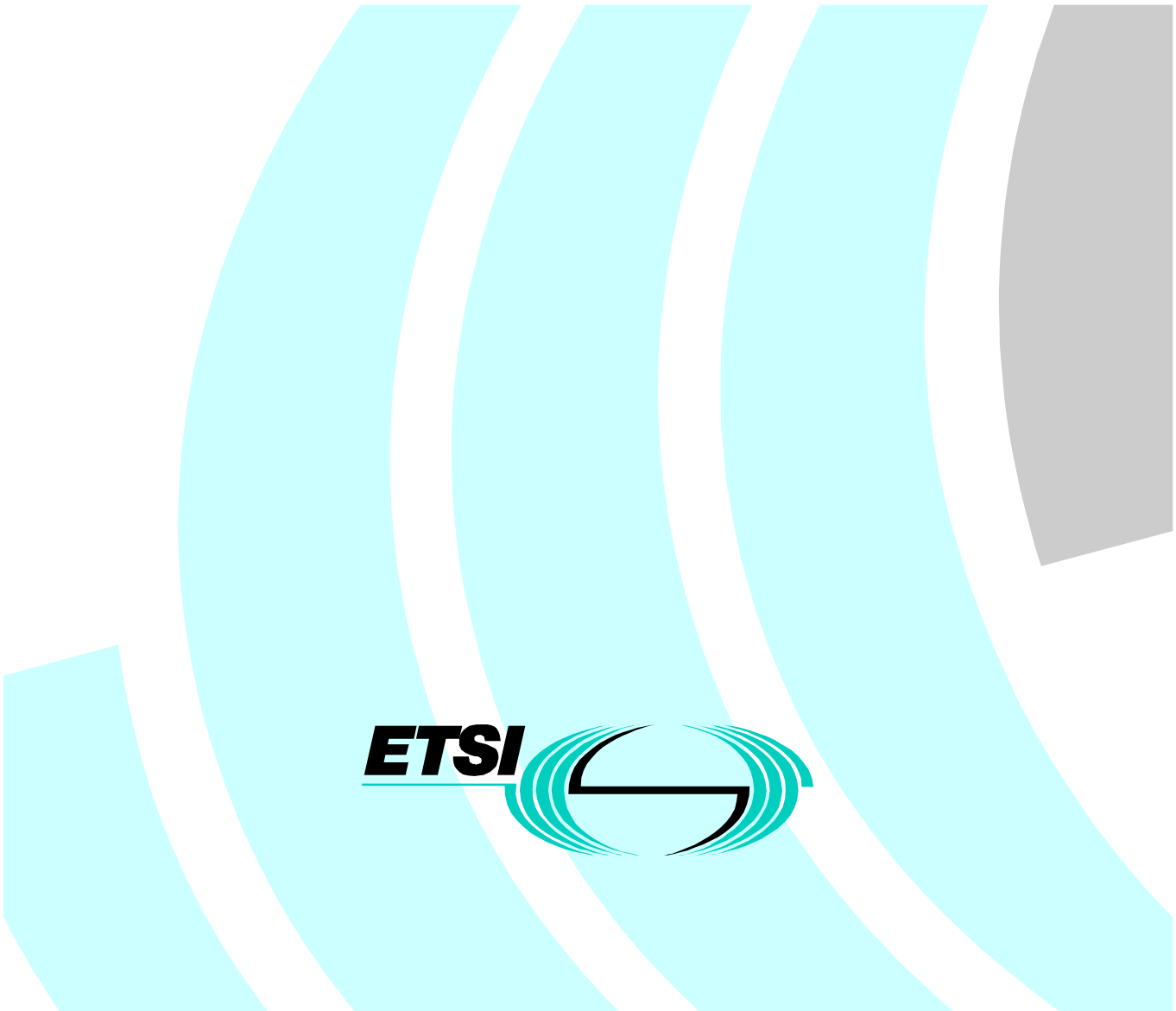
ETSI TS 101 659 V1.2.1 (2000-04)

Technical Specification

Terrestrial Trunked Radio (TETRA); Digital Advanced Wireless Service (DAWS); Medium Access Control (MAC) Service Description



Reference
RTS/TETRA-04047

Keywords
TETRA, PDO, data, radio

ETSI

Postal address
F-06921 Sophia Antipolis Cedex - FRANCE

Office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16
Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Internet
secretariat@etsi.fr
Individual copies of this ETSI deliverable
can be downloaded from
http://www.etsi.org
If you find errors in the present document, send your
comment to: editor@etsi.fr

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/ipr).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI Project Terrestrial Trunked Radio (TETRA).

The present document is one of the three documents specifying the requirements for TETRA Digital Advanced Wireless Service (DAWS):

- TS 101 658: Logical Link Control (LLC) Service Description;
- **TS 101 659: Medium Access Control (MAC) Service Description;**
- TS 101 660: Physical Layer (PHY) Service Description.

An overview of the requirements for DAWS can be found in TR 101 156 [1].

Introduction

The DAWS protocol architecture is provided in [1]. The Medium Access Controller (MAC) provides services to the Logical Link Controller (LLC) and requests services from the Physical layer (PHY). The present document describes the services the MAC shall provide to function within a DAWS network.

The prefix MAC will be used when a requirement applies to both the BS and MS MAC layers. The prefix BS_MAC or MS_MAC will be used when a requirement applies only to the BS or MS MAC layers, respectively.

As shown in figure 1, the LLC accesses MAC services via service access points (SAPs) A and B. MAC_SAP_A is for data transfer service primitives and MAC_SAP_B is for local control and status service primitives.

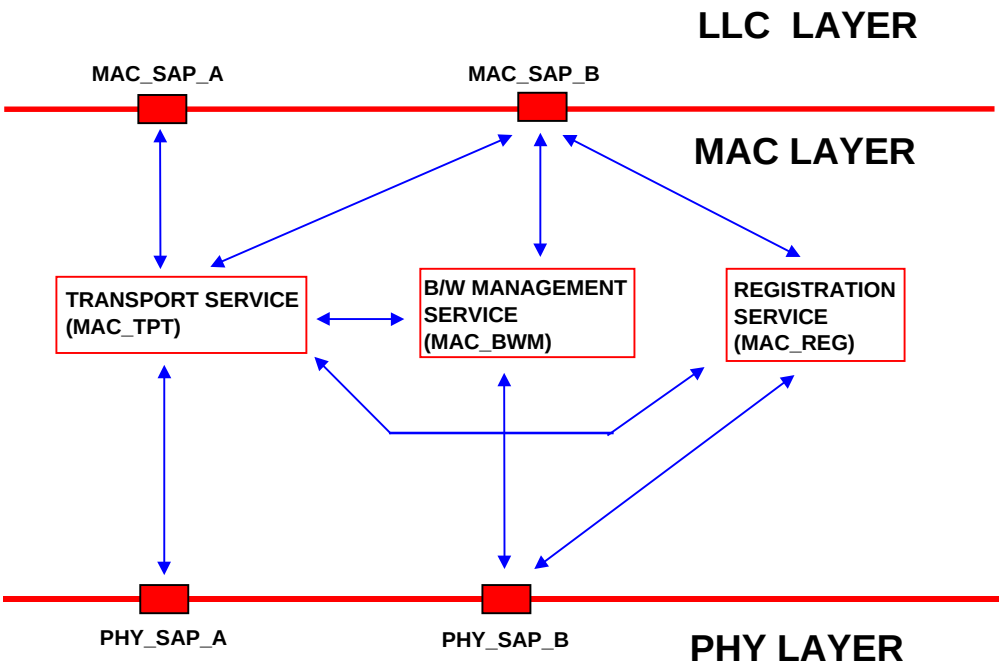


Figure 1: DAWS MAC Architecture

The MAC accesses PHY services via service access points A and B. PHY_SAP_A is for data transfer service primitives and PHY_SAP_B is for control and status service primitives.

The services provided by the MAC can be divided into three major areas: registration, bandwidth management, and transport. Requirements for each of these services are provided in clauses 4, 5, and 6. Service primitives and associated service data units are provided in clause 7.

1 Scope

The present document specifies the service requirements for the Digital Advanced Wireless Service (DAWS) Medium Access Control (MAC) layer. The present document provides a conceptual architecture useful for specifying service requirements but is not intended to imply a particular implementation. The present document contains preliminary MAC protocol requirements which will be moved into the formal MAC protocol specification document (Part 5) when it is drafted.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, subsequent revisions do apply.
- A non-specific reference to an ETS shall also be taken to refer to later versions published as an EN with the same number.

[1] ETSI TR 101 156: "Terrestrial Trunked Radio (TETRA); Technical requirements specification for Digital Advanced Wireless Service (DAWS)".

[2] ETSI TS 101 660: "Terrestrial Trunked Radio (TETRA); Digital Advanced Wireless Service (DAWS); Physical Layer (PHY) service description".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

base station: piece of equipment providing simultaneous, bi-directional network access to mobile stations

block: fixed-length sequence of bytes from a MAC PDU

contention-free: physical layer access method in which there is no possibility that two or more correctly operating mobile stations will transmit simultaneously in a manner which leads to mutually destructive interference between the transmissions

contention-possible: physical layer access method in which there exists the possibility that two or more correctly operating mobile stations will transmit simultaneously in a manner which leads to mutually destructive interference between the transmissions

contention-reduced: contention-possible physical layer access method designed to have reduced possibility of mutually destructive interference between two or more correctly operating mobile stations

downlink: general term meaning "from the base station to the mobile station"

frame: time period consisting of an integral number of slots between base station broadcasts specifying mobile station bandwidth assignments

mobile station: piece of equipment able to create and consume data but only having network access via a base station

protocol data unit: set of parameters and/or data passed from peer to peer by a protocol primitive

protocol instance: two protocol processes which exchange messages in order to transfer data from one protocol process to the other

protocol primitive: request, response, or informative message sent from peer to peer

protocol process: entity created to manage one end of a peer-to-peer protocol. For unidirectional data flows, a protocol process can be further described as either a sender process or a receiver process

serving cell: physical area serviced by a base station

service data unit: set of parameters and/or data passed between adjacent layers by a service primitive

service primitive: request, response, or informative message sent between adjacent layers

slot: minimum time period reserved for transmission by a single mobile station on a single frequency

sub-protocol: portion of a protocol performing a clearly identifiable operation

uplink: general term meaning "from the mobile station to the base station"

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ACK	Acknowledged
ARQ	Automatic Repeat Request
BE	Best-Effort
BS	Base Station
CL	Controlled-Load
DAWS	Digital Advanced Wireless Services
DL	Downlink
DQOS	Data Integrity Quality Of Service
IP	Internet Protocol
LLC	Logical Link Controller
MAC	Medium Access Controller
MAC_BWM	MAC Bandwidth Management Service
MAC_REG	MAC Registration Service
MAC_TPT	MAC Transport Service
MPDU	MAC Protocol Data Unit
MS	Mobile Station
MSH	Mobile Station Handle
MSI	Mobile Station Identifier
MTU	Maximum Transmission Unit
PDU	Protocol Data Unit
PHY	Physical Layer
QOS	Quality Of Service
RSVP	Resource Reservation Protocol
SAP	Service Access Point
SDU	Service Data Unit
TQOS	Timing Quality Of Service
UNACK	Unacknowledged
UL	Uplink

4 Registration Services

The MAC registration service (MAC_REG) is responsible for interacting with the PHY layer to maintain the highest possible signal quality for the current serving cell, as well as performing adjacent cell scans when requested by LLC_REG. This section will be expanded after a DAWS PHY layer is defined.

5 Bandwidth Management Services

The MAC bandwidth management service (MAC_BWM) is responsible for allocating bandwidth over the physical medium for MS in full-power and power-saving modes of operation.

5.1 Base station bandwidth management

BS_MAC_BWM shall allocate bandwidth on a per-flow basis. BS_MAC_BWM shall consider the current state of all flow input queues and QoS contracts, and then shall dynamically allocate a portion of available free bandwidth to each flow. During frame N, BS_MAC_BWM shall prepare and send to the PHY layer a **PHY_configure_sysinfo_request** primitive [2] containing slot assignments for frame N+2. PHY_LNK will send the system information PDU associated with **PHY_configure_sysinfo_request** to all MS in the cell during slot 0 of frame N+1. The MS will then have an entire frame to prepare for activity during frame N + 2.

BS_MAC_BWM shall share bandwidth allocation information with BS_MAC_TPT. BS_MAC_TPT shall issue **PHY_transfer_request** primitives to supply the BS_PHY layer with downlink PDUs.

5.2 Mobile station bandwidth management

MS_MAC_BWM shall convey the current state of flow input queues to BS_MAC_BWM so that BS_MAC_BWM can allocate bandwidth effectively for the flows.

MS_MAC_BWM shall share bandwidth allocation information with MS_MAC_TPT. MS_MAC_TPT shall issue **PHY_transfer_request** primitives to supply the MS_PHY layer with uplink PDUs.

5.3 Power management

MAC-BWM shall support a power conservation strategy which allows the MS to remain in a low power consumption state for a considerable portion of the time. A power conserving MS shall resume normal operation before attempting an MPDU transfer. The QoS delivered to an MPDU from a power conserving MS shall be equivalent to that delivered to an MPDU from an MS operating in full-power mode, except that the downlink and uplink MPDU transfer establishment latency shall be longer.

6 Transport Services

The MAC transport service (MAC_TPT) is responsible for the transfer of MPDUs over the physical medium. This clause discusses the architecture of MAC_TPT, provides requirements for the protocols in the MAC_TPT protocol suite, and describes MAC_TPT error handling.

6.1 Architecture

As shown in figures 2 and 3, MAC_TPT is composed of a suite of six separate protocols:

- UNACK_DL: unacknowledged downlink;
- UNACK_UL: unacknowledged uplink;
- ACK_BE_DL: acknowledged downlink, best-effort QoS;
- ACK_BE_UL: acknowledged uplink, best-effort QoS;
- ACK_RS_DL: acknowledged downlink, reserved QoS;
- ACK_RS_UL: acknowledged uplink, reserved QoS.

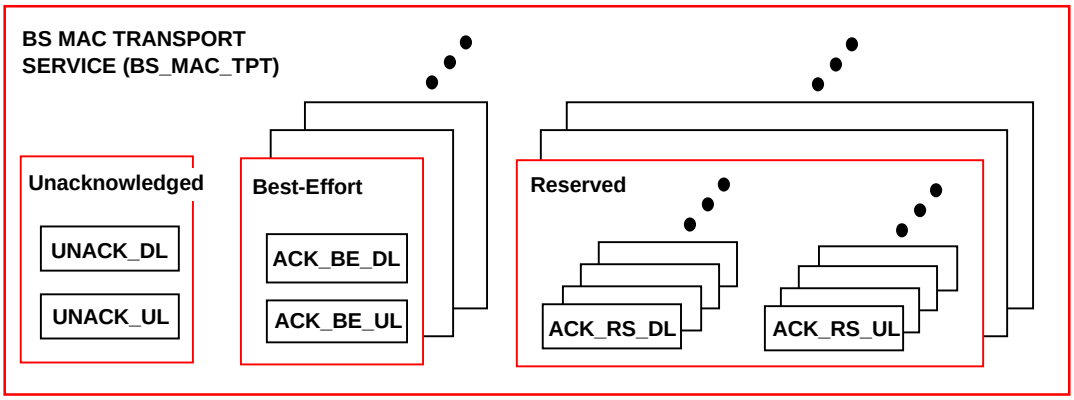


Figure 2: BS_MAC Transport Service Architecture

MAC_TPT shall create and maintain a pair of protocol instances, UNACK_DL and UNACK_UL, upon power-up.

BS_MAC_TPT shall contain one pair of protocol processes, UNACK_BE_DL and UNACK_BE_UL, for each MS registered with the BS. These protocol processes manage the transfer of PDUs with best-effort QoS. These protocol processes shall be created upon MS registration and destroyed upon MS de-registration. MS_MAC_TPT shall contain only one pair of these protocols, as shown in figure 3.

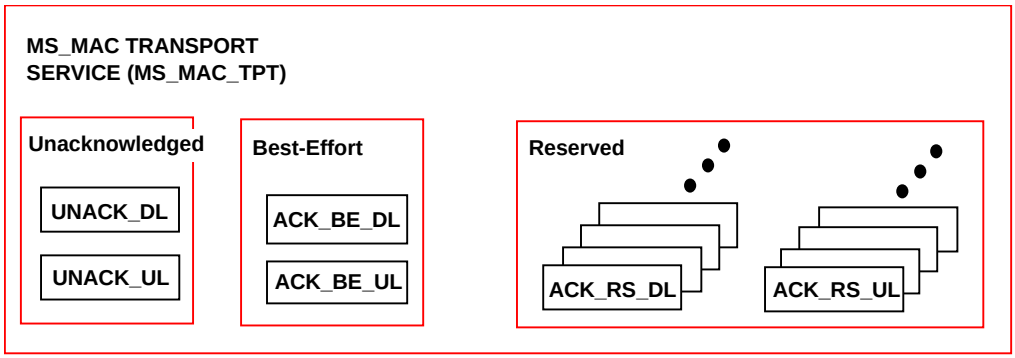


Figure 3: MS_MAC Transport Service Architecture

BS_MAC_TPT shall contain multiple dynamically expanding and contracting sets of protocol processes handling downlink and uplink traffic with reserved QoS, one set for each registered MS. These processes shall be created and destroyed dynamically in response to requests issued by LLC_TPT. MS_MAC_TPT shall contain only one set protocol processes managing reserved QoS.

MAC_TPT is not required to support "direct-mode" MPDU transfers, i.e. direct transfers between two or more MS within a single cell without intermediate handling by a BS. However, it is recommended that the design of MAC_TPT not preclude the addition of a direct-mode protocol at a later date.

6.2 Transport protocol suite

This subclauses describes the six protocols in the MAC_TPT protocol suite in more detail.

6.2.1 Unacknowledged downlink

UNACK_DL shall utilize a contention-free PHY access method available via PHY_SAP_A. A BS shall use UNACK_DL for system information and broadcast MPDUs. UNACK_DL PDUs may carry either a unicast or broadcast MSH as the destination address.

6.2.2 Unacknowledged uplink

UNACK_UL shall utilize a contention-possible PHY access method available via PHY_SAP_A. UNACK_UL PDUs may carry either a MSI or a unicast MSH as the source address. The MSI is used by an unregistered MS; the unicast MSH is used by a registered MS.

6.2.3 Acknowledged protocols

The ACK protocols shall implement a selective retransmission, ARQ strategy for MPDU transfer. The ACK protocols shall employ extensive error-recovery procedures to minimize transfer failures.

PDUs transferred by an ACK protocol may only carry a unicast MSH as the source or destination address.

The ACK protocols shall be responsible for data integrity QoS (DQOS). The DQOS supplied to an MPDU can range from error-free transfer (all corrupted blocks retransmitted until successfully conveyed) to partially error-free transfer (some corrupted blocks retransmitted; others left "as is") to open loop (no corrupted blocks retransmitted).

The DQOS provided to the MPDUs in a flow can either be fixed at protocol instantiation or determined dynamically. Dynamic determination of DQOS can be useful for time-sensitive traffic. In this case, MAC_BWM (TQOS) and MAC_TPT (DQOS) work together to manage the QoS supplied to a flow. For example, in order to maintain a certain TQOS, the DQOS can be lowered to compensate for deteriorating link quality.

The ACK protocols shall not discard MPDUs, duplicate MPDUs, or change the transmission order of MPDUs queued for transmission.

The ACK protocols shall include receiver flow control capability, by which the MPDU receiver can inform BS_MAC_BWM of near-term reception capability. BS_MAC_BWM shall utilize receiver flow control information in addition to any existing resource reservation commitments when performing bandwidth allocations among registered MS. For downlink transfers, receiver flow control information shall be signalled between MS_MAC_TPT and BS_MAC_BWM. For uplink transfers, receiver flow control information shall be passed via an internal interface between BS_MAC_TPT and BS_MAC_BWM. Receiver flow control may result in MPDU transfer suspension for a brief period.

The exact mechanism(s) by which MS_MAC_BWM communicates current uplink queue states to BS_MAC_BWM will be defined in a future version of the present document.

6.2.3.1 Acknowledged Transfers, Best-effort QOS

BS_MAC_BWM shall allocate bandwidth to acknowledged downlink and uplink best-effort flows on an as-demanded basis. If there are no pending MPDUs in the sender transmit queue, then no bandwidth is scheduled.

6.2.3.2 Acknowledged Transfers, Reserved QOS

The performance requirements associated with Reserved QoS will be defined in a future version of the present document.

7 Service Primitives

7.1 Primitive Definitions

7.1.1 MAC_transfer_request

MAC_transfer_request	
Usage	BS and MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_instance_ID*
	LPDU

This primitive is used by the LLC layer to pass a LPDU to the MAC layer for transfer to a peer MAC SAP B. The protocol instance ID parameter indicates which protocol instance should handle the transfer.

7.1.2 MAC_transfer_confirm

MAC_transfer_confirm	
Usage	BS and MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*transfer_receipt_ack*

This primitive acknowledges the receipt of the LPDU associated with a MAC_transfer_request. It does not indicate that the LPDU has been transferred to one or more peer SAPs.

7.1.3 MAC_transfer_indication

MAC_transfer_indication	
Usage	BS and MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*protocol_instance_ID*
	ack_transfer_result
	LPDU

This primitive passes a received LPDU to the LLC layer. The protocol instance ID of the protocol instance which performed the transfer is provided.

7.1.4 MAC_create_protocol_request

MAC_create_protocol_request	
Usage	BS and MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*protocol_instance_ID*
	MS_handle
	protocol_type
	protocol_parameters

This primitive requests the allocation of resources for a new acknowledged protocol instance.

BS_MAC_BWM will not initially allocate bandwidth to a new protocol instance. BS_LLC shall use other MAC primitives to enable bandwidth scheduling for the new protocol instance. The MS_handle is the MSH of the MS with which the protocol instance has been established.

If the MS LLC originates the request, the MS_handle is the MSH assigned to the MS by the BS with which the protocol instance has been established.

7.1.5 MAC_create_protocol_confirm

MAC_create_protocol_confirm	
Usage	BS and MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*create_protocol_result*

This primitive confirms the creation of the requested protocol.

7.1.6 MAC_delete_protocol_request

MAC_delete_protocol_request	
Usage	BS and MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	Yes
SDU Parameters	*protocol_instance_ID*

This primitive requests the deletion of a protocol instance. Any PDUs queued for transmission will be discarded.

7.1.7 MAC_delete_protocol_confirm

MAC_delete_protocol_confirm	
Usage	BS and MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*protocol_instance_ID*
	delete_protcol_result

This primitive confirms the deletion of a protocol instance.

7.1.8 MAC_configure_scheduling_request

MAC_configure_scheduling_request	
Usage	BS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	Yes
SDU Parameters	*protocol_instance_ID*
	new_scheduling_state

This primitive enables and disables bandwidth scheduling for a protocol instance.

7.1.9 MAC_configure_scheduling_confirm

MAC_configure_scheduling_confirm	
Usage	BS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*protocol_instance_ID*
	configure_scheduling_result

This primitive confirms a change in bandwidth scheduling status for a protocol instance.

7.1.10 MAC_queue_empty_notify_request

MAC_queue_empty_notify_request	
Usage	BS and MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	Yes
SDU Parameters	*protocol_instance_ID*

This primitive requests notification when the input queue for a protocol instance is empty.

7.1.11 MAC_queue_empty_notify_confirm

MAC_queue_empty_notify_confirm	
Usage	BS and MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*protocol_instance_ID*
	queue_empty_result

This primitive confirms that the input queue for a particular protocol instance is empty.

7.1.12 MAC_service_request

MAC_service_request	
Usage	MS
Source	LLC Layer
Destination	MAC Layer
Service Access Point	B
Multiple Outstanding	No
SDU Parameters	*base_station_ID*

This primitive tells the MAC to camp on the BS specified by *base_station_ID*.

7.1.13 MAC_service_confirm

MAC_service_confirm	
Usage	MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*service_result*

This primitive confirms a service request.

7.1.14 MAC_service_indication

MAC_service_indication	
Usage	MS
Source	MAC Layer
Destination	LLC Layer
Service Access Point	B
SDU Parameters	*service_status*

This primitive is used by the MAC to provide the LLC with the latest service status.

7.2 Parameter Definitions

7.2.1 *base_station_ID*

This parameter specifies a particular DAWS BS.

7.2.2 *ack_transfer_receipt_ack*

ack_transfer_receipt_ack	
0	success: receipt acknowledged
1	failure: transfer request already pending

7.2.3 *ack_transfer_result*

ack_transfer_result	
0	success: transfer OK
1	failure: transfer failed or aborted

7.2.4 *configure_scheduling_result*

configure_scheduling_result	
0	success: scheduler configured as requested
1	failure: specified protocol instance does not exist

7.2.5 *create_protocol_result*

create_protocol_result	
0	success: requested protocol created
1	failure: create protocol request already pending
2	failure: requested resources unavailable

7.2.6 *delete_protocol_result*

delete_protocol_result	
0	success: requested protocol deleted
1	failure: protocol instance does not exist

7.2.7 *LPDU*

Definition of this parameter is beyond the scope of the present document. Most often, it will consist of a LLC layer header and an IPv6 datagram.

7.2.8 *MS_handle*

This parameter is an identifier used to identify a particular MS while it is registered with a BS.

7.2.9 *new_scheduling_state*

new_scheduling_state	
0	scheduling disabled
1	scheduling enabled

7.2.10 *protocol_instance_ID*

This parameter uniquely identifies a protocol instance.

7.2.11 *protocol_parameters*

The format of this parameter depends upon the value of the *protocol_type* field.

If *protocol_type* is 0 or 1 (best-effort protocols), *protocol_parameters* is null.

If *protocol_type* is 2 or 3 (controlled-load protocols), *protocol_parameters* has the following format:

protocol_parameters (controlled-load)	
0	Source IP address
1	Flow label
2	Reservation specification

7.2.12 *protocol_type*

protocol_type	
0	Best-effort downlink
1	Best-effort uplink
2	Controlled-load downlink
3	Controlled-load uplink

7.2.13 *queue_empty_result*

queue_empty_result	
0	success: input queue of protocol instance is empty
1	failure: protocol instance does not exist

7.2.14 *service_result*

service_result	
0	success: requested service now available
1	failure: could not complete request

7.2.15 *service_status*

This parameter indicates whether service is currently provided, and if so, the *base_station_ID* and current signal strength and quality of the service. This parameter will be further defined after the DAWS PHY is defined.

7.2.16 *unack_transfer_receipt_ack*

unack_transfer_receipt_ack	
0	success: receipt acknowledged
1	failure: transfer request already pending

History

Document history		
V1.1.1	April 1999	Publication
V1.2.1	April 2000	Publication