para conseguir las 19 portadoras se han usado 9 portadoras dobles simplex y la portadora simplex en el slot 10 que pertenece a una de las dos portadoras lógicas dúplex. Para la portadora “n” se ha empleado la portadora simplex en la dirección inversa en el slot 22.
2. La conexión para el enlace “uplink” consta de 1 portadora para la dirección principal (portadora “k”) y 1 portadora “n”. La portadora “k” y “n” se corresponden con las portadoras simplex en los slots 9 y 21 respectivamente las cuales constituyen una de las portadoras dúplex.

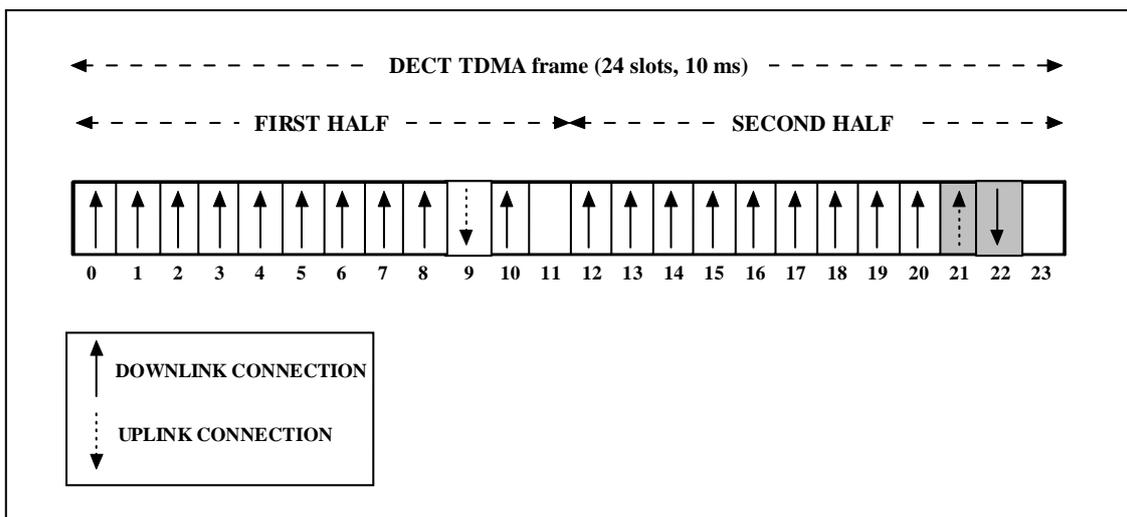


Figura 4. 2 Configuración de las portadoras con una portadora “n” y con el enlace downlink usando la conexión de mayor capacidad (Modelo 1)

Para el caso de emplear el servicio de corrección de errores a nivel de MAC, se realizará una retransmisión automática de todos los paquetes sobre las portadoras en la dirección principal (portadoras “k”) en dos situaciones.

1. Si hay un fallo en la cabecera de la portadora “n” que contiene los reconocimientos y se pierden todos los subcampos que contienen la información de control para MAC y DLC.
2. Si llegan con errores todos los subcampos con información de control para MAC.

Si alguno o todos los paquetes llegaron correctos al receptor o fueron corregidos y se pierden los reconocimientos a nivel de MAC, se realizarán retransmisiones innecesarias. Estas retransmisiones innecesarias consumirán un ancho de banda. Una manera de reducir este fenómeno consiste en emplear dos portadoras “n” en lugar de una única portadora “n”. Por este motivo se optó por crear una nueva configuración con 2 portadoras “n” como se muestra en la figura 4.3. La configuración de la figura 4.2 con una portadora “n” se denominará “modelo 1” y la configuración con 2 portadoras se denominará “modelo 2”. Para el modelo 2 (figura 4.3) se han empleado:

- 9 portadoras lógicas dobles simples
- 3 portadoras lógicas dúplex

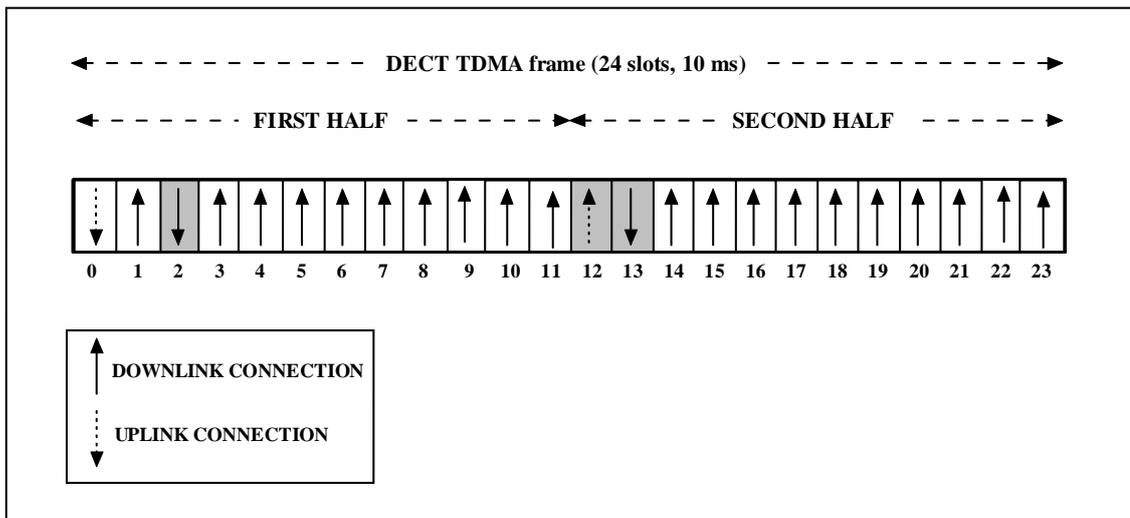


Figura 4. 3 Configuración de las portadoras con dos portadoras “n” y con el enlace downlink usando la conexión de mayor capacidad (Modelo 2)

Veamos distribución de portadoras lógicas para cada conexión.

1. La conexión para el enlace “downlink” consta de 20 portadoras para la dirección principal (portadoras “k”) y 2 portadoras “n” con información de control. Para conseguir las 20 portadoras se han usado 9 portadoras lógicas dobles simplex más 2 portadoras simplex en la dirección principal en los slots 1 y 14 que pertenecen respectivamente a 2 de las 3 portadoras lógicas dúplex. Para las 2 portadoras “n” se han empleado las portadoras simplex en la dirección inversa en los slots 2 y 13 que pertenecen a las 2 portadoras lógicas dúplex mencionadas anteriormente.

2. La conexión para el enlace "uplink" consta de 1 portadora en la dirección principal (portadora "k") y 1 portadora "n" con la información de control. La portadora "k" y "n" se corresponden con las portadoras simplex en los slots 0 y 12 respectivamente las cuales constituyen la tercera portadora dúplex.

A la hora de realizar la distribución de las portadoras en los slots se hicieron para ambos modelos las siguientes consideraciones.

1. La capa MAC realiza las peticiones a la capa DLC antes de que comience la mitad de una TDMA por lo que la información de control para las portadoras que se van a transmitir en esa TDMA debe llegar antes de que empiece la mitad de la TDMA.
2. La posición de las portadoras "k" y "n" determina en cada modelo para que portadoras de que mitad de la TDMA llevará la información de control MAC, la asignación de las portadoras "k" y "n" a cada slot no se ha realizado de manera arbitraria si no que se ha buscado en cada modelo la distribución más adecuada.
3. Si una portadora "k" se encuentra en el slot contiguo a la portadora "n" es muy posible que no le de tiempo al receptor a incluir la información de para esta portadora.
4. Se supone un retardo de procesamiento de la información equivalente a 1 slot.

El caso 3 se da en ambos modelos para la conexión del enlace downlink por lo que cuando ha sido posible se ha introducido una portadora de la conexión del enlace "uplink" entre un portadora "k" de la conexión del enlace "downlink" y la portadora "n" de dicho enlace. Se explica a continuación como se ha resuelto esta situación para cada modelo. Así pues para el caso del modelo 1 se ha evitado por completo la situación del caso 3 insertando la portadora "n" en el slot 21 separando así las portadoras en los slots 20 y 22. En el caso del modelo 2 no ha sido posible solventar por completo esta situación.

Para el modelo 1 (figura 4.2) se decidió que la portadora con la información de control debería ir colocada al final de la TDMA, para que de este modo pudiera transportar la información de control para todas las portadoras de esa TDMA. De esta manera la portadora "n" en el slot 22 contiene la información de control para las 18 portadoras "k" de la misma TDMA. Para la conexión del enlace "uplink", la portadora "n" en el slot 12 contiene la información de control MAC para la portadora en el slot 0 de la primera mitad de la TDMA.

Para el modelo 2 (figura 4.3) la portadora "n" de la primera mitad de la TDMA contiene la información de control para las portadoras "k" de la primera y la segunda mitad de la anterior TDMA (excepto para la portadora en el slot 1). La portadora "n" de la segunda mitad de la TDMA contiene la información para la primera mitad de la actual TDMA y la segunda mitad de la anterior TDMA.

La consideración de un retardo de procesamiento en los equipos provoca que para el modelo 1, los slots 11 y 23 permanezcan sin portadora de datos ya que las peticiones del enlace "downlink" para la siguiente TDMA se hacen cuando llega la portadora "n" en el slot 22 y se considera que la capa MAC tarda 1 slot en procesar la información entregada por las capas superiores.

4.2.2.2 Procedimiento de intercambio de conexiones

Los dos modelos descritos en el apartado anterior funcionarán correctamente para el caso de que la capacidad ofrecida por la conexión del enlace “uplink” sea suficiente para soportar el tráfico entregado por la parte portátil. Sin embargo, en determinados momentos puede ser que de repente aumente el tráfico ofrecido (por acumulación de ACKs). En estas condiciones el tamaño del buffer de SDUs del extremo transmisor del enlace puede aumentar de manera más o menos exponencial pudiendo llegar a saturarlo. Para remediar este incremento se podrá limitar el tiempo de vida de la SDUs usando el procedimiento que se ha descrito en la capa DLC. Cuando una SDU exceda el tiempo máximo permitido será eliminada provocando una pérdida de un reconocimiento TCP para el servidor. Como la fuente emisora de paquetes implementada en el servidor es una fuente “TCP Reno”, la pérdida de varios reconocimientos TCP podrá provocar que pase a un estado de “Congestion Avoidance” o en el peor caso “Slow Start”. Todo ello se traducirá en una reducción de la ventana de transmisión de paquetes IP en el transmisor, provocando una reducción de la tasa de llegada de los paquetes al sistema DECT. Con objeto de solucionar esta situación se podrá implementar un procedimiento de intercambio de las conexiones MAC entre los enlace dentro del sistema DECT. Este procedimiento se puede dividir en dos fases. Una primera fase de petición del intercambio. El procedimiento para esta primera fase consta de 3 pasos:

1. El extremo transmisor del enlace que usa la conexión de menor capacidad comprueba el número de SDUs en su buffer después de obtener la información de control DLC. Si el nivel del buffer se encuentra por encima de un límite permitido (“threshold”), mandará una petición al otro enlace para que le ceda la conexión de mayor capacidad.
2. Cuando el extremo transmisor del enlace que usa la conexión MAC de mayor capacidad recibe una petición para ceder la conexión, comprueba si el número de SDUs en su buffer se encuentra por debajo de un límite (“threshold”). En caso afirmativo cederá la conexión. En caso negativo se la niega.
3. Cuando el extremo transmisor que pidió la conexión recibe una confirmación afirmativa libera su conexión y comienza el procedimiento de la fase 2 para apoderarse de la conexión de mayor capacidad. El otro enlace comienza el proceso para apoderarse de la conexión de menor capacidad que se acaba de liberar. En caso de recibir una confirmación negativa, seguirá mandando peticiones hasta que sea necesario o hasta que reciba una confirmación afirmativa.

Para implementar en el simulador esta primera fase se han hecho las siguientes consideraciones con objeto de simplificar el procedimiento:

1. Se ha supuesto que los mensajes de petición y aceptación o rechazo serán recibidos por los módulos en la primera portadora “n” con información de control para el enlace. Además por tratarse de mensajes del plano de control se va a suponer que siempre llegan correctos.
2. Las portadoras “n” siempre transmiten información de control DLC en cada TDMA.

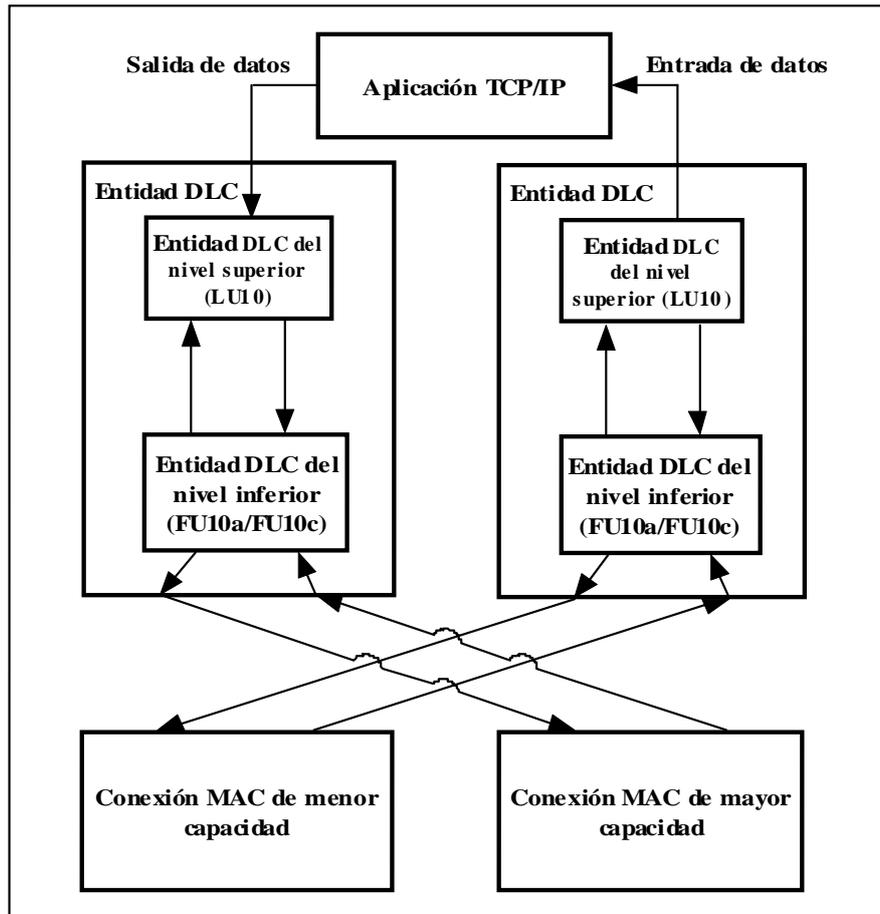


Figura 4. 4 Reasociación de las entidades DLC a las conexiones MAC

La fase 2 consiste en realizar el intercambio de las conexiones. Para este intercambio será necesario suspender el plano de usuario, liberar la conexión, modificar la nueva conexión y resumir el plano de usuario. Estos pasos aparecen descritos en el procedimiento de suspensión y resumen descrito en el apartado del estándar DLC, así como los mensajes del plano de control necesarios para llevarlo a cabo. Para poder simplificar el procedimiento se han hecho las siguientes consideraciones respecto a esta segunda fase:

1. Cuando un enlace acepta cambiar su conexión por la de menor capacidad comenzará el procedimiento de suspensión y resumen. Para simplificar todo el procedimiento (así como los mensajes del plano de control) se ha decidido que las transmisiones de paquetes permanezcan detenidas el tiempo mínimo que se considera que se va a tardar en completar el procedimiento de modificación de la conexión (suponiendo que los mensajes del plano de control llegan correctos).
2. Durante el procedimiento de modificación de la conexión el enlace que recibe la conexión de mayor capacidad deberá cambiar la dirección de las portadoras "k" para lo cual empleará el procedimiento "Fast Release" descrito en [7]. Dicho procedimiento establece que únicamente se puede cambiar la dirección de las portadoras dobles simplex, pero no el de las portadoras dúplex. Las portadoras simplex que pertenezcan a las portadoras dúplex seguirán teniendo la misma

dirección y únicamente cambiará la portadora que se empleará como portadora “n”.

3. Para el caso del enlace que pasa a usar la conexión de mayor capacidad va a tener que esperar como mínimo un número de 4 TDMA que es el tiempo mínimo que establece el estándar para invertir la dirección de las portadoras “k”. El enlace que va a usar la conexión de menor capacidad deberá emplear un tiempo mínimo de 3 TDMA que se ha estimado como el tiempo que tardará en apoderarse de la conexión liberada por el otro enlace y resumir el plano de usuario.

En la figura 4.4 se muestra la reasociación de las entidades del nivel de enlace a la nueva conexión. Para simplificar se ha eliminado en la figura la presencia de la entidad de gerencia del nivel de enlace (LLME). Esta entidad es la que se encarga de manejar todo el procedimiento para el intercambio de las conexiones. Si se desea conocer el resto de funciones que puede hacer la LLME acudir al apartado 2.7.1 del capítulo 2.

En la figura 4.5 y la figura 4.6 se muestra el aspecto de las nuevas configuraciones de las portadoras para los modelos 1 y 2 respectivamente después de aceptar un intercambio de la conexión.

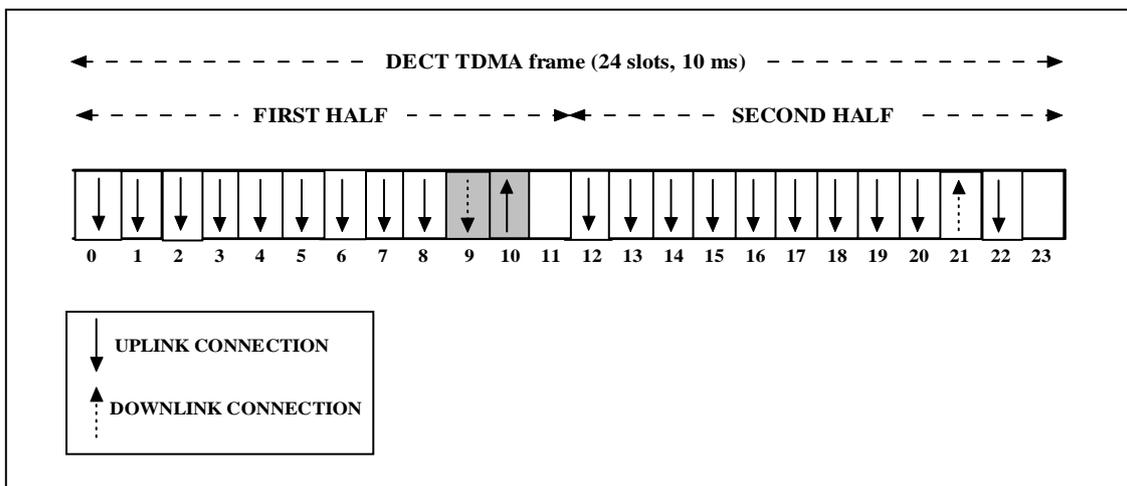


Figura 4. 5 Configuración de las portadoras con una portadora “n” y con el enlace uplink usando la conexión de mayor capacidad (Modelo 1)

Si se comparan las configuraciones antes y después de intercambiar a conexiones para ambos modelos, se observa que ha cambiado la dirección de las portadoras dobles simplex, mientras que las parejas de portadoras que constituyen las portadoras lógicas dúplex mantienen la misma dirección. En la figura 4.5 (modelo 1) y la figura 4.6 (modelo 2) se ve como la conexión de mayor capacidad pertenece ahora al enlace “uplink” y la conexión de menos capacidad al enlace “downlink”.

La diferencia fundamental respecto a la configuración de la figura 4.2 (modelo 1) y la figura 4.3 (modelo 2) es que además de la dirección, ha variado también la portadora que se emplea como portadora “n” en ambas conexiones y por tanto los instantes en que se recibe la información de control para cada conexión.

Para el caso de la conexión de mayor capacidad del modelo 1 la antigua portadora “n” en el slot 22 pasa ahora a tener la misma dirección que el resto de las portadoras simplex y se emplea para transportar datos del enlace uplink. Por otro lado la

portadora en el slot 10 que tenía antes la misma dirección que el resto de portadoras, ahora tiene la dirección inversa y se usa ahora para transportar la información de control. Para la conexión de menor capacidad ahora la portadora “n” será la portadora en el slot 9 y la portadora en el slot 21 pasará a transportar la información del enlace “downlink”.

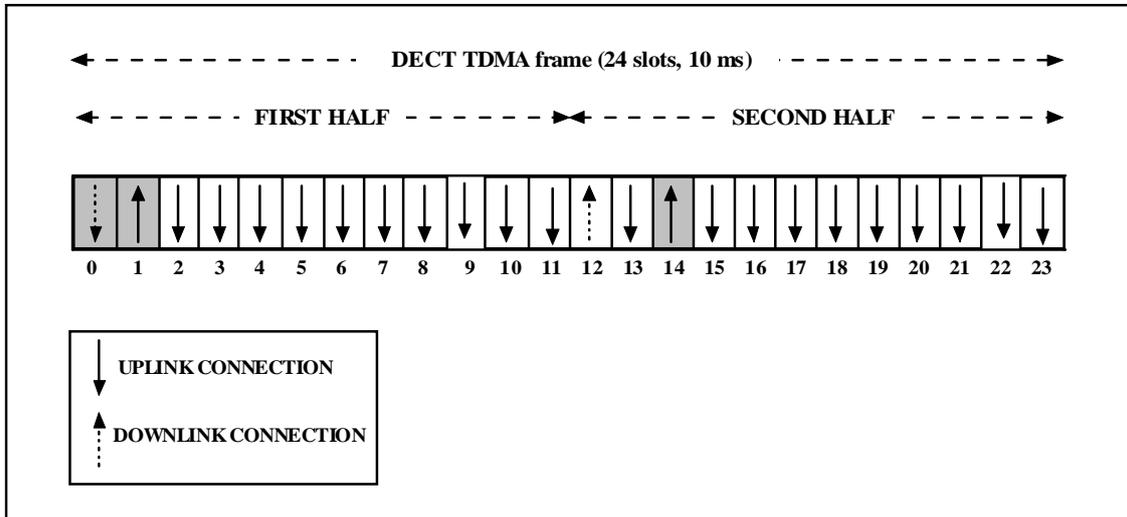


Figura 4. 6 Configuración de las portadoras con dos portadoras “n” y con el enlace uplink usando la conexión de mayor capacidad (Modelo 2)

Para el modelo 2 las antiguas portadoras “n” de la conexión de mayor capacidad de los slots 2 y 13, pasan a tener ahora la misma dirección que el resto de portadoras simplex, mientras que sus respectivas parejas en los slots 1 y 14, pasan a tener la dirección inversa y se convierten en las nuevas portadoras “n”. Por su parte la pareja de portadora que constituyen la portadora lógica dúplex de la conexión de menor capacidad intercambian también sus papeles siendo en la nueva configuración la portadora en el slot 0 la nueva portadora “n” y la portadora en el slot 12, la nueva portadora “k”.

En un apartado anterior se ha comentado la importancia de la distribución de las portadoras. Como se ha comentado antes, al intercambiar las conexiones, se ha cambiado también la posición de las portadoras “n”. A continuación se expondrán las consecuencias de variar la posición de la portadora “n” para cada modelo en el caso de use el modo de corrección de errores en la capa MAC.

Como la nueva portadora “n” de la conexión de mayor capacidad para el modelo 1 de la figura 4.5 está situada ahora en la primera mitad de la TDMA, no transporta información de control para las portadoras de la actual TDMA sino que transporta información de control para las portadoras de la primera mitad de la actual TDMA y para las portadoras de la segunda mitad de la anterior TDMA. En cuanto a la situación comentada en el apartado 4.2.2.1 de que una portadora “k” se encuentre en el slot anterior al de la portadora “n”, vemos como sigue perfectamente resuelto gracias a que la portadora “n” de la conexión de menor capacidad separa la portadora “k” del slot 8 de la portadora “n” del slot 10.

Para el modelo 2 de la figura 4.6, las portadoras “n” siguen estando en la primera y segunda mitad de la TDMA, pero desplazadas un slot. Este desplazamiento provoca una variación como consecuencia de la situación comentada en el apartado 4.2.2.1 sobre la incapacidad de resolver por completo el problema de tener una portadora “k”

en el slot anterior al de la portadora “n”. Así pues la portadora “n” de la primera mitad de la TDMA seguirá incluyendo ahora la información de control para las portadoras de la primera y la segunda mitad de la anterior TDMA. La portadora de la segunda mitad de la TDMA seguirá incluyendo la información de control para la segunda mitad de la anterior TDMA y para la primera mitad de la actual TDMA, pero no así la información de la portadora en el slot 13 por encontrarse en el slot anterior al de la portador “n”.

4.2.2.3 Servicios implementados

Los procedimientos implementados para la capa MAC se corresponden con los siguientes servicios:

Servicio IP_corrección_error. Las principales características de este servicio son:

- Retransmisiones automáticas a nivel de MAC usando el procedimiento MOD-2. Dicho procedimiento ha sido descrito en profundidad en el capítulo 2 en el apartado dedicado a la capa MAC.
- Los paquetes con errores en el campo de datos no son entregados a DLC.
- Servicio de mínimo retardo. Si el campo de datos es correcto se entrega la PDU a DLC en el momento en el que se tiene la información
- En caso de usar el formato de protección de múltiples subcampos existe la posibilidad de almacenamiento y posterior corrección de paquetes en el receptor mediante la comparación de subcampos.
- Cuando se usa para los paquetes el formato de protección de un solo campo, se pierde la posibilidad de poder corregir los paquetes en el receptor por comparación de subcampos, pero a cambio aumenta la cantidad de información que se puede introducir en un paquete.

Servicio IP_detección_error. Las principales características de este servicio son:

- No existen retransmisiones automáticas a nivel de MAC.
- Los paquetes con errores en el campo de datos no son entregados a DLC.
- No es un servicio de mínimo retardo. Si el campo de datos es correcto se entrega la PDU a DLC cuando se alcanza la mitad o el extremo de la TDMA.
- En caso de usar el formato de protección de múltiples subcampos no existe la posibilidad de corregir comparando subcampos puesto que los paquetes no se retransmiten.
- Cuando se usa para los paquetes el formato de protección de un solo campo aumenta la cantidad de información que se puede incluir en un paquete.

Para consultar sobre el resto de servicios que proporciona la capa MAC así como sobre sus características, consultar el apartado del capítulo 2 del presente proyecto dedicado a la capa MAC o bien referirse al estándar [6].

4.2.3 Implementación de la capa DLC

Se han implementado los procedimientos que se describen en el apartado 2.7 de capítulo 2.

4.2.3.1 Implementación del servicio LU10

Siguiendo las recomendaciones del perfil DPRS[15] que se encuentran resumidas en el capítulo 2 del presente documento, el servicio a implementar por la capa DLC será el servicio LU10 cuyas principales características han sido comentadas ya en apartado 2.7.2 del capítulo 2.

Fragmentación de SDUs en PDUs:

En la figura 3.5 y la se muestran gráficamente ejemplos de la fragmentación de un paquete de las capas superiores entregado a DLC en PDUs para cada tipo de formato de los paquetes físicos a nivel MAC. La influencia del tipo de formato en el tamaño de las PDUs ha sido descrita en el apartado 3.4 del capítulo 3. El procedimiento de fragmentación de LU10 ha sido implementado en una función dentro del módulo "DlcTx". Este módulo se explica en el apartado 4.3.2. Como el servicio LU10 permite incluir fragmentos de distintas SDUs en una misma PDU, la función que implemente el procedimiento debe incluir esta característica. A la hora de implementar estos procedimientos se han seguido las siguientes condiciones:

1. Se obviará el empleo de los indicadores de longitud intermedios conservándose únicamente el indicador de longitud primero. Este indicador de longitud es usado cuando la información a introducir es mayor o igual que la capacidad de la PDU.
2. El tamaño de las SDUs será siempre el mismo y será igual o mayor que el espacio disponible en una PDU vacía. De esta manera se justificará el uso del indicador de longitud primero.
3. Para el caso de las PDUs que estén incompletas, si todavía no han sido incluidas en la ventana de transmisión, serán completadas con información de otras SDUs. Además siguiendo estos principios el número de SDUs distintas por PDU será siempre de 2.

Mencionar que haciendo todas estas consideraciones, únicamente se cometerá un error en el caso de que las PDUs que deberán usar indicadores intermedios y que son aquellas que según estas condiciones contengan información de distintas SDUs. El obviar el uso de los indicadores de longitud representa un porcentaje en bits muy reducido, por lo que no se considera una influencia apreciable a la hora de obtener resultados concluyentes para el simulador.

4.2.3.2 Implementación de la clase 2

Siguiendo los requisitos del perfil DPRS recogidas en la tabla 9 incluida en [15] se han implementado los procedimientos de transmisión y recepción de la clase 2, que es el procedimiento que establece DPRS para el servicio LU10. La descripción de estos procedimientos se ha realizado en el apartado 2.1.1.1 del capítulo 2 del presente documento. En los siguientes apartados se describirá el procedimiento usado para simular los procedimientos de esta clase.

Implementación del esquema de retransmisión:

El estándar [7] define dos procedimientos posibles a la hora de implementar el procedimiento de petición de retransmisiones:

- **Go-Back-N:** en el mensaje de petición de retransmisión se manda el número de secuencia del último paquete que le ha llegado correcto. Cuando el transmisor recibe este mensaje, comienza a retransmitir todos los paquetes desde ese número de secuencia hasta el número de secuencia del último paquete perdido.
- **SRP:** el mensaje de petición de retransmisión, incluye un NACK por cada paquete perdido (fuera de secuencia) que incluye el número de secuencia de dicho paquete. Cuando el transmisor recibe un mensaje NACK, retransmite sólo y exclusivamente el paquete con ese número de secuencia

La desventaja de usar el primer procedimiento, es que en muchos casos se realizan retransmisiones innecesarias, por lo que si el ancho de banda es limitado y el canal tiene demasiados errores, puede revelarse como un procedimiento menos eficiente.

Dado que en el caso de los sistemas DECT, el ancho de banda es un recurso limitado y que la interfaz aérea está sometida a interferencias que generan errores en los paquetes, el mismo estándar recomienda el uso de SRP. Aunque SRP sea más difícil de implementar, será el más adecuado para las condiciones en las que se desarrolla nuestro simulador. Para su implementación se ha seguido el procedimiento descrito en [21].

Sincronización de la ventana:

Para limitar el tiempo máximo de vida de los paquetes IP entregados por las capas superiores a la capa DLC (SDUs) en el extremo transmisor de la capa DCL se ha implementado el procedimiento de Time out. El funcionamiento de dicho procedimiento es el siguiente:

Los paquetes IP entregados por las capas superiores a DLC se almacenan en un Buffer. Cuando DLC recibe un mensaje de inicio de trama, busca en el buffer e incrementa el contador para cada una de los paquetes. A continuación busca si algún paquete ha excedido el número máximo de tramas TDMA permitidas. En caso afirmativo, se manda un auto mensaje de Time out indicando que ha expirado un paquete y se elimina del buffer. Cuando se recibe este auto mensaje, se buscan en el buffer de fragmentos las PDUs que contiene información de un paquete que ha expirado y se marcan como pertenecientes a dicho paquete. Si un fragmento (PDU) contiene información de un paquete que no ha expirado, entonces no se marcará como expirada. El módulo "DlcTx" entra en estado de petición de sincronización.

Cuando llega un mensaje al módulo "DlcTx" con la petición de datos de MAC, comienza el procedimiento de sincronización de la ventana. Si han expirado todos los fragmentos (PDUs) de la ventana, se manda un mensaje de sincronización para incluir en todas las portadoras MAC libres con el número de secuencia de la última PDU que ha expirado para que actualice su ventana de recepción. En el caso de que no hayan expirado todas las PDUs de la ventana de transmisión, la última PDU expirada será sustituida por un mensaje de sincronismo y será tratada como una PDU normal. El transmisor almacena en una variable el número de secuencia de la última PDU que ha expirado.

Cuando el receptor recibe un mensaje para sincronizar la ventana, entonces entra en estado de recepción de mensaje de sincronización, actualiza la ventana de recepción y

manda confirmación de sincronización al transmisor en un mensaje especial que contiene el número de secuencia de la PDU que se recibió. Si en el tiempo en que el transmisor está esperando el mensaje de sincronización expira un nuevo paquete entregado por las capas superiores (SDU), se volverá a realizar el mismo procedimiento, actualizando el valor de la última PDU que expiró.

Cuando llegue el mensaje de confirmación de sincronización el transmisor comparará el valor almacenado de la última PDU expirada con el valor devuelto por el receptor. Si los valores no coinciden, se considerará que se han producido nuevas expiraciones mientras se esperaba la confirmación de sincronización. En el caso de que coincidan ambos números, se eliminarán las PDUs que han expirado y se sincronizará la ventana del transmisor. El transmisor pasará entonces a estado normal.

4.2.4 Implementación del tráfico TCP/IP

El servidor implementará los procedimientos de una fuente TCP reno. Este tipo de fuentes usan los siguientes algoritmos:

- Slow Start
- Congestion Avoidance
- Fast Retransmit
- Fast Recovery

Para la implementación de cada uno de ellos se ha consultado el libro titulado "TCP/IP Illustrated" [30]. El algoritmo Slow Start emplea en el servidor de nuestro modelo una ventana de congestión (cwnd). Esta ventana de congestión indica el número de paquetes que nuestro servidor puede mandar sin tener confirmación (ACK) para ellos. Al inicio de la conexión la ventana de congestión vale 1. El servidor manda el primer paquete y recibe el paquete ACK para este paquete aumenta de 1 a 2 el tamaño de la ventana de congestión de manera que la próxima vez el servidor transmite dos paquetes. Cuando el servidor recibe la confirmación para estos dos paquetes, incrementa entonces la ventana del transmisor de 2 a 4. Este procedimiento proporciona un incremento exponencial del tráfico ofrecido por el servidor. Existe un tamaño máximo de ventana de congestión permitido (rwnd) necesario para evitar la congestión del receptor. Este tamaño es proporcionado por el receptor al transmisor durante la fase de establecimiento de la conexión. Como en nuestro modelo se considera que la conexión TCP entre el servidor y la aplicación ha sido ya establecida, todos los valores que deben ser intercambiados durante esta fase se consideran ya conocidos por ambos extremos.

Como se ha comentado anteriormente, la pérdida de paquetes IP de datos y de los reconocimientos para estos paquetes no va a tener lugar en la red IP si no que va a estar motivada por los errores introducidos en el canal de un sistema DECT. Esta pérdida de paquetes requerirá una retransmisión de los paquetes tras un tiempo que se denomina RTT. Los procedimientos implementados en el simulador para la implementación del RTT se pueden consultar en [30]. Si hay pérdidas de paquetes será necesario que el tamaño de la ventana del receptor disminuya para evitar congestionar el buffer del router que da acceso al sistema DECT (puesto que se ha supuesto que el resto de routers de la red IP no pierden paquetes). Para implementar este procedimiento se usará el algoritmo "Congestion Avoidance" en combinación con "Slow Start". El funcionamiento combinado es el siguiente:

Cuando se recibe un ACK duplicado de la aplicación del terminal móvil se almacena el tamaño de la mitad de la ventana en una variable denominada "ssthresh". Si lo que se

produce es un time out para un paquete en el servidor entonces se fija el valor de $cwnd$ a 1. Cuando se reconocen nuevos paquetes se incrementa $cwnd$ y se compara con el valor de "ssthresh". Si $cwnd$ es menor o igual a "ssthresh", el servidor pasará al estado "Slow Start" o permanecerá en este estado si ya estaba antes. Si $cwnd$ es mayor que "ssthresh", el servidor pasará al estado "Congestion Avoidance". En dicho estado el tamaño de la ventana de congestión se incrementa en $1/cwnd$ cada vez que se recibe un nuevo reconocimiento. El crecimiento de la ventana será ahora un crecimiento lineal en comparación con el crecimiento exponencial de "Slow Start". Podemos afirmar que la variable "ssthresh" del servidor actúa como una indicadora de la última vez que se produjo una congestión. La aplicación receptora del terminal móvil, genera un ACK duplicado cada vez que tiene paquetes fuera de secuencia en su ventana de recepción. Debido a que los paquetes pueden sufrir distintos retardos a la hora de viajar por la red, se asume en el transmisor (servidor) que el receptor puede emitir hasta dos ACK duplicados antes de que llegue el paquete retardado. Según lo descrito en [30] el transmisor debe asumir que cuando reciba más de 3 ACKS duplicados, el paquete se ha perdido en la red y debe ser retransmitido de manera automática. A este procedimiento de retransmisión en el que no se espera a que venza el timer del paquete se le conoce con el nombre de "Fast Retransmit". Si se emplea este procedimiento, el servidor no pasa al estado "Slow Start" si no que pasa al estado "Congestion Avoidance". A este procedimiento se le conoce como "Fast Recovera". El principio en el que se basa el algoritmo "Fast Recovera" es que si llega una ACK duplicado es porque han llegado nuevos paquetes al receptor y el pasar a "Slow Start" supondría una reducción muy drástica del tráfico ofrecido por el servidor teniendo en cuenta que habría una gran cantidad de paquetes circulando por la red. El procedimiento combinado de "Fast Retransmit" y "Fast Recovera" es el siguiente:

- Cuando el servidor recibe el tercer ACK duplicado, fija el valor de la variable "ssthresh" a la mitad del tamaño de la ventana, retransmite el paquete perdido y fija $cwnd = ssthresh + 3$.
- Cada vez que el servidor recibe un nuevo ACK duplicado, incrementa en 1 el tamaño de $cwnd$ y transmite un nuevo paquete si el nuevo valor de $cwnd$ se lo permite.
- Cuando llega un nuevo ACK hace $cwnd = ssthresh$ y se introduce entonces en el estado de "Congestion Avoidance".

4.2.5 Modelo de errores del canal

Los errores que se producen en los paquetes al viajar a través de la interfaz aérea se han implementado usando un modelo de cadena de Markov [27] de dos estados: bueno (good) y malo (bad). Los dos estados se corresponden con el estado en que se puede encontrar el canal en un determinado momento. En el caso de que el canal se encuentre en estado malo, habrá errores en los paquetes. La posibilidad de permanecer en el mismo estado o realizar una transición entre estados se contempla cada trama TDMA.

En cuanto a la generación de los errores en los paquetes cuando el canal se encuentra en estado malo se ha adoptado dos procedimientos diferentes en función del formato de protección de paquete. A la hora de implementar ambos procedimientos en el simulador, se han hecho las siguientes consideraciones:

- Los campos de control CRC detectan todos los errores (lo que implica que los errores no pueden caer en estos campos).

- Se han omitido los errores de sincronismo y sincronización que se pueden producir cuando hay fallos en la cabecera.
- El tamaño del paquete se expresará en símbolos ya que dicho tamaño únicamente se corresponde con bits reales para el caso de que el nivel de modulación sea igual a 2.
- Cuando se habla de la cabecera del paquete MAC, se considera que es de tamaño igual al del campo A.
- Cuando se habla del campo de datos del paquete MAC se considera que el tamaño es igual al del campo B.
- El nivel de modulación de la cabecera es siempre de 2 independientemente del nivel de modulación del campo de datos.

El procedimiento para cada formato así como el modelo de la cadena de Markov de dos estados se explican en los apartados siguientes

4.2.5.1 Cadena de Markov de dos estados

En la figura 4.7 se ilustra la cadena de Markov:

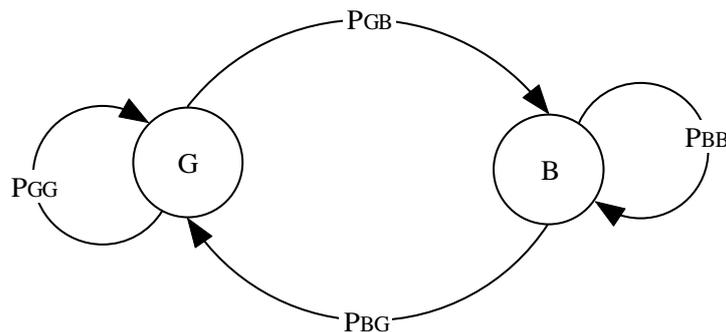


Figura 4. 7 Cadena de Markov de 2 estados

En el modelo se contemplan las siguientes probabilidades:

P_{GG}: "probabilidad de permanecer en estado bueno".

P_{GB}: "probabilidad de pasar de estado bueno a malo".

P_{BB}: "probabilidad de permanecer en estado malo".

P_{BG}: "probabilidad de pasar de estado malo a bueno".

Estas probabilidades se encuentran relacionadas de la siguiente manera:

$$P_{BB} = (1 - P_{BG})$$

$$P_{GG} = (1 - P_{GB})$$

En el fichero de entrada de parámetros del simulador (omnet.ini) se especifican

PGB y PBG y a partir de estos valores se calculan PGG y PBB usando las relaciones matemáticas que se han mostrado. Para modelar la interfaz aérea usando OMNET++ se ha creado un módulo “Channel” donde se han implementado el procedimiento para la modelación de errores basado en el uso de la cadena de Markov. Los valores que se han empleado en las simulaciones realizadas con una canal con errores han sido PGB = 0.16 y PBG = 0.84 tal y como se indica en [28]. Por tanto el canal tendrá cada nueva TDMA una posibilidad del 16% de pasar de estado bueno a malo y una probabilidad del 84% de pasar de estado malo a bueno.

4.2.5.2 Formato de protección de un solo campo

Para este formato como se ha mencionado anteriormente en el capítulo dedicado a la capa MAC, el paquete MAC consta de una cabecera, y un único campo de datos. Cuando se produzca un error podrá caer en la cabecera del paquete o bien en el campo de datos. Para modelar la generación de errores se han considerado dos probabilidades:

P_{HF} : “probabilidad de que el error caiga en la cabecera”.

P_{DF} : “probabilidad de que el error caiga en el campo de datos”.

Para el cálculo de estas probabilidades se han empleado las siguientes fórmulas:

$$PktSize = HeadSize + DataSize$$

$$P_{HF} = \frac{HeadSize}{PktSize}$$

$$P_{DF} = \frac{DataSize}{PktSize}$$

Donde:

- HeadSize: “tamaño de la cabecera del paquete”.
- DataSize: “tamaño del campo de datos del paquete”.
- PktSize: “tamaño del paquete MAC”.

El procedimiento a seguir es el siguiente: cada vez que el canal tiene que marcar un paquete con errores, genera un valor siguiendo una distribución uniforme entre 0 y 1. Si este valor es menor o igual que la probabilidad de que haya error en la cabecera, se marcará con un error. En caso contrario se marcará un error en el campo de datos. En la tabla 4.1 se muestran los valores usados en las simulaciones. Los tamaños de los distintos campos se expresan en símbolos ya que un mismo símbolo puede representar varios bits en función del nivel de modulación. Estos valores pueden ser consultados en el apartado 2.6 dedicado a la capa MAC del capítulo 2 o más a fondo en el estándar [6].

Head size	104
Data size	320
PktSize	424
PHF	0.245 (24.5 %)
PDF	0.754 (75.4%)

Tabla 4. 1 Probabilidad de que haya errores en la cabecera o en el campo de datos

4.2.5.3 Formato de protección de subcampos.

Como se ha comentado en el capítulo dedicado a la capa MAC, en este formato el campo de datos se divide en varios subcampos de longitud fija (8 bytes) cada uno con su CRC particular. Cuando se produzca un error, este podrá caer en la cabecera o en uno de los subcampos de datos.

Para modelar la generación de errores en este caso, se ha considerado que los errores para el campo de datos caen en los subcampos. Es necesario conocer por ello el número de subcampos en que se divide el campo de datos para cada nivel de modulación. Los valores que nos interesan se pueden consultar en la tabla 2.2. El procedimiento empleado para asignar el fallo cuando cae en la cabecera es el mismo que para el caso del formato de protección de un solo campo. El procedimiento varía para el campo de datos puesto que su tamaño (es menor que para el caso de un solo campo) se considerará como la suma del tamaño de todos los subcampos expresado en símbolos. Cuando el fallo se localiza en el campo de datos entonces se volverá a utilizar una variable uniforme distribuida entre 0 y 1 para asignar los errores a cada uno de los subcampos de datos. En la tabla 4.1 se muestran los valores usados en las simulaciones.

4.3 Descripción de los módulos

En el siguiente apartado se va proceder a mostrar los módulos que se han empleado para construir el simulador indicando las relaciones existentes entre ellos y la información que viaja por los canales que interconectan los módulos. Para definir dichos módulos se ha empleado el lenguaje .ned proporcionado por OMNET++. En la figura 4.8 se muestra un esquema donde aparece un esquema de la composición de los módulos.

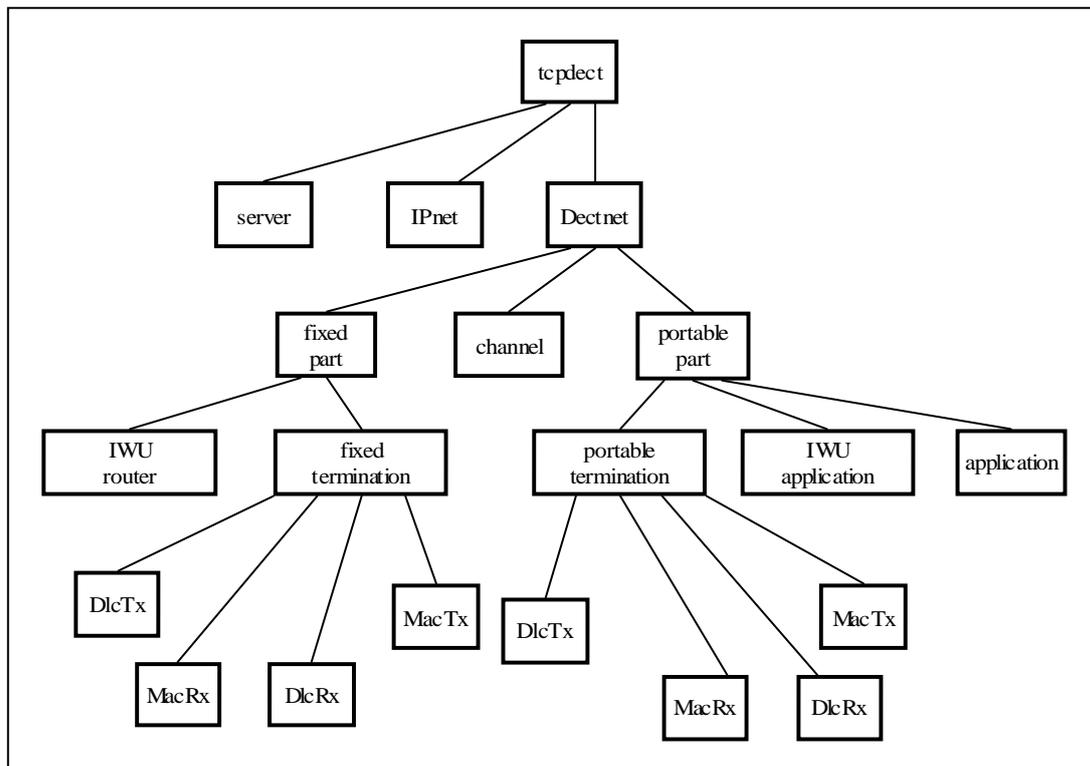


Figura 4. 8 Esquema que muestra la relación de jerarquía entre los módulos del simulador

4.3.1 Módulo principal “tcpdect”

El nombre del módulo principal que contiene a todos los demás es “tcpdect”. Para diseñar el contenido del módulo nos basamos en la figura 3.2 del capítulo 3 en la que se muestra un servidor que se comunica con un sistema DECT a través de una red IP. Así pues como se muestra en la figura 4.9 tenemos un módulo “server” que implementa las funciones del servidor descrita en el apartado 3.2 del capítulo 3. Puesto que la función de encaminamiento de red IP no es objeto de implementación, la única función de este módulo será la de introducir introduce un retardo. Se ha realizado una asignación de velocidad al enlace que une los módulos “server” e “IPnet” que representa la velocidad de transferencia del enlace del servidor.

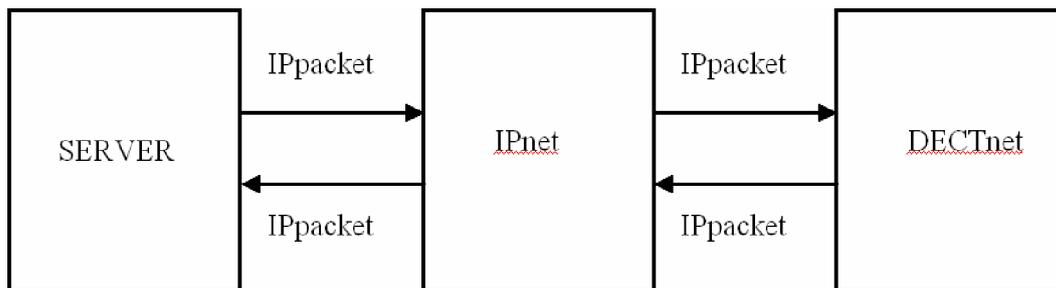


Figura 4. 9 Contenido del módulo “tcpdect”

4.3.2 Módulo “dectnet”

Los componentes del sistema DECT se encuentra incluidos en el módulo “dectnet”. Para el diseño de los módulos que componen el sistema DECT, se ha usado la del capítulo 2 en donde se describen los componentes de un sistema DECT. Así pues se ha creado un módulo para la parte fija llamado “fixedpart” y un módulo para la parte portátil llamado “portablepart”.

- El módulo “fixedpart” por un lado recibe/transmite paquetes de/hacia la red IP. Cuando recibe un paquete IP, transmite paquetes MAC y recibe confirmaciones de estos paquetes para la conexión “downlink”. Cuando recibe paquetes MAC, comprueba si forman un paquete IP, en cuyo caso lo envía hacia la red IP y transmite confirmaciones para la conexión “uplink”.
- El módulo “portablepart” para la conexión “downlink”, recibe los paquetes MAC y manda confirmaciones y para la conexión “uplink” manda los paquetes y recibe las confirmaciones.

Ambos módulos se comunican a través de una interfaz aérea que se ha implementado empleando un módulo canal. Este módulo recibe paquetes de datos y reconocimientos de una y otra conexión y los reenvía con o sin errores hacia el módulo destino. Todo lo descrito se muestra en la figura 4.10.

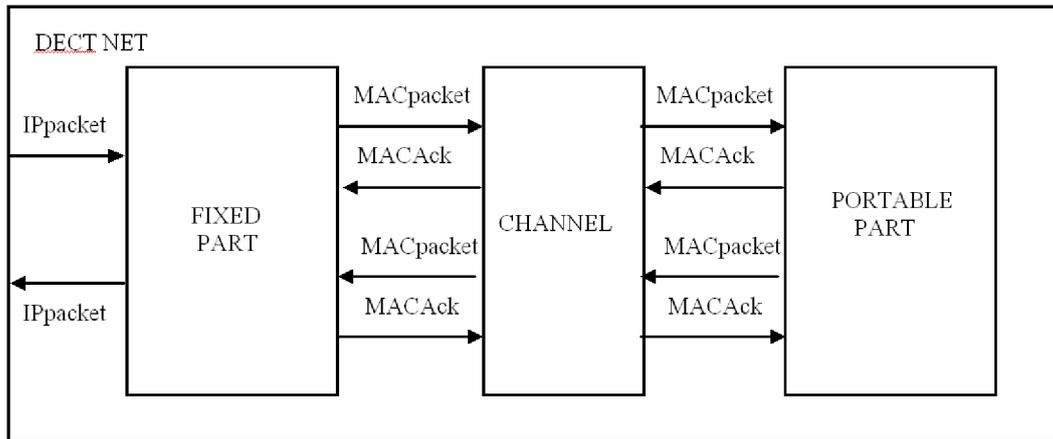


Figura 4. 10 Contenido del módulo “dectnet”

Dentro del módulo “fixedpart” tal como se muestra en la figura 4.11 nos encontramos dos módulos que han sido creados siguiendo la figura 3.2 del capítulo 3. Estos módulos son:

- “IWUrouter”, modela la IWU que actúa como router tal y como se comentó en el apartado 3.2 del capítulo 3. Por un lado recibe de la red IP los paquetes IP con datos del módulo “server” y transmite los paquetes que contienen los reconocimientos TCP para estos paquetes junto con las peticiones de datos. Por otro lado convierte los paquetes IP en SDUs que envía a la terminación fija y recibe SDUs de la terminación fija que convierte en paquetes IP que envía ala red IP.
- “fixedtermination”, modela la terminación fija y de acuerdo con la figura 3.2 del capítulo 3 e implementa las funciones de las capas del protocolo del sistema DECT. Estas funciones permiten que reciba SDUs que transmite hacia la terminación portátil en forma de paquetes físicos MAC, y también reciba paquetes físicos MAC que une y transforma en SDUs que son entregadas al módulo “IWUrouter”. Así mismo realiza el control de flujo de estos paquetes MAC mediante el empleo de un mecanismo de transmisión y recepción de paquetes MAC con reconocimientos a nivel MAC y a nivel DLC.

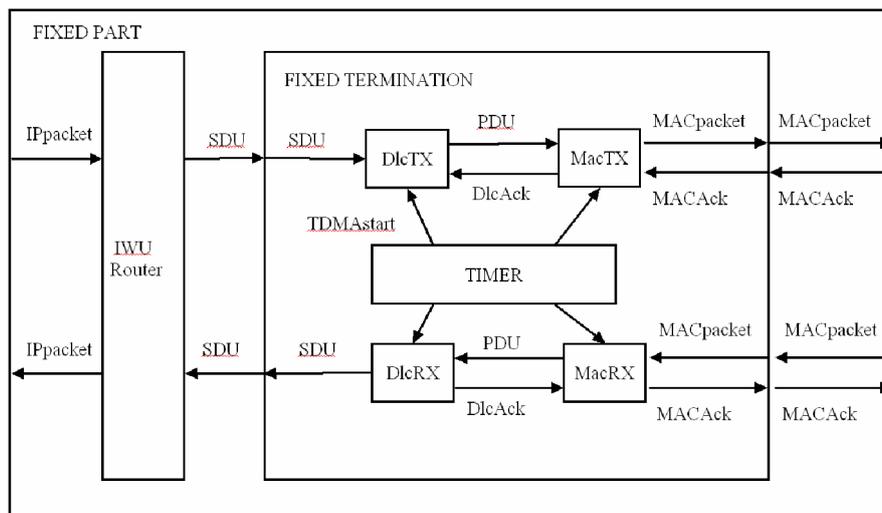


Figura 4. 11 Contenido del módulo “fixedpart” y “fixedtermination”

El contenido del módulo “portablepart” se muestra en la figura 4.12. De acuerdo con la estructura descrita en el apartado 2.2.1 del capítulo 2, se han creado tres módulos:

- El modulo “portabletermination”, que realiza las funciones de la terminación portátil establecidas por el estándar base DECT. Estas funciones permiten:
 1. Que las SDUs que recibe del módulo “IWUapplication” sean fragmentadas en paquetes físicos MAC que envía hacia la terminación fija
 2. Que los paquetes físicos MAC procedentes de la terminación portátil que recibe a través del canal sean ensamblados y transmitidos en forma de SDUs hacia la “IWUapplication”.
 3. Así mismo al igual que el módulo “fixedtermination” realiza el control de flujo de estos paquetes MAC mediante el empleo de un mecanismo de transmisión y recepción de paquetes MAC con reconocimientos a nivel MAC y a nivel DLC.
- El módulo “IWUapplication”, es necesario para realizar la conversión del formato de los paquetes que viajan por la interfaz aérea del sistema DECT a un formato de paquetes comprensible entre la aplicación TCP/IP y la terminación portátil. Por un lado recibe las SDUs de la terminación portátil que transforma en paquetes IP con datos del módulo “server” para la aplicación. Por otro lado recibe los paquetes IP con reconocimientos para los paquetes de datos y peticiones de nuevos datos al servidor y los envía a la terminación portátil en forma de SDUs.
- El módulo “application” representa una aplicación TCP/IP que actúa como un programa que emplea TCP /IP como protocolo a nivel de transporte. Esta aplicación recibe de la “IWUapplication” datos del servidor en forma de paquetes IP y entrega paquetes IP ala “IWUapplication” para el servidor que contienen reconocimientos y nuevas peticiones.

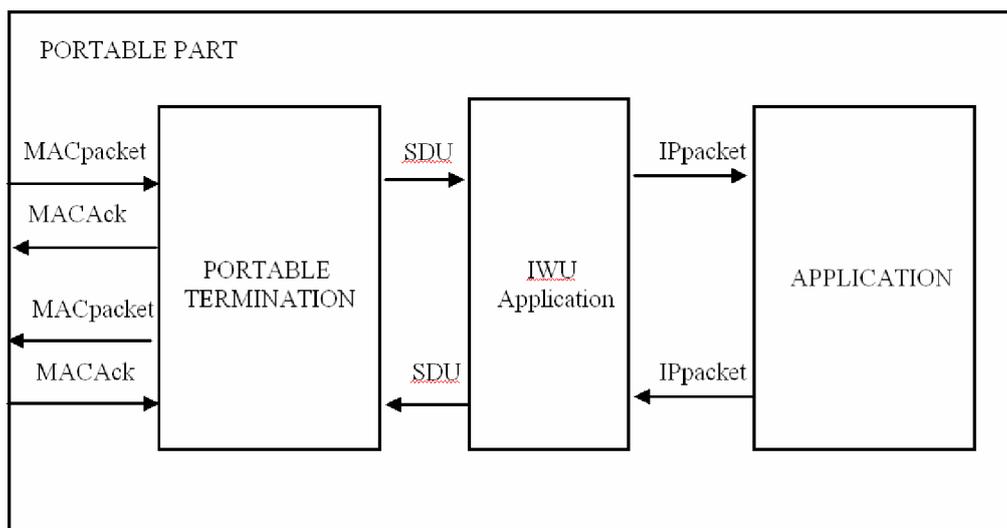


Figura 4. 12 Módulo “portablepart”

En el interior del módulo “fixedtermination” según la figura 3.2 del capítulo 3, nos encontramos con la entidad que se encarga de las funciones de control de la parte fija

(CCFP) y la estación que contiene las antenas para el acceso vía radio (RFP). Es en este punto donde se crean los módulos que van a contener las funciones de las capas que implementan la terminación fija. Para el caso del módulo “portabletermination” deberemos emplear estos mismos módulos para contener las funciones del protocolo en la terminación portátil.

Así pues para modelar los extremos que constituyen el enlace a nivel de la capa DLC se crearon dos tipos de módulos:

- El módulo “DlcTx” que representa el extremo transmisor del enlace DLC. Este módulo recibe por un lado SDUs del módulo que representa la IWU (router o application) y las fragmenta en PDUs que entrega a la capa MAC para que las envíe al extremo receptor del enlace DLC. Por otro lado recibe a través de la capa MAC la información de control (DlcAck) generada por el extremo receptor del enlace DLC para el control de flujo.
- El módulo “DlcRx” que representa el extremo receptor del enlace DLC. Este módulo por un lado recibe a través de la capa MAC las PDUs generadas por el extremo transmisor del enlace y las une para formar SDUs. A su vez genera información para el control de flujo (DlcAck) que envía al extremo transmisor del enlace a través de la capa MAC. Por otro lado entrega las SDUs al módulo que representa la IWU (router o application).

Los módulos DLC emplean el servicio LU10 (EFREL) (apartado 2.7.2) para procesar las PDUs y realizar el control de flujo. Los detalles de implementación de este servicio dentro de los módulos de cada extremo son descritos en el apartado 4.2.3 en el que se explica la implementación de los procedimientos de la capa DLC. Para implementar los procedimientos de los extremos que constituyen la conexión a nivel de la capa MAC se crearon dos tipos de módulos:

- El módulo “MacTx” representa el extremo transmisor de la conexión MAC. Por un lado se comunica con el módulo “DlcTx”, del cual recibe PDUs para entregar al extremo receptor del enlace. Así mismo entrega la información a “DlcTx” para el control de flujo del enlace. Por otro lado introduce las PDUs en paquetes físicos MAC que transmite hacia el extremo receptor de la conexión. Del extremo receptor de la conexión recibe la información para el control de flujo de los paquetes físicos MAC.
- El módulo “MacRx” contiene las funciones del extremo receptor de la conexión. Recibe paquetes físicos MAC que entrega al extremo receptor del enlace (“DlcRx”) de la capa DLC en forma de PDUs. Genera la información para el control de flujo de los paquetes físicos MAC que recibe del extremo transmisor de la conexión. Finalmente recibe la información para el control de flujo a nivel de la capa DLC la cual incluye en los paquetes físicos MAC de reconocimiento (MACAck).

El formato de los paquetes físicos MAC depende del nivel de protección y del nivel de modulación seleccionados. Los detalles de implementación de los procedimientos implementados en cada uno de los módulos de los extremos de la conexión MAC se describen en el apartado 4.2.2 dedicado a la implementación de la capa MAC. Como se ha comentado anteriormente los módulos que representan los extremos de la comunicación tanto a nivel MAC como DLC han ido instanciados tanto en el interior del módulo “fixedtermination” como en el módulo “portabletermination” como se muestra en la figura 4.11 y la figura 4.13 respectivamente. En el interior de estos módulos se encuentra también un módulo “Timer” que se encarga de generar el sincronismo para

la trama TDMA. De acuerdo con los descrito en [5] se ha instanciado un módulo "Timer" para la parte portátil y otro para la parte fija. El procedimiento implementado dentro del módulo "Timer" se encuentra descrito en detalle dentro del apartado 4.2.1.

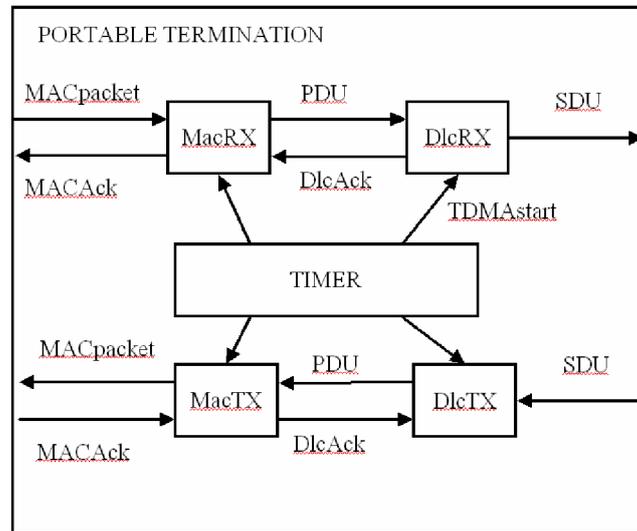


Figura 4. 13 Contenido del módulo "portabletermination"

4.4 Implementación de los mensajes

Todos los mensajes han sido implementados en OMNET++ como objetos derivados de la clase "cMessage". Para la generación automática del código en C++ se han definido las primitivas especificando su contenido (parámetros) en ficheros con extensión ".msg" y se ha hecho uso del comando "opp_msgc" incluido en OMNET++.

4.4.1 Implementación de las primitivas

En este apartado se van a describir las primitivas del estándar que se han implementado para la comunicación entre las capas del protocolo. En la figura 4.14 se representa un ejemplo de comunicación entre las distintas capas. A cada primitiva se le ha asignado en la figura un número para poder explicarla a continuación en los apartados siguientes.

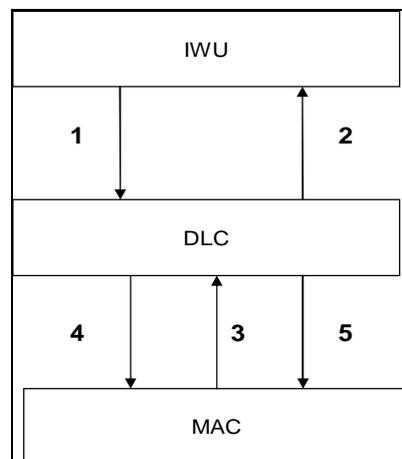


Figura 4. 14 Comunicación entre las entidades

Las primitivas son según la numeración de la figura 4.14:

- 1.- DL_U_DATA-req (IWU à DLC)
- 2.- DL_U_DATA-ind (DLC ß IWU)
- 3.- MAC_CO_DTR-ind (MAC à DLC)
- 4.- MAC_CO_DATA-req (MAC ß DLC)
- 5.- MAC_CO_DATA-ind (MAC à DLC)

4.4.2 Comunicación entre DLC y MAC

Junto al nombre de la primitiva aparece entre paréntesis el número que se le ha asignado para identificarla en la figura 4.14.

4.4.2.1 Primitivas MAC à DLC

MAC_CO_DTR-ind (3):

La capa MAC usa esta primitiva para pedir fragmentos de datos a la capa DLC. Esta primitiva incluye los siguientes parámetros:

- Identificador de la conexión.
- Número de fragmentos que solicita (0...30).

MAC_CO_DATA-ind (5):

Cada vez que llegan datos de un extremo de la conexión, con esta primitiva se entregan los fragmentos a la capa DLC. Los parámetros que contiene son:

- Identificador de la conexión.
- Número de fragmentos.
- Fragmentos.
- Resultado del CRC.

4.4.2.2 Primitivas DLC à MAC

MAC_CO_DATA-req (4):

Con esta primitiva la capa DLC le entrega los fragmentos que solicita la capa MAC para transmitirlos por el enlace. Los parámetros son:

- Identificador de la conexión.
- Número de fragmentos (0...29).
- Fragmentos.

4.4.3 Comunicación entre IWU y DLC

Junto al nombre de la primitiva aparece entre paréntesis el número que se le ha asignado para identificarla en la figura 4.14.

4.4.3.1 Primitivas IWU à DLC

DL_U_DATA-req (1):

La IWU usa esta primitiva para entregar fragmentos SDU a la capa DLC. Esta primitiva contiene:

- Número de fragmentos.
- Fragmentos.

4.4.3.2 Primitivas DLC à IWU

DL_U_DATA-ind (2):

Con esta primitiva la capa DLC le entrega los fragmentos SDU a la IWU. Esta primitiva contiene:

- Número de fragmentos.
- Fragmentos.

4.4.4 Implementación de los paquetes físicos que viajan a través de la red

Los paquetes IP que viajan entre el servidor y el router que da acceso al sistema DECT han sido implementados según el formato descrito en [30]. Los paquetes que viajan a través de la interfaz aérea del sistema DECT han sido implementados según los formatos descritos en el apartado 2.1.1 del capítulo 2.

4.4.5 Descripción de la comunicación entre los módulos del simulador

Para la descripción del paso de mensajes entre los módulos del simulador se ha considerado un ejemplo de transmisión de un paquete IP y la recepción de un reconocimiento para dicho paquete.

4.4.5.1 Transmisión de un paquete IP

Para explicar el procedimiento de transmisión de un paquete IP, se usará el diagrama temporal representado por la figura 4.15, figura 4.16 y figura 4.17 en el que se describe un ejemplo de transmisión de un paquete IP desde el servidor hasta la aplicación. En la figura 4.15 se muestra el tránsito de información entre los componentes del sistema, desde el servidor hasta el extremo transmisor de la parte fija y viceversa. Notar que no parece el módulo "Ipnet" puesto que se supone que únicamente introduce un retardo en los paquetes. En la figura 4.16 se muestra el tránsito de paquetes físicos a través del canal entre el extremo transmisor de la parte fija y el extremo receptor de la parte portátil viceversa. Por último en la figura 4.17 se muestra el tránsito de unidades del protocolo a través de primitivas entre el extremo receptor de la parte fija y la aplicación. La configuración de las portadoras para este ejemplo se corresponde con una configuración con dos portadoras "n" como la del modelo 2. El servicio seleccionado en a nivel de MAC es de corrección de errores. El módulo "canal" no introduce errores en la TDMA en la que se realizan las transmisiones. A continuación se describe paso a paso todo el procedimiento:

Cuando el módulo “IWUrouter” recibe el paquete IP (IP packet (0)) del módulo “server” (a través del módulo “IPnet”), extrae su información y lo reenvía en forma de SDU (SDU 0) al módulo “DlcTx” haciendo uso de la primitiva “DL_U_DATA-req” (figura 4.15). Cuando el módulo “DlcTx” de la FP recibe la SDU la fragmenta en “m + n” PDUs que almacena en el buffer en espera de la petición de datos de MAC (figura 4.15). Mientras espera la petición de MAC el módulo “DlcTx” recibe un reconocimiento (DlcAck) con información de control para la ventana. Antes de que comience la primera mitad de la nueva TDMA, el módulo “MacTx” de la FP manda la primitiva “MAC_CO_DTR-ind” al módulo “DlcTx” solicitando “m” PDUs para transmitir las “m” portadoras libres de la primera mitad de la nueva TDMA. El módulo “DlcTx” le responde con la primitiva “MAC_CO_DATA-req” que contiene las “m” PDUs (PDU 0...PDU m-1) (figura 4.15). Cuando comienza la primera mitad de la TDMA, el módulo “MacTx” transmite para cada slot que corresponda uno de los “m” paquetes físicos (MAC PKT 0...MAC PKT m-1) (figura 4.16). Cuando el primer paquete físico (MAC PKT 0) llega al módulo “MacRx” en la PP, éste comprueba que tanto la cabecera como el campo de datos es correcto (). Introduce la información del paquete físico (PDU 0) en una primitiva del tipo “MAC_CO_DATA-ind” y se la envía al módulo “DlcRx” (figura 4.17). Esta operación la repite para todos los paquetes que le llegan en la primera mitad de la TDMA. En este punto hay que matizar que para el caso de que el servicio sea de detección de errores, la información de todos los paquetes físicos se entrega justo en la mitad de la TDMA y no en el momento de recibir el paquete.

El módulo “DlcRx” de la PP obtiene las PDUs de la primitivas “MAC_CO_DATA-ind” y actualiza la ventana de recepción, de manera que la nueva PDU que espera será la PDU m. El módulo “MacRx” en la PP se prepara para transmitir un paquete físico sobre la portadora “n” para la primera mitad de la TDMA (MAC PKT ACK) y solicita la información de control al módulo “DlcRx” haciendo uso de la primitiva “MAC_CO_DTR-ind”. Usando la primitiva “MAC_CO_DATA-req”, el módulo “DlcTx” le entrega la información de control que consiste en un reconocimiento del la última PDU recibida correctamente (DLC ACK m-1) (figura 4.17). El módulo “MacRx” incluye esta información en los subcampos de datos del paquete físico junto con la información de control para las portadoras MAC (BCK y Q2) y lo transmite sobre la portadora “n” en el slot que corresponda (figura 4.16).

El módulo “MacTx” en la FP recibe el paquete físico (MAC PKT ACK) del módulo “MacRx” (figura 4.16) y comprueba que la cabecera es correcta. Como es correcta comprueba los subcampos de datos y obtiene la información de cada uno de ellos. Así pues obtiene la información de control para las portadoras (BCK y Q2) lo que le informa de que todas llegaron correctas y obtiene también la información de control para la capa DLC (Dlc ACK m-1), que envía al módulo “DlcTx” utilizando la primitiva “MAC_CO_DATA-ind”. El módulo “DlcTx” recibe la información de control con el reconocimiento de “DlcRx” y actualiza el borde inferior de su ventana de transmisión confirmando las PDUs de 0 a m-1 (figura 4.15). Antes de que comience la segunda mitad de la TDMA, el módulo “MacTx” solicita al módulo “DlcTx” “n” PDUs para introducir las “n” portadoras de la segunda mitad de la TDMA. El módulo “DlcTx” entrega “n” PDUs al módulo “MacTx” (PDU m...PDU m+n-1). El resto del procedimiento es igual que para las PDUs de la primera mitad. Cuando el módulo “DlcRx” recibe de “MacRx” la última PDU (PDU m+n-1) de la SDU, vuelve a juntar la información útil de las PDUs (PDU 0...PDU m+n-1) en una SDU (SDU 0) que incluye en la primitiva “DL_U_DATA-ind” que envía al módulo “IWUapplication” (figura .17).

Cuando el módulo “IWUapplication” en la PP recibe la SDU, entrega la información en forma de paquete IP (IP packet 0) al módulo “application” (figura 4.17). En este momento comienza la transmisión del reconocimiento hacia el Servidor. Este reconocimiento viaja en dirección de la PP a la FP usando el enlace “uplink”. El

procedimiento general será el mismo que el que se acaba de explicar para la dirección FP a PP y se describe en el siguiente apartado

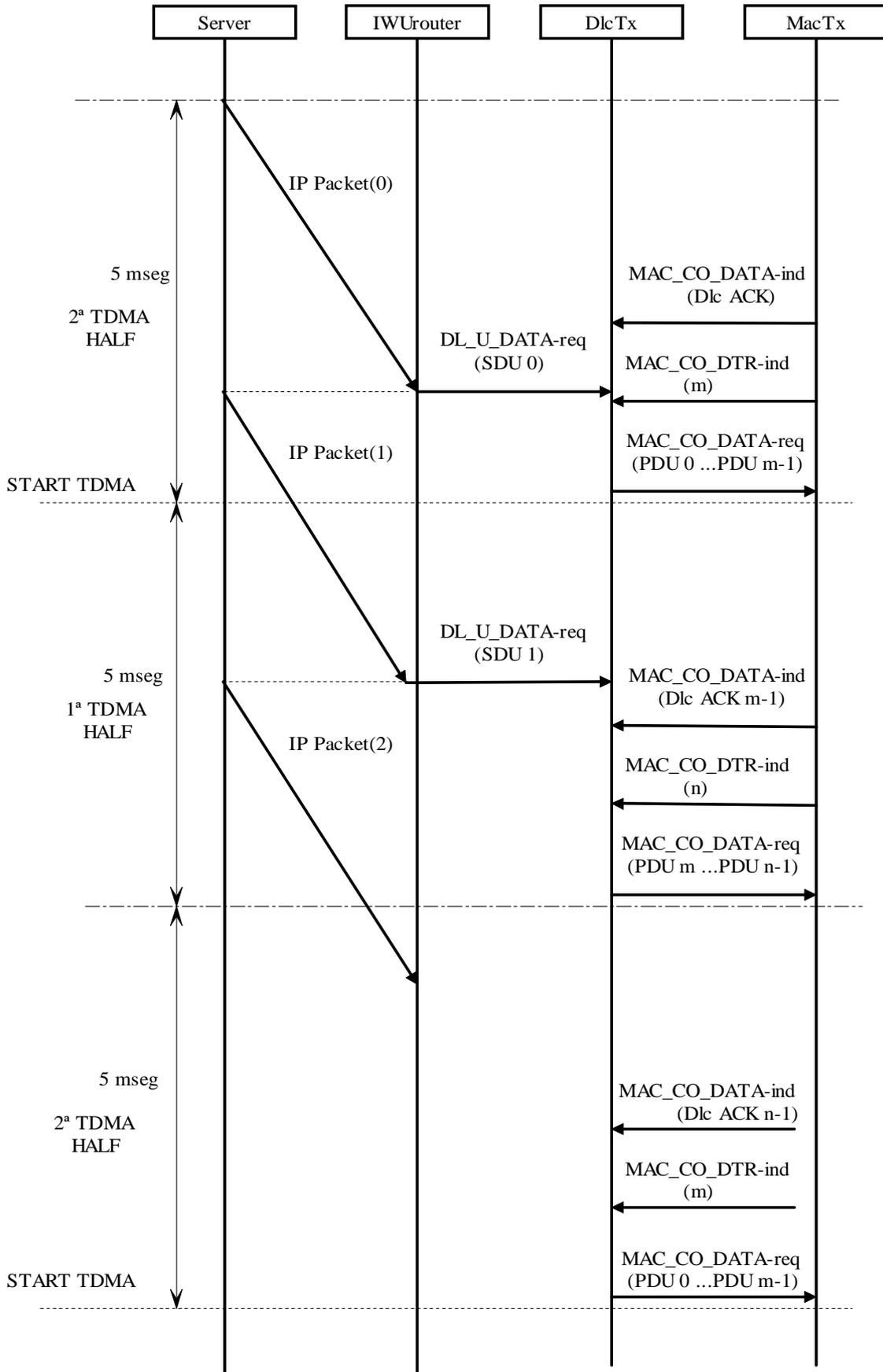


Figura 4. 15 Procedimiento de transmisión (entre el servidor y la parte fija)

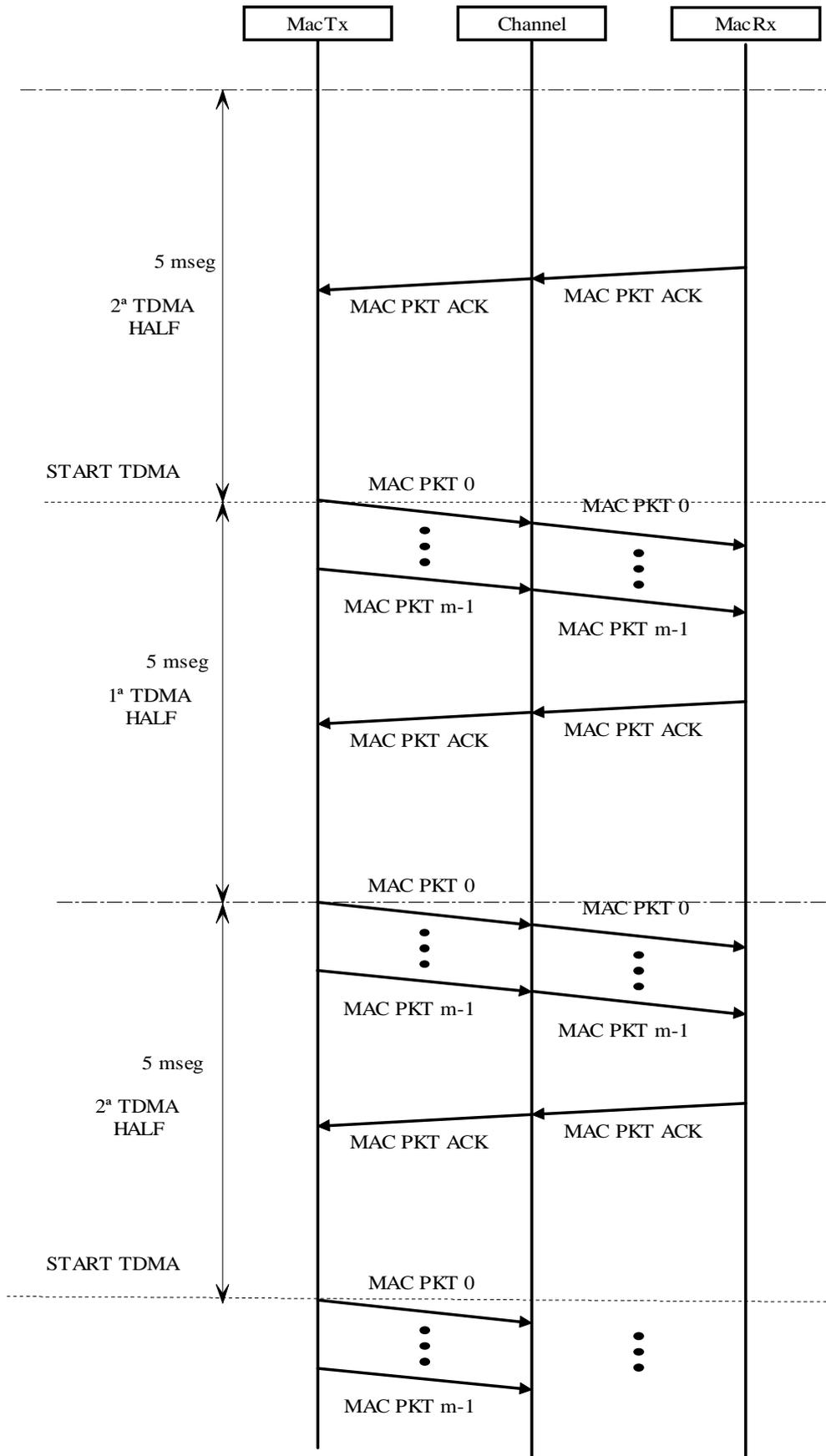


Figura 4. 16 Procedimiento de transmisión (entre la parte fija y la parte portátil)

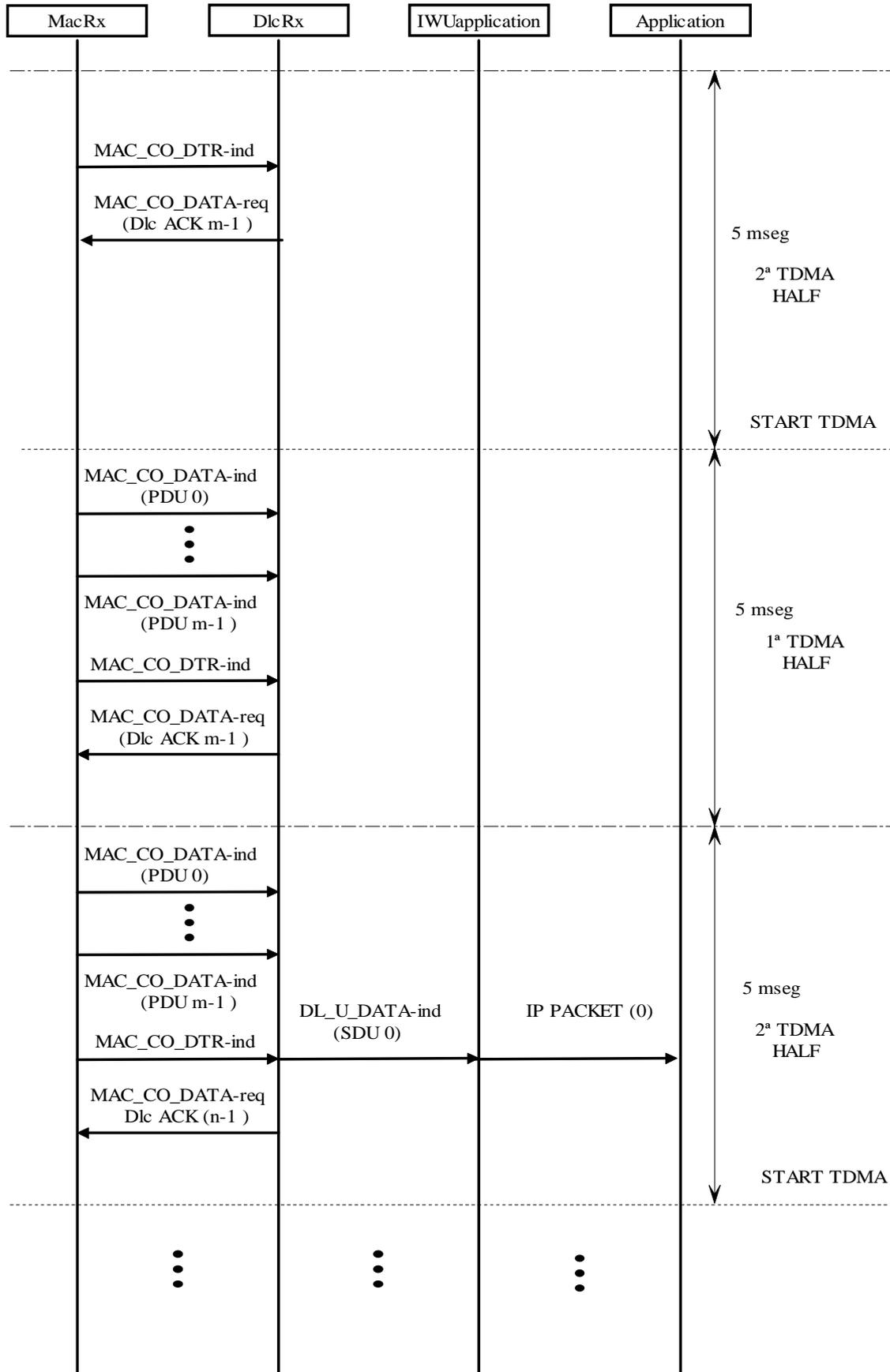


Figura 4. 17 Procedimiento de transmisión (entre la terminación portátil y la aplicación)

4.4.5.2 Transmisión un reconocimiento para el servidor

Para explicar el procedimiento de transmisión de un paquete IP con un reconocimiento para el servidor, se usará el diagrama temporal representado por la figura 4.18, la figura 4.19 y la figura 4.20 en el que se describe un ejemplo de transmisión de un paquete IP desde la aplicación hasta el servidor. En la figura 4.18 se muestra el tránsito de información entre los componentes del sistema, desde la aplicación hasta el extremo transmisor de la parte portátil y viceversa. En la figura 4.19 se muestra el tránsito de paquetes físicos a través del canal entre el extremo transmisor de la parte portátil y el extremo receptor de la parte fija y viceversa. Por último en la figura 4.20 se muestra el tránsito de unidades del protocolo usando primitivas a través de la parte portátil y finalmente la transmisión del paquete IP con el reconocimiento desde el router hasta el servidor. En esta representación se ha eliminado el módulo "IPdect" cuya única funcionalidad es introducir un retardo en los paquetes. La configuración de las portadoras para este ejemplo se corresponde con una configuración con una sola portadora "n" y una sola portadora "k" como la configuración de la conexión para el enlace "uplink" de la figura 4.2. El servicio seleccionado en el nivel de MAC es de corrección de errores. El módulo "canal" no introduce errores en la TDMA en la que se realizan las transmisiones. El nivel de modulación empleado para el ejemplo permite introducir una SDU entera en una sola PDU. A continuación al igual que se ha hecho para el enlace "downlink", se describe paso a paso todo el procedimiento:

Cuando el módulo "IWUapplication" en la PP recibe el paquete IP (IP packet (ACK 0)) del módulo "application", extrae su información y la reenvía en forma de SDU (SDU 0) al módulo "DlcTx" haciendo uso de la primitiva "DL_U_DATA-req" (figura 4.18).

Cuando el módulo "DlcTx" de la PP recibe la SDU la fragmenta en una PDU que almacena en el buffer en espera de la petición de datos de MAC (figura 4.18). Mientras espera la petición de MAC el módulo "DlcTx" recibe un reconocimiento (DlcAck) con información de control para la ventana.

Antes de que comience la primera mitad de la nueva TDMA, el módulo "MacTx" en la PP manda la primitiva "MAC_CO_DTR-ind" al módulo "DlcTx" solicitando 1 PDU para transmitirla en la portadora 0 de la primera mitad de la nueva TDMA. El módulo "DlcTx" le responde con la primitiva "MAC_CO_DATA-req" que contiene la PDU (PDU 0) (figura 4.18). Cuando comienza la primera mitad de la nueva TDMA, el módulo "MacTx" transmite en el slot asignado a la portadora el paquete físico (MAC PKT 0) (figura 4.19).

Cuando el paquete físico (MAC PKT 0) llega al módulo "MacRx" comprueba que tanto la cabecera como el campo de datos es correcto (figura 4.19). Introduce la información del paquete físico (PDU 0) en una primitiva del tipo "MAC_CO_DATA-ind" y se la envía al módulo "DlcRx" (figura 4.20). Esta operación varía para el caso de que el servicio sea de detección de errores, ya que la información del paquete físico se entregará justo en la mitad de la TDMA y no en el momento de recibir el paquete. El módulo "DlcRx" obtiene la PDU de la primitiva "MAC_CO_DATA-ind" y actualiza la ventana de recepción, de manera que la nueva PDU que espera será la PDU 1.

El módulo "MacRx" se prepara para transmitir un paquete físico sobre la portadora "n" para la segunda mitad de la TDMA (MAC PKT ACK) y solicita la información de control al módulo "DlcRx" haciendo uso de la primitiva "MAC_CO_DTR-ind". Usando la primitiva "MAC_CO_DATA-req", el módulo "DlcTx" le entrega la información de control que consiste en un reconocimiento de la última PDU recibida correctamente (DLC ACK 0). El módulo "MacRx" incluye esta información en los subcampos de datos del

paquete físico junto con la información de control para la portadora MAC (BCK y Q2) y lo transmite sobre la portadora “n” en el slot que corresponda (figura 4.19).

El módulo “MacTx” en la FP recibe el paquete físico (MAC PKT ACK) del módulo “MacRx” (figura 4.19) y comprueba que la cabecera es correcta. Como es correcta comprueba los subcampos de datos y obtiene la información de cada uno de ellos. Así pues obtiene la información de control para la portadora 0 (BCK y Q2) lo que le informa de que llegó correcta y obtiene también la información de control para la capa DLC (Dlc ACK 0), que envía al módulo “DlcTx” utilizando la primitiva “MAC_CO_DATA-ind”. El módulo “DlcTx” en la FP recibe la información de control con el reconocimiento de “DlcRx” y actualiza el borde inferior de su ventana de transmisión confirmando la PDU de 0 (figura 4.20). El módulo “DlcRx” obtiene la información útil de la PDU (PDU 0) y la introduce en una SDU (SDU 0) que incluye en la primitiva “DL_U_DATA-ind” que envía al módulo “IWUrouter” (figura 4.20).

Cuando el módulo “IWUapplication” recibe la SDU, entrega la información en forma de paquete IP (IP packet (ACK0)) al módulo “server” a través del enlace proporcionado por la red IP (figura 4.20).

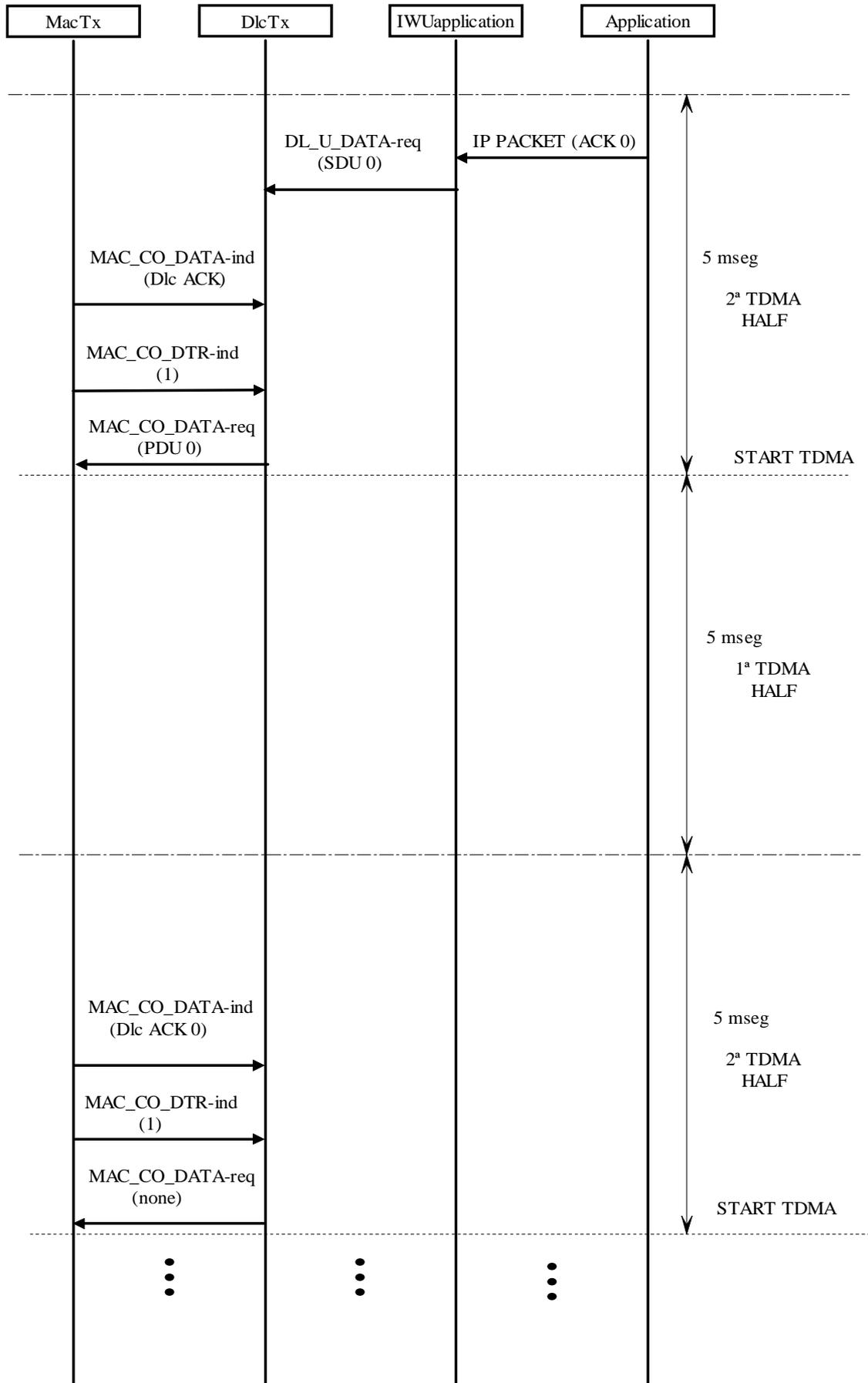


Figura 4. 18 Procedimiento de transmisión (entre la aplicación y la terminación portátil)

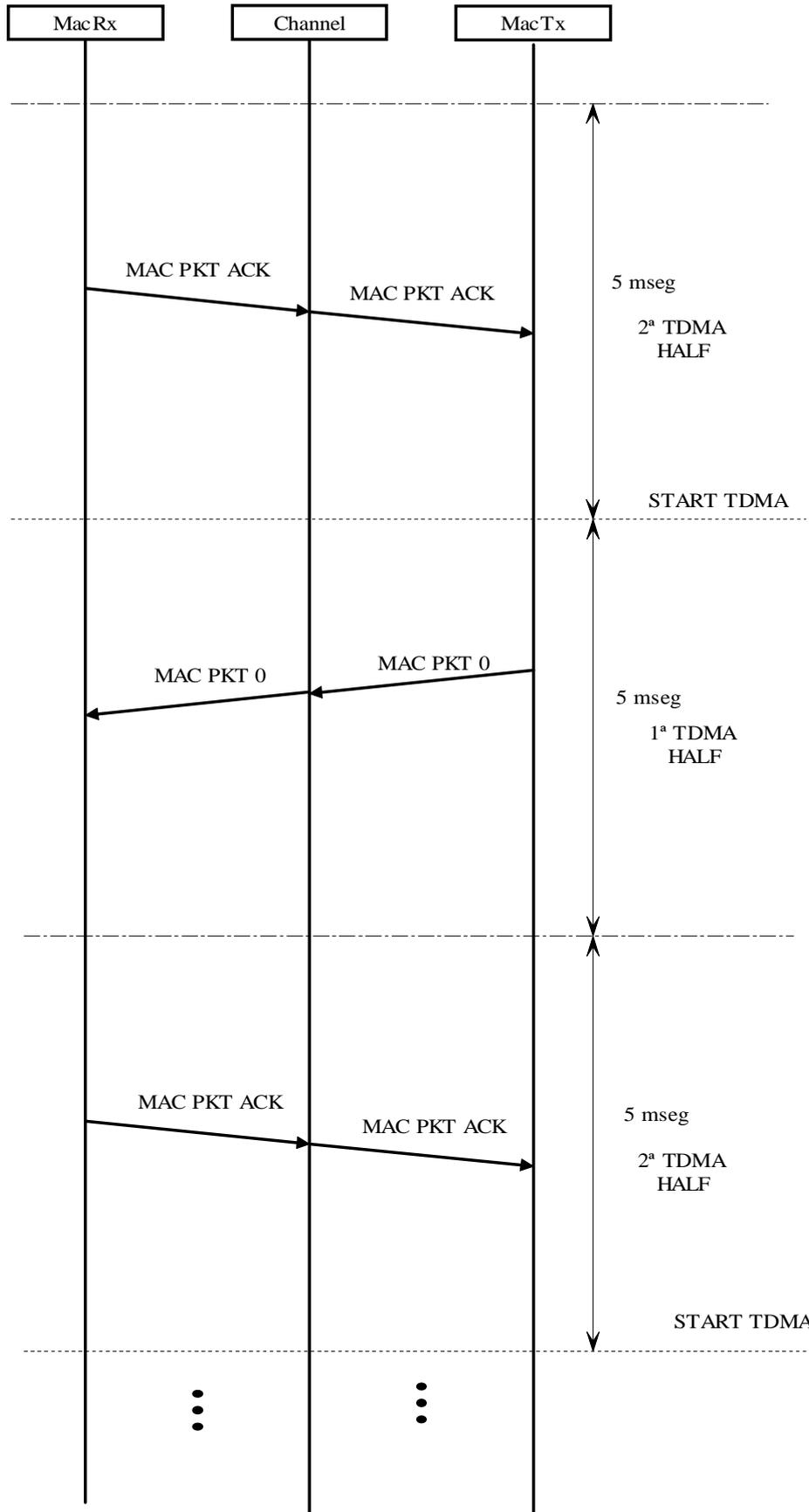


Figura 4. 19 Procedimiento de transmisión (entre la parte portátil y la parte fija)

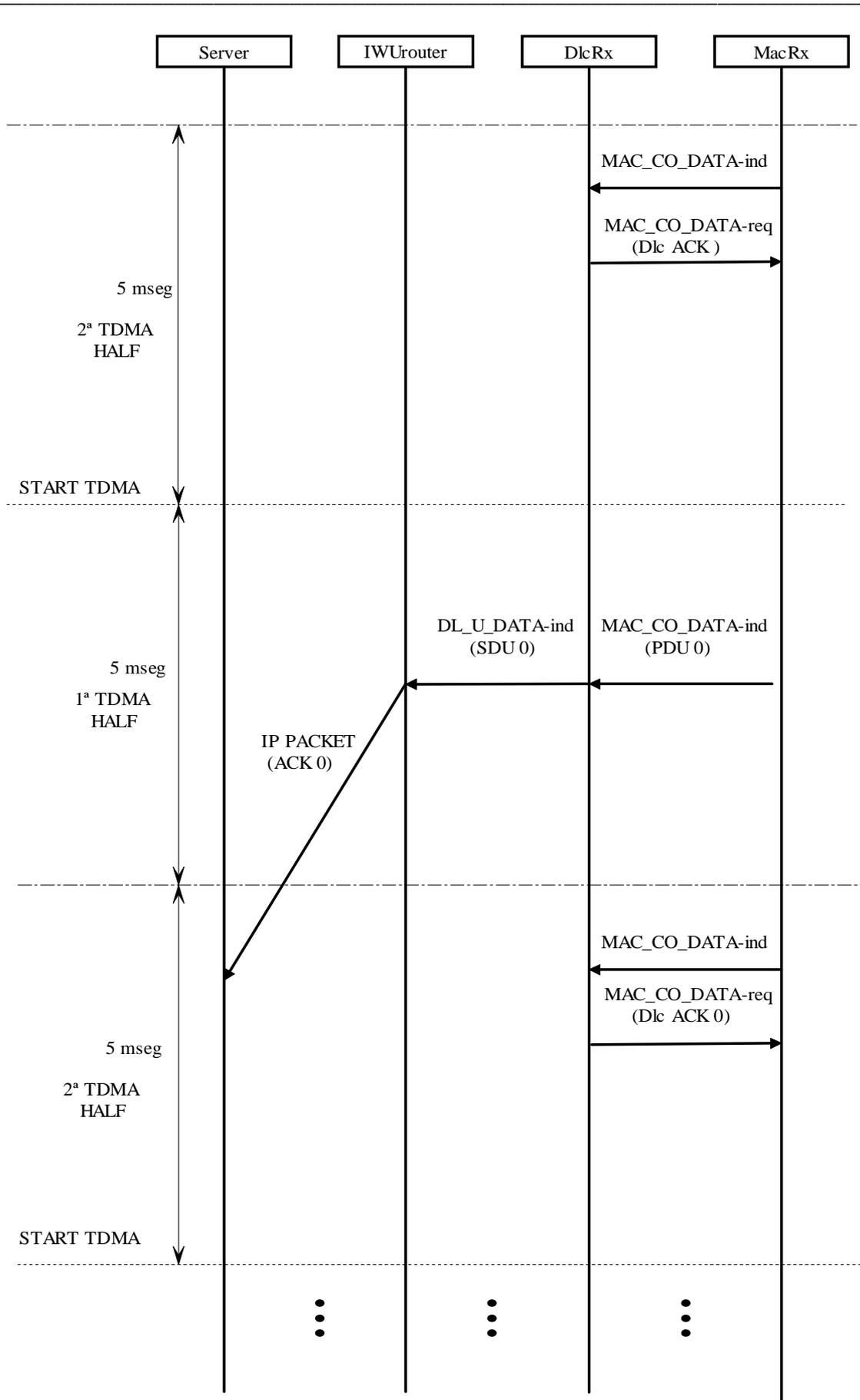


Figura 4.20 Procedimiento de transmisión (entre la terminación fija y el servidor)

Capítulo 5

Manejo del simulador

5.1. Instalación de OMNET++

Como se ha comentado en los capítulos anteriores para la realización de nuestro simulador se ha empleado la API de simulación proporcionada por OMNET++. Por tanto será necesario instalar el entorno de simulación OMNET++ para poder ejecutar nuestro modelo. La versión usada para el presente proyecto es la 2.3 y puede ser descargada de la página del autor [23]. Los pasos para la instalación de OMNET++ pueden ser consultados en [24].

5.2. Opciones de compilación

Para compilar y generar el fichero “make” con el código de nuestro simulador se usará el comando “opp_makemake”. Dependiendo de la opción que se use con este comando, se compilará el código incluyendo o no las librerías de TCL/TK. Si no se especifica ninguna opción acompañando al comando, las librerías de TCL/TK se añadirán por defecto. El uso de las librerías TCL/TK permite ejecutar el programa usando el entorno gráfico de OMNET++. En este entorno gráfico se puede observar el paso de mensajes entre módulos y se puede acceder directamente a gráficas en tiempo real y otro tipo de información en cada módulo. El entorno gráfico resulta de utilidad para la depuración del simulador. No obstante para el caso de simulaciones largas se aconseja no emplear dicho entorno y usar el entorno de consola. El comando que se debe usar para compilar el programa con el entorno de consola es:

```
opp_makemake -u Cmndenv
```

Una vez generado el fichero “make”, se usará el comando “make” para generar el fichero ejecutable.

5.3. Configuración de una simulación

Una vez compilado el código y obtenido el ejecutable ya podemos ejecutar nuestro simulador. Los ficheros con el código fuente del simulador se encuentran en la carpeta “tcpdectV6.4”. Dentro de esta carpeta tenemos el ejecutable de nuestro programa del mismo nombre. A la hora de realizar una simulación es necesario especificar los parámetros de entrada para la configuración de los módulos que componen el simulador. El fichero “omnetpp.ini” que es usado por defecto por OMNET++ será el empleado para especificar estos parámetros. Los parámetros de entrada se dividen en dos clases:

1. Parámetros para el tráfico TCP/IP.
2. Parámetros para el sistema DECT.

Para cada simulación el programa lee los parámetros de entrada del fichero “omnetpp.ini” proporcionado por OMNET++.

5.3.1. Parámetros de entrada TCP

A continuación se explican los parámetros de entrada para la configuración de los módulos que implementan el protocolo TCP reno.

- ID: Parámetro que se debe especificar para los módulos “server” y “application” y que sirve para identificarlos.
- Speed: Parámetro que define la capacidad del enlace TCP que une el servidor con el resto del sistema.
- Aggresive (true/false): Parámetro del módulo “server”. Contempla la opción de incrementar el tamaño de la ventana por cada ACK recibido. Si vale true se incrementa en uno por cada fragmento reconocido por el ACK. Si aggressive=false se incrementa en uno por cada ACK independientemente de los fragmentos que se reconozcan.
- SMSS: Parámetro del módulo “server”. Longitud del campo de datos del paquete IP (en bytes)
- ssthresh: Parámetro del módulo “server”. Valor (en paquetes) a partir del cual se entra en "Congestion Avoidance".
- rwnd: Parámetro del módulo “server”. Tamaño de la ventana del receptor (en bytes).
- fileSize: Parámetro del módulo “server”. Tamaño del fichero para descargar (en bytes).
- TCPdebug (true/false): Parámetro de los módulos “server” y “application” que contempla la opción de creación de los ficheros de depuración del protocolo TCP (.log).

5.3.2. Parámetros de entrada del sistema DECT

Los parámetros de entrada para configurar los distintos módulos que configuran el sistema DECT son:

- Model (1/2): Parámetro de los módulos “mactrx” y “macrx”. Se refiere al modelo de configuración de la conexión asimétrica (1 ó 2) descritos en el apartado 4.2.2.1 del capítulo 4.
- Errors (true/false): Parámetro del módulo “channel”. Determina si se usará un canal con o sin errores.
- probBG: (Requiere que “Errors = true”). Parámetro del módulo “channel” en el que se especifica la probabilidad de que el canal pase del estado malo (bad) al estado bueno (good).
- probGB: (Requiere que “Errors = true”). Parámetro del módulo “channel” en el que se especifica la probabilidad de que el canal pase del estado bueno (good) al estado malo (bad).

- **ModifyCnx (true/false):** Opción e hacer o no intercambio de conexiones.
- **DwnServiceType (7/8):** Parámetro de los módulos “mactrx” y “macrx”. Se refiere al tipo de servicio empleado para la protección de la pérdida de información para el enlace “downlink”. Puede tomar los valores 7 (detección de errores) u 8 (corrección de errores).
- **UpServiceType (7/8):** Igual que el parámetro “DwnServiceType” pero para el enlace “uplink”.
- **DwnCRC_Format (0/1):** Parámetro de los módulos “mactrx”, “macrx” y del módulo “dlctrx”. Se refiere al tipo de formato de protección empleado para proteger la información de los paquetes físicos que viajan por el enlace “downlink”. Puede tomar los valores 0 (protección de un solo campo) ó 1 (protección de múltiples subcampos).
- **UpCRC_Format (0/1):** Igual que el parámetro “DwnCRC_Format” pero para el enlace “uplink”.
- **LU10FillDwn (true/false):** Parámetro del módulo “dlctrx”. Activa (true) o desactiva (false) para el enlace “downlink” la opción de usar el servicio de relleno de PDUs con otras SDUs definido para LU10.
- **LU10FillUp (true/false):** Igual que el parámetro “DwnCRC_Format” pero para el enlace “LU10FillDwn”.
- **DwnModulationLevel (2/4/8):** Parámetro del módulo “dlctrx”. Se refiere al nivel de modulación de los datos en el enlace “downlink”. Puede tomar los valores 2, 4 y 8.
- **UpModulationLevel (2/4/8):** Igual que el parámetro “DwnModulationLevel” pero para el enlace “LU10FillDwn”.
- **SDUsExpireDwn (true/false):** Parámetro del módulo “dlctrx”. Activa (true) o desactiva (false) para el enlace “downlink” la opción de limitar el tiempo de vida de las SDUs en el buffer de transmisión.
- **SDUsExpireUp (true/false):** Igual que el parámetro “SDUsExpireUp”, pero para el enlace “uplink”.
- **T_R_Dwn:** (Requiere que “SDUsExpireDwn = true”) En este parámetro se especifica el número de TDMA's antes de que expire la SDU en el enlace “downlink”.
- **T_R_Up:** (Requiere que “SDUsExpireUp = true”). Igual que “T_R_Dwn”, pero para el enlace “uplink”.
- **Threshold:** (Requiere que “ModifyCnx = true”) Parámetro para el módulo “dlctrx” de la parte fija y de la parte portátil que especifica un límite de SDUs en el buffer. Si se supera este límite el enlace con la conexión de menor capacidad solicita la conexión de mayor capacidad. Si el número de SDUs en el buffer del enlace con la conexión de mayor capacidad es menor o igual a este límite, no cede la conexión. El límite puede tomar un valor distinto para cada enlace. Los valores que puede tomar es cualquier número entero positivo. Si se especifica un valor

negativo para el enlace que usa la conexión de mayor capacidad, nunca cederá dicha conexión.

- DECTdebug (true/false): Activa (true) o desactiva (false) para la red DECT la opción de generar ficheros de registro (.log) para depurar los módulos que forman dicha red.

5.4. Ejecución de una Simulación

En este apartado y con el objeto de mostrar gráficamente el aspecto que presentan los módulos y las gráficas que nos puede mostrar nuestro simulador, se ha compilado el código de manera que se use la interfaz gráfica tal. Para arrancar el entorno gráfico y poder comenzar con la simulación se escribe en la línea de comando el nombre del ejecutable. En este momento, se arranca el entorno gráfico de OMNET++ (implementado en TCL/TK). A continuación OMNET++ llama automáticamente al método de inicialización de los módulos que componen el simulador. Para cada módulo, se ha indicado en este método la lectura de los parámetros de configuración de fichero "omnetpp.ini". Si alguno de los parámetros especificados en el fichero toma un valor que no coincide con el tipo definido en el módulo para ese parámetro, OMNET++ mostrará un mensaje de error y la simulación será abortada de manera automática. Si todos los parámetros coinciden con el tipo especificado para ellos en cada módulo el entorno gráfico terminará de arrancar y se podrá comenzar con la ejecución de la simulación.

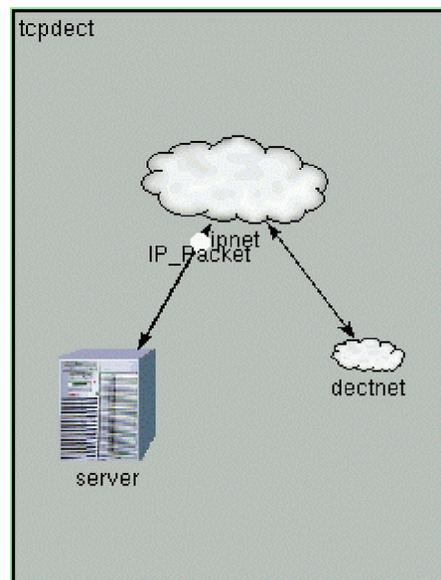


Figura 5. 1 Entorno gráfico (tcpdect)

No obstante si el parámetro es del tipo definido pero no se trata de un valor correcto, OMNET++ no lo detectará y permitirá comenzar la simulación. Esta situación puede provocar que el programa termine de manera inesperada o que los resultados obtenidos sean erróneos.

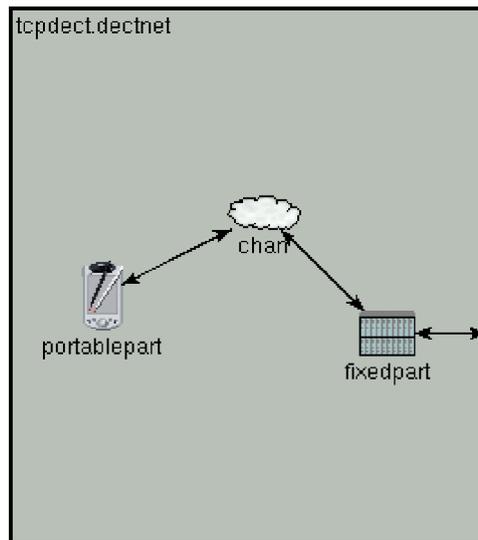


Figura 5. 2 Entorno gráfico (tcpdect.dectnet)

Para resolver este problema se ha incluido un código adicional en cada módulo que comprueba si el valor del parámetro leído del fichero “omnetpp.ini”, coincide con los valores permitidos. El manejo de la interfaz gráfica OMNET++ se encuentra fuera de los objetivos del proyecto. Si el lector desea conocer los aspectos de manejo de dicha interfaz puede acudir al manual [24].

5.4.1. Parámetros de salida para TCP

Durante la simulación con el entorno gráfico se puede seguir la evolución gráfica del valor de los siguientes parámetros del módulo “server” en función del tiempo de simulación.

- FRFR IN: Tamaño de la ventana antes de entrar en FRFR.
- FRFR OUT: Tamaño de la ventana antes de salir del estado FRFR.
- RTO indicador: Valor de RTO antes de que se produzca un Time out.
- Cwnd: tamaño de la ventana de congestión.
- Awnd: tamaño máximo permitido para la ventana de transmisión.
- Ssthresh: límite para entrar en CA.
- RTO value: Valor del timer para la retransmisión de un determinado fragmento
- Flight Size: Número de paquetes pendientes de reconocimiento.
- Throughput: Velocidad de transmisión (en bits) de la fuente TCP.
- Int SN: Número de secuencia del último paquete transmitido.

Para el parámetro “Throughput”, se puede consultar también su valor medio

En el módulo “application”, se ha creado un parámetro para tomar la medida del tiempo de retardo de los paquetes desde que son transmitidos por el módulo “server” hasta que son reconocidos por “application”. Para este parámetro se puede consultar:

- Su evolución temporal en función del tiempo de simulación.
- Su valor medio.
- El número de veces que un paquete ha sufrido el mismo retardo. Estos resultados se representan en forma de un histograma de frecuencias.

En la figura 5.3 se puede observar un ejemplo de toma de medias haciendo uso de la interfaz gráfica.

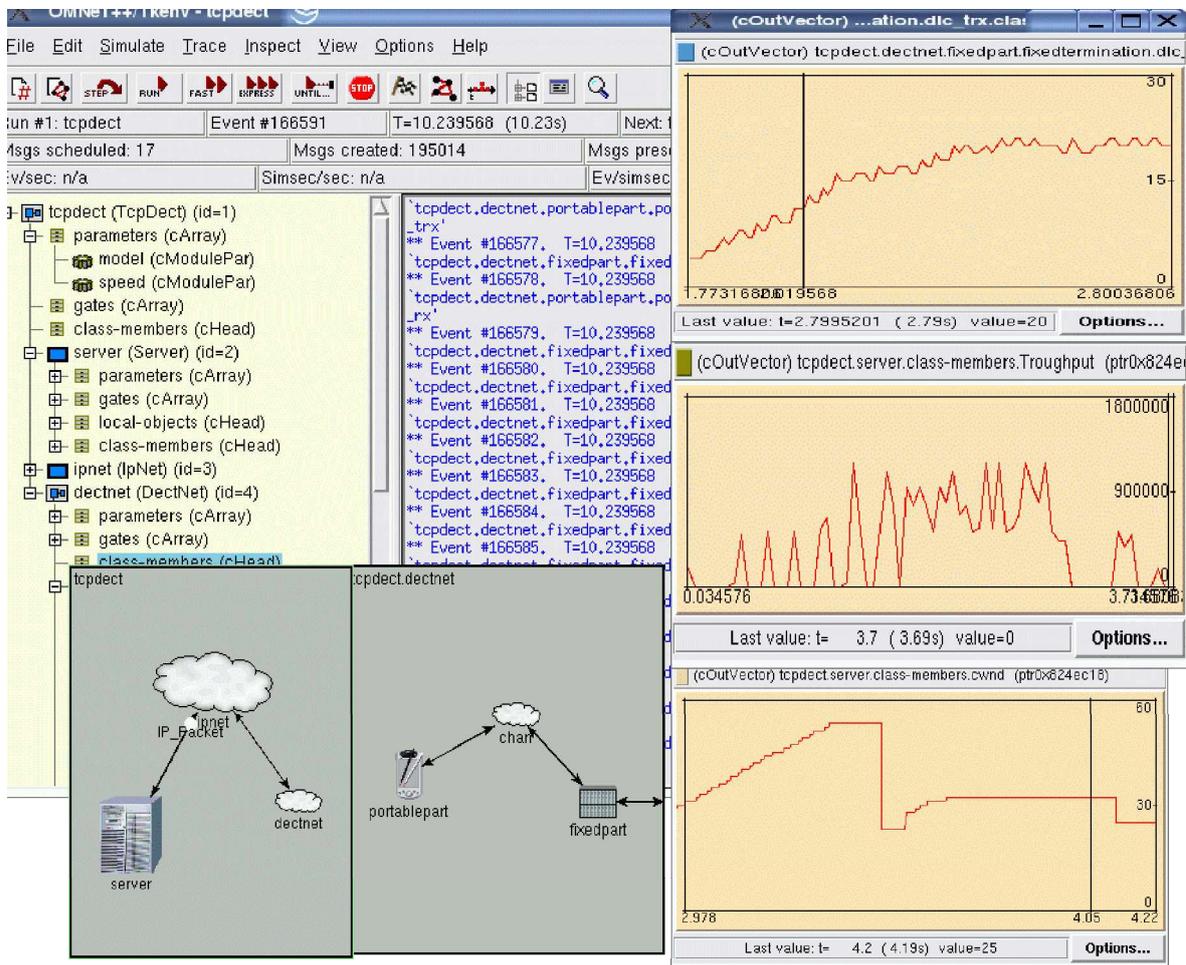


Figura 5. 3 Ejemplo de toma de medidas

5.4.2. Parámetros de salida del sistema DECT

Durante la simulación con el entorno gráfico se puede seguir la evolución gráfica del valor de los siguientes parámetros en función del tiempo de simulación en el módulo “dlctrx” (para cada enlace) dentro del módulo “dectnet”:

- numSDUs: Número de SDUs en el buffer del transmisor.
- numPDUs: Número de PDUs dentro de la ventana de transmisión.

5.5. Ficheros de registro de los módulos

Para la validación del simulador se ha hecho que cada uno de los módulos genere un fichero de registro de extensión .log donde aparecen todos los eventos generados por dicho módulo a lo largo del tiempo de simulación. Mediante el estudio de las trazas generadas por cada módulo en estos ficheros se han validado todos los procedimientos implementados. Sólo en el caso de que se haya activado la opción de depuración del programa se podrá consultar la lista de eventos que ocurren en cada módulo:

- Server.log: Fichero de registro del módulo “server”.
- Application.log: Fichero de registro del módulo “application”.
- DlcTxDwn.log: Fichero de registro del módulo “dlctx” del enlace “downlink”.
- DlcTxUp: Fichero de registro del módulo “dlctx” del enlace “uplink”.
- DlcRxDwn.log: Fichero de registro del módulo “dlcrx” del enlace “downlink”.
- DlcRxUp.log: Fichero de registro del módulo “dlcrx” del enlace “uplink”.
- MacTxDwn.log: Fichero de registro del módulo “mactrx” del enlace “downlink”.
- MacTxUp.log: Fichero de registro del módulo “mactrx” del enlace “uplink”.
- MacRxDwn.log: Fichero de registro del módulo “macrx” del enlace “downlink”.
- MacRxUp.log: Fichero de registro del módulo “macrx” del enlace “uplink”.
- Channel.log: Fichero de registro del módulo “channel”.
- Iwu_application.log: Fichero de registro del módulo “iwu_application”.
- Iwu_router.log: Fichero de registro del módulo “iwu_router”.
- Downlink.log: Fichero general de registro para el enlace “downlink”.
- Uplink.log: Fichero general de registro para el enlace “uplink”.

A continuación se proporciona la información sobre la información que se puede extraer de cada fichero y sobre el formato de presentación.

5.5.1. Fichero “server.log”

Consultando este fichero se puede obtener información sobre el control de flujo en el servidor. Se muestran mensajes para indicar el estado del servidor, la transmisión de un paquete ACK, la recepción de un paquete ACK y el tamaño de la ventana de gestión. El formato de cada mensaje es:

X à Server TCP IP: Y Z

Donde:

- X: Tiempo de simulación en el que se produjo el evento.
- Y: Identificador del servidor.
- Z: Mensaje del módulo

5.5.1.1. Ejemplo del fichero “server.log”

En la figura 5.4 se muestra como ejemplo un fragmento del fichero de registro del módulo “server”. El servidor TCP transmite un paquete con el número de segmento 10 y recibe una confirmación del paquete con número de segmento 9 actualizando el tamaño de la ventana de congestión (cwnd).

```
0.314272-->Server TCP IP: 0 Transmite SN: 10
0.314432--> Server TCP IP: 0 ha recibido ACK 9 --> NUEVO
0.314432-->Server TCP IP: 0 Estado: SS
Last ACK SN: 9 SN: 10 cwnd: 6
```

Figura 5. 4 Ejemplo del fichero “server.log”

5.5.2. Fichero “application.log”

En este fichero se muestran los mensajes generados por el módulo “application”. El formato de los mensajes del fichero es:

X à Application TCP IP: Y Z

Donde:

- X: Tiempo de simulación en el que se produjo el evento.
- Y: Identificador de la aplicación del terminal.
- Z: Mensaje del módulo

Leyendo los mensajes de este módulo se puede obtener información sobre el control de flujo de los paquetes IP en el extremo receptor de la conexión TCP.

5.5.2.1. Ejemplo del fichero “application.log”

En la figura 5.5 se muestra como ejemplo un fragmento del fichero de registro del módulo “application”. Se muestra la recepción de los paquetes TCP del 1 al 10.

```
El modulo application 1 está en marcha
0.0178561-->Application TCP IP 1 ha recibido el paquete: 1
0.252864-->Application TCP IP 1 ha recibido el paquete: 2
0.256192-->Application TCP IP 1 ha recibido el paquete: 3
```

```

0.272864-->Application TCP IP 1 ha recibido el paquete: 4
0.276192-->Application TCP IP 1 ha recibido el paquete: 5
0.278688-->Application TCP IP 1 ha recibido el paquete: 6
0.292864-->Application TCP IP 1 ha recibido el paquete: 7
0.296192-->Application TCP IP 1 ha recibido el paquete: 8
0.298688-->Application TCP IP 1 ha recibido el paquete: 9
0.322864-->Application TCP IP 1 ha recibido el paquete: 10

```

Figura 5. 5 Ejemplo del fichero “application.log”

5.5.3. Ficheros “iwu_application.log” e “iwu_router.log”

Los mensajes generados por el router (módulo “iwu_router”) que actúa como IWU entre el sistema DECT y la red IP pueden ser consultados en el fichero “iwu_router.log”. Por su parte los mensajes generados por la entidad que actúa como IWU (módulo “iwu_application”) entre la aplicación TCP/IP y el sistema DECT, se pueden consultar en el fichero “iwu_application.log”.

En ambos ficheros se muestra el tiempo en el que se produjo un evento (expresado en segundos) de llegada de un paquete y la acción efectuada.

5.5.3.1. Ejemplo del fichero “iwu_router.log”

En la figura 5.6 se muestra una porción del fichero de registro del router donde se puede obtener la siguiente información:

- Ha habido un evento de llegada de un paquete procedente de la red IP a los 23,04 seg. A continuación el router ha realizado la entrega del paquete al nivel de enlace del sistema DECT en forma de SDU.
- Se ha producido en 224,112 mseg la llegada de una primitiva que contiene una SDU de la capa de enlace del sistema DECT. El router ha entregado la información de la SDU a la red IP en forma de paquete IP.

```

Fichero de registro del router
Evento en: 0.002304
Llega paquete IP a router
Se entrega SDU a Dlc
Numero de SDUs enviadas a Dlc: 1
Evento en: 0.224112
Llega trama SDU a router
Se envía paquete IP para server por la red IP

```

Figura 5. 6 Ejemplo del fichero “iwu_router.log”

5.5.3.2. Ejemplo del fichero “iwu_application.log”

En la figura 5.7 se muestra una porción del fichero para la entidad IWU que actúa como traductora de protocolo entre el sistema DECT y la aplicación. La información que se puede obtener es:

- Ha habido un evento de llegada de una primitiva que contiene una SDU procedente del sistema DECT a los 17,8561 mseg. A partir de la SDU se ha construido un paquete IP que se ha entregado a la aplicación.
- Un evento en 217,856 mseg de llegada de un paquete IP procedente de la aplicación. La información ha sido entregada en forma de SDU al nivel de enlace del sistema DECT.

```
Fichero de registro de la aplicación
Evento en: 0.0178561
Llega trama SDU
Se entrega paquete IP a la aplicación
Evento en: 0.217856
Llega paquete IP de la aplicación
Se entrega SDU a Dlc
Numero de SDUs enviadas a Dlc: 1
```

Figura 5. 7 Ejemplo del fichero “iwu_application.log”

5.5.4. Ficheros “DlctrxDwn.log” y “DlcTrxUp.log”

En los ficheros “DlctrxDwn.log” y “DlcTrxUp.log” se muestran los eventos ocurridos en el extremo DLC que transmite la información del plano de usuario para los enlaces “downlink” e “uplink” respectivamente. El formato de los mensajes con la información sobre los eventos es el mismo en ambos ficheros. Esta información permite conocer la evolución de los procedimientos de control de flujo a nivel DLC. Como los eventos de petición de datos generados por MAC se producen antes de que empiece la mitad de una TDMA, con el objeto facilitar la lectura del fichero se ha decidido ordenar por tramas TDMA la información a mostrar en cada momento. Para ello se han empleado mensajes en el fichero que marcan el principio de una nueva TDMA e informan de la mitad de la TDMA en la que se encuentra el sistema.

En estos ficheros se puede conocer información referente al número de PDUs solicitadas por MAC y que se incluyen en la ventana de transmisión, el número de SDUs no expiradas en el Buffer, el instante en que se producen expiraciones de SDUs, etc.

5.5.4.1. Ejemplos del fichero “DlcTrxDwn.log”

En la figura 5.8 se muestra la llegada de una primitiva con un paquete IP al buffer de la estación base del sistema DECT :

```
Evento en: 0.002304
Ha llegado nueva SDU, número de SDUs en el buffer: 1
```

Figura 5. 8 Ejemplo de llegada de una primitiva con una SDU (“DlctrxDwn.log”)

En la figura 5.9 al inicio de una nueva trama TDMA se incrementa el número de TDMA para cada SDU en el Buffer:

```
Evento en: 0.27
Inicio de nueva trama Tdma
```

```
Se busca en SDU buffer de: 3 a:5
Tdma para SDU= 3 en buffer: 1
Tdma para SDU= 4 en buffer: 1
Número de SDUs en el Buffer = 2
```

Figura 5. 9 Ejemplo de inicio de trama TDMA (“DlcTrxDwn.log”)

En la figura 5.10 se muestra la petición de datos de MAC para transmitirlos en la siguiente TDMA. La capa MAC pide 10 segmentos PDU y la capa DLC fragmenta las SDUs e introduce la información en PDUs.

```
Evento en: 0.269568
MAC pide: 10 segmentos

Evento Entregar PDUs en: 0.269568
Se ha introducido el principio de SDU: 3 en PDU: 20
Número de SDUS en la PDU = 1
Introducida PDU en OutBuff[20]
Número de PDUs: 1
Asigna SN: 20 a PDU: 20
Entrega: 20----->
FlightSize = 1
El número de endSDUs en la PDU es:0
Dlc entrega datos a Mac
Ventana del transmisor: [20...20]

Evento Entregar PDUs en: 0.269568
Número de SDUS en la PDU = 1
Introducida PDU en OutBuff[21]
Número de PDUs: 2
Asigna SN: 21 a PDU: 21
Entrega: 21----->
FlightSize = 2
El número de endSDUs en la PDU es:0
Dlc entrega datos a Mac
Ventana del transmisor: [20...21]

Evento Entregar PDUs en: 0.269568
Número de SDUS en la PDU = 1
Introducida PDU en OutBuff[22]
Número de PDUs: 3
Asigna SN: 22 a PDU: 22
Entrega: 22----->
FlightSize = 3
El número de endSDUs en la PDU es:0
Dlc entrega datos a Mac
Ventana del transmisor: [20...22]
```

Figura 5. 10 Ejemplo de petición de datos para transmitir en la TDMA (“DlcTrxDwn.log”)

En la figura 5.11 se muestra la recepción de fragmentos (PDUs) procedentes de la capa MAC. Se muestra la llegada de una ACK que confirma las PDUs de la 20 a la 25.

```
Evento Recepción en: 0.27952
FP: Obtenida información de control
El mensaje DataInd contiene 1
```

```
<----- Ack: 38

Se Confirma SN: 20(PDU 20)
Se elimina PDU= 20
Ventana del transmisor: [21...38]

Se Confirma SN: 21(PDU 21)
Se elimina PDU= 21
Ventana del transmisor: [22...38]

Se Confirma SN: 22(PDU 22)
Se elimina PDU= 22
Ventana del transmisor: [23...38]

Se Confirma SN: 23(PDU 23)
Se elimina PDU= 23
Ventana del transmisor: [24...38]

Se Confirma SN: 24(PDU 24)
Se elimina PDU= 24
Ventana del transmisor: [25...38]

Se Confirma SN: 25(PDU 25)
Se elimina PDU= 25
Ventana del transmisor: [26...38]

Se Confirma SN: 26(PDU 26)
Se confirma la entrega de la SDU: 3
Antes OldSDU 3
Después OldSDU: 4
Retardo para la SDU: 0.27952
Numero de SDUs confirmadas: 4
Retardo medio acumulado: 0.81808
```

Figura 5. 11 Ejemplo de recepción de una ACK para el nivel de enlace (“DlcTrxDwn.log”)

5.5.5. Ficheros “DlcrxDwn.log” y “DlcrxUp.log”

En los ficheros “DlcrxDwn.log” y “DlcrxUp.log” se muestran los eventos ocurridos en el extremo DLC que transmite la información para el control de flujo del plano de usuario para los enlaces “downlink” e “uplink” respectivamente. El formato de los mensajes con la información sobre los eventos es el mismo en ambos ficheros. Esta información permite conocer la evolución de los procedimientos de control de flujo a nivel DLC. Con el objeto de facilitar la lectura del fichero se ha decidido ordenar por tramas TDMA la información a mostrar en cada momento. Para ello se han empleado mensajes en el fichero que marcan el principio de una nueva TDMA e informan de la mitad de la TDMA en la que se encuentra el sistema.

En estos ficheros se puede conocer información referente al estado de la ventana de recepción, el número de PDUs que han llegado en secuencia, el número de PDUs perdidas que van a ser retransmitidas y el número de SDUs confirmadas. Veamos algunos ejemplos del tipo de información mostrada:

5.5.5.1. Ejemplos del fichero “DlcrxDwn.log”

En la figura 5.12 se muestra un fragmento del fichero “DlcrxDwn.log” en el que aparece el inicio de la primera trama TDMA, la búsqueda en el buffer de las PDUs que han llegado fuera de secuencia y la recepción de dos PDUs en secuencia con la actualización de la ventana de recepción a nivel de enlace:

```

Evento en: 0.01
Inicio de nueva trama Tdma

Buscando PDUs fuera de secuencia
No hay PDUs fuera de secuencia
Evento en: 0.0153601
Llega Segmento de la capa Mac
Receptor Dlc: Tengo el SN= 0
----->0
Introducida PDU en InBuff[0]
Se actualiza ventana
Se espera PDU: 1
Número de PDUs que han llegado en secuencia: 1
Ventana de recepción: [1...64]
Evento en: 0.0157761
Llega Segmento de la capa Mac
Receptor Dlc: Tengo el SN= 1
----->1
Introducida PDU en InBuff[1]
Se actualiza ventana
Se espera PDU: 2
Número de PDUs que han llegado en secuencia: 2
Ventana de recepción: [2...65]

```

Figura 5. 12 Ejemplo de recepción de fragmentos de la capa MAC (“DlcRxDown.log”)

En la figura 5.13 al principio de la TDMA no hay PDUS fuera de secuencia. A continuación llega una PDU con SN = 1063, que no coincide con la PDU esperada y que se marca como fuera de secuencia. Con la llegada de la PDU fuera de secuencia se fijan las PDUs con SN =1060,1061 y 1063 como PDUs perdidas. Cuando MAC pide la información de control se entrega un ACK para el transmisor DLC reconociendo la última PDU que llegó en secuencia (borde inferior de la ventana)

```

Evento en: 0.97
Inicio de nueva trama Tdma

Buscando PDUs fuera de secuencia
No hay PDUs fuera de secuencia
Evento en: 0.972032
Llega Segmento de la capa Mac
Receptor Dlc: Tengo el SN= 1063
----->1063
PDU no esperada
Introducida PDU en InBuff[1063]
Ventana de recepción: [1060...1123]

-----
Evento en: 0.979152
Mac pide información de control
PDUs perdidas
PDU: 1060 perdida
PDU: 1061 perdida
PDU: 1062 perdida
<----- ACK: 1059
Dlc entrega datos de control a Mac con SN =1059

```

Figura 5. 13 Ejemplo de detección de pérdida de PDUs (“DlcRxDown.log”)

En la figura 5.14 se muestran los eventos ocurridos durante la TDMA número 100 que empieza en el instante 1 seg (10 mseg * 100 = 1seg). Al buscar en el buffer se encuentran PDUs perdidas con SN = 1061,SN = 1062 y 1064 lo que nos indica que la última PDU que ha llegado fuera de secuencia es la PDU con SN = 1065. Se informa

que la PDU lleva fuera de secuencia 3 tramas (cuando lleve 8 tramas, se mandará un mensaje Nack para pedir la retransmisión de las PDUS perdidas). A los 1,2 mseg de iniciarse la TDMA la capa MAC entrega a DLC las PDUs perdidas con SN=1061 y SN = 1062. DLC reconoce entonces las PDUs que están en secuencia (desde 1061 hasta 1063 (incluida)). La PDU 1063 contiene el final de una SDU por lo que se entrega la SDU a la capa superior.

```
Evento en: 1
Inicio de nueva trama Tdma

Buscando PDUs perdidas
PDU: 1061 perdida
PDU: 1062 perdida
PDU: 1064 perdida
Última PDU fuera de secuencia: 1065
Lleva fuera de secuencia: 3 tramas
Evento en: 1.0012
Llega Segmento de la capa Mac
Receptor Dlc: Tengo el SN= 1061
----->1061
Introducida PDU en InBuff[1061]
Se actualiza ventana
Se espera PDU: 1062
Número de PDUs que han llegado en secuencia: 1062
Ventana de recepción: [1062...1125]
Evento en: 1.00162
Llega Segmento de la capa Mac
Receptor Dlc: Tengo el SN= 1062
----->1062
Introducida PDU en InBuff[1062]
Se actualiza ventana
Se espera PDU: 1063
Número de PDUs que han llegado en secuencia: 1063
La PDU contiene el final de una SDU que no ha expirado
Entregada SDU a la capa superior
Número de SDUs entregadas: 173
Se actualiza ventana
Se espera PDU: 1064
Número de PDUs que han llegado en secuencia: 1064
```

Figura 5. 14 Ejemplo de recepción de PDUs perdidas(“DlcRxDown.log”)

5.5.6. Ficheros “MactrxDown.log” y “MactrxUp.log”

En los ficheros “MactrxDown.log” y “MactrxUp.log”, se muestran los eventos ocurridos en el extremo transmisor de datos de la capa MAC para las conexiones asociadas a los enlaces “downlink” e “uplink” respectivamente. El formato de los mensajes con la información sobre los eventos es el mismo en ambos ficheros. Esta información permite conocer la evolución de los procedimientos de control de flujo a nivel MAC. Con el objeto facilitar la lectura del fichero se ha decidido ordenar por tramas TDMA la información a mostrar en cada momento. Para ello se han empleado mensajes en el fichero que marcan el principio de una nueva TDMA e informan de la mitad de la TDMA en la que se encuentra el sistema. Además se informa también del slot en el que nos encontramos dentro de la TDMA.

En cada uno de estos ficheros se puede conocer el número de PDUs que han llegado de la capa DLC, la portadora sobre la que se va a re/transmitir la información de cada PDU. Sólo para el caso de que el servicio proporcionado por la capa MAC sea de

corrección de errores se muestra información referente a las retransmisiones. Esta información muestra el número de veces que ha sido retransmitido un paquete o si un paquete ha superado el tiempo máximo de vida permitido y debe ser descartado. Se proporcionan a continuación unos fragmentos de uno de estos ficheros.

5.5.6.1. Ejemplos del fichero “MacTrxDwn.log”

En la figura 5.15 se muestra un fragmento del fichero “MacTrxDwn.log” donde se ve un ejemplo de transmisión de dos paquetes sobre portadoras en los slots 0 y 1 de la primera mitad de una TDMA. El paquete sobre la portadora 0 contiene un fragmento DLC con SN = 1041. El número de secuencia asignado al paquete para el control de flujo a nivel MAC es SN = 1.

```

Evento en :0.94
Inicio de nueva trama Tdma

Primera mitad de la Tdma
Slot: 0
Hay que transmitir si es posible
Transmisión de paquete MAC sobre portadora: 0 en slot: 0
Se introduce Segmento de Dlc (SN: 1041)
Longitud del paquete Mac en símbolos es: 424
Número de secuencia es SN = 1
Transmito en:0.94

Slot: 1
Hay que transmitir si es posible
Transmisión de paquete MAC sobre portadora: 1 en slot: 1
Se introduce Segmento de Dlc (SN: 1042)
Longitud del paquete Mac en símbolos es: 424
Número de secuencia es SN= 1
Transmito en:0.940416
    
```

Figura 5. 15 Ejemplo de transmisión de paquetes MAC (“MacTrxDwn.log”)

En la figura 5.16 se muestra un ejemplo de fallo en la cabecera de la portadora “n” con la información para el control de flujo a nivel MAC (Sólo para el caso de que el servicio sea de corrección de errores)

```

Fallo en la cabecera del paquete del receptor
No se puede obtener tipo de mensaje ni SN
Se ha perdido información para las portadoras de la primera mitad de la Tdma

Número de reintentos del paquete con SN= 1 sobre la portadora: 0 son: 1
Número de reintentos del paquete con SN= 1 sobre la portadora: 1 son: 1
    
```

Figura 5. 16 Ejemplo de fallo en la recepción de los ACKs para MAC (“MacTxDwn.log”)

En la figura 5.17 se muestra un ejemplo de recepción de ACK para el caso de que el servicio sea de corrección de errores. El receptor MAC pide al transmisor MAC la retransmisión del paquete sobre la portadora “0” porque la cabecera llegó con errores. Para el resto de portadoras el transmisor recibe la confirmación del receptor y avanza normal asignando un nuevo SN a la portadora.

```

Evento en: 0.95952
Recibido Ack
    
```

```
Contiene información de control para las portadoras de la primera mitad de la
actual Tdma

El receptor pide retransmisión del paquete sobre la portadora: 0 porque A_CRC
le llegó incorrecto
Avanzo normal, el SN del nuevo paquete sobre la portadora: 1 es, SN= 0
Avanzo normal, el SN del nuevo paquete sobre la portadora: 2 es, SN= 0
Avanzo normal, el SN del nuevo paquete sobre la portadora: 3 es, SN= 1
Avanzo normal, el SN del nuevo paquete sobre la portadora: 4 es, SN= 0
Avanzo normal, el SN del nuevo paquete sobre la portadora: 5 es, SN= 1
Avanzo normal, el SN del nuevo paquete sobre la portadora: 6 es, SN= 1
Avanzo normal, el SN del nuevo paquete sobre la portadora: 7 es, SN= 1
Avanzo normal, el SN del nuevo paquete sobre la portadora: 8 es, SN= 0
Avanzo normal, el SN del nuevo paquete sobre la portadora: 9 es, SN= 0
```

Figura 5. 17 Ejemplo de recepción de ACKs para las portadoras a nivel MAC (“MacTrxDwn.log”)

5.5.7. Ficheros “MacrxDwn.log” y “MacrxUp.log”

En los ficheros “MacrxDwn.log” y “MacrxUp.log”, se muestran los eventos ocurridos en el extremo transmisor de la portadora “n” con la información control para las conexiones asociadas a los enlaces “downlink” e “uplink” respectivamente. El formato de los mensajes con la información sobre los eventos es el mismo en ambos ficheros. Esta información permite conocer la evolución de los procedimientos de control de flujo a nivel MAC. Con el objeto de facilitar la lectura del fichero se ha decidido ordenar por tramas TDMA la información a mostrar en cada momento. Para ello se han empleado mensajes en el fichero que marcan el principio de una nueva TDMA e informan de la mitad de la TDMA en la que se encuentra el sistema. Además se informa también del slot en el que nos encontramos dentro de la TDMA. A continuación se muestran algunos ejemplos de fragmentos de uno de estos ficheros:

5.5.7.1. Ejemplos del fichero “MacRxDwn.log”

En la figura 5.18 se muestra un ejemplo de recepción de paquetes para el caso de que el servicio sea de corrección de errores con múltiples subcampos. Se comprueba que los paquetes sobre las portadoras 0 y 1 han llegado sin errores y que el SN de los paquetes coincide con el que se esperaba por lo que el receptor avanza la ventana para cada portadora.

```
Evento en :0.61
Inicio de nueva trama Tdma

Slot: 0
Paquete del transmisor sobre portadora: 0
Se espera SN= 0
Receptor recibió nuevo paquete del transmisor
SN coincide con el que se esperaba
Se pasa a comprobar si el campo de datos es correcto
Todos los B-subfields correctos
Avanzo normal, NEXT_BCK= 1
Se entregan datos a Dlc de la portadora: 0

Slot: 1
Paquete del transmisor sobre portadora: 1
Se espera SN= 0
Receptor recibió nuevo paquete del transmisor
SN coincide con el que se esperaba
```

```

Se pasa a comprobar si el campo de datos es correcto
Todos los B-subfields correctos
Avanzo normal, NEXT_BCK= 1
Se entregan datos a Dlc de la portadora: 1
    
```

Figura 5. 18 Ejemplo de recepción de paquetes a nivel MAC (“MacRxDown.log”)

En la figura 5.19 se muestra un ejemplo de recepción de paquetes para el caso de que el servicio sea de corrección de errores con múltiples subcampos. El paquete sobre la portadora en el slot 0 es un nuevo paquete que ha llegado con la cabecera correcta, pero con error en varios de los subcampos B. El receptor almacena los subcampos y pide retransmisión del paquete. El paquete sobre la portadora en el slot 1 es un paquete retransmitido por el transmisor que tiene todos los campos correctos por lo que se actualiza la ventana y el receptor espera un nuevo paquete.

```

-----
Evento en :0.63
Inicio de nueva trama Tdma
-----

Slot: 0
Paquete del transmisor sobre portadora: 0
Se espera SN= 0
Receptor recibió nuevo paquete del transmisor
SN coincide con el que se esperaba
Se pasa a comprobar si el campo de datos es correcto
Algún B-subfield incorrecto
El paquete recibido tiene:

subcampo B[0]= 0
subcampo B[1]= 1
subcampo B[2]= 0
subcampo B[3]= 1
subcampo B[4]= 0
subcampo B[5]= 0
subcampo B[6]= 1
subcampo B[7]= 0
subcampo B[8]= 1
subcampo B[9]= 1
subcampo B[10]= 1
subcampo B[11]= 1

Almaceno B-subfields y pido retransmitir el paquete con SN= 0

Slot: 1
Paquete del transmisor sobre portadora: 1
Se espera SN= 1
Receptor recibe paquete retransmitido del transmisor
SN coincide con el que se esperaba
Se pasa a comprobar si el campo de datos es correcto
Todos los B-subfields correctos
Avanzo normal, NEXT_BCK= 0
Se entregan datos a Dlc de la portadora: 1
    
```

Figura 5. 19 Ejemplo de recepción de paquetes a nivel MAC (MacRxDown.log”)

5.5.8. Fichero “Channel.log”

El fichero “channel.log” recoge los eventos ocurridos en el módulo “channel”, el cual modela el canal aéreo de DECT. Consultando este fichero se puede obtener información sobre el estado del canal (bueno o malo) en cada instante de simulación y

los paquetes que han sufrido errores al viajar por el canal en el caso de que el estado sea malo.

5.5.8.1. Ejemplos del fichero “Channel.log”

En la figura 5.20 se muestra un ejemplo de cambio de estado del canal y recepción y procesado de un paquete del enlace “downlink” (de la FP a la PP).

```
Evento en: 0.000416
Cambio de estado bueno a malo
Evento en: 0.010416
Cambio de estado malo a bueno
Channel: Recibido paquete de FP para PP en:0.0153601
Paquete sin errores
-----
```

Figura 5. 20 Ejemplo de cambio del estado del canal

En la figura 5.21 se muestra un ejemplo de cambio recepción y procesado de un paquete del enlace “downlink” (de la FP a la PP). El canal se encuentra en estado malo por lo que se han generado los errores 0de acuerdo con el formato de subcampos del paquete.

```
Channel: Recibido paquete de FP para PP en:1.20578

Paquete con errores
Error en los subcampos de datos del paquete
Número de Errores= 5

Error en subcampo B: 4
Error en subcampo B: 5
Error en subcampo B: 3
Error en subcampo B: 1
Error en subcampo B: 2
```

Figura 5. 21 Ejemplo del canal en estado malo

5.5.9. Ficheros “Downlink.log” y “Uplink.log”

En los ficheros “Downlink.log” y “Uplink.log” se proporciona una visión general del viaje de los paquetes través de los diferentes módulos entre los extremos de la comunicación para los enlaces “uplink”y “downlink” respectivamente. Para proporcionar esta visión general se han seleccionado dentro de cada módulo los mensajes que permiten seguir el viaje del paquete. Estos ficheros son ideales para obtener una visión completa del paso de mensajes entre los módulos para cada enlace. El formato de los mensajes dentro de este módulo es el siguiente:

X à Y : Z

Donde:

- X: Tiempo de simulación en el que se produjo el evento.
- Y: Nombre del módulo.

- Z: Mensaje del módulo

5.5.9.1. Ejemplo del fichero “Downlink.log”

En la figura 5.22 se muestra un ejemplo completo de transmisión a través del enlace de descarga (“downlink”) de un paquete IP procedente del servidor.

```

0-->Server TCP IP: 0Transmite SN: 1
0.002304-->Iwu Router: Llega paquete IP
0.002304-->Iwu Router: Se entrega SDU a Dlc
0.002304-->Transmisor Dlc: Ha llegado nueva SDU, número de SDUs en el buffer:
1
0.01-->Transmisor Dlc: Tdmas para SDU= 0 en buffer: 1
0.01-->Transmisor Dlc: Número de SDUs en el Buffer = 1
0.01-->Receptor Dlc: No hay PDUs fuera de secuencia
0.014576-->Transmisor Mac: Petición de datos para la conexión
0.014576-->Transmisor Dlc: Mac pide 9 segmentos
Transmisor Dlc: Entrega: 0----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...0]
Transmisor Dlc: Entrega: 1----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...1]
Transmisor Dlc: Entrega: 2----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...2]
Transmisor Dlc: Entrega: 3----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...3]
Transmisor Dlc: Entrega: 4----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...4]
Transmisor Dlc: Entrega: 5----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...5]
Transmisor Dlc: Entrega: 6----->
0.014576-->Transmisor Dlc:Ventana del transmisor: [0...6]
0.014992-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 10 en
slot: 12
0.014992-->Transmisor Mac: Segmento Dlc (SN:0)
0.0153601-->Receptor Mac: Paquete del transmisor sobre portadora: 10
0.0153601-->Receptor Mac: Se espera SN= 0
0.0153601-->Receptor Mac: Receptor recibió nuevo paquete del transmisor
0.0153601-->Receptor Mac: SN coincide con el que se esperaba
0.0153601-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0153601-->Receptor Mac: Todos los B-subfields correctos
0.0153601-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0153601-->Receptor Mac: Se entregan datos a Dlc de la portadora: 10
0.0153601-->Receptor Dlc: Llega Segmento de la capa Mac
0.0153601-->Receptor Dlc: Receptor Dlc: Tengo el SN= 0
0.0153601-->Receptor Dlc: Se actualiza ventana
0.0153601-->Receptor Dlc: Se espera PDU: 1
0.0153601-->Receptor Dlc: Ventana de recepción: [1...64]
0.015408-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 11 en
slot: 13
0.015408-->Transmisor Mac: Segmento Dlc (SN:1)
0.0157761-->Receptor Mac: Paquete del transmisor sobre portadora: 11
0.0157761-->Receptor Mac: Se espera SN= 0
0.0157761-->Receptor Mac: Receptor recibió nuevo paquete del transmisor
0.0157761-->Receptor Mac: SN coincide con el que se esperaba
0.0157761-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0157761-->Receptor Mac: Todos los B-subfields correctos
0.0157761-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0157761-->Receptor Mac: Se entregan datos a Dlc de la portadora: 11
0.0157761-->Receptor Dlc: Llega Segmento de la capa Mac
0.0157761-->Receptor Dlc: Receptor Dlc: Tengo el SN= 1
0.0157761-->Receptor Dlc: Se actualiza ventana
0.0157761-->Receptor Dlc: Se espera PDU: 2
0.0157761-->Receptor Dlc: Ventana de recepción: [2...65]
0.015824-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 12 en
slot: 14

```

```

0.015824-->Transmisor Mac: Segmento Dlc (SN:2)
0.0161921-->Receptor Mac: Paquete del transmisor sobre portadora: 12
0.0161921-->Receptor Mac: Se espera SN= 0
0.0161921-->Receptor Mac: Receptor recibió nuevo paquete del transmisor
0.0161921-->Receptor Mac: SN coincide con el que se esperaba
0.0161921-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0161921-->Receptor Mac: Todos los B-subfields correctos
0.0161921-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0161921-->Receptor Mac: Se entregan datos a Dlc de la portadora: 12
0.0161921-->Receptor Dlc: Llega Segmento de la capa Mac
0.0161921-->Receptor Dlc: Receptor Dlc: Tengo el SN= 2
0.0161921-->Receptor Dlc: Se actualiza ventana
0.0161921-->Receptor Dlc: Se espera PDU: 3
0.0161921-->Receptor Dlc: Ventana de recepción: [3...66]
0.01624-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 13 en
slot: 15
0.01624-->Transmisor Mac: Segmento Dlc (SN:3)
0.0166081-->Receptor Mac: Paquete del transmisor sobre portadora: 13
0.0166081-->Receptor Mac: Se espera SN= 0
0.0166081-->Receptor Mac: Receptor recibió nuevo paquete del transmisor
0.0166081-->Receptor Mac: SN coincide con el que se esperaba
0.0166081-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0166081-->Receptor Mac: Todos los B-subfields correctos
0.0166081-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0166081-->Receptor Mac: Se entregan datos a Dlc de la portadora: 13
0.0166081-->Receptor Dlc: Llega Segmento de la capa Mac
0.0166081-->Receptor Dlc: Receptor Dlc: Tengo el SN= 3
0.0166081-->Receptor Dlc: Se actualiza ventana
0.0166081-->Receptor Dlc: Se espera PDU: 4
0.0166081-->Receptor Dlc: Ventana de recepción: [4...67]
0.016656-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 14 en
slot: 16
0.016656-->Transmisor Mac: Segmento Dlc (SN:4)
0.0170241-->Receptor Mac: Paquete del transmisor sobre portadora: 14
0.0170241-->Receptor Mac: Se espera SN= 0
0.0170241-->Receptor Mac: Receptor recibió nuevo paquete del transmisor
0.0170241-->Receptor Mac: SN coincide con el que se esperaba
0.0170241-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0170241-->Receptor Mac: Todos los B-subfields correctos
0.0170241-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0170241-->Receptor Mac: Se entregan datos a Dlc de la portadora: 14
0.0170241-->Receptor Dlc: Llega Segmento de la capa Mac
0.0170241-->Receptor Dlc: Receptor Dlc: Tengo el SN= 4
0.0170241-->Receptor Dlc: Se actualiza ventana
0.0170241-->Receptor Dlc: Se espera PDU: 5
0.0170241-->Receptor Dlc: Ventana de recepción: [5...68]
0.017072-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 15 en
slot: 17
0.017072-->Transmisor Mac: Segmento Dlc (SN:5)
0.0174401-->Receptor Mac: Paquete del transmisor sobre portadora: 15
0.0174401-->Receptor Mac: Se espera SN= 0
0.0174401-->Receptor Mac: Receptor recibió nuevo paquete del transmisor
0.0174401-->Receptor Mac: SN coincide con el que se esperaba
0.0174401-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0174401-->Receptor Mac: Todos los B-subfields correctos
0.0174401-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0174401-->Receptor Mac: Se entregan datos a Dlc de la portadora: 15
0.0174401-->Receptor Dlc: Llega Segmento de la capa Mac
0.0174401-->Receptor Dlc: Receptor Dlc: Tengo el SN= 5
0.0174401-->Receptor Dlc: Se actualiza ventana
0.0174401-->Receptor Dlc: Se espera PDU: 6
0.0174401-->Receptor Dlc: Ventana de recepción: [6...69]
0.017488-->Transmisor Mac:Transmisión de paquete MAC sobre portadora: 16 en
slot: 18
0.017488-->Transmisor Mac: Segmento Dlc (SN:6)
0.0178561-->Receptor Mac: Paquete del transmisor sobre portadora: 16
0.0178561-->Receptor Mac: Se espera SN= 0
0.0178561-->Receptor Mac: Receptor recibió nuevo paquete del transmisor

```

```

0.0178561-->Receptor Mac: SN coincide con el que se esperaba
0.0178561-->Receptor Mac: Se pasa a comprobar si el campo de datos es correcto
0.0178561-->Receptor Mac: Todos los B-subfields correctos
0.0178561-->Receptor Mac: Avanzo normal, NEXT_BCK= 1
0.0178561-->Receptor Mac: Se entregan datos a Dlc de la portadora: 16
0.0178561-->Receptor Dlc: Llega Segmento de la capa Mac
0.0178561-->Receptor Dlc: Receptor Dlc: Tengo el SN= 6
0.0178561-->Receptor Dlc: La PDU contiene el final de una SDU que no ha
expirado
0.0178561-->Receptor Dlc: Entregada SDU a la capa superior
0.0178561-->Receptor Dlc: Número de SDUs entregadas: 1
0.0178561-->Receptor Dlc: Se actualiza ventana
0.0178561-->Receptor Dlc: Se espera PDU: 7
0.0178561-->Receptor Dlc: Ventana de recepción: [7...70]
0.0178561-->Iwu : Llega trama SDU
0.0178561-->Iwu : Se envía paquete TCP/IP a la aplicación
0.0178561-->Application TCP IP 1 ha recibido el paquete: 1
0.019152-->Receptor Mac: Mac pide datos de control a dlc
0.019152-->Receptor Dlc: Mac pide información de control
0.019152-->Receptor Dlc: <----- ACK: 6
0.019152-->Receptor Mac: Llegan a Mac los datos de Dlc
0.019152-->Receptor Mac: Se obtiene la información del canal Gf
0.019152-->Receptor Mac: Transmisión de paquete Ack
0.0195201-->Transmisor Mac: Recibido Ack
0.0195201-->Transmisor Mac: Se entregan datos de control a Dlc
0.0195201-->Transmisor Dlc: <----- Ack: 6
0.0195201-->Transmisor Dlc: Se Confirma: 0
0.0195201-->Transmisor Dlc: Se Confirma: 1
0.0195201-->Transmisor Dlc: Se Confirma: 2
0.0195201-->Transmisor Dlc: Se Confirma: 3
0.0195201-->Transmisor Dlc: Se Confirma: 4
0.0195201-->Transmisor Dlc: Se Confirma: 5
0.0195201-->Transmisor Dlc: Se Confirma: 6
0.0195201-->Transmisor Dlc: Se confirma la entrega de la SDU: 0
0.019568-->Transmisor Mac: Petición de datos para la conexión
0.019568-->Transmisor Dlc: Mac pide 10 segmentos
0.019568-->Transmisor Mac: Mac recibe mensaje con 0 segmentos de Dlc
0.019568-->Transmisor Mac: Se cancelarán todas las retransmisiones en curso

```

Figura 5. 22 transmisión de un paquete IP del servidor a la parte portátil

Capítulo 6

Evaluación de prestaciones mediante simulación

6.1. Introducción:

Nuestro objetivo es evaluar el comportamiento de los servicios definidos en las capas DLC y MAC del estándar DECT implementados en nuestro simulador. Para la realización de los experimentos se usará el escenario de referencia que ha sido descrito en el apartado 3.2 del capítulo 3. Los experimentos que se van a realizar son:

- Experimentos para un canal con errores.
- Experimentos para un canal sin errores.

Los resultados y conclusiones obtenidas para cada experimento se incluyen en los apartados dedicados a cada uno de ellos.

6.2. Servicios asimétricos para simular:

En este apartado se van a definir los servicios que se van a simular en los experimentos que se van a llevar cabo con nuestro simulador. La notación que se ha empleado para identificar el tipo de servicio está basada en la empleada en el estándar. La notación es:

X f Y (ssub/msub)

- **X**: representa el tipo de servicio de protección. Puede tomar 2 valores:
 - a) 8 → modo protegido con corrección de errores
 - b) 7 → modo protegido con detección de errores
- **f**: indica que el formato de slot empleado es el de slot completo.
- **Y**: representa el nivel de modulación del campo de datos del paquete. Puede tomar los valores:
 - a) 2 → dos niveles de modulación.
 - b) 4 → cuatro niveles de modulación.
 - c) 8 → ocho niveles de modulación.
- **ssub**: se empleará cuando se quiera indicar que el formato de protección es de un solo campo.
- **msub**: se empleará cuando se quiera indicar que el formato de protección es de múltiples subcampos.

De los 12 servicios que se van a simular, hay 10 servicios asimétricos que se encuentran recogidos en unas tablas incluidas en el documento del estándar que describe la capa MAC[6] y son los siguientes:

- 8f2 msub à corrección de errores, formato de slot completo, 2 niveles de modulación, protección de múltiples subcampos.
- 8f4 msub à corrección de errores, formato de slot completo, 4 niveles de modulación, protección de múltiples subcampos.
- 8f8 msub à corrección de errores, formato de slot completo, 8 niveles de modulación, protección de múltiples subcampos.
- 7f2 msub à detección de errores, formato de slot completo, 2 niveles de modulación, protección de múltiples subcampos.
- 7f4 msub à detección de errores, formato de slot completo, 4 niveles de modulación, protección de múltiples subcampos.
- 7f8 msub à detección de errores, formato de slot completo, 8 niveles de modulación, protección de múltiples subcampos.
- 8f2 ssub à corrección de errores, formato de slot completo, 2 niveles de modulación, protección de un solo campo.
- 7f2 ssub à detección de errores, formato de slot completo, 2 niveles de modulación, protección de un solo campo.
- 7f4 ssub à detección de errores, formato de slot completo, 4 niveles de modulación, protección de un solo campo.
- 7f8 ssub à detección de errores, formato de slot completo, 8 niveles de modulación, protección de un solo campo.

Existen además dos servicios adicionales que no se encuentran recogidos en las tablas del estándar pero que si aparecen definidos como tales en el resto de documentos del estándar. Estos servicios son:

- 8f4 ssub à corrección de errores, formato de slot completo, 4 niveles de modulación, protección de un solo campo.
- 8f8 ssub à corrección de errores, formato de slot completo, 8 niveles de modulación, protección de un solo campo.

A pesar de no aparecer en la tabla del estándar, estos servicios serán también simulados siguiendo las reglas que especifica el estándar para ellos. Para las simulaciones se ha considerado que el servicio ofrecido por cada conexión completamente asimétrica va a ser el mismo para el enlace al que esté asignada (“downlink” o “uplink”).

6.3. Cálculo teórico del Throughput

Para realizar los cálculos de la velocidad del campo de datos del paquete MAC se han considerado únicamente los bits que incluyen información de la PDU con la información TCP/IP (despreciando la información de control de la PDU), despreciando los bits que contienen información de protección de CRC. A continuación se muestra un ejemplo del cálculo de la velocidad para el caso del servicio 8f2 con $k=1, m=0$ y $n=1$. En primer lugar debemos obtener el tamaño máximo del campo de datos para la PDU

para el tipo de servicio solicitado. Este valor se consulta en la tabla 13.3.2.2.1 del estándar dedicado a la capa Dlc [7] y se obtiene que para 8f2 es 30 bytes. Ahora se implementa para calcular la velocidad en bits se realiza la operación:

$$Throughput_{8f2} = \frac{MaxPDUPayload}{TDMAtime} = \frac{(30 \times 8)}{10} = 24Kbps$$

Si se deseara calcular la velocidad teniendo en cuenta la información de control de la cabecera de la PDU se debe consultar la tabla 12.2.1.1 del estándar [7]. En esta tabla obtenemos el tamaño máximo de la PDU que es de 32 bytes. Como se observa para obtener el valor anterior simplemente habrá que restarle a 32 los dos bytes de la cabecera de la PDU. Si consideramos la cabecera de la PDU el valor obtenido es de:

$$Throughput_{8f2} = \frac{MaxPDUSize}{TDMAtime} = \frac{(32 \times 8)}{10} = 25,6Kbps$$

Este valor de la velocidad coincide con los valores que podemos encontrar en la tabla 5.4 del estándar [6].

En las simulaciones que se van a realizar se desea comparar este valor teórico con el valor de "throughput" del tráfico TCP/IP, por lo que se usará el cálculo de la velocidad sin tener en cuenta la cabecera de la PDU.

En la tabla 6.1 se muestra el cálculo de este "throughput" teórico para cada servicio para el caso de la conexión de menor capacidad (k = 1, n = 1). En la columna "PDU Payload" se muestra el tamaño (en bytes) del campo de datos de la PDU extraído de la tabla 13.3.2.2.1 del estándar. El "throughput" se muestra en Kbits por segundo(Kbps)

Tipo de servicio	PDU Payload (en bytes)	Throughput (en Kbps)
8f2 msub	30	24 Kbps
8f4 msub	62	49,6 Kbps
8f8 msub	94	75,2 Kbps
7f2 msub	30	24 Kbps
7f4 msub	62	49,6 Kbps
7f8 msub	94	75,2 Kbps
8f2 ssub	36	28,8 Kbps
8f4 ssub	74	59,2 Kbps
8f8 ssub	114	91,2 Kbps
7f2 ssub	36	28,8 Kbps
7f4 ssub	74	59,2 Kbps
7f8 ssub	114	91,2 Kbps

Tabla 6. 1 Throughput teórico para la conexión de menor capacidad

Para el caso de la conexión de mayor capacidad, el "throughput" máximo o teórico depende del número de portadoras "k" en la dirección principal. Como el número de portadoras "k" varía en función del modelo de configuración se ha elaborado una tabla para cada modelo. En la y tabla 6.2 la tabla 6.3 se muestran los valores de

"throughput" teórico calculados para las conexiones de mayor capacidad del modelo 1 ($k = 19, m = 0, n = 1$) y del modelo 2 ($k = 20, m = 0, n = 2$) respectivamente.

Tipo de Servicio	Bytes en la Tdma	Throughput total
8f2 msub	570	456 Kbps
8f4 msub	1178	942 Kbps
8f8 msub	1786	1,4288 Mbps
7f2 msub	570	456 Kbps
7f4 msub	1178	942 Kbps
7f8 msub	1786	1,4288 Mbps
8f2 ssub	684	547,2 Kbps
8f4 ssub	1406	1,1248 Mbps
8f8 ssub	2166	1,7328 Mbps
7f2 ssub	684	547,2 Kbps
7f4 ssub	1406	1,1248 Mbps
7f8 ssub	2166	1,7328 Mbps

Tabla 6. 2 Throughput teórico para la conexión de mayor capacidad (modelo 1)

Tipo de Servicio	Bytes en la Tdma	Total Throughput
8f2 msub	600	480 Kbps
8f4 msub	1240	992 Kbps
8f8 msub	1880	1,5040 Mbps
7f2 msub	600	480 Kbps
7f4 msub	1240	992 Kbps
7f8 msub	1880	1,5040 Mbps
8f2 ssub	720	576 Kbps
8f4 ssub	1480	1,184 Mbps
8f8 ssub	2280	1,824 Mbps
7f2 ssub	720	576 Kbps
7f4 ssub	1480	1,184 Mbps
7f8 ssub	2280	1,824 Mbps

Tabla 6. 3 Throughput teórico para la conexión de mayor capacidad (modelo 2)

En la columna "Total bytes in Tdma" de ambas tablas se almacenan el número máximo de bits que se pueden transportar en una trama Tdma (10 mseg) y se calcula multiplicando el número de bits de información útil de cada portadora por el número de portadoras "k" de la conexión de mayor capacidad de cada modelo. El número de bytes útiles para una portadora se puede extraer de la segunda columna de la tabla 6.1. El mismo procedimiento se sigue para calcular los valores de la columna "Total Throughput".

6.4. Ventana de recepción de la aplicación

Como se ha comentado anteriormente, el receptor indica al transmisor el tamaño de su ventana de recepción (*rwnd*). Este tamaño será usado por el transmisor para fijar el valor máximo de la ventana de congestión y el nivel inicial de "ssthresh". En nuestro simulador se ha suprimido la fase de establecimiento de las conexiones por lo que se supondrá que el tamaño de la ventana de recepción es ya conocido por el transmisor. El valor a emplear para *rwnd* será de 64 Kbytes tal y como se encuentra expresado en [30]. A partir de *rwnd* se hace el cálculo del parámetro "ssthresh". El valor de "ssthresh" se expresa en paquetes y se ha obtenido dividiendo el tamaño en bytes de la *rwnd* entre el tamaño en bytes del paquete IP. A continuación se muestran los valores de "ssthresh" calculados para cada uno de los dos tamaños de paquetes IP que se van a emplear en las simulaciones:

Para SMSS = 1024:

$$ssthresh_{1024} = \frac{rwnd}{SMSS} = \frac{65536}{1024} = 64$$

Para SMSS = 536:

$$ssthresh_{1024} = \frac{rwnd}{SMSS} = \frac{65536}{536} = 122,2686$$

Redondeando se obtienen unos valores de 64 y 122.

6.5. Tamaño del Buffer de la FT

El buffer de la FT donde se almacenan en forma de SDUs los paquetes IP que llegan a través de la IWU procedentes del servidor debe tener en principio un tamaño limitado. Cuando un paquete IP del servidor llegue a la FT, será descartado en el caso de que no haya espacio en el buffer para almacenarlo. Esta situación dará lugar a la pérdida de paquetes IP lo que provocará una reacción por parte del servidor para evitar la congestión. Dado que se van a emplear paquetes IP de dos tamaños diferentes, en nuestras simulaciones las SDUs que se van a almacenar en el buffer de FT van a ser también de dos tamaños. El tamaño del buffer que se va emplear en las simulaciones para SDUs de 1064 bytes será de 20 SDUs. A partir de este tamaño de buffer expresado en paquetes, se puede calcular el tamaño en bytes de la memoria como:

$$buffer_{FT} = 20 \times (1024 + 40) = 21280 \text{ bytes}$$

Partiendo del tamaño del buffer expresado en bytes (redondeando a 21 Kbytes), se ha calculado el número de SDUs de 576 bytes que caben en el buffer:

$$SDUs = \frac{21504}{576} = 37,33$$

Eliminando la parte decimal, se calcula que el número de SDUs de 576 bytes que se pueden almacenar en el buffer de 21 Kbytes es de 37.

6.6. Selección del tiempo de vida de las SDUs

Los procedimientos de la clase 2 seleccionada a nivel DLC, ofrecen la opción de limitar el tiempo de vida de las SDUs en el buffer del transmisor. El tiempo de vida se expresa en TDMA's y su valor se deja al criterio del administrador. Al limitar el tiempo de vida de las SDUs se evitan situaciones como por ejemplo que el número de SDUs en el buffer del transmisor DLC crezca de manera exponencial provocando una saturación del buffer o que se ocupe el ancho de banda en la retransmisión de SDUs que ya han sido entregadas al receptor. Las PDUs que contienen información de las SDUs que han expirado no volverán a ser retransmitidas y en el caso de que llegue una petición de su retransmisión, se sustituirá la última PDU expirada por un mensaje de sincronización para la ventana del receptor. El limitar el tiempo de vida de los paquetes puede provocar la pérdida de SDUs en el caso de que durante su tiempo de vida no hayan sido incluidas completamente en la ventana de transmisión. Nuestro objetivo es seleccionar un tiempo de vida que nos permita determinar la reacción de la fuente TCP/IP del servidor cuando la FT decide eliminar las SDUs expiradas en su buffer. Para realizar un valor del tiempo de vida a asignar a cada SDU se ha tenido en cuenta que la parte transmisora de datos del enlace "downlink" (la FT) posee un buffer limitado. Para evitar saturar este buffer se ha estimado (suponiendo que llega una SDU por cada TDMA) un tiempo de vida de las SDUs igual al tamaño del buffer.

6.7. Experimentos para un canal sin errores

El objetivo de estos experimentos es evaluar el "throughput" medio que alcanza la fuente TCP reno para cada servicio y validar el simulador, mediante su comparación con los valores de "throughput" teóricos calculados en el apartado 6.3. Los experimentos a realizar se muestran en la tabla 6.4:

Experimento	Tamaño de los segmentos TCP (SMSS)	Modelo	Buffer Limitado	Expiración de SDUs	Modificación de la conexión
1	536 bytes	1	NO	NO	NO
2	536 bytes	2	NO	NO	NO
3	1024 bytes	1	NO	NO	NO
4	1024 bytes	2	NO	NO	NO
5	536 bytes	1	SI	NO	NO
6	536 bytes	2	SI	NO	NO
7	1024 bytes	1	SÍ	NO	NO
8	1024 bytes	2	SI	NO	NO

Tabla 6. 4 Experimentos para un canal sin errores

Los resultados obtenidos para cada experimento se muestran en los apartados siguientes.

6.7.1. Experimento 1

En este experimento se han empleado segmentos TCP con un tamaño de 536 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 1 suponiendo un buffer ideal para almacenar los paquetes que llegan a la

parte fija del sistema DECT procedentes de la red IP. Además los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.5:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	541,489 Kbps	0,890245 s
7f2msub	454,245 Kbps	1,079720 s
7f4ssub	1,0967 Mbps	0,399161 s
7f4msub	921,494 Kbps	0,348477 s
7f8ssub	1,6742 Mbps	0,184997 s
7f8msub	1,3862 Mbps	0,231532 s
8f2ssub	542,511 Kbps	0.892945 s
8f2msub	454,345 Kbps	1,081910 s
8f4ssub	1,0981 Mbps	0,401870 s
8f4msub	922,514 Kbps	0,351298 s
8f8ssub	1,6779 Mbps	0,231351 s
8f8msub	1,3881 Mbps	0,300015 s

Tabla 6. 5 Resultados del experimento 1

Vemos como para el caso de suponer un buffer ideal (ilimitado) en la estación base del sistema DECT para almacenar los paquetes IP que vienen del servidor que los valores obtenidos de throughput se aproximan para cada tipo de servicio al valor teórico calculado en el apartado 6.3. La razón por la que no se alcanza el valor teórico exacto se debe al retardo inicial introducido por el algoritmo "Slow Start".

Si comparamos los servicios proporcionados vemos como el throughput aumenta conforme aumenta el nivel de modulación seleccionado. En cuanto al tipo de control de flujo seleccionado a nivel de MAC se observa que, manteniendo el nivel de modulación y el formato de protección, no existe mucha variación entre emplear la corrección o la detección de errores, puesto que se ha supuesto un canal sin errores. Las ligeras variaciones entre ambos resultados surgen a causa del instante en que la capa MAC entrega la información a la capa DLC tal y como se ha comentado en el apartado 4.2.2.3 del capítulo 4. Por último se pueden comentar las diferencias surgidas para el caso de servicios que tienen el mismo nivel de modulación, el mismo control de errores, pero diferente formato de protección de la información. Así pues,

para el caso de emplear una protección de múltiples subcampos (msub) se puede observar como el throughput es menor puesto que al usar múltiples CRCs, disminuye el espacio disponible en el paquete para los datos. El uso de múltiples CRCs también afecta al retardo de tránsito de los paquetes. Vemos que como para el caso de msub, este retardo es mayor, ya que al haber menos espacio en el paquete para la información, se incluirán menos paquetes enteros de las capas superiores.

6.7.2. Experimento 2

En este experimento se han empleado segmentos TCP con un tamaño de 536 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 2 suponiendo un buffer ideal para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. Además los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.6:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	569,137 Kbps	0.841921 s
7f2msub	477,182 Kbps	1,022570 s
7f4ssub	1,15302 Mbps	0,278850 s
7f4msub	969,930 Kbps	0,464892 s
7f8ssub	1,7569 Mbps	0,173569 s
7f8msub	1,4579 Mbps	0,218560 s
8f2ssub	567,701 Kbps	0,545846 s
8f2msub	478,102 Kbps	1,029790 s
8f4ssub	1,1579 Mbps	0,381394 s
8f4msub	971,056 Kbps	0,472005 s
8f8ssub	1,7765 Mbps	0,219091 s
8f8msub	1,4613 Mbps	0,284805 s

Tabla 6. 6 Resultados del experimento 2

Se observa como para el caso de suponer un buffer ideal (ilimitado) en la estación base del sistema DECT para almacenar los paquetes IP que vienen del servidor que los valores obtenidos de throughput se aproximan para cada tipo de servicio al valor

teórico calculado en el apartado 6.3. La razón por la que no se alcanza el valor teórico exacto se debe al retardo inicial introducido por el algoritmo “Slow Start”. Las conclusiones obtenidas comparando los tipos de servicios son las mismas que para el experimento 1.

Si comparamos los resultados con los obtenidos para el experimento 1 se puede verificar que en el experimento 2 el throughput sale más alto ya que el modelo de configuración de las portadoras (modelo 2) usa una portadora más para el enlace downlink. Así pues en el experimento 2 gracias al ancho de banda extra proporcionado por la portadora adicional, se puede comprobar como el retardo de tránsito de los paquetes es menor que el retardo para los paquetes del experimento 1.

6.7.3. Experimento 3

En este experimento se han empleado segmentos TCP con un tamaño de 1024 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 1 suponiendo un buffer ideal para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. Además los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.7:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	543,818 Kbps	0,856366 s
7f2msub	455672 Kbps	1,039660 s
7f4ssub	1,0966 Mbps	0,293146 s
7f4msub	925,724 Kbps	0,474018 s
7f8ssub	1,6853 Mbps	0,219000 s
7f8msub	1,3929 Mbps	0,284807 s
8f2ssub	544,196 Kbps	0,859917 s
8f2msub	456,429 Kbps	1,042950 s
8f4ssub	1,1027 Mbps	0,387139 s
8f4msub	927,611 Kbps	0,476865 s
8f8ssub	1,6853 Mbps	0,221647 s
8f8msub	1,3944 Mbps	0,287651 s

Tabla 6. 7 Resultados del experimento 3

Se observa como para el caso de suponer un buffer ideal (ilimitado) en la estación base del sistema DECT para almacenar los paquetes IP que vienen del servidor que los valores obtenidos de throughput se aproximan para cada tipo de servicio al valor teórico calculado en el apartado 6.3. La razón por la que no se alcanza el valor teórico exacto se debe al retardo inicial introducido por el algoritmo “Slow Start”. Las

conclusiones obtenidas comparando los tipos de servicios son las mismas que para el experimento 1.

Si comparamos los resultados de este experimento con los del experimento 1 vemos que para un mismo modelo de configuración de las portadoras (modelo 1) al usar segmentos TCP de 1024 bytes obtenemos mejores resultados que al usar segmentos TCP de 536 bytes.

6.7.4. Experimento 4

En este experimento se han empleado segmentos TCP con un tamaño de 1024 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 2 suponiendo un buffer ideal para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. Además los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.8:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	571,812 Kbps	0,810468 s
7f2msub	478,749 Kbps	0,985090 s
7f4ssub	1,15838 Mbps	0,360895 s
7f4msub	973,391 Kbps	0,446332 s
7f8ssub	1,77238 Mbps	0,201687 s
7f8msub	1,46482 Mbps	0,266223 s
8f2ssub	571,813 Kbps	0,817575 s
8f2msub	479,506 Kbps	0,992268 s
8f4ssub	1,1597 Mbps	0,36798 s
8f4msub	975,468 Kbps	0,453413 s
8f8ssub	1,77446 Mbps	0,208905 s
8f8msub	1,46891 Mbps	0,273291 s

Tabla 6. 8 resultados del experimento 4

Se observa como para el caso de suponer un buffer ideal (ilimitado) en la estación base del sistema DECT para almacenar los paquetes IP que vienen del servidor que los valores obtenidos de throughput se aproximan para cada tipo de servicio al valor teórico calculado en el apartado 6.3. La razón por la que no se alcanza el valor teórico exacto se debe al retardo inicial introducido por el algoritmo "Slow Start". Las conclusiones obtenidas comparando los tipos de servicios son las mismas que para el experimento 1.

Si comparamos los resultados de este experimento con los de los experimento 2 vemos que para un mismo modelo de configuración de las portadoras (modelo 2) al

usar segmentos TCP de 1024 bytes obtenemos mejores resultados que al usar segmentos TCP de 536 bytes.

6.7.5. Experimento 5

En este experimento se han empleado segmentos TCP con un tamaño de 536 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 1 suponiendo un buffer real (limitado) para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. El tamaño del buffer será el especificado en el apartado 6.5. Los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.9:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	417,791 Kbps	0,230112 s
7f2msub	407,243 Kbps	0,266701 s
7f4ssub	746,903 Kbps	0,131021 s
7f4msub	710,756 Kbps	0,14591 s
7f8ssub	1,088 Mbps	0,10997 s
7f8msub	423,727 Kbps	0,244448 s
8f2ssub	458,748 Kbps	0,264521 s
8f2msub	386,557 Kbps	0,259345 s
8f4ssub	759,290 Kbps	0,129201 s
8f4msub	723,757 Kbps	0,14504 s
8f8ssub	1,5929 Mbps	0,114895 s
8f8msub	886,162 Kbps	0,116386 s

Tabla 6. 9 Resultados del experimento 5

Comparando estos resultados con los obtenidos para el caso de que se suponga un buffer ideal (experimento 1) donde no se descartan paquetes, se observa una disminución lógica del throughput ocasionada por la pérdida de paquetes. Los paquetes perdidos son paquetes descartados por la parte transmisora del enlace “downlink” del sistema DECT, la cual no dispone de espacio suficiente en el buffer para almacenarlos. El servidor gracias al algoritmo de TCP reno implementado consigue detectar la pérdida de paquetes en el receptor y regula el tamaño de su ventana para evitar la congestión con lo que disminuye el throughput. Al limitar el buffer, disminuye también el tiempo de retardo extremo a extremo de los paquetes puesto que el tiempo máximo que va a tener que estar un paquete en el buffer de transmisión del sistema DECT antes de ser transmitido por el enlace downlink va a ir en función del número máximo de SDUs que puedan almacenarse en dicho buffer. Los

resultados obtenidos comparando los servicios son los mismos que para el caso del experimento 1.

6.7.6. Experimento 6

En este experimento se han empleado segmentos TCP con un tamaño de 536 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 2 suponiendo un buffer real (limitado) para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. El tamaño del buffer será el especificado en el apartado 6.5. Los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.10:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	436,016 Kbps	0,219656 s
7f2msub	371,199 Kbps	0,256175 s
7f4ssub	781,821 Kbps	0,132776 s
7f4msub	684,746 Kbps	0,143346 s
7f8ssub	1,13899Mbps	0,108315 s
7f8msub	1,4028 Mbps	0,129854 s
8f2ssub	447,382 Kbps	0,217828 s
8f2msub	378,572 Kbps	0,251943 s
8f4ssub	814,385 Kbps	0,129708 s
8f4msub	690,993 Kbps	0,142952 s
8f8ssub	1,17698 Mbps	0,108809 s
8f8msub	962,352 Kbps	0,116194 s

Tabla 6. 10 Resultados del experimento 6

Comparando estos resultados con los obtenidos para el caso de que se suponga un buffer ideal (experimento 2) donde no se descartan paquetes, se observa una disminución lógica del throughput ocasionada por la pérdida de paquetes. Los paquetes perdidos son paquetes descartados por la parte transmisora del enlace "downlink" del sistema DECT, la cual no dispone de espacio suficiente en el buffer para almacenarlos. El servidor gracias al algoritmo de TCP reno implementado consigue detectar la pérdida de paquetes en el receptor y regula el tamaño de su ventana para evitar la congestión con lo que disminuye el throughput. Al limitar el buffer, disminuye también el tiempo de retardo extremo a extremo de los paquetes puesto que el tiempo máximo que va a tener que estar un paquete en el buffer de transmisión del sistema DECT antes de ser transmitido por el enlace downlink va a ir en función del número máximo de SDUs que puedan almacenarse en dicho buffer. Las

conclusiones obtenidas comparando los tipos de servicio son las mismas que para el experimento 1.

Si comparamos los resultados de throughput de cada servicio con los obtenidos para el experimento 5 vemos como para el experimento 6 se obtienen peores resultados a pesar de emplear un modelo de portadoras con mayor ancho de banda (modelo 2). Este fenómeno está relacionado con el tamaño del buffer de la estación base del sistema DECT y el tamaño de los segmentos TCP procedentes de servidor (536 bytes).

6.7.7. Experimento 7

En este experimento se han empleado segmentos TCP con un tamaño de 1024 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 1 suponiendo un buffer real (limitado) para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. El tamaño del buffer será el especificado en el apartado 6.5. Los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.11:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	460,212 Kbps	0,241648 s
7f2msub	415,571 Kbps	0,293629 s
7f4ssub	885,056 Kbps	0,145175 s
7f4msub	762,673 Kbps	0,163397 s
7f8ssub	1,2819 Mbps	0,110952 s
7f8msub	1,0713 Mbps	0,124075 s
8f2ssub	476,290 Kbps	0,244743 s
8f2msub	417,841 Kbps	0,296367 s
8f4ssub	889,780 Kbps	0,14682 s
8f4msub	778,936 Kbps	0,16706 s
8f8ssub	1,3017 Mbps	0,112741 s
8f8msub	1,1048 Mbps	0,126375 s

Tabla 6. 11 resultados del experimento 7

Comparando estos resultados con los obtenidos para el caso de que se suponga un buffer ideal (experimento 3) donde no se descartan paquetes, se observa una disminución lógica del throughput ocasionada por la pérdida de paquetes. Los paquetes perdidos son paquetes descartados por la parte transmisora del enlace "downlink" del sistema DECT, la cual no dispone de espacio suficiente en el buffer

para almacenarlos. El servidor gracias al algoritmo de TCP reno implementado consigue detectar la pérdida de paquetes en el receptor y regula el tamaño de su ventana para evitar la congestión con lo que disminuye el throughput. Al limitar el buffer, disminuye también el tiempo de retardo extremo a extremo de los paquetes puesto que el tiempo máximo que va a tener que estar un paquete en el buffer de transmisión del sistema DECT antes de ser transmitido por el enlace downlink va a ir en función del número máximo de SDUs que puedan almacenarse en dicho buffer. Las conclusiones obtenidas comparando los tipos de servicio son las mismas que para el experimento 1.

Comparando los resultados obtenidos en el experimento 5 con los resultados obtenidos en este experimento estamos comparando los resultados obtenidos para SMSS = 536 con los obtenidos para SMSS = 1024. Se puede ver como para el caso de usar segmentos TCP de tamaño 1024 bytes se obtienen en la mayoría de servicios mejores resultados.

6.7.8. Experimento 8

En este experimento se han empleado segmentos TCP con un tamaño de 1024 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 2 suponiendo un buffer real (limitado) para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. El tamaño del buffer será el especificado en el apartado 6.5. Los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados se muestran en la tabla 6.12:

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	502,960 Kbps	0,236149 s
7f2msub	401,387 Kbps	0,273758 s
7f4ssub	901,134 Kbps	0,141315 s
7f4msub	788,394 Kbps	0,160365 s
7f8ssub	1,36532 Mbps	0,114158 s
7f8msub	1,14647 Mbps	0,124384 s
8f2ssub	503,907 Kbps	0,240477 s
8f2msub	417,842 Kbps	0,277942 s
8f4ssub	935,178 Kbps	0,144977 s
8f4msub	804,094 Kbps	0,165518 s
8f8ssub	1,13802 Mbps	0,114259 s
8f8msub	1,1202 Mbps	0,124741 s

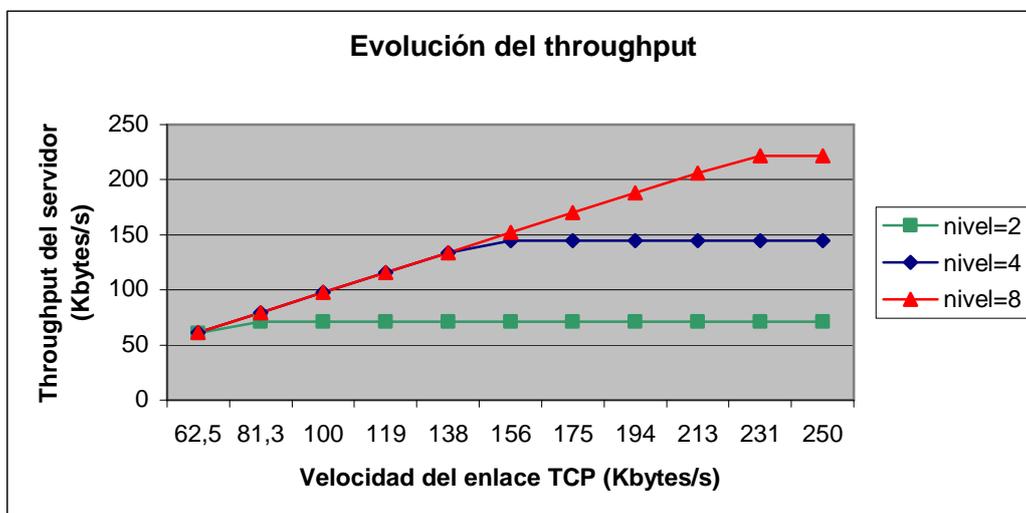
Tabla 6. 12 Resultados del experimento 8

Comparando estos resultados con los obtenidos para el caso de que se suponga un buffer ideal (experimento 4) donde no se descartan paquetes, se observa una disminución lógica del throughput ocasionada por la pérdida de paquetes. Los paquetes perdidos son paquetes descartados por la parte transmisora del enlace “downlink” del sistema DECT, la cual no dispone de espacio suficiente en el buffer para almacenarlos. El servidor gracias al algoritmo de TCP reno implementado consigue detectar la pérdida de paquetes en el receptor y regula el tamaño de su ventana para evitar la congestión con lo que disminuye el throughput. Al limitar el buffer, disminuye también el tiempo de retardo extremo a extremo de los paquetes puesto que el tiempo máximo que va a tener que estar un paquete en el buffer de transmisión del sistema DECT antes de ser transmitido por el enlace downlink va a ir en función del número máximo de SDUs que puedan almacenarse en dicho. Las conclusiones obtenidas comparando los tipos de servicio son las mismas que para el experimento 1.

Si comparamos los resultados de este experimento con los de los experimento 6 vemos que para un mismo modelo de configuración de las portadoras (modelo 2) al usar segmentos TCP de 1024 bytes obtenemos mejores resultados que al usar segmentos TCP de 536 bytes.

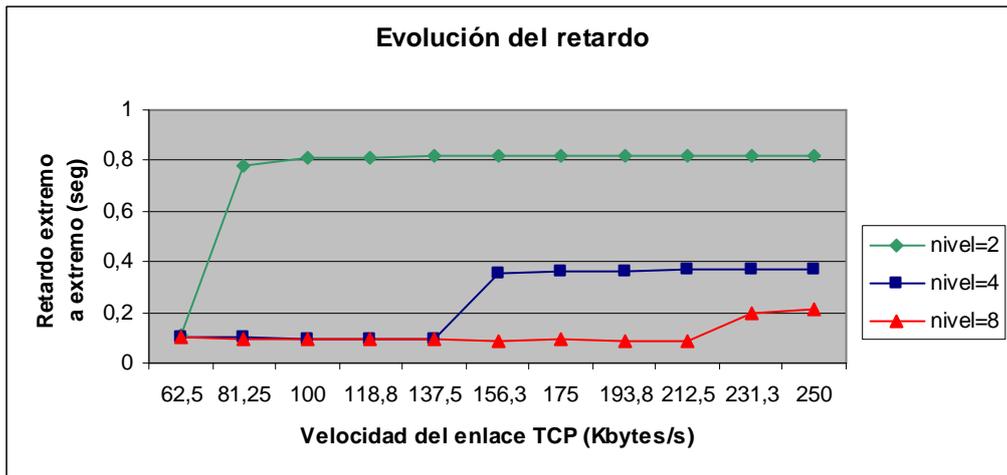
6.7.9. Experimento 9

En este experimento va a variar la velocidad del enlace TCP para poder validar la evolución del tiempo de retardo extremo a extremo y del throughput de nuestro simulador. Se van a comparar los resultados variando el nivel de modulación para el servicio de corrección de errores con un sólo CRC. Las gráficas corresponden al modelo 2 con una SMSS = 1024.



Gráfica 6. 1 Evolución del throughput

El throughput para cada nivel de modulación tiende a seguir un crecimiento lineal a medida que aumenta la velocidad del enlace TCP, hasta que se estabiliza en torno a un valor que se considera como el máximo valor de throughput posible dentro de ese tipo de servicio para el sistema DECT. Gracias a esta gráfica, se verifica la adaptación de la velocidad de la fuente TCP del servidor a la velocidad proporcionada por el sistema DECT al variar la velocidad del enlace TCP en función del nivel de modulación empleado.

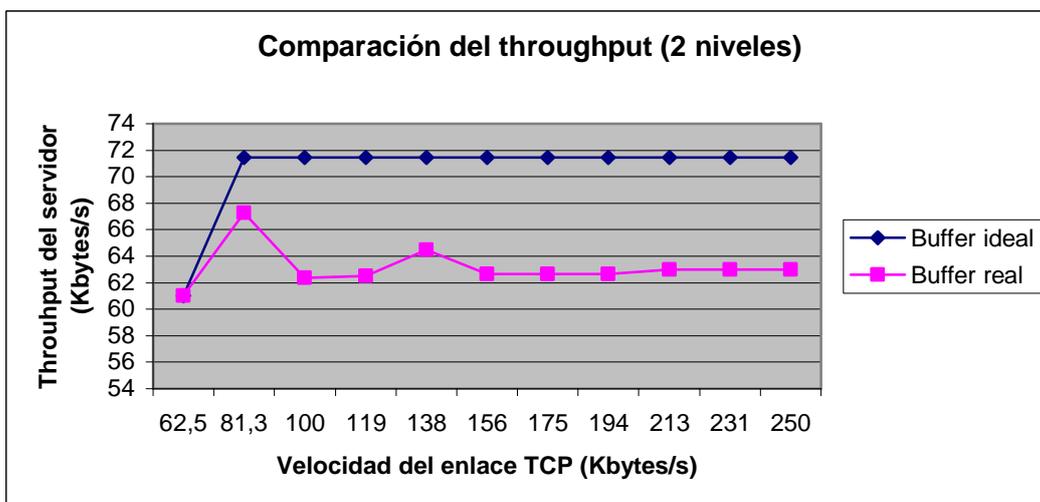


Gráfica 6. 2 Evolución del retardo extremo a extremo

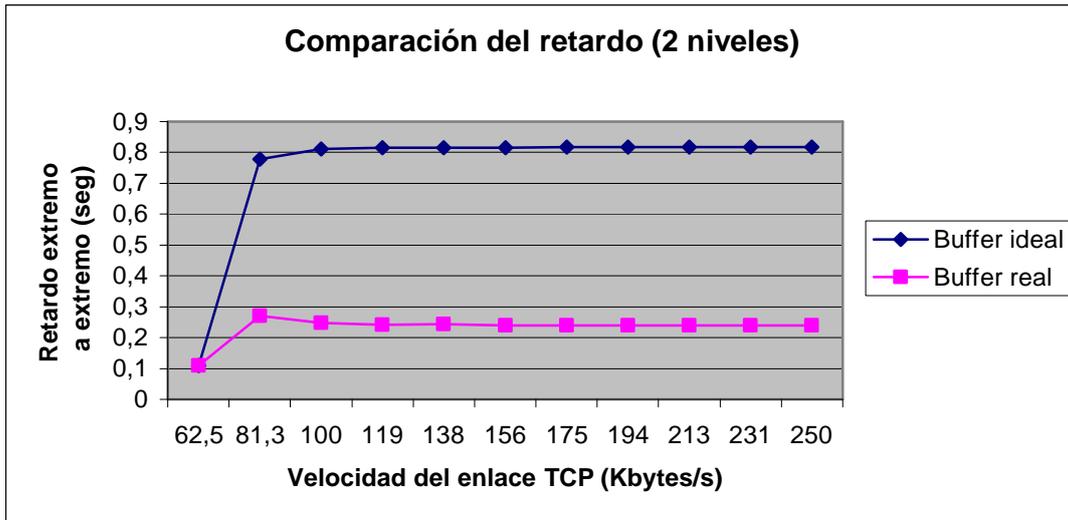
En la se puede observar la evolución del tiempo de retardo extremo a extremo expresado en segundos para cada uno de los niveles de modulación. Para cada uno de dichos niveles se puede ver como a medida que aumenta la velocidad del enlace TCP el retardo extremo a extremo crece de manera lineal hasta estabilizarse en torno a un valor más o menos constante. El valor de la velocidad del enlace para el que se produce esta estabilización coincide para cada nivel de modulación con el valor de máxima velocidad ofrecida por el servicio del sistema DECT tal y como se puede apreciar en la . Así pues en el momento en que la velocidad del enlace TCP supera la máxima velocidad ofrecida por el enlace del sistema DECT, aumenta el número de SDUs en el buffer del transmisor DECT, con lo que cada SDU tarda más en ser transmitida al otro extremo.

6.7.10. Experimento 10

En este experimento se compararán los valores de throughput y retardo extremo a extremo obtenidos para el caso de variar la velocidad del enlace TCP para cada nivel de modulación. En la gráfica 6.3 y la gráfica 6.4 se compara la diferencia entre emplear un buffer ideal y emplear un buffer real para el caso de usar dos niveles de modulación:

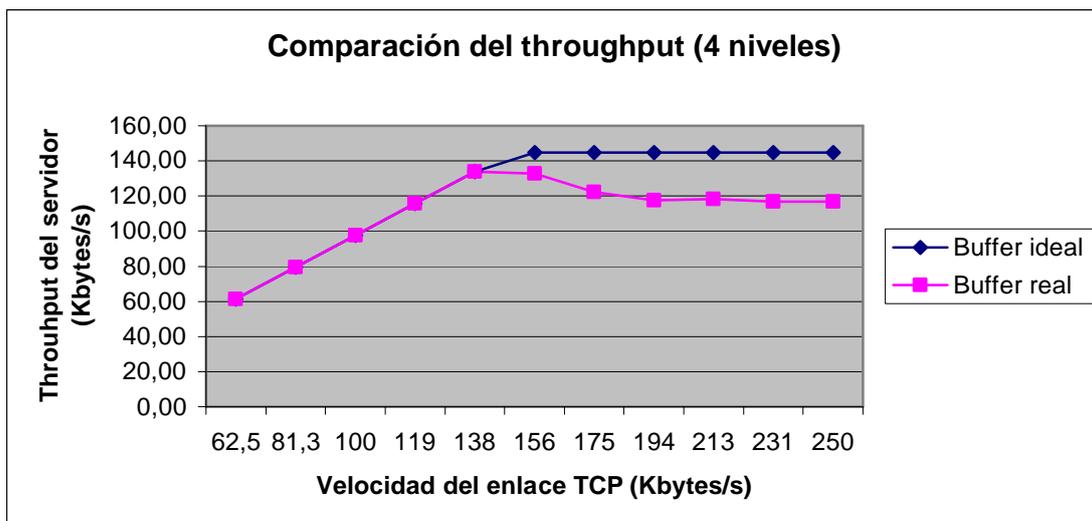


Gráfica 6. 3 Comparativa del throughput (2 niveles de modulación)

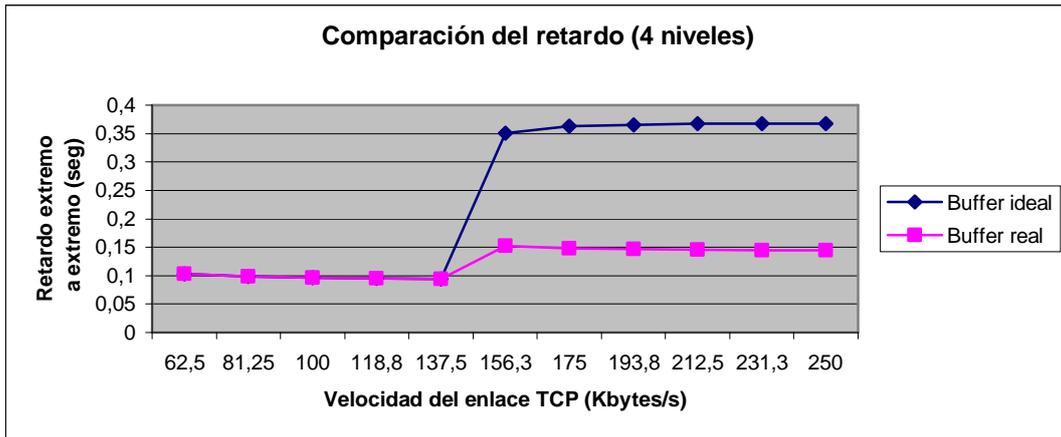


Gráfica 6. 4 Comparativa del retardo (2 niveles de modulación)

Se observa como para el caso de un buffer real donde se descartan paquetes, el throughput máximo que puede llegar a proporcionar el sistema DECT es inferior al proporcionado para el caso de considerar un buffer ideal. En cuanto al tiempo de retardo de los paquetes, vemos como experimenta una mejora tal y como se ha comentado anteriormente. Para el caso de la gráfica 6.5 y la gráfica 6.6 (4 niveles de modulación) se puede apreciar como la evolución de los valores de throughput y retardo respectivamente siguen para un buffer limitado (real) la misma tendencia que para el caso de un buffer ideal. Justo para el momento en el que el sistema DECT alcanza la máxima velocidad posible, se produce una diferenciación entre ambas curvas. La velocidad del enlace TCP es mayor que la velocidad que puede alcanzar el sistema DECT y por este motivo comienza la pérdida de paquetes en el buffer con la consiguiente variación del throughput del servidor gracias al algoritmo TCP reno implementado. Esta situación también ocurría para el caso de emplear dos niveles de modulación pero se daba mucho antes.

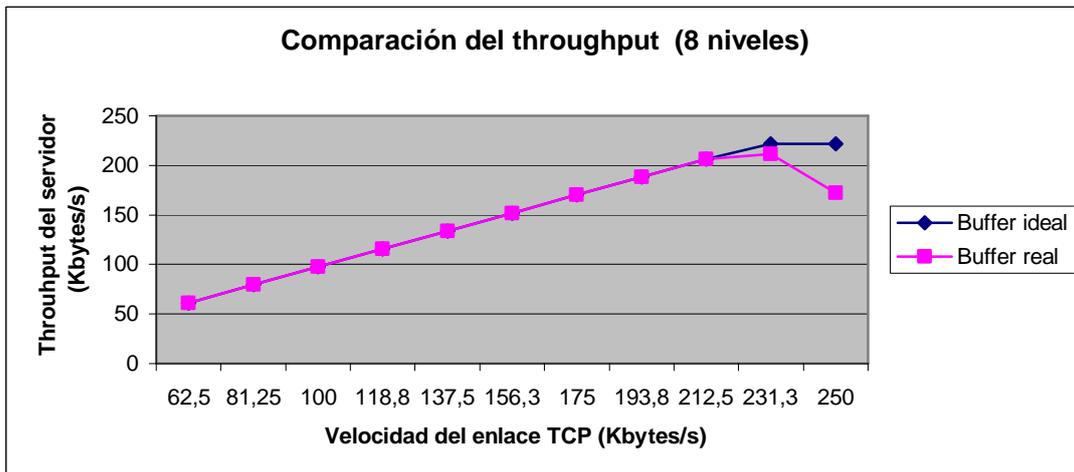


Gráfica 6. 5 Comparativa del throughput (4 niveles de modulación)

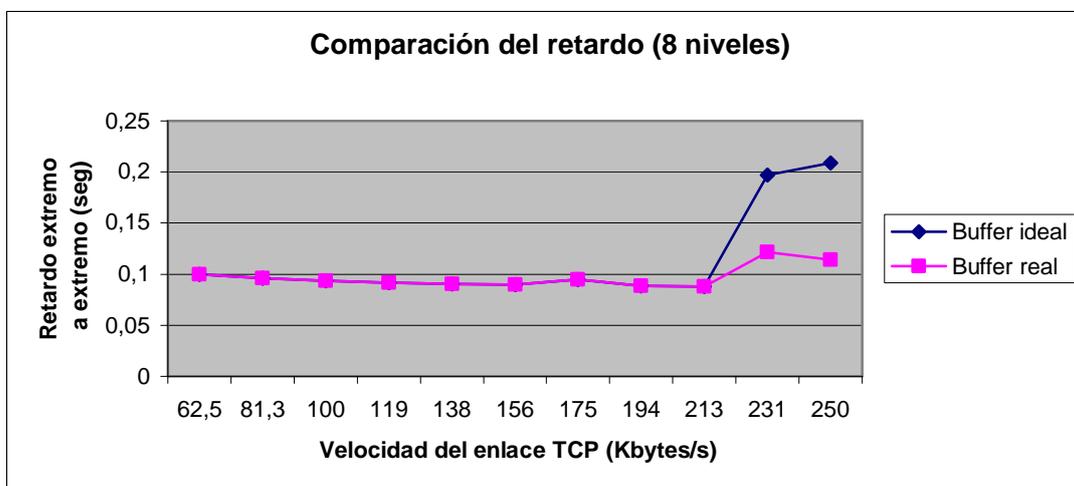


Gráfica 6. 6 Comparativa del retardo (4 niveles de modulación)

Para el caso de la gráfica 6.7 y la gráfica 6.8 (8 niveles de modulación) se puede observar el mismo comportamiento que para las gráficas de los niveles de modulación anteriores:



Gráfica 6. 7 Comparativa del throughput (8 niveles de modulación)



Gráfica 6. 8 Comparativa del retardo (8 niveles de modulación)

6.8. Experimentos para un canal con errores

El objetivo de estos experimentos es evaluar el “throughput” y el retardo extremo a extremo medio que alcanza la fuente TCP reno para cada servicio y observar el comportamiento del simulador en un canal con errores. En la tabla 6.13 se muestran los experimentos que se van a realizar. Para los experimentos 11 y 12 se van a simular los servicios descritos en el apartado 6.2 empleando un SMSS = 1024, ya que ha sido para el que se han obtenido mejores resultados en los experimentos para un canal sin errores. Como el modelo del canal emplea un cálculo de probabilidades para simular los errores, para poder extraer unos resultados que no estén influenciados por un comportamiento particular del canal es necesario usar semillas diferentes en el fichero de configuración de cada simulación consiguiendo así variar el comportamiento del módulo “Channel”. En nuestro caso se han simulado todos los servicios empleando 4 semillas diferentes. Como se han empleado semillas distintas para un mismo servicio se obtienen resultados que varían, por lo que se ha calculado el intervalo de confianza para cada valor estimado haciendo uso de la distribución t-student con 3 grados de libertad y considerando un nivel de confianza del 90%.

Para los experimentos 13, 14 se va a emplear un servicio de corrección de errores con 8 niveles de modulación y protección de un solo campo (8f8ssub). Para el experimento 13 se van a probar el procedimiento implementado para la expiración de los paquetes IP procedentes del servidor en el buffer de la estación base del sistema DECT. Para el experimento 14 se va a aprobar el procedimiento implementado para el intercambio de conexiones entre los enlaces. Para el experimento 15 se va a modificar el tipo de servicio empleado para cada enlace.

Experimento	Tamaño de los segmentos TCP (SMSS)	Modelo	Buffer Limitado	Expiración de SDUs	Modificación de la conexión
11	1024 bytes	1	SI	NO	NO
12	1024 bytes	2	SI	NO	NO
13	1024 bytes	2	SI	SI	NO
14	1024 bytes	2	SI	NO	SI
15	1024 bytes	2	SI	NO	NO

Tabla 6. 13 Experimentos para un canal con errores

En los apartados siguientes se mostrarán los resultados obtenidos para los experimentos.

6.8.1. Experimento 11

En este experimento se han empleado segmentos TCP con un tamaño de 1024 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 1 suponiendo un buffer real (limitado) para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. El tamaño del buffer será el especificado en el apartado 6.5. Los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados de la han sido calculados considerando un nivel de confianza del 90% se muestran en la tabla 6.14.

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	314,18525 ± 1,16 Kbps	0,311396 ± 0,000603 s
7f2msub	258,85725 ± 2,05 Kbps	0,367235 ± 0,001303 s
7f4ssub	327,83813 ± 16,54 Kbps	0,161949 ± 0,000595 s
7f4msub	401,00750 ± 7,57 Kbps	0,183706 ± 0,000340 s
7f8ssub	411,646 ± 16,26 Kbps	0,140308 ± 0,000922 s
7f8msub	480,02675 ± 8,13 Kbps	0,141799 ± 0,000136 s
8f2ssub	417,132 ± 3,11 Kbps	0,274792 ± 0,001877 s
8f2msub	331,11425 ± 0,19 Kbps	0,325649 ± 0,000204 s
8f4ssub	732,97175 ± 2,29 Kbps	0,155099 ± 0,000366 s
8f4msub	624,72825 ± 3,04 Kbps	0,179893 ± 0,000647 s
8f8ssub	966,34025 ± 4,07 Kbps	0,114691 ± 0,000284 s
8f8msub	855,96925 ± 2,65 Kbps	0,130059 ± 0,000366 s

Tabla 6. 14 Resultados del experimento 11

Comparando los resultados obtenidos para cada servicio en este experimento se puede observar como se produce una disminución considerable del throughput para el caso de usar un servicio de detección de errores, por lo que se verifica que este tipo de servicio no resulta adecuado para un canal ruidoso. En cuanto a los servicios de corrección de errores se puede ver que el mecanismo de retransmisión automática a nivel MAC permite mantener unos niveles de throughput más o menos constantes en torno a un valor tal y como revelan los intervalos de confianza. En cuanto a formato de protección, mirando los resultados se puede llegar a una conclusión acerca del compromiso que existe entre proteger la información usando múltiples CRCs o uno sólo. Se puede observar que a pesar de emplear un solo CRC para proteger toda la información, el formato ssub proporciona en todos los casos mejores resultados que el formato msub. A la vista de estos resultados se llega a la conclusión que aunque el nuevo formato ssub introducido en el estándar, aunque no permite realizar

correcciones por comparación de subcampos (como es el caso de msub), resulta en la práctica mejor formato puesto que permite incluir mayor cantidad de información.

6.8.2. Experimento 12

En este experimento se han empleado segmentos TCP con un tamaño de 1024 bytes. Se han realizado la simulación de los servicios descritos en el apartado 6.2 para el modelo 2 suponiendo un buffer real (limitado) para almacenar los paquetes que llegan a la parte fija del sistema DECT procedentes de la red IP. El tamaño del buffer será el especificado en el apartado 6.5. Los paquetes del buffer no expiran y no se permite realizar un intercambio de conexiones entre los enlaces. Los resultados de la tabla 6.15 han sido calculados considerando un nivel de confianza del 90%.

Tipo de servicio	Throughput	Retardo extremo a extremo
7f2ssub	296,40525 ± 2,12 Kbps	0,286510 ± 0,000647 s
7f2msub	268,788 ± 1,45 Kbps	0,337727 ± 0,001827 s
7f4ssub	440,06675 ± 7,49 Kbps	0,159190 ± 0,000218 s
7f4msub	446,20475 ± 2,03 Kbps	0,182543 ± 0,000727 s
7f8ssub	472,60225 ± 9,72 Kbps	0,133267 ± 0,000400 s
7f8msub	373,105875 ± 20,11 Kbps	0,161024 ± 0,002759 s
8f2ssub	397,3655 ± 0,71 Kbps	0,272935 ± 0,000734 s
8f2msub	328,088 ± 0,30 Kbps	0,324774 ± 0,000728 s
8f4ssub	725,97275 ± 2,14 Kbps	0,157222 ± 0,000244 s
8f4msub	626,951 ± 2,85 Kbps	0,178089 ± 0,000237 s
8f8ssub	1011,076 ± 3,11 Kbps	0,116763 ± 0,000169 s
8f8msub	799,9795 ± 4,25 Kbps	0,132480 ± 0,000376 s

Tabla 6. 15 Resultados del experimento 12

Las conclusiones obtenidas comparando los resultados de los servicios en este experimento son las mismas que para el experimento 12.

Si comparamos los resultados de este experimento con los del experimento 12 estaremos comparando los resultados para el caso de usar el modelo 1 de configuración de portadoras con el modelo 2. Se puede ver que excepto para el servicio 8f8ssub el throughput para el modelo 1 es superior que para el modelo 2. En principio el modelo 2 debería ofrecer mejores medidas de throughput puesto que emplea una portadora más que el modelo 1 y además cuenta con dos portadoras para transportar la información de control para la capa DLC y MAC. No obstante después de observar en profundidad los ficheros de registro (.log) generados por los módulos que componen el nivel MAC de nuestro simulador se ha encontrado el motivo de esta diferencia de velocidades. En primer lugar las portadoras “n” del modelo 2 llevan información de control para las portadoras de la TDMA anterior tal y como se ha explicado en detalle en el apartado 4.2.2.1 del capítulo 4. La portadora “n” del modelo lleva información de control para las portadoras que se encuentran en la misma TDMA que la portadora “n”.

Después de revisar y comparar los ficheros de registro de los módulos para ambos modelos se han detectado la situación que se expone a continuación. Para el caso del modelo 1 la portadora “n” que lleva incluida la información de control para las portadoras de la segunda mitad de la TDMA llega correcta y antes de que comience la primera mitad de la nueva trama TDMA, se hace la petición a DLC para transmitir en la primera mitad de esa trama. En el modelo 2 esta misma información de control es transportada en la portadora “n” que se encuentra al comienzo de la primera mitad de la nueva TDMA. Para ambos modelos la nueva TDMA encuentra el canal en estado malo. En el modelo 1 como se ha obtenido toda la información de control en la portadora “n” de la anterior TDMA, se hará una nueva petición de datos a DLC antes de que comience la segunda mitad de la nueva TDMA. Para el modelo 2 la portadora “n” que contenía información de control para las portadoras de la anterior TDMA se pierde (por un error en la cabecera o por error en todos los subcampos) por lo que para este modelo no se hará petición de nuevos datos a DLC si no que se preparará una retransmisión automática para esos paquetes en la segunda mitad de la TDMA (paquetes que habían sido ya confirmados para el modelo 1). Por este motivo, el modelo 1 se adelanta al modelo 2 a la hora de transmitir nuevos paquetes. Además de esto suponiendo que para el caso del modelo 2 la portadora de la segunda mitad de la TDMA que contiene replicada la información de control consiguiese entregar esa información, al ser entregada después del comienzo de la segunda mitad, aunque se pueda acceder a esta información de control, las peticiones que se hagan a DLC, serán para la segunda mitad de la siguiente TDMA, por lo que en el mejor caso lo que se hará es cancelar las retransmisiones automáticas innecesarias en esa segunda mitad de la actual TDMA, en el caso de que sea necesario por que así lo indique la información de control.

La idea de emplear una configuración con dos portadoras “n” para proporcionar un mayor nivel de protección de la información fue extraída de un artículo [28]. En dicho artículo no se hace ningún tipo de recomendación sobre cual debe ser la posición de las portadoras “n”, siendo su ubicación arbitraria. Tampoco en el estándar aparece nada que haga mención a como se deben situar dichas portadoras. Después de los resultados obtenidos para el modelo 2 se puede afirmar que la posición de las portadoras “n” dentro de la TDMA además de determinar para que TDMA transportan la información de control (puede ser para la actual TDMA o para la anterior), determinan también en cierta medida la respuesta del protocolo en un entorno ruidoso.

6.8.3. Experimento 13

En este experimento se va a limitar el tiempo de vida de los paquetes IP con datos del servidor en el buffer de la estación base del sistema DECT según el procedimiento descrito en el apartado 6.6. El valor de este tiempo de vida se expresa en TDMA y su determinación queda a elección del administrador del sistema DECT. El modelo a usar será el modelo 2 y el tamaño de los segmentos TCP será de 1024 bytes. Como ejemplo para validar comprobar el impacto de limitar el tiempo de vida de las SDUs se va a simular el servicio 8f8ssub, limitando el tiempo de vida a 20 TDMA.

En la tabla 6.16 se muestra una comparación entre ambas opciones y se ve claramente como disminuye el throughput como consecuencia de limitar el tiempo de vida de las SDUs. A la vista de los resultados se puede afirmar que limitar el tiempo de vida de las SDUs en el buffer no resulta muy adecuado para el tipo de tráfico TCP que tiene que soportar el servicio 8f8ssub en nuestro simulador, puesto que provoca numerosas pérdidas de paquetes en el buffer que tienen como consecuencia una disminución del throughput de la fuente del servidor.

Tipo de servicio	Throughput	Retardo extremo a extremo
8f8ssub SDUExpire=false	1011,076 ± 3,11 Kbps	0,116763 ± 0,000169 s
8f8ssub SDUExpire=true	971,3535 ± 4,91 Kbps	0,117022 ± 0,000191 s

Tabla 6. 16 Resultados del experimento 13

6.8.4. Experimento 14

En este experimento se va a permitir el intercambio de conexiones entre los enlaces tal y como se ha descrito en el apartado 4.2.2.2 del capítulo 4. Esta característica resulta interesante para el caso de que el usuario del terminal móvil desee transmitir gran cantidad de información al servidor en un momento determinado. En nuestro simulador se ha considerado que el terminal portátil genera paquetes que contienen únicamente una cabecera con ACKs para el servidor. En el siguiente experimento se va a simular para el modelo 2 el servicio 8f8ssub suponiendo segmentos TCP de 1024 bytes. Se va a considerar la siguiente configuración para los thresholds:

Parte Fija → Threshold = 2
 Parte Portátil → Threshold = 2

Tipo de servicio	Throughput	Retardo extremo a extremo
8f8ssub, MdfCnx=false	1011,076 ± 3,11 Kbps	0,116763 ± 0,000169 s
8f8ssub, MdfCnx=true	734,2963 ± 8,56 Kbps	0,114746 ± 0,000161 s

Tabla 6. 17 Resultados del experimento 14

Para esta situación la parte portátil solicitará la conexión de mayor capacidad cuando el número de SDUs en el buffer sea mayor o igual que 2. La parte fija cederá la conexión de menor capacidad cuando el número de SDUs en el buffer sea menor o igual a 2. Cuando el número de SDUs en el buffer sea mayor o igual a 2, si la parte fija desea recuperar la conexión de mayor capacidad, deberá esperar que la parte portátil se la conceda. Para que la parte portátil conceda la conexión de mayor capacidad, el número de SDUs en el buffer tiene que ser menor o igual a 2.

A la vista de los resultados de la tabla 6.17 se puede afirmar que la configuración probada para el intercambio de las conexiones no favorece la mejora de los parámetros de throughput y de retardo. Se han hecho simulaciones aumentando el valor de threshold en la FP y a medida que se aumentaba los parámetros han empeorado. La razón del empeoramiento de los parámetros es que la FP cede más fácilmente la conexión de mayor capacidad, pero le cuesta más recuperarla a medida que aumentamos su threshold, por lo que los valores de throughput y de retardo extremo a extremo se ven gravemente afectados. Por este motivo se aconseja que esta solución sea adoptada sólo cuando sea completamente necesaria para el caso de que el tráfico TCP generado por la aplicación en la PP contenga además de los ACKs de la cabecera información en el campo de datos para el servidor.

6.8.5. Experimento 15

En este experimento se va a variar el tipo de servicio a emplear por cada uno de los enlaces. Como ejemplo de una posible variación supóngase un caso en que el enlace “uplink” usa el servicio “8f8ssub” y el enlace “downlink” emplea el servicio “8f8msub”. El modelo usado será el modelo 2 y el tamaño de los segmentos TCP será de 1024 bytes.

Tipo de servicio	Throughput	Retardo extremo a extremo
8f8msub, 8f8msub	799,9795 ± 4,25 Kbps	0,132480 ± 0,000376 s
8f8msub, 8f8ssub	808681,5 ± 7,57 Kbps	0,12595 ± 0,000220 s

Tabla 6. 18 Resultados del experimento 15

En la tabla 6.18 donde se comparan los resultados con los obtenidos para el caso de que los dos servicios sean 8f8msub. Se puede observar que existe una mejora del valor medio de ambos parámetros.

Capítulo 7

Conclusiones y Líneas Futuras

7.1 Introducción

El estándar base DECT proporciona, desde sus últimas versiones una gran número de reglas y procedimientos para la integración de los servicios de datos de banda ancha con los servicios de voz. Los procedimientos del estándar base necesarios para cada tipo particular de servicio aparecen recogidos en los perfiles para los servicios de datos (DSPs) tal y como se ha explicado en el apartado 2.4 del capítulo 2. En nuestro proyecto partiendo del estándar base y de los perfiles se ha realizado un trabajo de síntesis para el diseño de un escenario de referencia en el cual los usuarios de los servicios de voz conviven con los usuarios de los servicios de banda ancha. A partir de este escenario se ha construido un simulador que implementa todos los procedimientos y características necesarios para la integración de los servicios de voz y datos. En los apartados 7.2 más adelante y 7.3 más adelante se incluyen respectivamente las conclusiones y líneas futuras. En los apartados 7.4 más adelante y 7.5 más adelante se incluye una refelexión sobre el futuro de DECT y una descripción de otros proyectos relacionados con DECT.

7.2 Conclusiones

A partir del análisis de los documentos del estándar base y de los perfiles DECT se ha propuesto una solución para la integración de los servicios de voz y datos. Se ha diseñado un escenario donde se implementa la solución de integración propuesta. Todos los nuevos procedimientos y características de las últimas versiones del estándar DECT que son necesarios para la implementación del escenario han sido resumidos en el capítulo 2. Entre las características se encuentra el uso de nuevos niveles de modulación y formatos de paquetes a nivel MAC, junto con nuevos procedimientos a nivel DLC.

Se ha desarrollado un simulador en C++ mediante el empleo de las librerías de OMNET++ que implementa el escenario de referencia diseñado. Las ventajas de haber empleado esta técnica de implementación son:

- Orientación a objetos: permite el desarrollo y validación por separado de cada módulo aprovechando la arquitectura de capas.
- Librerías de OMNET++: permiten la reutilización de módulos y favorece la escalabilidad del simulador.

Las características y funcionalidades ofrecidas por el simulador implementado son:

- Capacidad para simular una amplia variedad de servicios DECT.
- Capacidad para generar ficheros de registro para cada módulo.
- Capacidad para obtener gráficas y resultados.

Las conclusiones sobre los experimentos realizados y los resultados obtenidos se pueden resumir en:

- Cada servicio del sistema DECT seleccionado en el simulador se comporta de la manera esperada proporcionando unos valores de throughput y de retardo extremo a extremo en concordancia con el tipo de servicio.
- El tráfico ofrecido por el servidor y el retardo extremo a extremo de los paquetes se ajusta en todo momento al nivel de modulación seleccionado para la conexión del sistema DECT.
- Para un canal con errores el servicio de corrección de errores a nivel MAC funciona mejor que el servicio de detección de errores.

Las utilidades del proyecto realizado son:

- Desde el punto de vista de la investigación: evaluación de prestaciones, desarrollo de algoritmos y protocolos, etc.
- Desde el punto de vista de la aplicación docente: conocimiento teórico del estándar DECT, conocimiento práctico del funcionamiento real de los procedimientos del protocolo en un sistema DECT, etc.

7.3 Líneas futuras del proyecto

Se proponen como líneas futuras de nuestro proyecto la incorporación de nuevas características al simulador tales como:

1. El nuevo formato que usa el principio conocido como principio del turbo código permite al igual que el formato de protección de un solo campo (ssub) transportar una gran cantidad de información y además incorpora la facilidad de corregir los paquetes en el receptor. Se propone como una línea futura el incorporar este formato en nuestro simulador y comprobar en la práctica las mejoras que ofrece frente a ssub.
2. La posibilidad de poder realizar conexiones simétricas para poder simular el funcionamiento de las fuentes de tráfico TCP/IP con aplicaciones para las cuales resulta más adecuada este tipo de configuración simétrica.
3. La incorporación de un modelo de pérdida de paquetes dentro de la red IP que une el servidor con el sistema DECT y la toma de resultados, puede ser una buena línea a seguir en un futuro.
4. Probar una nueva configuración que use dos portadoras “n” en los dos últimos slots de la segunda mitad de la TDMA después de todas las portadoras “k”.

7.4 Futuro de DECT

La tecnología DECT es una tecnología madura y consolidada que cuenta en la actualidad con un total aproximado de 120 millones de sistemas en todo el mundo. En la figura 7.1 se muestra la comparación entre el número de terminales móviles de DECT vendidos para su uso empresarial en el mundo frente al número de terminales Wi-Fi.

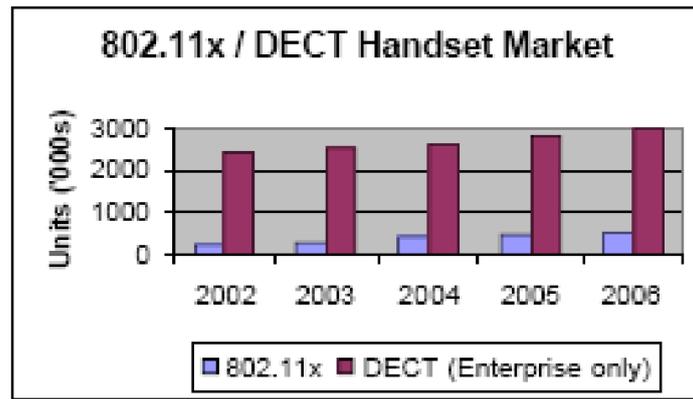


Figura 7. 1 Comparación de las ventas entre los terminales DECT y los terminales Wi-Fi (datos extraídos de [34]).

El número de terminales DECT vendidos en el año 2004 es 7 veces superior al número de terminales Wi-Fi. Para el año 2006 se prevé que el número de terminales DECT será 6 veces superior al número de terminales Wi-Fi. Las ventajas de DECT al tener una mayor cuota de mercado frente a Wi-Fi son:

- Hay una demanda continua de productos DECT, lo que favorece la mejora continua de los productos.
- Hay un gran número de vendedores de sistemas DECT en Europa y Asia que desarrollan productos que cumplen con el estándar DECT y por tanto son compatibles entre sí.
- Los precios de los terminales DECT son menores que los de los terminales Wi-Fi. Además se espera que el precio de los terminales Wi-Fi baje en los próximos años

7.5 Otros proyectos relacionados con DECT

Por último y si el lector está interesado en conocer otros estudios que se están realizando en la actualidad sobre DECT, comentar la existencia de un proyecto europeo conocido como WinDECT [29], el cual pretende demostrar de manera práctica los beneficios de DECT frente a la tecnología de voz sobre IP. Para más información sobre el proyecto WinDECT se puede consultar su página web oficial [31]. Otras páginas de Internet en donde se puede encontrar información general sobre la tecnología DECT son "DECT web" [32] y el "DECT Forum" [33].

Referencias:

- [1] Página principal de la web oficial de la ETSI
<http://www.etsi.org/>
- [2] Página de descarga de estándares de libre acceso de la ETSI
http://www.etsi.org/services_products/freestandard/home.htm
- [3] Documentos antiguos del estándar base DECT
ETSI EN 300 175 (Partes 1 – 5) (1996)
- [4] Documento del estándar base con la descripción general del protocolo DECT.
ETSI EN 300 175-1: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 1: Overview".
- [5] Documento del estándar base con la descripción de la capa PHL de DECT.
ETSI EN 300 175-2 V1.7.1: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 2: Physical Layer (PHL)" (Julio 2003).
- [6] Documento del estándar base con la descripción de la capa MAC de DECT.
ETSI EN 300 175-3 V1.8.0: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 3: Data Link Control (DLC) layer" (Julio 2004).
- [7] Documento del estándar base con la descripción de la capa DLC de DECT.
ETSI EN 300 175-4 V1.8.0: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 5: Data Link Control (DLC) layer" (Julio 2004).
- [8] Documento del estándar base con la descripción de la capa NWK de DECT.
ETSI EN 300 175-6 v1.8.0: "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 5: Network (NWK) layer" (Julio 2004).
- [9] Perfil de acceso genérico (GAP).
ETSI EN 300 444 V1.4.1: "Digital Enhanced Cordless Telecommunications (DECT); Generic AccessProfile (GAP)" (Septiembre 2001).
- [10] ISO/IEC 8802-3: "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications".
- [11] ISO/IEC 8802-5: "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 5: Token ring access method and physical layer specifications".
- [12] IETF RFC 791 (1981): "Internet Protocol", J. Postel.
- [13] IETF RFC 1661 (1994): "The Point-to-Point Protocol (PPP)", W. Simpson.
- [14] Perfil de acceso multimedia (DMAP) y el perfil de acceso específico (ASAP).
ETSI EN 301 650 v1.2.1: "Digital Enhanced Cordless Telecommunications (DECT); DECT Multimedia Access Profile(DMAP); Application Specific Access Profile (ASAP)" (Abril 2002).

- [15] Perfil del servicio de radio paquetes (DPRS).
ETSI EN 301 649 V1.4.0: "Digital Enhanced Cordless Telecommunications (DECT); DECT Packet Radio Services (DPRS)" (Agosto 2004).
- [16] Servicio de mensajes cortos usando DECT.
ETSI EN 300 757: "Digital Enhanced Cordless Telecommunications (DECT); Low Rate Messaging Service (LRMS) including Short Messaging Service (SMS)".
- [17] Documento sobre IDBS.
ETSI EN 301 238: "Digital Enhanced Cordless Telecommunications (DECT); Data Services Profile (DSP); Isochronous data bearer services with roaming mobility (service type D, mobility class 2)".
- [18] Informe Técnico sobre los servicios de datos sobre DECT.
ETSI TR 102 185 v1.2.1: "Digital Enhanced Cordless Telecommunications (DECT); Data services overview" (Mayo 2001).
- [19] Documento con especificaciones técnicas sobre el acceso de DECT a redes IP.
ETSI TS 102 265 v1.1.1: "Digital Enhanced Cordless Telecommunications (DECT); DECT access to IP Networks" (Octubre 2003).
- [20] Informe Técnico sobre el acceso de DECT a redes IP.
ETSI TR 102 010 v1.1.1: "Digital Enhanced Cordless Telecommunications (DECT); DECT access to IP Networks" (Octubre 2003).
- [21] "Redes de Ordenadores". 2ª ed.
Ed. Prentice-Hall Hispanoamericana, S.A.
Aut. Andrew S. Tanenbaum
- [22] Personal Wireless Communication with DECT and PWT.
Ed. Artech House Publishers
Aut. Jhon Philips, Gerard Mac Namee
- [23] Página web oficial de OMNET++.
<http://www.omnetpp.org/>
- [24] Manual de OMNET ++;
Ed. Manual on-line
Aut. Andrea Vargas
- [25] C++ How To Program 2ª ed.
Ed. Deitel and Deitel
Aut. H.M. Deitel/ P.J. deitel
- [26] Estándar en el que se basan los estándares Wi-Fi
IEEE 802.11
- [27] On the accuracy of a first-order Markov model for data block transmisión on fading.
Ed. Artículo IEEE ICUPC'95, pp 211-215. Nov 1995
Aut. M.Zozzi y otros
- [28] TCP/IP Data Transfer over the DECT Air Interface with Multibearer Capability and Support of Asymmetric Flows.
Ed. Kluwer Academic Publishers.
Aut. Andrea Baiocchi.

[29] Wireless LAN with Integration of Professional-Quality DECT Telephony.
Aut. Marjan Spegel y otros

[30] TCP/IP Illustrated, Vol. 1, The Protocols.
Ed. Addison Wesley, 1994
Aut. W.R. Stevens

[31] Página web oficial del proyecto WinDECT.
http://www.windect.ethz.ch/project_summary.html

[32] Página web con información general sobre DECT.
<http://www.dectweb.com>

[33] Página web oficial del foro de DECT.
<http://www.dect.org>

[34] VoWLAN Vs DECT
Aut. Eric Stridbaek y otros.