WiMAX Reference Manual

Generated by Doxygen 1.5.1

Fri Feb 8 16:42:33 2008

Contents

Chapter 1

WiMAX Hierarchical Index

1.1 WiMAX Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

WiMAX Class Index

2.1 WiMAX Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

WiMAX File Index

3.1 WiMAX File List

Here is a list of all files with brief descriptions:

Chapter 4

WiMAX Class Documentation

4.1 active_flows Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **acflows_ptr next**
- u_int16_t **flow_id**
- u_int16_t **backlogged_tmp**
- int16_t **granted_bw_tmp**

4.1.1 Member Data Documentation

4.1.1.1 acflows_ptr active_flows::next

4.1.1.2 u_int16_t active_flows::flow_id

4.1.1.3 u_int16_t active_flows::backlogged_tmp

4.1.1.4 int16_t active_flows::granted_bw_tmp

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.2 allocation_time Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **start_time**
- double **end_time**
- u_int32_t **num_slots**
- **aptr next**
- u_char **type**
- u_char **used**

4.2.1 Member Data Documentation

4.2.1.1 double allocation_time::start_time

4.2.1.2 double allocation_time::end_time

4.2.1.3 u_int32_t allocation_time::num_slots

4.2.1.4 aptr allocation_time::next

4.2.1.5 u_char allocation_time::type

4.2.1.6 u_char allocation_time::used

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.3 bs_conf_param Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **mgmt_conf_param mgtparam**
- **map_conf_param mapparam**

4.3.1 Detailed Description

This structure contains all the configurable parameters of the BS node

4.3.2 Member Data Documentation

4.3.2.1 struct mgmt_conf_param bs_conf_param::mgtparam

4.3.2.2 struct map_conf_param bs_conf_param::mapparam

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.4 bs_snd_timer_list Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **expiration_time**
- **bs_sptr next**
- int **cindex**
- int **findex**

4.4.1 Member Data Documentation

4.4.1.1 double bs_snd_timer_list::expiration_time

4.4.1.2 bs_sptr bs_snd_timer_list::next

4.4.1.3 int bs_snd_timer_list::cindex

4.4.1.4 int bs_snd_timer_list::findex

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.5 bs_statistics Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **avg_interdpr_map**
- u_int32_t **avg_datagrants**
- u_int32_t **num_req**
- u_int32_t **avg_contentionslots**
- u_int32_t **data_sent**
- u_int32_t **fragmented_count**
- u_int32_t **concatenation_count**
- u_char **num_IE**
- u_int32_t **num_creq**
- u_int32_t **num_piggyreq**
- u_int32_t **num_creqgrant**
- u_int32_t **num_creqdenied**
- u_int32_t **total_num_cslots**
- u_int32_t **total_num_mgmtslots**

4.5.1 Detailed Description

Simulation output statistics variable

4.5.2 Member Data Documentation

4.5.2.1 double bs_statistics::avg_interdpr_map

4.5.2.2 u_int32_t bs_statistics::avg_datagrants

4.5.2.3 u_int32_t bs_statistics::num_req

4.5.2.4 u_int32_t bs_statistics::avg_contentionslots

4.5.2.5 u_int32_t bs_statistics::data_sent

4.5.2.6 u_int32_t bs_statistics::fragmented_count

4.5.2.7 u_int32_t bs_statistics::concatenation_count

4.5.2.8 u_char bs_statistics::num_IE

4.5.2.9 u_int32_t bs_statistics::num_creq

4.5.2.10 u_int32_t bs_statistics::num_piggyreq

4.5.2.11 u_int32_t bs_statistics::num_creqgrant

4.5.2.12 u_int32_t bs_statistics::num_creqdenied

4.5.2.13 u_int32_t bs_statistics::total_num_cslots

4.5.2.14 u_int32_t bs_statistics::total_num_mgmtslots

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.6 bsqnode Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- Packet * **pkt**
- double **enq_time**
- int **cindex**
- int **findex**
- **qlist next**

4.6.1 Member Data Documentation

4.6.1.1 Packet* bsqnode::pkt

4.6.1.2 double bsqnode::enq_time

4.6.1.3 int bsqnode::cindex

4.6.1.4 int bsqnode::findex

4.6.1.5 qlist bsqnode::next

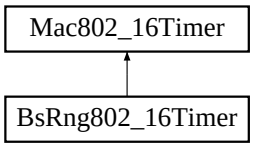
The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.7 BsRng802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsRng802_16Timer::



Public Member Functions

- **BsRng802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.7.1 Constructor & Destructor Documentation

4.7.1.1 BsRng802_16Timer::BsRng802_16Timer (Mac802_16 * *m*) [inline]

4.7.2 Member Function Documentation

4.7.2.1 void BsRng802_16Timer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

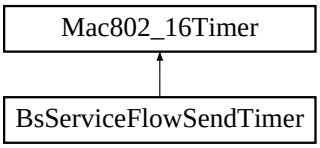
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.8 BsServiceFlowSendTimer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsServiceFlowSendTimer::



Public Member Functions

- **BsServiceFlowSendTimer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.8.1 Constructor & Destructor Documentation

4.8.1.1 BsServiceFlowSendTimer::BsServiceFlowSendTimer (Mac802_16 * *m*) [inline]

4.8.2 Member Function Documentation

4.8.2.1 void BsServiceFlowSendTimer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

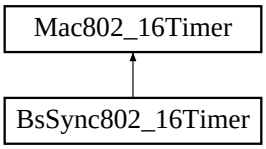
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.9 BsSync802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsSync802_16Timer::



Public Member Functions

- **BsSync802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.9.1 Constructor & Destructor Documentation

4.9.1.1 BsSync802_16Timer::BsSync802_16Timer (Mac802_16 * *m*) [inline]

4.9.2 Member Function Documentation

4.9.2.1 void BsSync802_16Timer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

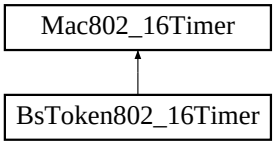
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.10 BsToken802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsToken802_16Timer::



Public Member Functions

- **BsToken802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.10.1 Constructor & Destructor Documentation

4.10.1.1 BsToken802_16Timer::BsToken802_16Timer (Mac802_16 * *m*) [inline]

4.10.2 Member Function Documentation

4.10.2.1 void BsToken802_16Timer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

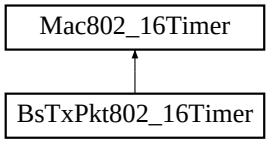
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.11 BsTxPkt802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsTxPkt802_16Timer::



Public Member Functions

- **BsTxPkt802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.11.1 Constructor & Destructor Documentation

4.11.1.1 BsTxPkt802_16Timer::BsTxPkt802_16Timer (Mac802_16 * *m*) [inline]

4.11.2 Member Function Documentation

4.11.2.1 void BsTxPkt802_16Timer::handle (Event * *e*) [virtual]

Send Timer

Reimplemented from **Mac802_16Timer** (p. 86).

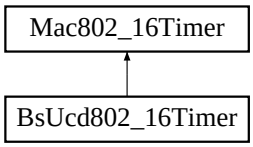
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.12 BsUcd802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsUcd802_16Timer::



Public Member Functions

- **BsUcd802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.12.1 Detailed Description

Timers to schedule transmition of Management messages

4.12.2 Constructor & Destructor Documentation

4.12.2.1 BsUcd802_16Timer::BsUcd802_16Timer (Mac802_16 * *m*) [inline]

4.12.3 Member Function Documentation

4.12.3.1 void BsUcd802_16Timer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

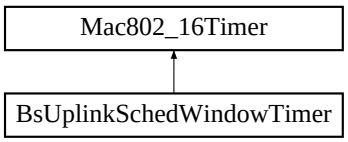
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.13 BsUplinkSchedWindowTimer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for BsUplinkSchedWindowTimer::



Public Member Functions

- **BsUplinkSchedWindowTimer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.13.1 Detailed Description

Window used to control the provision of the minimum BW requested by rtPS and nrtPS services.

4.13.2 Constructor & Destructor Documentation

4.13.2.1 BsUplinkSchedWindowTimer::BsUplinkSchedWindowTimer (Mac802_16 * *m*) [inline]

4.13.3 Member Function Documentation

4.13.3.1 void BsUplinkSchedWindowTimer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.14 downstream_channel Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **data_rate**
- double **prop_delay**
- u_int32_t **overhead_bytes**

4.14.1 Member Data Documentation

4.14.1.1 double downstream_channel::data_rate

4.14.1.2 double downstream_channel::prop_delay

4.14.1.3 u_int32_t downstream_channel::overhead_bytes

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.15 downstream_flow_record Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **flow_classifier classifier**
- **SchedType sched_type**
- **plist packet_list**
- u_int16_t **max_qsize**
- u_char **state**
- Packet * **pkt**
- u_char **pending**
- **allocation_time * alloc_list**
- double **ugsjitter**
- int **jitterSamples**
- double **last_granttime**
- double **nominal_alloctime**
- double **last_jittercaltime**
- double **ginterval**
- u_int16_t **gsize**
- double **latency**
- u_int32_t **min_bw**
- u_int16_t **flow_id**
- **PhsType PHS_profile**
- PacketQueue * **tokenq_**
- double **tokens_**
- double **rate_**
- int **bucket_**
- int **tokenqlen_**
- double **lastupdatetime_**
- int **init_**
- Event **intr**
- char **ratecontrol**
- double **util_total_bytes_DS**
- int **util_total_pkts_DS**
- int **total_pkts_dropped**
- int **dropped_tokenq**
- double **dsq_delay**
- int **num_qsamples**
- int **dropped_dsq**

4.15.1 Detailed Description

This structure defines attributes of downstream service flow of a SS

4.15.2 Member Data Documentation

4.15.2.1 struct flow_classifier downstream_flow_record::classifier

4.15.2.2 SchedType downstream_flow_record::sched_type

4.15.2.3 plist downstream_flow_record::packet_list

4.15.2.4 u_int16_t downstream_flow_record::max_qsize

4.15.2.5 u_char downstream_flow_record::state

4.15.2.6 Packet* downstream_flow_record::pkt

4.15.2.7 u_char downstream_flow_record::pending

4.15.2.8 struct allocation_time* downstream_flow_record::alloc_list

4.15.2.9 double downstream_flow_record::ugsjitter

4.15.2.10 int downstream_flow_record::jitterSamples

4.15.2.11 double downstream_flow_record::last_granttime

4.15.2.12 double downstream_flow_record::nominal_alloctime

4.15.2.13 double downstream_flow_record::last_jittercaltime

4.15.2.14 double downstream_flow_record::ginterval

4.15.2.15 u_int16_t downstream_flow_record::gsize

4.15.2.16 double downstream_flow_record::latency

4.15.2.17 u_int32_t downstream_flow_record::min_bw

4.15.2.18 u_int16_t downstream_flow_record::flow_id

4.15.2.19 PhsType downstream_flow_record::PHS_profile

4.15.2.20 PacketQueue* downstream_flow_record::tokenq_

4.15.2.21 double downstream_flow_record::tokens_

4.15.2.22 double downstream_flow_record::rate_

4.15.2.23 int downstream_flow_record::bucket_

4.15.2.24 int downstream_flow_record::tokenqlen_

4.15.2.25 double downstream_flow_record::lastupdatetime_

4.15.2.26 int downstream_flow_record::init_

4.15.2.27 Event downstream_flow_record::intr

4.15.2.28 char downstream_flow_record::ratecontrol

4.15.2.29 double downstream_flow_record::util_total_bytes_DS

4.15.2.30 int downstream_flow_record::util_total_pkts_DS

- ns-2.28/mac/**hdr-802_16.h**

4.16 downstream_sflow Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **downstream_flow_record downstream_record**

4.16.1 Detailed Description

To store the Upstream flow information so that it could be sent to BS during the Registration Phase

4.16.2 Member Data Documentation

4.16.2.1 struct downstream_flow_record downstream_sflow::downstream_record

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.17 flow_classifier Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- int32_t **src_ip**
- int32_t **dst_ip**
- packet_t **pkt_type**

4.17.1 Detailed Description

This defines the classifier to be used for matching packets to upstream/downstream flows

4.17.2 Member Data Documentation

4.17.2.1 int32_t flow_classifier::src_ip

4.17.2.2 int32_t flow_classifier::dst_ip

4.17.2.3 packet_t flow_classifier::pkt_type

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.18 hdr_mac802_16 Struct Reference

`#include <hdr-802_16.h>`

Public Member Functions

- **mac802_16_frame_hdr** & **dshdr** ()
- int & **dt_conv_overhead** ()

Static Public Member Functions

- static int & **offset** ()
- static **hdr_mac802_16** * **access** (const Packet *p)

Public Attributes

- **mac802_16_frame_hdr dshdr_**
- int **dt_conv_overhead_**

Static Public Attributes

- static int **offset_**

4.18.1 Detailed Description

802_16 packet header

4.18.2 Member Function Documentation

4.18.2.1 static int& hdr_mac802_16::offset () [inline, static]

4.18.2.2 static hdr_mac802_16* hdr_mac802_16::access (const Packet * *p*) [inline, static]

4.18.2.3 struct mac802_16_frame_hdr& hdr_mac802_16::dshdr () [inline]

4.18.2.4 int& hdr_mac802_16::dt_conv_overhead () [inline]

4.18.3 Member Data Documentation

4.18.3.1 struct mac802_16_frame_hdr hdr_mac802_16::dshdr_

4.18.3.2 int hdr_mac802_16::dt_conv_overhead_

4.18.3.3 int hdr_mac802_16::offset_ [static]

The documentation for this struct was generated from the following files:

- ns-2.28/mac/**hdr-802_16.h**

- ns-2.28/mac/**mac-802_16-bs.cc**

4.19 hdr_mac802_16extd Struct Reference

`#include <hdr-802_16.h>`

Public Member Functions

- int & **num_hdr_** ()
- **mac802_16_extended_header_element** * **exthdr** ()

Static Public Member Functions

- static int & **offset** ()
- static **hdr_mac802_16extd** * **access** (const Packet *p)

Public Attributes

- **mac802_16_extended_header_element exthdr_** [NUM_EXTENDED_HDR_ELEMENTS]
- int **num_hdr**

Static Public Attributes

- static int **offset_**

4.19.1 Detailed Description

802_16 extended packet header

4.19.2 Member Function Documentation

4.19.2.1 static int& hdr_mac802_16extd::offset () [inline, static]

4.19.2.2 static hdr_mac802_16extd* hdr_mac802_16extd::access (const Packet * *p*) [inline, static]

4.19.2.3 int& hdr_mac802_16extd::num_hdr_ () [inline]

4.19.2.4 struct mac802_16_extended_header_element* hdr_mac802_16extd::exthdr () [inline]

4.19.3 Member Data Documentation

4.19.3.1 struct mac802_16_extended_header_element hdr_mac802_16extd::exthdr_[NUM_EXTENDED_HDR_ELEMENTS]

4.19.3.2 int hdr_mac802_16extd::num_hdr

4.19.3.3 int hdr_mac802_16extd::offset_ [static]

The documentation for this struct was generated from the following files:

- ns-2.28/mac/**hdr-802_16.h**
- ns-2.28/mac/**mac-802_16-bs.cc**

4.20 hdr_mac802_16map Struct Reference

802_16 map packet header

`#include <hdr-802_16.h>`

Public Member Functions

- double & **allocstarttime_** ()
- double & **allocendtime_** ()
- double & **acktime_** ()
- u_char & **numIE_** ()
- u_char & **bkoff_start_** ()
- u_char & **bkoff_end_** ()

Static Public Member Functions

- static int & **offset** ()
- static **hdr_mac802_16map** * **access** (const Packet *p)

Public Attributes

- double **allocstarttime**
- double **allocendtime**
- double **acktime**
- u_char **numIE**
- u_char **bkoff_start**
- u_char **bkoff_end**

Static Public Attributes

- static int **offset_**

4.20.1 Detailed Description

802_16 map packet header

The information elements will be send in DATA portion of packet

4.20.2 Member Function Documentation

4.20.2.1 static int& hdr_mac802_16map::offset () [inline, static]

4.20.2.2 static hdr_mac802_16map* hdr_mac802_16map::access (const Packet * *p*) [inline, static]

4.20.2.3 double& hdr_mac802_16map::allocstarttime_ () [inline]

4.20.2.4 double& hdr_mac802_16map::allocendtime_ () [inline]

4.20.2.5 double& hdr_mac802_16map::acktime_ () [inline]

4.20.2.6 u_char& hdr_mac802_16map::numIE_ () [inline]

4.20.2.7 u_char& hdr_mac802_16map::bkoff_start_ () [inline]

4.20.2.8 u_char& hdr_mac802_16map::bkoff_end_ () [inline]

4.20.3 Member Data Documentation

4.20.3.1 double hdr_mac802_16map::allocstarttime

4.20.3.2 double hdr_mac802_16map::allocendtime

4.20.3.3 double hdr_mac802_16map::acktime

4.20.3.4 u_char hdr_mac802_16map::numIE

4.20.3.5 u_char hdr_mac802_16map::bkoff_start

4.20.3.6 u_char hdr_mac802_16map::bkoff_end

4.20.3.7 int hdr_mac802_16map::offset_ [static]

The documentation for this struct was generated from the following files:

- ns-2.28/mac/**hdr-802_16.h**
- ns-2.28/mac/**mac-802_16-bs.cc**

4.21 hdr_mac802_16mgmt Struct Reference

`#include <hdr-802_16.h>`

Public Member Functions

- int & **dstaddr_** ()
- int & **srcaddr_** ()
- u_char & **type_** ()

Static Public Member Functions

- static int & **offset** ()
- static **hdr_mac802_16mgmt** * **access** (const Packet *p)

Public Attributes

- int **dstaddr**
- int **srcaddr**
- u_char **msg_payload_** [MANAGEMENT_MSG_PAYLOAD_SIZE]
- u_char **type**

Static Public Attributes

- static int **offset_**

4.21.1 Detailed Description

802_16 management packet header

4.21.2 Member Function Documentation

4.21.2.1 static int& hdr_mac802_16mgmt::offset () [inline, static]

4.21.2.2 static hdr_mac802_16mgmt* hdr_mac802_16mgmt::access (const Packet * *p*) [inline, static]

4.21.2.3 int& hdr_mac802_16mgmt::dstaddr_ () [inline]

4.21.2.4 int& hdr_mac802_16mgmt::srcaddr_ () [inline]

4.21.2.5 u_char& hdr_mac802_16mgmt::type_ () [inline]

4.21.3 Member Data Documentation

4.21.3.1 int hdr_mac802_16mgmt::dstaddr

4.21.3.2 int hdr_mac802_16mgmt::srcaddr

4.21.3.3 u_char hdr_mac802_16mgmt::msg_payload_[MANAGEMENT_MSG_PAYLOAD_SIZE]

4.21.3.4 u_char hdr_mac802_16mgmt::type

4.21.3.5 int hdr_mac802_16mgmt::offset_ [static]

The documentation for this struct was generated from the following files:

- ns-2.28/mac/**hdr-802_16.h**
- ns-2.28/mac/**mac-802_16-bs.cc**

4.22 job Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **release_time**
- double **period**
- double **deadline**
- u_int32_t **mini_slots**
- **jptr next**
- u_int16_t **flow_id**
- **SchedType sclass**
- u_char **type**
- u_char **flag**
- u_char **retry_count**
- double **ugsjitter**
- int **jitterSamples**
- double **last_jittercaltime**

4.22.1 Member Data Documentation

4.22.1.1 double job::release_time

4.22.1.2 double job::period

4.22.1.3 double job::deadline

4.22.1.4 u_int32_t job::mini_slots

4.22.1.5 jptr job::next

4.22.1.6 u_int16_t job::flow_id

4.22.1.7 SchedType job::sclass

4.22.1.8 u_char job::type

4.22.1.9 u_char job::flag

4.22.1.10 u_char job::retry_count

4.22.1.11 double job::ugsjitter

4.22.1.12 int job::jitterSamples

4.22.1.13 double job::last_jittercaltime

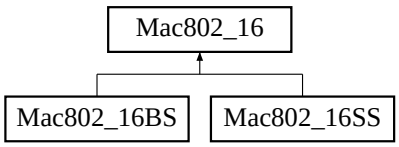
The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.23 Mac802_16 Class Reference

`#include <mac-802_16.h>`

Inheritance diagram for Mac802_16::



Public Member Functions

- **Mac802_16** ()
- virtual ~**Mac802_16** ()
- virtual void **BsUcdHandler** (Event *e)
- virtual void **BsSyncHandler** (Event *e)
- virtual void **BsRngHandler** (Event *e)
- virtual void **BsMapHandler** (Event *e)
- virtual void **SSRngHandler** (Event *e)
- virtual void **SSSndTimerHandler** (Event *e)
- virtual void **SSReqTimerHandler** (Event *e)
- virtual void **RecvFrame** (Packet *p, int i)
- virtual void **BsSendHandler** (Event *e)
- virtual void **BsSndTimerHandler** (Event *e)
- virtual void **BsWindowTimerHandler** (Event *e)
- virtual void **BsTokenHandler** (Event *e)
- int **command** (int argc, const char *const *argv)
- void **recvHandler** (Event *e)
- void **sendHandler** (Event *e)
- void **recv** (Packet *p, Handler *h)
- void **configure_upstream** ()
- void **set_bit** (u_char *, int, int)
- void **dump_pkt** (Packet *)
- char **ClassifyDataMgmt** (Packet *)
- int **is_idle** ()
- int **match** (Packet *, struct **flow_classifier**)
- int **bit_on** (u_char, int)
- u_int32_t **calculate_slots** (double, double)
- u_int32_t **power** (u_int32_t, u_int32_t)
- double **TX_Time** (Packet *p, int)
- Packet * **AllocPkt** (int)
- void **insert_alloclist** (**aptr** &, u_int16_t, double, double, u_int16_t, u_int32_t)
- int **CanBeSent** (**aptr** &, Packet *, u_int32_t)
- int **len_queue** (**plist**)
- int **CanErtPSBeSent** (**aptr** &, Packet *)

Public Attributes

- **TxPkt802_16Timer mhTxPkt_**
- **RxPkt802_16Timer mhRxPkt_**
- **upstream_channel upchannel**
- **downstream_channel downchannel**
- char **collision**
- u_int16_t **bytes_pminislot**
- u_int32_t **minislots_psec**
- double **size_mslots**
- u_int16_t **size_ureqgrant**
- u_int32_t **avg_pkts**
- u_int32_t **avg_bytes**
- u_int32_t **num_pkts**
- double **num_bytes**
- u_int32_t **total_num_sent_pkts**
- double **total_num_sent_bytes**
- u_int32_t **total_num_mgt_pkts_US**
- u_int32_t **total_num_rng_pkts_US**
- u_int32_t **total_num_concat_pkts_US**
- u_int32_t **total_num_frag_pkts_US**
- u_int32_t **total_num_req_pkts_US**
- u_int32_t **total_num_plaindata_pkts_US**
- u_int32_t **total_num_concatdata_pkts_US**
- u_int32_t **total_num_frames_US**
- u_int32_t **total_num_BE_pkts_US**
- u_int32_t **total_num_RTVBR_pkts_US**
- u_int32_t **total_num_UGS_pkts_US**
- u_int32_t **total_num_OTHER_pkts_US**
- u_int32_t **total_num_rx_pkts**
- double **total_num_rx_bytes**
- double **total_num_BW_bytesUP**
- double **total_num_BW_bytesDOWN**
- double **total_num_appbytesUS**
- double **total_num_appbytesDS**
- u_int32_t **total_packets_dropped**
- double **last_BWCalcTime**
- double **last_rtime**
- u_int32_t **avg_mgmtpkts**
- u_int32_t **num_mgmtpkts**
- u_int32_t **avg_mgmtbytes**
- u_int32_t **num_mgmtbytes**
- double **last_mmgmttime**
- MacState **rx_state_**
- MacState **tx_state_**
- int **tx_active_**

Static Public Attributes

- static int **lan_num**
- static **Mac802_16BS** * **bs_arr** [NUM_802_16_LANS]

Friends

- class **TxPkt802_16Timer**
- class **RxPkt802_16Timer**

4.23.1 Detailed Description

Mac802_16 (p. 36) class

4.23.2 Constructor & Destructor Documentation

4.23.2.1 Mac802_16::Mac802_16 ()

Constructor function

4.23.2.2 virtual Mac802_16::~Mac802_16 () [inline, virtual]

4.23.3 Member Function Documentation

4.23.3.1 virtual void Mac802_16::BsUcdHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 51), and **Mac802_16SS** (p. 72).

4.23.3.2 virtual void Mac802_16::BsSyncHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 51), and **Mac802_16SS** (p. 72).

4.23.3.3 virtual void Mac802_16::BsRngHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 51), and **Mac802_16SS** (p. 72).

4.23.3.4 virtual void Mac802_16::BsMapHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 50), and **Mac802_16SS** (p. 71).

4.23.3.5 virtual void Mac802_16::SSRngHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 50), and **Mac802_16SS** (p. 72).

4.23.3.6 virtual void Mac802_16::SSSndTimerHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 50), and **Mac802_16SS** (p. 72).

4.23.3.7 virtual void Mac802_16::SSReqTimerHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 50), and **Mac802_16SS** (p. 72).

4.23.3.8 virtual void Mac802_16::RecvFrame (Packet * p, int i) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 51), and **Mac802_16SS** (p. 74).

4.23.3.9 virtual void Mac802_16::BsSendHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 54), and **Mac802_16SS** (p. 72).

4.23.3.10 virtual void Mac802_16::BsSndTimerHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 59).

4.23.3.11 virtual void Mac802_16::BsWindowTimerHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 51).

4.23.3.12 virtual void Mac802_16::BsTokenHandler (Event * e) [inline, virtual]

Reimplemented in **Mac802_16BS** (p. 51), and **Mac802_16SS** (p. 72).

4.23.3.13 int Mac802_16::command (int *argc*, const char *const * *argv*)

Reimplemented in **Mac802_16BS** (p. 50), and **Mac802_16SS** (p. 71).

4.23.3.14 void Mac802_16::recvHandler (Event * e)

4.23.3.15 void Mac802_16::sendHandler (Event * e)

4.23.3.16 void Mac802_16::recv (Packet * p, Handler * h)

To handle incoming packet.

4.23.3.17 void Mac802_16::configure_upstream ()

To configure the mini-slot parameters

4.23.3.18 void Mac802_16::set_bit (u_char *, int, int)

4.23.3.19 void Mac802_16::dump_pkt (Packet *)

4.23.3.20 char Mac802_16::ClassifyDataMgmt (Packet * p)

This returns a 0 if this frame contains data

else it returns a 1 in which case it contains a MAC specific header (i.e., request)

4.23.3.21 int Mac802_16::is_idle ()

Test if the channel is idle.

4.23.3.22 int Mac802_16::match (Packet *, struct *flow_classifier*)

4.23.3.23 int Mac802_16::bit_on (u_char, int)

4.23.3.24 u_int32_t Mac802_16::calculate_slots (double *stime*, double *etime*)

Parameters:

stime starting time for the map

etime ending time

Returns:

the number of slots in the map.

4.23.3.25 u_int32_t Mac802_16::power (u_int32_t *bs*, u_int32_t *in*)

Calculate bs raised to in

4.23.3.26 double Mac802_16::TX_Time (Packet * *p*, int)

4.23.3.27 Packet * Mac802_16::AllocPkt (int)

4.23.3.28 void Mac802_16::insert_alloclist (aptr &, u_int16_t, double, double, u_int16_t, u_int32_t)

4.23.3.29 int Mac802_16::CanBeSent (aptr &, Packet *, u_int32_t)

4.23.3.30 int Mac802_16::len_queue (plist)

4.23.3.31 int Mac802_16::CanErtPSBeSent (aptr & *alloc_list*, Packet * *p*)

Check if there is a grant for the ertPS service and if this grant is greater than the packet size.

Parameters:

p,: The frame that already has been adjusted for all headers

Returns:

0: If there is no grant for ertPS service; 1: If there is a grant, but it is lesser than the packet size; 2: If there is a grant greater or equal to the packet size;

4.23.4 Friends And Related Function Documentation

4.23.4.1 friend class TxPkt802_16Timer [friend]

4.23.4.2 friend class RxPkt802_16Timer [friend]

4.23.5 Member Data Documentation

4.23.5.1 TxPkt802_16Timer Mac802_16::mhTxPkt_

4.23.5.2 RxPkt802_16Timer Mac802_16::mhRxPkt_

4.23.5.3 struct upstream_channel Mac802_16::upchannel

4.23.5.4 struct downstream_channel Mac802_16::downchannel

4.23.5.5 char Mac802_16::collision

4.23.5.6 u_int16_t Mac802_16::bytes_pminislot

4.23.5.7 u_int32_t Mac802_16::minislots_psec

4.23.5.8 double Mac802_16::size_mslots

4.23.5.9 u_int16_t Mac802_16::size_ureqgrant

4.23.5.10 u_int32_t Mac802_16::avg_pkts

4.23.5.11 u_int32_t Mac802_16::avg_bytes

4.23.5.12 u_int32_t Mac802_16::num_pkts

4.23.5.13 double Mac802_16::num_bytes

4.23.5.14 u_int32_t Mac802_16::total_num_sent_pkts

4.23.5.15 double Mac802_16::total_num_sent_bytes

4.23.5.16 u_int32_t Mac802_16::total_num_mgt_pkts_US

4.23.5.17 u_int32_t Mac802_16::total_num_rng_pkts_US

4.23.5.18 u_int32_t Mac802_16::total_num_concat_pkts_US

4.23.5.19 u_int32_t Mac802_16::total_num_frag_pkts_US

4.23.5.20 u_int32_t Mac802_16::total_num_req_pkts_US

4.23.5.21 u_int32_t Mac802_16::total_num_plaindata_pkts_US

4.23.5.22 u_int32_t Mac802_16::total_num_concatdata_pkts_US

4.23.5.23 u_int32_t Mac802_16::total_num_frames_US

4.23.5.24 u_int32_t Mac802_16::total_num_BE_pkts_US

4.23.5.25 u_int32_t Mac802_16::total_num_RTVBR_pkts_US

4.23.5.26 u_int32_t Mac802_16::total_num_UGS_pkts_US

4.23.5.27 u_int32_t Mac802_16::total_num_OTHER_pkts_US

- ns-2.28/mac/**mac-802_16.h**
- ns-2.28/mac/**mac-802_16-base.cc**
- ns-2.28/mac/**mac-802_16-bs.cc**
- ns-2.28/mac/**mac-802_16-FSM.cc**

4.24 mac802_16_extended_header_element Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- u_char **eh_type**: 4
- u_char **eh_len**: 4
- u_int16_t **eh_data** [6]

4.24.1 Detailed Description

802_16 Extended header element

4.24.2 Member Data Documentation

4.24.2.1 u_char mac802_16_extended_header_element::eh_type

4.24.2.2 u_char mac802_16_extended_header_element::eh_len

4.24.2.3 u_int16_t mac802_16_extended_header_element::eh_data[6]

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.25 mac802_16_frame_hdr Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- u_char **fc_type**: 2
- u_char **fc_parm**: 5
- u_char **ehdr_on**: 1
- u_int16_t **mac_param**
- u_int16_t **len**

4.25.1 Detailed Description

802_16 header format

4.25.2 Member Data Documentation

4.25.2.1 u_char mac802_16_frame_hdr::fc_type

4.25.2.2 u_char mac802_16_frame_hdr::fc_parm

4.25.2.3 u_char mac802_16_frame_hdr::ehdr_on

4.25.2.4 u_int16_t mac802_16_frame_hdr::mac_param

4.25.2.5 u_int16_t mac802_16_frame_hdr::len

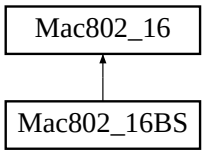
The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.26 Mac802_16BS Class Reference

`#include <mac-802_16.h>`

Inheritance diagram for Mac802_16BS::



Public Member Functions

- int **register_to_bs** (int macaddr, u_int16_t priority, u_char def_up, u_char def_dn, struct **upstream_sflow** *UpEntry, u_char UpSize, struct **downstream_sflow** *DownEntry, u_char DownSize)
- int **compare_priority** (**priority_array_p** A, **priority_array_p** B)
- **Mac802_16BS** ()
- ~**Mac802_16BS** ()

Public Attributes

- **ss_record** * **SSRecord**
- u_int16_t **CurrIndexSSTable**

Protected Member Functions

- int **command** (int argc, const char *const *argv)

Private Member Functions

- void **SSRngHandler** (Event *e)
- void **SSSndTimerHandler** (Event *e)
- void **SSReqTimerHandler** (Event *e)
- void **UnlockQueue** ()
- void **BsMapHandler** (Event *e)

  This is called when a MAP is to be sent.

- void **BsSyncHandler** (Event *e)
- void **BsUcdHandler** (Event *e)
- void **BsRngHandler** (Event *e)
- void **BsWindowTimerHandler** (Event *e)
- void **RateControl** (Packet *, int, int, int)
- double **getupdatedtokens** (int, int)
- void **BsTokenHandler** (Event *)
- void **insert_tokenlist** (double, int, int)
- void **sendUp** (Packet *p)
- void **sendDown** (Packet *p)

- void **RecvFrame** (Packet *, int)
- void **PassDownstream** (Packet *)
- void **HandleInData** (Packet *, int, int)
- void **HandleInMgmt** (Packet *)
- void **HandleFrag** (Packet *)
- void **HandleConcat** (Packet *, int, int)
- void **ReleaseJobs** ()
- void **fill_job** (struct **job** *, char, char, double, u_int32_t, double, double, u_int16_t)
- void **InsertJob** (struct **job** *, int)
- void **HandleReq** (u_int16_t, int, int, bool)
- void **tune_parameters** ()
- void **alloc_bw** ()
- void **delete_joblist** (int)
- void **PhUnsupress** (Packet *, int, int)
- void **ApplyPhs** (Packet *, int, int)
- void **MakeAperiodicAlloc** (struct **job** *, **mapptr**, int)
- void **AllocMemSSrecord** ()
- void **MakePeriodicAllocation** (int, **mapptr**, double)
- void **ReOrder** (**mapptr**)
- void **MarkOtherSlots** (**mapptr**, u_int32_t, int)
- void **jobdrop** (**mapptr**)
- void **MarkGrantPending** (u_int32_t *, double)
- void **SendMap** ()
- u_int32_t **MarkUnusedSlots** (u_int32_t)
- void **find_flowindex** (u_int16_t, int *, int *)
- void **SendFrame** (Packet *, int)
- void **BsSendHandler** (Event *)
- void **dump_stats** ()
- void **ParseExtHdr** (Packet *)
- void **dumpUGSJITTER** (char *)
- void **UpdateJitter** (int, u_int16_t, double)
- int **find_size_map** ()
- int **NumContSlots** (**mapptr** *)
- int **ChkQoSJobs** (double, double)
- int **classify** (Packet *, char, int *)
- int **insert_mapjob** (int, **jptr**, u_int32_t *, int)
- int **insert_mapjob** (struct **job** *, u_int32_t *)
- int **find_ss** (int)
- u_int32_t **MarkOtherAlloc** (u_int32_t)
- u_int32_t **FillMap** (int, int, int, int, int)
- **mapptr find_best_hole** (u_int32_t)
- **mapptr find_next** (**mapptr**)
- **mapptr find_prv** (**mapptr**)
- **mapptr TryAlloc** (double, u_int32_t)
- double **determine_deadline** (char, int, int)
- double **FitMap** (**mapptr**, double)
- double **CalculateAckTime** ()
- void **print_job_list** (int)
- void **print_map_list** ()
- void **print_short_map_list** (Packet *Pkt)

This method dumps all IE's associated with the map.

- void **delete_maplist** ()
- void **dumpBWBS** (char *fileName)
- void **dumpFinalBSStats** (char *fileName, int DSBW, int USBW)
- void **dumpFinalSIDStats** (int cindex, int findex, char *fileName, int DSBW)
- void **dumpFinalDSSIDS** (char *fileName, int DSBW)
- void **dump802_16QueueStats** (char *fileName, int channelBW)
- void **dump802_16UtilStats** (char *fileName, int channelDSBW, int channelUSBW)
- void **insert_pkt_at_Head** (Packet *, int, int)
- int **len_mgmtqueue** ()
- Packet * **deque_pkt** ()
- int **packetTrace** (Packet *p, int direction)
- void **HandleOutData** (Packet *, int, int)
- int **down_idle** (int, int, **EventType**, Packet *)
- int **down_decision** (int, int, **EventType**, Packet *)
- int **down_tosend** (int, int, **EventType**, Packet *)
- int **down_waitformap** (int, int, **EventType**, Packet *)
- void **UpdateAllocationTable** (int, int)
- void **UpdateJitter** (int, int)
- bool **insert_pkt** (Packet *, int, int)
- double **timer_expiration** (int, int, Packet *)
- Packet * **deque_pkt** (int, int)
- void **insert_mgmtpkt** (Packet *, int, int)
- void **insert_sndlist** (double, int, int)
- void **BsSndTimerHandler** (Event *e)
- void **alloc_ul_bw** (double, double, u_int32_t)
- void **alloc_dl_bw** (double, double, u_int32_t)
- void **checkDeadline** (double, double, u_int32_t &)
- void **checkMinimumBW** (u_int32_t &, **priority_array_p** &, int &)
- void **MakeAllocation** (**jptr**, int, **mapptr**, char, int)
- int **CanBeSent** (**aptr** &, Packet *)

Private Attributes

- double **window_**
- **Map802_16Timer mhMap_**
- **BsRng802_16Timer mhRng_**
- **BsUcd802_16Timer mhUcd_**
- **BsSync802_16Timer mhSync_**
- **BsTxPkt802_16Timer mhBsTxPkt_**
- **BsServiceFlowSendTimer mhBsSend_**
- **BsUplinkSchedWindowTimer mhWdw_**
- **BsToken802_16Timer mhToken_**
- **tkptr TokenList**
- **bs_conf_param Conf_Table_**
- u_int16_t **SizeSSTable**
- **bs_statistics SSStatistics_**
- **job * job_list** [8]
- **mapptr mptr**

- **bs_sptr BsSndList**
- double **map_stime**
- double **omap_stime**
- double **map_etime**
- double **omap_etime**
- double **map_acktime**
- double **AckTime**
- double **max_slots_pmap**
- double **next_map**
- int **numIE**
- int **map_lookahead**
- int **MAP_LOOKAHEAD**
- int **max_burst_slots**
- int **rem_overhead**
- int **MapPropDelay**
- int **size_rtqueue**
- int **avg_szrtqueue**
- double **last_mrqtime**
- int **size_bfqueue**
- int **avg_szbfqueue**
- double **last_mbfqtime**
- int **num_bfreq**
- int **avg_bfreq**
- double **last_mbfreq**
- int **num_rtreq**
- int **avg_rtreq**
- double **last_mrtreq**
- double **last_dmptime**
- u_int32_t **num_dgrant**
- u_int32_t **num_contention**
- u_int32_t **num_req**
- u_int32_t **num_gpend**
- u_int32_t **avg_dgrant**
- u_int32_t **avg_contention**
- u_int32_t **avg_req**
- u_int32_t **avg_gpend**
- u_int32_t **dropped_tokenq**
- u_int32_t **dropped_dsq**
- double **proportion**
- u_char **contention_thrhold**
- u_char **network_status**
- u_int32_t **rtpoll_ddlinemiss**
- u_int32_t **beffort_ddlinemiss**
- int32_t **num_rtslots**
- int32_t **num_nrtslots**
- int32_t **num_befslots**
- u_int32_t **num_adjust_slots**
- Event **intr**
- Event **uintr**
- Event **sintr**

- Event **rintr**
- Event **rxintr_**
- Event **wintr**
- **qlist txq_**
- int **qnp_**
- int **qnb_**
- int **max_qnp_**
- int **max_qnb_**
- int **min_qnp_**
- int **min_qnb_**
- int **queue_total_bytes_in**
- int **queue_total_bytes_out**
- double **util_total_bytes_US**
- double **util_total_bytes_DS**
- int **util_total_pkts_US**
- int **util_total_pkts_DS**
- double **util_bytes_US**
- double **util_bytes_DS**
- int **qlim_**
- double **lastDumpTime**
- double **lastUtilDumpTime**
- int(Mac802_16BS::* **DOWNswitch** [DOWNSTATES])(int, int, **EventType**, Packet *)

Static Private Attributes

- static int **next_flowid**

Friends

- class **Map802_16Timer**
- class **BsUcd802_16Timer**
- class **BsRng802_16Timer**
- class **BsSync802_16Timer**
- class **BsTxPkt802_16Timer**
- class **BsServiceFlowSendTimer**
- class **BsToken802_16Timer**

4.26.1 Detailed Description

Mac802_16 (p. 36) class

4.26.2 Constructor & Destructor Documentation

4.26.2.1 Mac802_16BS::Mac802_16BS ()

Constructor

4.26.2.2 Mac802_16BS::~Mac802_16BS () [inline]

4.26.3 Member Function Documentation

4.26.3.1 int Mac802_16BS::register_to_bs (int *macaddr*, u_int16_t *priority*, u_char *def_up*, u_char *def_dn*, struct upstream_sflow * *UpEntry*, u_char *UpSize*, struct downstream_sflow * *DownEntry*, u_char *DownSize*)

4.26.3.2 int Mac802_16BS::compare_priority (priority_array_p *A*, priority_array_p *B*)

4.26.3.3 int Mac802_16BS::command (int *argc*, const char *const * *argv*) [protected]

Reimplemented from **Mac802_16** (p. 39).

4.26.3.4 void Mac802_16BS::SSRngHandler (Event * *e*) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.26.3.5 void Mac802_16BS::SSSndTimerHandler (Event * *e*) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.26.3.6 void Mac802_16BS::SSReqTimerHandler (Event * *e*) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.26.3.7 void Mac802_16BS::UnlockQueue () [inline, private]

4.26.3.8 void Mac802_16BS::BsMapHandler (Event * *e*) [private, virtual]

This is called when a MAP is to be sent.

This is called when a MAP is to be sent.

- Overview:
- `Figure` out the start/stop times for this MAP
- `Call` calculate_slots to make sure the MAP does not go > 4096 slots in the future (this would violate 802_16).
- `Call` **alloc_bw()** (p. 53) to assign slots
- `Figure` out when to reset the bs 802_16 timer for the next MAP

Parameters:

e timer event

Reimplemented from **Mac802_16** (p. 38).

4.26.3.9 void Mac802_16BS::BsSyncHandler (Event * e) [private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.26.3.10 void Mac802_16BS::BsUcdHandler (Event * e) [private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.26.3.11 void Mac802_16BS::BsRngHandler (Event * e) [private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.26.3.12 void Mac802_16BS::BsWindowTimerHandler (Event * e) [private, virtual]

Reimplemented from **Mac802_16** (p. 39).

4.26.3.13 void Mac802_16BS::RateControl (Packet * p, int *cindex*, int *findex*, int *type*) [private]

The type filed is used to indicate whether packet was picked from IFQ queue or not..If yes, then unlock the IFQ

4.26.3.14 double Mac802_16BS::getupdatedtokens (int, int) [private]

4.26.3.15 void Mac802_16BS::BsTokenHandler (Event *) [private, virtual]

Reimplemented from **Mac802_16** (p. 39).

4.26.3.16 void Mac802_16BS::insert_tokenlist (double, int, int) [private]

4.26.3.17 void Mac802_16BS::sendUp (Packet * p) [private]

Packet coming up from PHY layer...

4.26.3.18 void Mac802_16BS::sendDown (Packet * p) [private]

Send packet down to the physical layer...only packets passed by upper layers will be sent by this function..No Management messages..

4.26.3.19 void Mac802_16BS::RecvFrame (Packet * p, int *concat_flag*) [private, virtual]

Called from the recvHandler when a frame has arrived on the upstream channel.

Needed to make sure we increment byte stats only once as this routine is reentered if the frame is concatonated static int RecvFrameEntered = 0; This routine will invoke HandleInData or HandleInMgmt ....

Parameters:

p A packet that already has been adjusted for all headers

concat_flag true if we are being reentered due to a concatonated frame. In this case, we've already counted the packet, but we need to count the size.

Reimplemented from **Mac802_16** (p. 39).

4.26.3.20 void Mac802_16BS::PassDownstream (Packet * *p*) [private]

A simple ethernet PDU is carried in the 802_16 payload.. No Fragmentation or Concatenation

4.26.3.21 void Mac802_16BS::HandleInData (Packet * *p*, int *cindex*, int *findex*) [private]

overHeadBitsStripped is 1 if the packet already has had the OH bits taken off (This occurs when multiple frames are concatonated)

4.26.3.22 void Mac802_16BS::HandleInMgmt (Packet * *p*) [private]

This routine receives a Mgt message of the following types:

- case 2: request header
- case 3: Fragmentation frame header
- case 26: Concatenation frame header
- case 1: RNG msg

Note: if it's a concat frame- this means its a concat request.

Parameters:

p the packet

4.26.3.23 void Mac802_16BS::HandleFrag (Packet * *p*) [private]

Parameters:

p The frame containing a fragment

4.26.3.24 void Mac802_16BS::HandleConcat (Packet *, int, int) [private]

4.26.3.25 void Mac802_16BS::ReleaseJobs () [private]

Function called during the starting of simulation to release all the UGS jobs & VBR unicast req jobs

4.26.3.26 void Mac802_16BS::fill_job (struct job * *node*, char *sclass*, char *type*, double *period*, u_int32_t *size*, double *release*, double *deadline*, u_int16_t *flow_id*) [private]

Parameters:

node job queue element (i.e., a job)

4.26.3.27 void Mac802_16BS::InsertJob (struct job * *node*, int *index*) [private]

a job (bw request or periodic grant) in the job_list

4.26.3.28 void Mac802_16BS::HandleReq (u_int16_t *req*, int *cindex*, int *findex*, bool *aggregate*) [private]

Is called when there has been a request for bw from a SS. The routine creates a new job entry (node)

4.26.3.29 void Mac802_16BS::tune_parameters () [private]

4.26.3.30 void Mac802_16BS::alloc_bw () [private]

This is called when a MAP is to be sent. It allocates the bw.

4.26.3.31 void Mac802_16BS::delete_joblist (int) [private]

4.26.3.32 void Mac802_16BS::PhUnsupress (Packet *, int, int) [private]

4.26.3.33 void Mac802_16BS::ApplyPhs (Packet *, int, int) [private]

4.26.3.34 void Mac802_16BS::MakeAperiodicAlloc (struct job *, mapptr, int) [private]

4.26.3.35 void Mac802_16BS::AllocMemSSrecord () [private]

4.26.3.36 void Mac802_16BS::MakePeriodicAllocation (int, mapptr, double) [private]

4.26.3.37 void Mac802_16BS::ReOrder (mapptr) [private]

4.26.3.38 void Mac802_16BS::MarkOtherSlots (mapptr *t*, u_int32_t *num*, int *jk*) [private]

This method marks the cr and mgt slots in the map. jk will be used as sid for distinguishing between various slots. a 0 says its mgt , a 1 says it CR.

Parameters:

t The map

num Number of slots.

jk Indicates if mgt or CR slots (value 0 and 1 respectively)

4.26.3.39 void Mac802_16BS::jobdrop (mapptr) [private]

4.26.3.40 void Mac802_16BS::MarkGrantPending (u_int32_t *, double) [private]

4.26.3.41 void Mac802_16BS::SendMap () [private]

This method creates the MAP message, creates the packet that contains the message and sends it.

4.26.3.42 u_int32_t Mac802_16BS::MarkUnusedSlots (u_int32_t *num_slots*) [private]

This method fills unused slots of the current map as unused. Or, if FILLWITHCONTOPS is defined, it fills with CR slots. The global mptr points to the current map.

Parameters:

num_slots Number of unused slots in the map

Returns:

The final number of unused slots (hopefully 0 if FILLWITHCONTOPS is on)

4.26.3.43 void Mac802_16BS::find_flowindex (u_int16_t, int *, int *) [private]

4.26.3.44 void Mac802_16BS::SendFrame (Packet * *p*, int *type*) [private]

All frames that are to be sent over the DS channel will be sent down through this method.

4.26.3.45 void Mac802_16BS::BsSendHandler (Event *) [private, virtual]

Reimplemented from **Mac802_16** (p. 39).

4.26.3.46 void Mac802_16BS::dump_stats () [private]

4.26.3.47 void Mac802_16BS::ParseExtHdr (Packet *) [private]

4.26.3.48 void Mac802_16BS::dumpUGSJITTER (char * *outputFile*) [private]

This is called periodically (as of 3/26/05 it is piggybacked off the dumpBWBS mechanism). It computes a jitter statistic based on stats gathered since the last time this method was called. Each time a UGS allocation is made, the jitter is updated. This running total of jitter is simply the difference between the scheduled allocation time (based on the UGS grant interval) and the actual.

4.26.3.49 void Mac802_16BS::UpdateJitter (int *ugs*, u_int16_t *flow_id*, double *t*) [private]

Called when the BS has allocated BW for this job in the next map.

Parameters:

flow_id The flow or job to look at

t The allocated time (j->alloc_stime) of the job

4.26.3.50 int Mac802_16BS::find_size_map () [private]

Calculate the size of MAP, all the IEs plus data-grant pending sizes..

4.26.3.51 int Mac802_16BS::NumContSlots (mapptr *) [private]

4.26.3.52 int Mac802_16BS::ChkQoSJobs (double, double) [private]

4.26.3.53 int Mac802_16BS::classify (Packet * *p*, char *dir*, int * *find*) [private]

Examines the packet and matches it to a flow in the SSRecord. The dir param determines if the upstream or downstream flow are looked at.

Parameters:

p The packet

dir DOWNSTREAM or UPSTREAM

find This ptr variable is set with the index of the flow table entry.

Returns:

The index of the matching ss's SSRecord table entry.

4.26.3.54 int Mac802_16BS::insert_mapjob (int, jptr, u_int32_t *, int) [private]

4.26.3.55 int Mac802_16BS::insert_mapjob (struct job *, u_int32_t *) [private]

4.26.3.56 int Mac802_16BS::find_ss (int *t*) [private]

Given the mac address parameter, this routine searches the SS table to return the SSRecord table index of the correct SS.

4.26.3.57 u_int32_t Mac802_16BS::MarkOtherAlloc (u_int32_t *num_slots*) [private]

This function will mark contention-slots and SM slots in map list

Parameters:

num_slots Number of slots in the map

4.26.3.58 u_int32_t Mac802_16BS::FillMap (int *i*, int *ugs*, int *rt_poll*, int *nrt_poll*, int *be*) [private]

This is called when a MAP is to be sent. The mptr contains the start of the map.

Returns:

The number of slots used in the allocation

4.26.3.59 mapptr Mac802_16BS::find_best_hole (u_int32_t *n*) [private]

This function returns a pointer to the map node where the best hole begins

Parameters:

n The number of slots that we need to allocate

Returns:

A ptr to a MAP

4.26.3.60 mapptr Mac802_16BS::find_next (mapptr *m*) [private]

Returns:

The next job in the map list.

4.26.3.61 mapptr Mac802_16BS::find_prv (mapptr) [private]

4.26.3.62 mapptr Mac802_16BS::TryAlloc (double, u_int32_t) [private]

4.26.3.63 double Mac802_16BS::determine_deadline (char, int, int) [private]

4.26.3.64 double Mac802_16BS::FitMap (mapptr *j*, double *t*) [private]

Try to fit job 'j' starting at 't'. Keep on trying till you reach the end of the MAP..

4.26.3.65 double Mac802_16BS::CalculateAckTime () [private]

4.26.3.66 void Mac802_16BS::print_job_list (int *i*) [private]

Prints to stdout the contents of the desired job queue:

- job_list[0] which is UGS periodic
- job_list[1] which is rt-VBR periodic
- job_list[2] which is rt-VBR non periodic
- job_list[3] which is best effort

Parameters:

i Indicates which job list to print

4.26.3.67 void Mac802_16BS::print_map_list () [private]

4.26.3.68 void Mac802_16BS::print_short_map_list (Packet * *p*) [private]

This method dumps all IE's associated with the map.

This method dumps all IE's associated with the map. The format: each MAP will begin with a code of 0 and have zero or more codes of 1,2,3.

- 0 timestamp mapstart mapstop numberIE's
- 1 timestamp flow_id start stop release time deadline IUC numberslots

Parameters:

p The packet that contains the map message

4.26.3.69 void Mac802_16BS::delete_maplist () [private]

4.26.3.70 void Mac802_16BS::dumpBWBS (char * *fileName*) [private]

4.26.3.71 void Mac802_16BS::dumpFinalBSStats (char * *fileName*, int *DSBW*, int *USBW*) [private]

4.26.3.72 void Mac802_16BS::dumpFinalSIDStats (int *cindex*, int *findex*, char * *fileName*, int *DSBW*) [private]

4.26.3.73 void Mac802_16BS::dumpFinalDSSIDS (char * *fileName*, int *DSBW*) [private]

4.26.3.74 void Mac802_16BS::dump802_16QueueStats (char * *outputFile*, int *channelBW*) [private]

Called periodically by a TCL script to monitor the queue in the DS direction. This method also computes the utilization of the DS channel for each iteration.

4.26.3.75 void Mac802_16BS::dump802_16UtilStats (char * *outputFile*, int *DSBW*, int *USBW*) [private]

Called periodically by a TCL script to monitor the queue in the DS direction.

4.26.3.76 void Mac802_16BS::insert_pkt_at_Head (Packet * *p*, int *cindex*, int *findex*) [private]

This method inserts the packet in the downstream Tx queue. It inserts at the head of the queue. If the Q is full, the packet at the tail is removed.

Parameters:

p ptr to the frame

cindex Holds the SSRecord table index associated with the SS that is the target.

findex Holds the flow record entry index.

4.26.3.77 int Mac802_16BS::len_mgmtqueue () [private]

4.26.3.78 Packet * Mac802_16BS::deque_pkt () [private]

4.26.3.79 int Mac802_16BS::packetTrace (Packet * *p*, int *direction*) [private]

Parameters:

p A 802_16 frame

direction direction is 0 when sent DS and 1 when arrival at US

Returns:

a 0 on success.

4.26.3.80 void Mac802_16BS::HandleOutData (Packet * *p*, int *cindex*, int *findex*) [private]

Packet enters the BS for respective flow from this function

4.26.3.81 int Mac802_16BS::down_idle (int *cindex*, int *findex*, EventType *e*, Packet * *p*) [private]

Parameters:

tbindex denotes the service-flow entry on which packet has been mapped

4.26.3.82 int Mac802_16BS::down_decision (int, int, EventType, Packet *) [private]

4.26.3.83 int Mac802_16BS::down_tosend (int, int, EventType, Packet *) [private]

4.26.3.84 int Mac802_16BS::down_waitformap (int *cindex*, int *findex*, EventType *e*, Packet * *p*) [private]

This is called when certain events occur while a downstream flow is waiting for a MAP.

4.26.3.85 void Mac802_16BS::UpdateAllocationTable (int *cindex*, int *findex*) [private]

This function updates the allocation table for a service-flow when a MAP is received

4.26.3.86 void Mac802_16BS::UpdateJitter (int *cindex*, int *findex*) [private]

This is called whenever a MAP is received by a UGS flow. The statistic adds all error (scheduled allocation time from actual). Then the dumpUGSJitter method periodically obtains the mean jitter = runningJitter / samples

4.26.3.87 bool Mac802_16BS::insert_pkt (Packet *, int, int) [private]

4.26.3.88 double Mac802_16BS::timer_expiration (int, int, Packet *) [private]

4.26.3.89 Packet * Mac802_16BS::deque_pkt (int, int) [private]

4.26.3.90 void Mac802_16BS::insert_mgmtpkt (Packet * *p*, int *cindex*, int *findex*) [private]

This method inserts the packet in the downstream Tx queue. It drops the packet if it's full.

Parameters:

p ptr to the frame

cindex Holds the SSRecord table index associated with the SS that is the target.

findex Holds the flow record entry index.

4.26.3.91 void Mac802_16BS::insert_sndlist (double, int, int) [private]

4.26.3.92 void Mac802_16BS::BsSndTimerHandler (Event * *e*) [private, virtual]

Send timer to data packets from QoS scheduling services

Reimplemented from **Mac802_16** (p. 39).

4.26.3.93 void Mac802_16BS::alloc_ul_bw (double *map_stime*, double *map_etime*, u_int32_t *num_slots*) [private]

This is called when a MAP is to be sent. It allocates bw to the UL subframe.

4.26.3.94 void Mac802_16BS::alloc_dl_bw (double *map_stime*, double *map_etime*, u_int32_t *num_slots*) [private]

This is called when a MAP is to be sent. It allocates bw to the DL subframe.

4.26.3.95 void Mac802_16BS::checkDeadline (double *map_stime*, double *subframe_duration*, u_int32_t & *num_slots*) [private]

check which rtPS requests must be sent in the next frame in order to satisfy the maximum latency request.

4.26.3.96 void Mac802_16BS::checkMinimumBW (u_int32_t & *num_slots*, priority_array_p & *pri_array*, int & *size*) [private]

Assign priority values for the requests in the intermetiate queue to decide wich ones will be scheduled. The priority computation considers the min_bw request, backlogged BW and granted BW in order to provide the min_bw request of each flow. The higher priority requests will be inserted in the high queue.

Allocate the array. Warning. You should deallocate this array after using it.

4.26.3.97 void Mac802_16BS::MakeAllocation (jptr, int, mapptr, char, int) [private]

4.26.3.98 int Mac802_16BS::CanBeSent (aptr & *alloc_list*, Packet * *p*) [private]

Parameters:

p The frame that already has been adjusted for all headers

Returns:

1, if this frame can be send; 0, if this frame can NOT be send

4.26.4 Friends And Related Function Documentation

4.26.4.1 friend class Map802_16Timer [friend]

4.26.4.2 friend class BsUcd802_16Timer [friend]

4.26.4.3 friend class BsRng802_16Timer [friend]

4.26.4.4 friend class BsSync802_16Timer [friend]

4.26.4.5 friend class BsTxPkt802_16Timer [friend]

4.26.4.6 friend class BsServiceFlowSendTimer [friend]

4.26.4.7 friend class BsToken802_16Timer [friend]

4.26.5 Member Data Documentation

4.26.5.1 struct ss_record* Mac802_16BS::SSRecord

4.26.5.2 u_int16_t Mac802_16BS::CurrIndexSSTable

4.26.5.3 double Mac802_16BS::window_ [private]

4.26.5.4 Map802_16Timer Mac802_16BS::mhMap_ [private]

4.26.5.5 BsRng802_16Timer Mac802_16BS::mhRng_ [private]

4.26.5.6 BsUcd802_16Timer Mac802_16BS::mhUcd_ [private]

4.26.5.7 BsSync802_16Timer Mac802_16BS::mhSync_ [private]

4.26.5.8 BsTxPkt802_16Timer Mac802_16BS::mhBsTxPkt_ [private]

4.26.5.9 BsServiceFlowSendTimer Mac802_16BS::mhBsSend_ [private]

4.26.5.10 BsUplinkSchedWindowTimer Mac802_16BS::mhWdw_ [private]

4.26.5.11 BsToken802_16Timer Mac802_16BS::mhToken_ [private]

4.26.5.12 tkptr Mac802_16BS::TokenList [private]

4.26.5.13 struct bs_conf_param Mac802_16BS::Conf_Table_ [private]

4.26.5.14 u_int16_t Mac802_16BS::SizeSSTable [private]

4.26.5.15 struct bs_statistics Mac802_16BS::SSStatistics_ [private]

4.26.5.16 struct job* Mac802_16BS::job_list[8] [private]

4.26.5.17 mapptr Mac802_16BS::mptr [private]

4.26.5.18 bs_sptr Mac802_16BS::BsSndList [private]

4.26.5.19 double Mac802_16BS::map_stime [private]

4.26.5.20 double Mac802_16BS::omap_stime [private]

4.26.5.21 double Mac802_16BS::map_etime [private]

4.26.5.22 double Mac802_16BS::omap_etime [private]

- ns-2.28/mac/**mac-802_16.h**
- ns-2.28/mac/**mac-802_16-bs.cc**
- ns-2.28/mac/**mac-802_16-FSM.cc**

4.27 Mac802_16BsClass Class Reference

Public Member Functions

- **Mac802_16BsClass** ()
- TclObject * **create** (int, const char *const *)

4.27.1 Detailed Description

TCL Hooks for the simulator

4.27.2 Constructor & Destructor Documentation

4.27.2.1 Mac802_16BsClass::Mac802_16BsClass () [inline]

4.27.3 Member Function Documentation

4.27.3.1 TclObject* Mac802_16BsClass::create (int, const char *const *) [inline]

The documentation for this class was generated from the following file:

- ns-2.28/mac/**mac-802_16-bs.cc**

4.28 Mac802_16EHeaderClass Class Reference

Public Member Functions

- **Mac802_16EHeaderClass** ()

4.28.1 Detailed Description

Mac802_16EHeaderClass (p. 64) class

4.28.2 Constructor & Destructor Documentation

4.28.2.1 Mac802_16EHeaderClass::Mac802_16EHeaderClass () [inline]

The documentation for this class was generated from the following file:

- ns-2.28/mac/**mac-802_16-bs.cc**

4.29 Mac802_16HeaderClass Class Reference

Public Member Functions

- **Mac802_16HeaderClass** ()

4.29.1 Detailed Description

Mac802_16HeaderClass (p. 65) class

4.29.2 Constructor & Destructor Documentation

4.29.2.1 Mac802_16HeaderClass::Mac802_16HeaderClass () [inline]

The documentation for this class was generated from the following file:

- ns-2.28/mac/**mac-802_16-bs.cc**

4.30 Mac802_16MapHeaderClass Class Reference

Public Member Functions

- **Mac802_16MapHeaderClass** ()

4.30.1 Detailed Description

Mac802_16MHeaderClass (p. 67) class

4.30.2 Constructor & Destructor Documentation

4.30.2.1 Mac802_16MapHeaderClass::Mac802_16MapHeaderClass () [inline]

The documentation for this class was generated from the following file:

- ns-2.28/mac/**mac-802_16-bs.cc**

4.31 Mac802_16MHeaderClass Class Reference

Public Member Functions

- **Mac802_16MHeaderClass** ()

4.31.1 Detailed Description

Mac802_16MHeaderClass (p. 67) class

4.31.2 Constructor & Destructor Documentation

4.31.2.1 Mac802_16MHeaderClass::Mac802_16MHeaderClass () [inline]

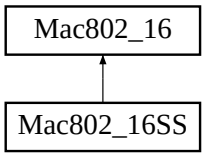
The documentation for this class was generated from the following file:

- ns-2.28/mac/**mac-802_16-bs.cc**

4.32 Mac802_16SS Class Reference

`#include <mac-802_16.h>`

Inheritance diagram for Mac802_16SS::



Public Member Functions

- **Mac802_16SS** ()
- ~**Mac802_16SS** ()

Protected Member Functions

- int **command** (int argc, const char *const *argv)

Private Member Functions

- void **BsMapHandler** (Event *e)
- void **BsSyncHandler** (Event *e)
- void **BsUcdHandler** (Event *e)
- void **BsRngHandler** (Event *e)
- void **BsSendHandler** (Event *e)
- void **BsTokenHandler** (Event *e)
- void **SSRngHandler** (Event *e)
- void **SSSndTimerHandler** (Event *e)
- void **SSReqTimerHandler** (Event *e)
- void **sendUp** (Packet *p)
- void **sendDown** (Packet *p)
- void **Initialize_entry** (char direction, char index)
- void **HandleMap** (Packet *)
- void **HandleOutData** (Packet *, char)
- void **SendData** (Packet *, char)
- int **SendReq** (char, Packet *)
- void **decide_frag** (Packet *, char)
- void **send_frag_data** (Packet *, char)
- void **fill_extended_header** (int, Packet *, char)
- void **send_concat** (Packet *, char)
- void **RecvFrame** (Packet *, int)
- void **handle_indata** (Packet *, char)
- void **handle_inmgmt** (Packet *, char)
- void **SetDefaultFlow** ()
- void **PhUnsupress** (Packet *, char)
- void **ApplyPhs** (Packet *, char)

- void **print_alloclist** (char)
- void **print_short_map_list** (Packet *Pkt)
- void **print_classifiers** ()
- void **dump_stats** ()
- char **classify** (Packet *, char)
- int **check_concat_req** (char)
- int **check_frag_req** (u_int32_t, char)
- int **fill_piggyback_req** (char)
- void **MarkUsedContentionSlots** (int, char)
- void **turn_off_contention** (char)
- int **back_off** (char, Packet *)
- void **insert_sndlist** (double, char)
- void **insert_reqlist** (double, char)
- void **refresh_reqlist** (char)
- void **reinsert_reqlist** (char)
- void **insert_pkt** (Packet *, char)
- void **UpdateAllocationTable** (char)
- void **HandleOtherMgmt** (Packet *)
- void **print_ssalloclist** ()
- void **HandleOutMgmt** (Packet *, char)
- void **FillPiggyReq** (char, Packet *)
- int **FillPiggyExtHdr** (char, Packet *, int)
- void **UpdateJitter** (char)
- int **CanBeSent** (**aptr** &, Packet *, char)
- int **CanUnicastReqBeSent** (char)
- int **CanContentionReqBeSent** (char)
- int **DataGrantPending** (char)
- int **MapSentAfterReq** (char)
- int **NumContentionSlots** (char)
- void **USRateMeasure** (char, Packet *)
- void **us_getupdatedtokens** (char)
- int **look_at_queue** (char)
- int **filterACKPackets** (Packet *pkt, char tbindex)
- int **packetMatch** (Packet *pkt1, Packet *pkt2)
- double **find_contention_slot** (char, u_int32_t)
- double **timer_expiration** (char, Packet *, int)
- Packet * **deque_pkt** (char)
- Packet * **pkt_lookup** (char tbindex, int i)
- void **dumpBWSS** (char *fileName)
- void **dumpUGSJITTER** (char, char *filename)
- void **dumpFinalSSStats** (char *fileName)
- void **dump802_16QueueStats** (char *fileName, int channelBW)
- void **dump802_16UtilStats** (char *fileName, int channelDSBW, int channelUSBW)
- int **packetTrace** (Packet *p, int direction)
- int **timingsTrace** (Packet *p, int id)
- int **ugs_idle** (char, **EventType**, Packet *)
- int **ugs_decision** (char, **EventType**, Packet *)
- int **ugs_tosend** (char, **EventType**, Packet *)
- int **ugs_waitformap** (char, **EventType**, Packet *)
- int **rtpoll_idle** (char, **EventType**, Packet *)

- int **rtpoll_decision** (char, **EventType**, Packet *)
- int **rtpoll_tosend** (char, **EventType**, Packet *)
- int **rtpoll_waitformap** (char, **EventType**, Packet *)
- int **rtpoll_tosendreq** (char, **EventType**, Packet *)
- int **ertpoll_idle** (char, **EventType**, Packet *)
- int **ertpoll_decision** (char, **EventType**, Packet *)
- int **ertpoll_tosend** (char, **EventType**, Packet *)
- int **ertpoll_waitformap** (char, **EventType**, Packet *)
- int **ertpoll_tosendreq** (char, **EventType**, Packet *)
- int **nrtpoll_idle** (char, **EventType**, Packet *)
- int **nrtpoll_decision** (char, **EventType**, Packet *)
- int **nrtpoll_tosend** (char, **EventType**, Packet *)
- int **nrtpoll_waitformap** (char, **EventType**, Packet *)
- int **nrtpoll_tosendreq** (char, **EventType**, Packet *)
- int **nrtpoll_reqsent** (char, **EventType**, Packet *)
- int **nrtpoll_contention** (char, **EventType**, Packet *)
- int **beffort_idle** (char, **EventType**, Packet *)
- int **beffort_decision** (char, **EventType**, Packet *)
- int **beffort_tosend** (char, **EventType**, Packet *)
- int **beffort_waitformap** (char, **EventType**, Packet *)
- int **beffort_tosendreq** (char, **EventType**, Packet *)
- int **beffort_reqsent** (char, **EventType**, Packet *)
- int **beffort_contention** (char, **EventType**, Packet *)
- int **beffort_ratecheck** (char, **EventType**, Packet *)
- void **FillErtPSPiggyReq** (char, Packet *)
- int **SendErtPSReq** (char, Packet *)

Private Attributes

- RNG * **rng_**
- **SSRng802_16Timer mhSSRng_**
- **SSServiceFlowSendTimer mhSSSend_**
- **SSServiceFlowRequestTimer mhReq_**
- Packet * **map_**
- char **debug_ss**
- u_int32_t **ss_id**
- u_char **my_lan**
- u_char **SizeUpFlowTable**
- **upstream_sflow UpFlowTable** [MAX_NUM_UPFLOWS_PERSS]
- u_char **SizeDownFlowTable**
- **downstream_sflow DownFlowTable** [MAX_NUM_DOWNFLOWS_PERSS]
- u_char **default_upstream_index_**
- u_char **default_dstream_index_**
- u_int16_t **priority**
- double **rng_freq**
- int **bs_addr**
- double **map_acktime**
- u_int32_t **total_num_frag**
- u_int32_t **total_num_collisions**
- u_int32_t **total_num_bkoff**

- u_int32_t **total_collision_drops**
- u_int32_t **total_queue_drops**
- **sptr SndList**
- **rptr ReqList**
- **rptr tempReqList**
- int(Mac802_16SS::* **UGSswitch** [UGSSTATES])(char, **EventType**, Packet *)
- int(Mac802_16SS::* **RTPOLLswitch** [RTPOLLSTATES])(char, **EventType**, Packet *)
- int(Mac802_16SS::* **ERTPOLLswitch** [ERTPOLLSTATES])(char, **EventType**, Packet *)
- int(Mac802_16SS::* **NRTPOLLswitch** [NRTPOLLSTATES])(char, **EventType**, Packet *)
- int(Mac802_16SS::* **BEFFORTswitch** [BEFFORTSTATES])(char, **EventType**, Packet *)
- Event **rintr**
- double **last_dmptime**
- int **reqFlag**
- int **reqFlagCounter**

Friends

- class **SSRng802_16Timer**
- class **SSServiceFlowSendTimer**
- class **SSServiceFlowRequestTimer**

4.32.1 Detailed Description

Mac802_16SS (p. 68) class

4.32.2 Constructor & Destructor Documentation

4.32.2.1 Mac802_16SS::Mac802_16SS ()

Constructor Function

4.32.2.2 Mac802_16SS::~Mac802_16SS () [inline]

4.32.3 Member Function Documentation

4.32.3.1 int Mac802_16SS::command (int *argc*, const char *const * *argv*) [protected]

Reimplemented from **Mac802_16** (p. 39).

4.32.3.2 void Mac802_16SS::BsMapHandler (Event * *e*) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.32.3.3 void Mac802_16SS::BsSyncHandler (Event * e) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.32.3.4 void Mac802_16SS::BsUcdHandler (Event * e) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.32.3.5 void Mac802_16SS::BsRngHandler (Event * e) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.32.3.6 void Mac802_16SS::BsSendHandler (Event * e) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 39).

4.32.3.7 void Mac802_16SS::BsTokenHandler (Event * e) [inline, private, virtual]

Reimplemented from **Mac802_16** (p. 39).

4.32.3.8 void Mac802_16SS::SSRngHandler (Event * e) [private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.32.3.9 void Mac802_16SS::SSSndTimerHandler (Event * e) [private, virtual]

Timer handler function

Reimplemented from **Mac802_16** (p. 38).

4.32.3.10 void Mac802_16SS::SSReqTimerHandler (Event * e) [private, virtual]

Reimplemented from **Mac802_16** (p. 38).

4.32.3.11 void Mac802_16SS::sendUp (Packet * p) [private]

So a packet has been received by the PHY, and we want to send it up to the 802_16 link...

4.32.3.12 void Mac802_16SS::sendDown (Packet * p) [private]

Send packets passed by LLC to physical layer..So, NO MAC level Management Messages will be sent via this fucntion

4.32.3.13 void Mac802_16SS::Initialize_entry (char *direction*, char *index*) [private]

4.32.3.14 void Mac802_16SS::HandleMap (Packet * *p*) [private]

This routine invokes every flow handler (in whatever state they are in) to process the map. If a flow sees an opportunity to send- it updates the SS's allocation table.

Parameters:

p The packet containing the MAP

4.32.3.15 void Mac802_16SS::HandleOutData (Packet * *p*, char *tbindex*) [private]

Packet enters the SM for respective flow from this function

4.32.3.16 void Mac802_16SS::SendData (Packet *, char) [private]

4.32.3.17 int Mac802_16SS::SendReq (char *tbindex*, Packet * *p*) [private]

This is called when the SS wants to send a request to the BS It is invoked off the REQ_TIMER thread. The routine check_concat_req(tbindex) is called to see if we can send a concat request. The routine also calls beffort_ratecheck to see if it is even allowed to request bandwidth yet (subject to rate control).

Parameters:

p This is the current packet that we want to send.

Returns:

A 0 if an error occurs and the request is not sent, else a 1.

4.32.3.18 void Mac802_16SS::decide_frag (Packet * *p*, char *tbindex*) [private]

This routine is called at the slot time that has been allocated as the beginning transmission time for this packet.

Parameters:

p frame (not concatonated)

tbindex index to US flow table

4.32.3.19 void Mac802_16SS::send_frag_data (Packet * *p*, char *tbindex*) [private]

This routine is called when the packet p is a part of a frag or must be sent as a frag.

Parameters:

p frame (not concatonated)

tbindex index to US flow table

4.32.3.20 void Mac802_16SS::fill_extended_header (int *i*, Packet * *p*, char *tbindex*) [private]

Parameters:

i if (i == 1) it's a fragment, else not.

4.32.3.21 void Mac802_16SS::send_concat (Packet * *p*, char *tbindex*) [private]

This is called by SendData when a concatonated frame is ready to be sent. The number to send depends on the grant size

Parameters:

p Packet to send

tbindex Flow ptr

4.32.3.22 void Mac802_16SS::RecvFrame (Packet *, int) [private, virtual]

Reimplemented from **Mac802_16** (p. 39).

4.32.3.23 void Mac802_16SS::handle_indata (Packet *, char) [private]

4.32.3.24 void Mac802_16SS::handle_inmgmt (Packet *, char) [private]

4.32.3.25 void Mac802_16SS::SetDefaultFlow () [private]

4.32.3.26 void Mac802_16SS::PhUnsupress (Packet *, char) [private]

4.32.3.27 void Mac802_16SS::ApplyPhs (Packet *, char) [private]

4.32.3.28 void Mac802_16SS::print_alloclist (char) [private]

4.32.3.29 void Mac802_16SS::print_short_map_list (Packet * *p*) [private]

This method dumps all IE's associated with the map. The format: each MAP will begin with a code of 0 and have zero or more codes of 1,2,3.

- 0 timestamp SSID mapstart mapstop numberIE's maplength in slots
- 1 timestamp flowid (null IE)
- 2 timestamp flowid (indicates a grant is pending to the SS)
- 3 timestamp flowid starttime endtime iuccode slots

Parameters:

p The packet that contains the map message

4.32.3.30 void Mac802_16SS::print_classifiers () [private]

4.32.3.31 void Mac802_16SS::dump_stats () [private]

4.32.3.32 char Mac802_16SS::classify (Packet *, char) [private]

4.32.3.33 int Mac802_16SS::check_concat_req (char *tbindex*) [private]

returns a 1 if there is at least one packet waiting in the queue AND concat is enabled.

4.32.3.34 int Mac802_16SS::check_frag_req (u_int32_t, char) [private]

4.32.3.35 int Mac802_16SS::fill_piggyback_req (char) [private]

4.32.3.36 void Mac802_16SS::MarkUsedContentionSlots (int, char) [private]

4.32.3.37 void Mac802_16SS::turn_off_contention (char) [private]

4.32.3.38 int Mac802_16SS::back_off (char, Packet *) [private]

4.32.3.39 void Mac802_16SS::insert_sndlist (double, char) [private]

4.32.3.40 void Mac802_16SS::insert_reqlist (double, char) [private]

4.32.3.41 void Mac802_16SS::refresh_reqlist (char) [private]

4.32.3.42 void Mac802_16SS::reinsert_reqlist (char) [private]

4.32.3.43 void Mac802_16SS::insert_pkt (Packet *, char) [private]

4.32.3.44 void Mac802_16SS::UpdateAllocationTable (char *tbindex*) [private]

This function updates the allocation table for a service-flow when a MAP is received

4.32.3.45 void Mac802_16SS::HandleOtherMgmt (Packet *) [private]

4.32.3.46 void Mac802_16SS::print_ssalloclist () [private]

4.32.3.47 void Mac802_16SS::HandleOutMgmt (Packet *, char) [private]

4.32.3.48 void Mac802_16SS::FillPiggyReq (char *tbindex*, Packet * *p*) [private]

This is called to see if a request can be piggybacked to this packet, p. Nothing happens if (len_queue(UpFlowTable[tbindex].packet_list) = 0). IF there is at least one packet in the queue, deque it and save it in the flow table: UpFlowTable[tbindex].pkt = deque_pkt(tbindex);

This routine pulls this next packet, sees how many slots are required and then calls

ratecheck_returnval = FillPiggyExtHdr(tbindex,p,slots);

This sets the current packet with a piggyback request for bandwidth for the next packet that was just dequeued

4.32.3.49 int Mac802_16SS::FillPiggyExtHdr (char, Packet *, int) [private]

4.32.3.50 void Mac802_16SS::UpdateJitter (char *tbindex*) [private]

This is called whenever a MAP is received by a UGS flow. The statistic adds all error (scheduled allocation time from actual). Then the dumpUGSJitter method periodically obtains the mean jitter = runningJitter / samples

4.32.3.51 int Mac802_16SS::CanBeSent (aptr &, Packet *, char) [private]

4.32.3.52 int Mac802_16SS::CanUnicastReqBeSent (char) [private]

4.32.3.53 int Mac802_16SS::CanContentionReqBeSent (char) [private]

4.32.3.54 int Mac802_16SS::DataGrantPending (char) [private]

4.32.3.55 int Mac802_16SS::MapSentAfterReq (char) [private]

4.32.3.56 int Mac802_16SS::NumContentionSlots (char) [private]

4.32.3.57 void Mac802_16SS::USRateMeasure (char *tbindex*, Packet * *p*) [private]

This routine updates the available tokens subject to the BW rate taking into account the transmission of this packet.

4.32.3.58 void Mac802_16SS::us_getupdatedtokens (char *tbindex*) [private]

This routine computes the available tokens subject to BW rate limits.

4.32.3.59 int Mac802_16SS::look_at_queue (char *tbindex*) [private]

This routine is for debug purposes. It displays packets in the queue.

Parameters:

tbindex Indicates the service-flow entry to which the packet has been mapped

4.32.3.60 int Mac802_16SS::filterACKPackets (Packet * *pkt*, char *tbindex*) [private]

This routine is called only if you want to do ack filtering at a SS. Before the packet is inserted in the queue (to await upstream transmission) packets that are already in the queue are scanned. All older ACK packets for this TCP connection (identified by matching the flowid, src/dst addr/port) and with a seqno less than the seqno of this ACK are removed from the queue and the packets are freed.

Parameters:

pkt ptr to a packet that has arrived and is to be queued

tbindex Indicates the service-flow entry to which the packet has been mapped

Returns:

The number of ACK packets that are remmoved from the Queue.

4.32.3.61 int Mac802_16SS::packetMatch (Packet * *pkt1*, Packet * *pkt2*) [private]

This routine compares the 2 packets and sees it their flowid, src/dst addr/port all match. If so, they are from the same flow.

Parameters:

pkt1 : ptr to a one packet

pkt2 : ptr to the second packet

Returns:

a 1 is pkt2 and pkt1 are from the same flow. Else a 0.

4.32.3.62 double Mac802_16SS::find_contention_slot (char *tbindex*, u_int32_t *num*) [private]

Find the starting time of the 'num' contention slot

4.32.3.63 double Mac802_16SS::timer_expiration (char, Packet *, int) [private]

4.32.3.64 Packet * Mac802_16SS::deque_pkt (char) [private]

4.32.3.65 Packet * Mac802_16SS::pkt_lookup (char *tbindex*, int *i*) [private]

4.32.3.66 void Mac802_16SS::dumpBWSS (char * *fileName*) [private]

4.32.3.67 void Mac802_16SS::dumpUGSJITTER (char *tbindex*, char * *outputFile*) [private]

This is called periodically (as of 3/26/05 it is piggybacked off the dumpBWSS mechanism). It computes a jitter statistic based on stats gathered since the last time this method was called. Each time a UGS allocation is made, the jitter is updated. This running total of jitter is simply the difference between the scheduled allocation time (based on the UGS grant interval) and the actual.

4.32.3.68 void Mac802_16SS::dumpFinalSSStats (char * *fileName*) [private]

4.32.3.69 void Mac802_16SS::dump802_16QueueStats (char * *outputFile*, int *channelBW*) [private]

Is called via a tcl script to dump the queue length of The first upstream flow in the table

4.32.3.70 void Mac802_16SS::dump802_16UtilStats (char * *fileName*, int *channelDSBW*, int *channelUSBW*) [private]

4.32.3.71 int Mac802_16SS::packetTrace (Packet * *p*, int *direction*) [private]

Parameters:

p A 802_16 frame.

direction direction is 0 when sent DS and 1 when arrival at US.

Returns:

0 on success.

4.32.3.72 int Mac802_16SS::timingsTrace (Packet * *p*, int *id*) [private]

Parameters:

p A 802_16 frame

id The ID that goes in the first field of the trace line

4.32.3.73 int Mac802_16SS::ugs_idle (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

tbindex denotes the service-flow entry on which packet has been mapped

4.32.3.74 int Mac802_16SS::ugs_decision (char, EventType, Packet *) [private]

4.32.3.75 int Mac802_16SS::ugs_tosend (char, EventType, Packet *) [private]

4.32.3.76 int Mac802_16SS::ugs_waitformap (char *tbindex*, EventType *e*, Packet * *p*) [private]

This is called when certain events occur while a UGS flow is waiting for a MAP.

4.32.3.77 int Mac802_16SS::rtpoll_idle (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

tbindex denotes the service-flow entry on which packet has been mapped

4.32.3.78 int Mac802_16SS::rtpoll_decision (char, EventType, Packet *) [private]

4.32.3.79 int Mac802_16SS::rtpoll_tosend (char, EventType, Packet *) [private]

4.32.3.80 int Mac802_16SS::rtpoll_waitformap (char, EventType, Packet *) [private]

4.32.3.81 int Mac802_16SS::rtpoll_tosendreq (char, EventType, Packet *) [private]

4.32.3.82 int Mac802_16SS::ertpoll_idle (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

tbindex denotes the service-flow entry on which packet has been mapped

4.32.3.83 int Mac802_16SS::ertpoll_decision (char, EventType, Packet *) [private]

4.32.3.84 int Mac802_16SS::ertpoll_tosend (char, EventType, Packet *) [private]

4.32.3.85 int Mac802_16SS::ertpoll_waitformap (char, EventType, Packet *) [private]

4.32.3.86 int Mac802_16SS::ertpoll_tosendreq (char, EventType, Packet *) [private]

4.32.3.87 int Mac802_16SS::nrtpoll_idle (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

tbindex denotes the service-flow entry on which packet has been mapped

4.32.3.88 int Mac802_16SS::nrtpoll_decision (char, EventType, Packet *) [private]

4.32.3.89 int Mac802_16SS::nrtpoll_tosend (char, EventType, Packet *) [private]

4.32.3.90 int Mac802_16SS::nrtpoll_waitformap (char, EventType, Packet *) [private]

4.32.3.91 int Mac802_16SS::nrtpoll_tosendreq (char, EventType, Packet *) [private]

4.32.3.92 int Mac802_16SS::nrtpoll_reqsent (char, EventType, Packet *) [private]

4.32.3.93 int Mac802_16SS::nrtpoll_contention (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

p The frame that already has bee adjusted for all headers

4.32.3.94 int Mac802_16SS::beffort_idle (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

tbindex denotes the service-flow entry on which packet has been mapped

4.32.3.95 int Mac802_16SS::beffort_decision (char, EventType, Packet *) [private]

4.32.3.96 int Mac802_16SS::beffort_tosend (char, EventType, Packet *) [private]

4.32.3.97 int Mac802_16SS::beffort_waitformap (char, EventType, Packet *) [private]

4.32.3.98 int Mac802_16SS::beffort_tosendreq (char, EventType, Packet *) [private]

4.32.3.99 int Mac802_16SS::beffort_reqsent (char, EventType, Packet *) [private]

4.32.3.100 int Mac802_16SS::beffort_contention (char *tbindex*, EventType *e*, Packet * *p*) [private]

Parameters:

p The frame that already has been adjusted for all headers

4.32.3.101 int Mac802_16SS::beffort_ratecheck (char *tbindex*, EventType *e*, Packet * *p*) [private]

This is called at before sending a request for bandwidth (including a piggy req) to see if the rate control has provided enough tokens. The dshdr_.mac_param contains the number of slots in the request for the case of a REQ_TIMER. Normally this is 1 or 6 for a piggy request. For the case of a PIGGY_REQ, the exthrd_[].eh_data[1] is set to # slots requested.

Parameters:

e Specifies the asynch event that led to this invocation: REQ_TIMER, PIGGYBACK_REQ

Returns:

If 1, then can send, if -1, then can NOT send

4.32.3.102 void Mac802_16SS::FillErtPSPiggyReq (char *tbindex*, Packet * *p*) [private]

This is called to send a request piggybacked to packet p. The request informs the BS the new packet size of the ertPS service.

4.32.3.103 int Mac802_16SS::SendErtPSReq (char *tbindex*, Packet * *p*) [private]

This is called when the ertPS service wants to send a request to the BS. It is invoked off the REQ_TIMER thread. The request will inform the new packet size to the BS.

Parameters:

p This is the current packet that we want to send.

Returns:

0 if an error occurs and the request is not sent, else a 1.

4.32.4 Friends And Related Function Documentation

4.32.4.1 friend class SSRng802_16Timer [friend]

4.32.4.2 friend class SSServiceFlowSendTimer [friend]

4.32.4.3 friend class SSServiceFlowRequestTimer [friend]

4.32.5 Member Data Documentation

4.32.5.1 RNG* Mac802_16SS::rng_ [private]

4.32.5.2 SSRng802_16Timer Mac802_16SS::mhSSRng_ [private]

4.32.5.3 SSServiceFlowSendTimer Mac802_16SS::mhSSSend_ [private]

4.32.5.4 SSServiceFlowRequestTimer Mac802_16SS::mhReq_ [private]

4.32.5.5 Packet* Mac802_16SS::map_ [private]

4.32.5.6 char Mac802_16SS::debug_ss [private]

4.32.5.7 u_int32_t Mac802_16SS::ss_id [private]

4.32.5.8 u_char Mac802_16SS::my_lan [private]

4.32.5.9 u_char Mac802_16SS::SizeUpFlowTable [private]

4.32.5.10 struct upstream_sflow Mac802_16SS::UpFlowTable[MAX_NUM_UPFLOWS_PERSS] [private]

4.32.5.11 u_char Mac802_16SS::SizeDownFlowTable [private]

4.32.5.12 struct downstream_sflow Mac802_16SS::DownFlowTable[MAX_NUM_DOWNFLOWS_PERSS] [private]

4.32.5.13 u_char Mac802_16SS::default_upstream_index_ [private]

4.32.5.14 u_char Mac802_16SS::default_dstream_index_ [private]

4.32.5.15 u_int16_t Mac802_16SS::priority [private]

4.32.5.16 double Mac802_16SS::rng_freq [private]

4.32.5.17 int Mac802_16SS::bs_addr [private]

4.32.5.18 double Mac802_16SS::map_acktime [private]

4.32.5.19 u_int32_t Mac802_16SS::total_num_frag [private]

4.32.5.20 u_int32_t Mac802_16SS::total_num_collisions [private]

4.32.5.21 u_int32_t Mac802_16SS::total_num_bkoff [private]

4.32.5.22 u_int32_t Mac802_16SS::total_collision_drops [private]

4.32.5.23 u_int32_t Mac802_16SS::total_queue_drops [private]

4.32.5.24 sptr Mac802_16SS::SndList [private]

4.32.5.25 rptr Mac802_16SS::ReqList [private]

- ns-2.28/mac/**mac-802_16.h**
- ns-2.28/mac/**mac-802_16-FSM.cc**
- ns-2.28/mac/**mac-802_16-ss.cc**

4.33 Mac802_16SSClass Class Reference

Public Member Functions

- **Mac802_16SSClass** ()
- TclObject * **create** (int, const char *const *)

4.33.1 Detailed Description

TCL Hooks for the simulator

4.33.2 Constructor & Destructor Documentation

4.33.2.1 Mac802_16SSClass::Mac802_16SSClass () [inline]

4.33.3 Member Function Documentation

4.33.3.1 TclObject* Mac802_16SSClass::create (int, const char *const *) [inline]

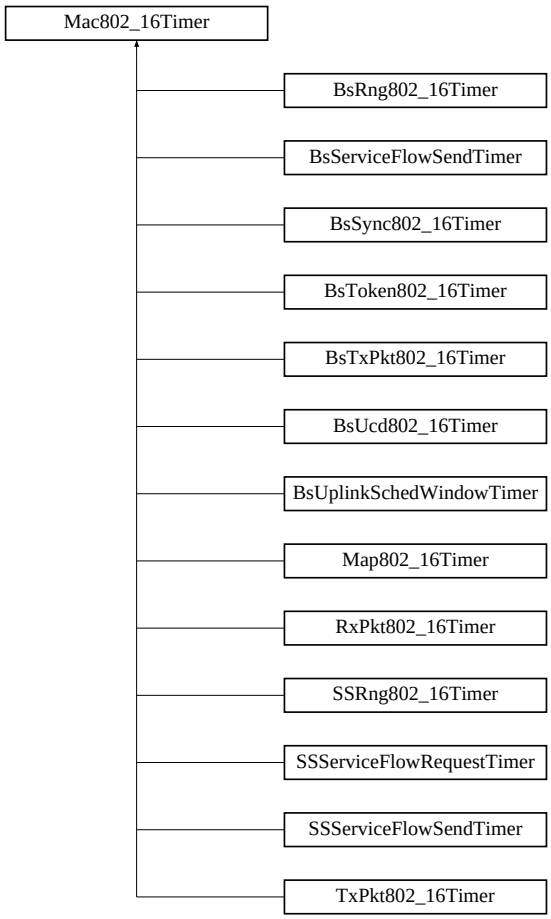
The documentation for this class was generated from the following file:

- ns-2.28/mac/**mac-802_16-ss.cc**

4.34 Mac802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for Mac802_16Timer::



Public Member Functions

- **Mac802_16Timer** (**Mac802_16** *m)
- virtual void **handle** (Event *e)
- void **start** (Packet *e, double time)
- virtual void **stop** (Packet *e)
- int **busy** (void)
- double **expire** (void)

Protected Attributes

- **Mac802_16** * **mac**
- int **busy_**
- int **paused_**
- Event **intr**

- double **stime**
- double **rtime**

4.34.1 Detailed Description

Base class for all the timer classes

4.34.2 Constructor & Destructor Documentation

4.34.2.1 Mac802_16Timer::Mac802_16Timer (Mac802_16 * *m*) [inline]

4.34.3 Member Function Documentation

4.34.3.1 virtual void Mac802_16Timer::handle (Event * *e*) [inline, virtual]

Reimplemented in **RxPkt802_16Timer** (p. 94), **TxPkt802_16Timer** (p. 101), **BsTxPkt802_16Timer** (p. 18), **Map802_16Timer** (p. 87), **BsUcd802_16Timer** (p. 19), **BsRng802_16Timer** (p. 14), **BsSync802_16Timer** (p. 16), **BsToken802_16Timer** (p. 17), **BsServiceFlowSendTimer** (p. 15), **BsUplinkSchedWindowTimer** (p. 20), **SSRng802_16Timer** (p. 97), **SSServiceFlowSendTimer** (p. 99), and **SSServiceFlowRequestTimer** (p. 98).

4.34.3.2 void Mac802_16Timer::start (Packet * *e*, double *time*)

4.34.3.3 void Mac802_16Timer::stop (Packet * *e*) [virtual]

4.34.3.4 int Mac802_16Timer::busy (void) [inline]

4.34.3.5 double Mac802_16Timer::expire (void) [inline]

4.34.4 Member Data Documentation

4.34.4.1 Mac802_16* Mac802_16Timer::mac [protected]

4.34.4.2 int Mac802_16Timer::busy_ [protected]

4.34.4.3 int Mac802_16Timer::paused_ [protected]

4.34.4.4 Event Mac802_16Timer::intr [protected]

4.34.4.5 double Mac802_16Timer::stime [protected]

4.34.4.6 double Mac802_16Timer::rtime [protected]

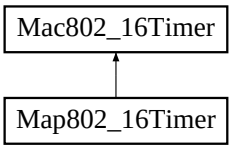
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.35 Map802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for Map802_16Timer::



Public Member Functions

- **Map802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.35.1 Detailed Description

Timers to schedule transmition of MAPS.

4.35.2 Constructor & Destructor Documentation

4.35.2.1 Map802_16Timer::Map802_16Timer (Mac802_16 * m) [inline]

4.35.3 Member Function Documentation

4.35.3.1 void Map802_16Timer::handle (Event * e) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.36 map_conf_param Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **time_covered**
- double **map_interval**
- u_int32_t **num_contention_slots**
- u_int32_t **num_sm_slots**
- u_char **bkoff_start**
- u_char **bkoff_end**

4.36.1 Member Data Documentation

4.36.1.1 double map_conf_param::time_covered

4.36.1.2 double map_conf_param::map_interval

4.36.1.3 u_int32_t map_conf_param::num_contention_slots

4.36.1.4 u_int32_t map_conf_param::num_sm_slots

4.36.1.5 u_char map_conf_param::bkoff_start

4.36.1.6 u_char map_conf_param::bkoff_end

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.37 map_list Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **alloc_stime**
- double **alloc_etime**
- double **release_time**
- double **deadline**
- u_int32_t **nslots**
- **mapptr next**
- u_int16_t **flow_id**
- u_int16_t **flag**

4.37.1 Member Data Documentation

4.37.1.1 double map_list::alloc_stime

4.37.1.2 double map_list::alloc_etime

4.37.1.3 double map_list::release_time

4.37.1.4 double map_list::deadline

4.37.1.5 u_int32_t map_list::nslots

4.37.1.6 mapptr map_list::next

4.37.1.7 u_int16_t map_list::flow_id

4.37.1.8 u_int16_t map_list::flag

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.38 mgmt_conf_param Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **sync_msg_interval**
- double **rng_msg_interval**
- double **ucd_msg_interval**

4.38.1 Detailed Description

The frequency of generation of management messages

4.38.2 Member Data Documentation

4.38.2.1 double mgmt_conf_param::sync_msg_interval

4.38.2.2 double mgmt_conf_param::rng_msg_interval

4.38.2.3 double mgmt_conf_param::ucd_msg_interval

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.39 pnode Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- Packet * **pkt**
- double **enq_time**
- **plist next**

4.39.1 Member Data Documentation

4.39.1.1 Packet* pnode::pkt

4.39.1.2 double pnode::enq_time

4.39.1.3 plist pnode::next

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.40 priority_array Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- int **priority**
- **jptr job_pointer**

4.40.1 Detailed Description

$A8 - auxiliar struct to compute the priority of the rtPS and nrtPS in the intermediate queue

4.40.2 Member Data Documentation

4.40.2.1 int priority_array::priority

4.40.2.2 jptr priority_array::job_pointer

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.41 req_timer_list Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **expiration_time**
- **rptr next**
- u_char **rindex**

4.41.1 Member Data Documentation

4.41.1.1 double req_timer_list::expiration_time

4.41.1.2 rptr req_timer_list::next

4.41.1.3 u_char req_timer_list::rindex

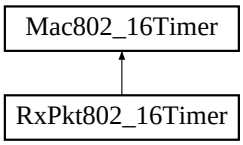
The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.42 RxPkt802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for RxPkt802_16Timer::



Public Member Functions

- **RxPkt802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.42.1 Detailed Description

Timers to control packet sending and receiving time.

4.42.2 Constructor & Destructor Documentation

4.42.2.1 RxPkt802_16Timer::RxPkt802_16Timer (Mac802_16 * m) [inline]

4.42.3 Member Function Documentation

4.42.3.1 void RxPkt802_16Timer::handle (Event * e) [virtual]

Receive Timer

Reimplemented from **Mac802_16Timer** (p. 86).

The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.43 snd_timer_list Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **expiration_time**
- **sptr next**
- u_char **tindex**

4.43.1 Member Data Documentation

4.43.1.1 double snd_timer_list::expiration_time

4.43.1.2 sptr snd_timer_list::next

4.43.1.3 u_char snd_timer_list::tindex

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.44 ss_record Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **downstream_flow_record d_rec** [MAX_NUM_DOWNFLOWS_PERSS]
- **up_flow_record u_rec** [MAX_NUM_UPFLOWS_PERSS]
- int **ss_macaddr**
- u_int16_t **priority**
- u_char **SizeDnFlTable**
- u_char **SizeUpFlTable**
- u_char **default_upstream_index_**
- u_char **default_downstream_index_**

4.44.1 Member Data Documentation

4.44.1.1 struct downstream_flow_record ss_record::d_rec[MAX_NUM_DOWNFLOWS_PERSS]

4.44.1.2 struct up_flow_record ss_record::u_rec[MAX_NUM_UPFLOWS_PERSS]

4.44.1.3 int ss_record::ss_macaddr

4.44.1.4 u_int16_t ss_record::priority

4.44.1.5 u_char ss_record::SizeDnFlTable

4.44.1.6 u_char ss_record::SizeUpFlTable

4.44.1.7 u_char ss_record::default_upstream_index_

4.44.1.8 u_char ss_record::default_downstream_index_

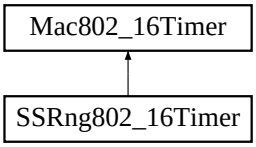
The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.45 SSRng802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for SSRng802_16Timer::



Public Member Functions

- **SSRng802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.45.1 Detailed Description

Timers to schedule transmition of Management messages

4.45.2 Constructor & Destructor Documentation

4.45.2.1 SSRng802_16Timer::SSRng802_16Timer (Mac802_16 * m) [inline]

4.45.3 Member Function Documentation

4.45.3.1 void SSRng802_16Timer::handle (Event * e) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

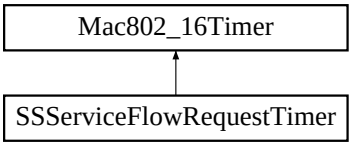
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.46 SSServiceFlowRequestTimer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for SSServiceFlowRequestTimer::



Public Member Functions

- **SSServiceFlowRequestTimer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.46.1 Detailed Description

REQUEST timer..(per service-flow timer)

4.46.2 Constructor & Destructor Documentation

4.46.2.1 SSServiceFlowRequestTimer::SSServiceFlowRequestTimer (Mac802_16 * *m*) [inline]

4.46.3 Member Function Documentation

4.46.3.1 void SSServiceFlowRequestTimer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

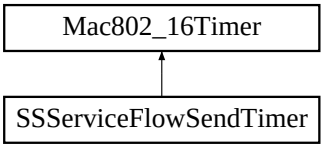
The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.47 SSServiceFlowSendTimer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for SSServiceFlowSendTimer::



Public Member Functions

- **SSServiceFlowSendTimer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.47.1 Detailed Description

SEND timer..(per service-flow timer)

4.47.2 Constructor & Destructor Documentation

4.47.2.1 SSServiceFlowSendTimer::SSServiceFlowSendTimer (Mac802_16 * *m*) [inline]

4.47.3 Member Function Documentation

4.47.3.1 void SSServiceFlowSendTimer::handle (Event * *e*) [virtual]

Reimplemented from **Mac802_16Timer** (p. 86).

The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.48 token_timer_list Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **expiration_time**
- **tkptr next**
- int **cindex**
- int **findex**

4.48.1 Member Data Documentation

4.48.1.1 double token_timer_list::expiration_time

4.48.1.2 tkptr token_timer_list::next

4.48.1.3 int token_timer_list::cindex

4.48.1.4 int token_timer_list::findex

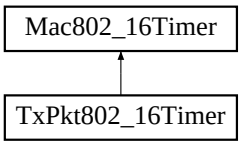
The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.49 TxPkt802_16Timer Class Reference

`#include <mac-802_16-timers.h>`

Inheritance diagram for TxPkt802_16Timer::



Public Member Functions

- **TxPkt802_16Timer** (**Mac802_16** *m)
- void **handle** (Event *e)

4.49.1 Constructor & Destructor Documentation

4.49.1.1 TxPkt802_16Timer::TxPkt802_16Timer (Mac802_16 * m) [inline]

4.49.2 Member Function Documentation

4.49.2.1 void TxPkt802_16Timer::handle (Event * e) [virtual]

Send Timer

Reimplemented from **Mac802_16Timer** (p. 86).

The documentation for this class was generated from the following files:

- ns-2.28/mac/**mac-802_16-timers.h**
- ns-2.28/mac/**mac-802_16-timers.cc**

4.50 up_flow_record Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **flow_classifier classifier**
- Packet * **frag_pkt**
- **SchedType sched_type**
- **PhsType PHS_profile**
- double **ginterval**
- u_int32_t **frag_data**
- u_int16_t **flow_id**
- u_int16_t **gsize**
- u_int16_t **seq_num**
- u_char **flag**
- double **latency**
- u_int32_t **min_bw**
- double **last_granttime**
- u_int16_t **backlogged**
- int16_t **granted_bw**

4.50.1 Detailed Description

This structure contains all the information excluding scheduling algorithm information about a SS at the BS node

4.50.2 Member Data Documentation

4.50.2.1 struct flow_classifier up_flow_record::classifier

4.50.2.2 Packet* up_flow_record::frag_pkt

4.50.2.3 SchedType up_flow_record::sched_type

4.50.2.4 PhsType up_flow_record::PHS_profile

4.50.2.5 double up_flow_record::ginterval

4.50.2.6 u_int32_t up_flow_record::frag_data

4.50.2.7 u_int16_t up_flow_record::flow_id

4.50.2.8 u_int16_t up_flow_record::gsize

4.50.2.9 u_int16_t up_flow_record::seq_num

4.50.2.10 u_char up_flow_record::flag

4.50.2.11 double up_flow_record::latency

4.50.2.12 u_int32_t up_flow_record::min_bw

4.50.2.13 double up_flow_record::last_granttime

4.50.2.14 u_int16_t up_flow_record::backlogged

4.50.2.15 int16_t up_flow_record::granted_bw

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.51 upstream_channel Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- double **data_rate**
- double **prop_delay**
- u_int32_t **max_burst_size**
- u_int32_t **overhead_bytes**
- u_char **physlots_p_minislot**

4.51.1 Detailed Description

This structure defines all the user-configurable paramters for upstream channel

4.51.2 Member Data Documentation

4.51.2.1 double upstream_channel::data_rate

4.51.2.2 double upstream_channel::prop_delay

4.51.2.3 u_int32_t upstream_channel::max_burst_size

4.51.2.4 u_int32_t upstream_channel::overhead_bytes

4.51.2.5 u_char upstream_channel::physlots_p_minislot

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.52 upstream_flow_record Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **flow_classifier classifier**
- **SchedType sched_type**
- **PhsType PHS_profile**
- double **ginterval**
- double **latency**
- u_int32_t **min_bw**
- u_int16_t **flow_id**
- u_int16_t **gsize**
- u_char **flag**

4.52.1 Detailed Description

This structure defines attributes of an upstream service flow of a SS node

4.52.2 Member Data Documentation

4.52.2.1 struct flow_classifier upstream_flow_record::classifier

4.52.2.2 SchedType upstream_flow_record::sched_type

4.52.2.3 PhsType upstream_flow_record::PHS_profile

4.52.2.4 double upstream_flow_record::ginterval

4.52.2.5 double upstream_flow_record::latency

4.52.2.6 u_int32_t upstream_flow_record::min_bw

4.52.2.7 u_int16_t upstream_flow_record::flow_id

4.52.2.8 u_int16_t upstream_flow_record::gsize

4.52.2.9 u_char upstream_flow_record::flag

The documentation for this struct was generated from the following file:

- ns-2.28/mac/**hdr-802_16.h**

4.53 upstream_sflow Struct Reference

`#include <hdr-802_16.h>`

Public Attributes

- **upstream_flow_record upstream_record**
- **allocation_time * alloc_list**
- **plist packet_list**
- u_int16_t **max_qsize**
- Packet * **pkt**
- Packet * **frag_pkt**
- char **debug**
- double **map_acktime**
- double **req_time**
- double **avg_queuing_delay**
- u_int32_t **queuing_samples**
- double **avg_req_stime**
- double **enque_time**
- u_int32_t **num_delay_samples**
- u_int32_t **curr_gsize**
- u_int32_t **num_pkt_snt**
- u_int32_t **drop_count**
- u_int32_t **seq_num**
- u_int32_t **frag_data**
- double **last_mfrtime**
- u_int32_t **avg_pkts**
- u_int32_t **avg_bytes**
- u_int32_t **num_bytes**
- u_int32_t **num_pkts**
- u_int32_t **SID_num_sent_bytes**
- u_int32_t **SID_num_sent_pkts**
- u_int32_t **avg_slotspermap**
- u_int32_t **totalACKs**
- u_int32_t **totalACKsFiltered**
- u_int32_t **total_piggyreq**
- u_int32_t **total_creq**
- u_int32_t **total_fcoll**
- u_int32_t **avg_fcont**
- u_int32_t **fcont_count**
- u_int32_t **total_collision_drops**
- u_int32_t **total_queue_drops**
- int **bs_addr**
- u_int16_t **max_concat_threshhold**
- u_char **state**
- u_int32_t **bk_offwin**
- u_int32_t **bk_offcounter**
- u_int32_t **bk_offend**
- u_int32_t **bk_offstart**
- u_char **num_retries**

- u_char **max_retries**
- u_char **contention_on**
- u_char **pending**
- Event **intr**
- Event **rintr**
- double **ugsjitter**
- int **jitterSamples**
- double **last_granttime**
- double **nominal_alloctime**
- double **last_jittercaltime**
- double **acceptance_rate**
- double **prev_acceptance_rate**
- float **wt_factor**
- u_int32_t **num_slots_req**
- double **tokens_**
- double **rate_**
- int **bucket_**
- double **lastupdatetime_**
- int **init_**
- char **ratecontrol**
- int **totalPacketsInConcatFrames**
- int **totalConcatFrames**
- int **not_requested_pkts**
- u_int16_t **aggreqinterval**
- u_int16_t **req_counter**

4.53.1 Member Data Documentation

4.53.1.1 struct upstream_flow_record upstream_sflow::upstream_record

4.53.1.2 struct allocation_time* upstream_sflow::alloc_list

4.53.1.3 plist upstream_sflow::packet_list

4.53.1.4 u_int16_t upstream_sflow::max_qsize

4.53.1.5 Packet* upstream_sflow::pkt

4.53.1.6 Packet* upstream_sflow::frag_pkt

4.53.1.7 char upstream_sflow::debug

4.53.1.8 double upstream_sflow::map_acktime

4.53.1.9 double upstream_sflow::req_time

4.53.1.10 double upstream_sflow::avg_queuing_delay

4.53.1.11 u_int32_t upstream_sflow::queuing_samples

4.53.1.12 double upstream_sflow::avg_req_stime

4.53.1.13 double upstream_sflow::enque_time

4.53.1.14 u_int32_t upstream_sflow::num_delay_samples

4.53.1.15 u_int32_t upstream_sflow::curr_gsize

4.53.1.16 u_int32_t upstream_sflow::num_pkt_snt

4.53.1.17 u_int32_t upstream_sflow::drop_count

4.53.1.18 u_int32_t upstream_sflow::seq_num

4.53.1.19 u_int32_t upstream_sflow::frag_data

4.53.1.20 double upstream_sflow::last_mfrtime

4.53.1.21 u_int32_t upstream_sflow::avg_pkts

4.53.1.22 u_int32_t upstream_sflow::avg_bytes

4.53.1.23 u_int32_t upstream_sflow::num_bytes

4.53.1.24 u_int32_t upstream_sflow::num_pkts

4.53.1.25 u_int32_t upstream_sflow::SID_num_sent_bytes

4.53.1.26 u_int32_t upstream_sflow::SID_num_sent_pkts

4.53.1.27 u_int32_t upstream_sflow::avg_slotspermap

4.53.1.28 u_int32_t upstream_sflow::totalACKs

4.53.1.29 u_int32_t upstream_sflow::totalACKsFiltered

4.53.1.30 u_int32_t upstream_sflow::total_piggyreq

- ns-2.28/mac/**hdr-802_16.h**

Chapter 5

WiMAX File Documentation

5.1 ns-2.28/mac/hdr-802_16.h File Reference

```
#include <arp.h>
#include <connector.h>
#include <delay.h>
#include <ll.h>
#include <mac.h>
#include "mac-802_16-timers.h"
#include <marshall.h>
#include <packet.h>
#include <queue.h>
#include <random.h>
```

Classes

- struct **mac802_16_frame_hdr**
- struct **mac802_16_extended_header_element**
- struct **hdr_mac802_16**
- struct **hdr_mac802_16extd**
- struct **hdr_mac802_16mgmt**
- struct **hdr_mac802_16map**

  802_16 map packet header

- struct **pnode**
- struct **flow_classifier**
- struct **allocation_time**
- struct **downstream_flow_record**
- struct **upstream_flow_record**
- struct **upstream_channel**
- struct **downstream_channel**
- struct **up_flow_record**

- struct **ss_record**
- struct **mgmt_conf_param**
- struct **map_conf_param**
- struct **bs_conf_param**
- struct **bs_statistics**
- struct **job**
- struct **map_list**
- struct **token_timer_list**
- struct **bsqnode**
- struct **bs_snd_timer_list**
- struct **active_flows**
- struct **priority_array**
- struct **snd_timer_list**
- struct **req_timer_list**
- struct **upstream_sflow**
- struct **downstream_sflow**

Defines

- #define **EPSILON** 0.0000000000005
- #define **UPSTREAM** 0
- #define **DOWNSTREAM** 1
- #define **OFF** 0
- #define **ON** 1
- #define **BROADCAST_SID** 1
- #define **DATA_PKT** 0
- #define **MGMT_PKT** 1
- #define **SUPRESS_ALL_SIZE** 100
- #define **SUPRESS_TCPIP_SIZE** 40
- #define **SUPRESS_UDPIP_SIZE** 28
- #define **SUPRESS_MAC_SIZE** 28
- #define **DATA_GRANT** 0
- #define **UREQ_GRANT** 1
- #define **CONTENTION_GRANT** 2
- #define **SM_GRANT** 3
- #define **ETHER_HDR_LEN** ((ETHER_ADDR_LEN << 1) + ETHER_TYPE_LEN)
- #define **MAC802_16_HDR_LEN** 6
- #define **MAX_NUM_IE** 240
- #define **MAX_GRANT** 255
- #define **MAX_RETRY** 10
- #define **SIZE_MGMT_HDR** 24
- #define **SIZE_MAP_HDR** 4
- #define **MANAGEMENT_MSG_PAYLOAD_SIZE** 5
- #define **NUM_EXTENDED_HDR_ELEMENTS** 2
- #define **MAX_NUM_DOWNFLOWS_PERSS** 10
- #define **MAX_NUM_UPFLOWS_PERSS** 10
- #define **NUM_802_16_LANS** 5
- #define **MAX_NUM_SS_NODES** 500
- #define **MAX_NUM_RETRIES** 15

- #define **FRAG_ENABLE_BIT** 8
- #define **CONCAT_ENABLE_BIT** 7
- #define **FRAG_ON_BIT** 6
- #define **CONCAT_ON_BIT** 5
- #define **PIGGY_ENABLE_BIT** 4
- #define **NO_PIGGY_BIT** 3
- #define **PIGGY_NOT_SEND** 2
- #define **CONCAT_THRESHHOLD** 1
- #define **QUEUE_THRESHOLD** 25
- #define **UGSSTATES** 4
- #define **RTPOLLSTATES** 6
- #define **ERTPOLLSTATES** 5
- #define **NRTPOLLSTATES** 7
- #define **BEFFORTSTATES** 8
- #define **DOWNSTATES** 4
- #define **UGS_IDLE** 0
- #define **UGS_DECISION** 1
- #define **UGS_TOSEND** 2
- #define **UGS_WAITFORMAP** 3
- #define **RTPOLL_IDLE** 0
- #define **RTPOLL_DECISION** 1
- #define **RTPOLL_TOSEND** 2
- #define **RTPOLL_WAITFORMAP** 3
- #define **RTPOLL_TOSENDREQ** 4
- #define **RTPOLL_REQSENT** 5
- #define **ERTPOLL_IDLE** 0
- #define **ERTPOLL_DECISION** 1
- #define **ERTPOLL_TOSEND** 2
- #define **ERTPOLL_WAITFORMAP** 3
- #define **ERTPOLL_TOSENDREQ** 4
- #define **NRTPOLL_IDLE** 0
- #define **NRTPOLL_DECISION** 1
- #define **NRTPOLL_TOSEND** 2
- #define **NRTPOLL_WAITFORMAP** 3
- #define **NRTPOLL_TOSENDREQ** 4
- #define **NRTPOLL_REQSENT** 5
- #define **NRTPOLL_CONTENTION** 6
- #define **BEFFORT_IDLE** 0
- #define **BEFFORT_DECISION** 1
- #define **BEFFORT_TOSEND** 2
- #define **BEFFORT_WAITFORMAP** 3
- #define **BEFFORT_TOSENDREQ** 4
- #define **BEFFORT_REQSENT** 5
- #define **BEFFORT_CONTENTION** 6
- #define **BEFFORT_RATECHECK** 7
- #define **DOWN_IDLE** 0
- #define **DOWN_DECISION** 1
- #define **DOWN_TOSEND** 2
- #define **DOWN_WAITFORMAP** 3
- #define **T16_TIMER** 1

Typedefs

- typedef **pnode** * **plist**
- typedef **allocation_time** * **aptr**
- typedef **job** * **jptr**
- typedef **map_list** * **mapptr**
- typedef **token_timer_list** * **tkptr**
- typedef **bsqnode** * **qlist**
- typedef **bs_snd_timer_list** * **bs_sptr**
- typedef **active_flows** * **acflows_ptr**
- typedef **priority_array** * **priority_array_p**
- typedef **snd_timer_list** * **sptr**
- typedef **req_timer_list** * **rptr**

Enumerations

- enum **SchedType** {

  UGS = 0x0000, **RT_POLL** = 0x0001, **ERT_POLL** = 0x0002, **NRT_POLL** = 0x0003,

  BEST_EFFORT = 0x0004 }
- enum **PhsType** {

  SUPRESS_ALL = 0x0000, **SUPRESS_TCPIP** = 0x0001, **SUPRESS_UDPIP** = 0x0002, **SUPRESS_MAC** = 0x0003,

  NO_SUPRESSION = 0x0004 }
- enum **EventType** {

  PKT_ARRIVAL = 0x0000, **MAP_ARRIVAL** = 0x0001, **SEND_TIMER** = 0x0002, **REQ_TIMER** = 0x0003,

  SEND_PKT = 0x0004, **SEND_UREQ** = 0x0006, **CONTENTION_ON** = 0x0007, **CONTENTION_SLOTS** = 0x0008,

  CONTENTION_BKOFF = 0x0009, **UNLOCK_QUEUE** = 0x00010, **PIGGYBACK_REQ** = 0x00011 }

5.1.1 Detailed Description

This file contains:

- `the` 802_16 frame formats (i.e., headers)
- `the` map structure
- `the` upstream flow table structure
- `All` global enum and defines needed by the code

5.1.2 Define Documentation

5.1.2.1 #define BEFFORT_CONTENTION 6

5.1.2.2 #define BEFFORT_DECISION 1

5.1.2.3 #define BEFFORT_IDLE 0

5.1.2.4 #define BEFFORT_RATECHECK 7

5.1.2.5 #define BEFFORT_REQSENT 5

5.1.2.6 #define BEFFORT_TOSEND 2

5.1.2.7 #define BEFFORT_TOSENDREQ 4

5.1.2.8 #define BEFFORT_WAITFORMAP 3

5.1.2.9 #define BEFFORTSTATES 8

5.1.2.10 #define BROADCAST_SID 1

5.1.2.11 #define CONCAT_ENABLE_BIT 7

5.1.2.12 #define CONCAT_ON_BIT 5

5.1.2.13 #define CONCAT_THRESHHOLD 1

5.1.2.14 #define CONTENTION_GRANT 2

5.1.2.15 #define DATA_GRANT 0

5.1.2.16 #define DATA_PKT 0

5.1.2.17 #define DOWN_DECISION 1

5.1.2.18 #define DOWN_IDLE 0

5.1.2.19 #define DOWN_TOSEND 2

5.1.2.20 #define DOWN_WAITFORMAP 3

5.1.2.21 #define DOWNSTATES 4

5.1.2.22 #define DOWNSTREAM 1

5.1.2.23 #define EPSILON 0.0000000000005

5.1.2.24 #define ERTPOLL_DECISION 1

5.1.2.25 #define ERTPOLL_IDLE 0

5.1.2.26 #define ERTPOLL_TOSEND 2

5.1.2.27 #define ERTPOLL_TOSENDREQ 4

5.1.2.28 #define ERTPOLL_WAITFORMAP 3

5.1.2.29 #define ERTPOLLSTATES 5

5.1.2.30 #define ETHER_HDR_LEN ((ETHER_ADDR_LEN << 1) + ETHER_TYPE_LEN)

5.1.3.2 typedef struct allocation_time* aptr

5.1.3.3 typedef struct bs_snd_timer_list* bs_sptr

5.1.3.4 typedef struct job* jptr

Scheduling algorithm data structures

5.1.3.5 typedef struct map_list* mapptr

5.1.3.6 typedef struct pnode* plist

5.1.3.7 typedef struct priority_array* priority_array_p

5.1.3.8 typedef struct bsqnode* qlist

Packet queue structure

5.1.3.9 typedef struct req_timer_list* rptr

5.1.3.10 typedef struct snd_timer_list* sptr

5.1.3.11 typedef struct token_timer_list* tkptr

List of token timer requests

5.1.4 Enumeration Type Documentation

5.1.4.1 enum EventType

Enumerator:

PKT_ARRIVAL

MAP_ARRIVAL

SEND_TIMER

REQ_TIMER

SEND_PKT

SEND_UREQ

CONTENTION_ON

CONTENTION_SLOTS

CONTENTION_BKOFF

UNLOCK_QUEUE

PIGGYBACK_REQ

5.1.4.2 enum PhsType

Enumerator:

SUPRESS_ALL
SUPRESS_TCPIP
SUPRESS_UDPIP
SUPRESS_MAC
NO_SUPRESSION

5.1.4.3 enum SchedType

Enumerator:

UGS
RT_POLL
ERT_POLL
NRT_POLL
BEST_EFFORT

5.2 ns-2.28/mac/mac-802_16-base.cc File Reference

```
#include "mac-802_16.h"
#include "mobilenode.h"
```

5.2.1 Detailed Description

This file contains methods for the base **Mac802_16** (p. 36) class.

5.3 ns-2.28/mac/mac-802_16-bs.cc File Reference

```
#include "mac-802_16.h"
#include "random.h"
#include "math.h"
#include "ping.h"
```

Classes

- class **Mac802_16EHeaderClass**
- class **Mac802_16HeaderClass**
- class **Mac802_16MHeaderClass**
- class **Mac802_16MapHeaderClass**
- class **Mac802_16BsClass**

Defines

- #define **FLOW_ID** 0
- #define **SS_ID** 0
- #define **DUMPBSMAP** 0
- #define **FILLWITHCONTOPS** 0
- #define **SHORT_STATS** 1

Functions

- int **compare_priority** (const void *a, const void *b)

Variables

- int **lan_num**
- **Mac802_16BS** * **bs_arr** [NUM_802_16_LANS]
- **Mac802_16EHeaderClass class_hdr_mac802_16extd**
- **Mac802_16HeaderClass class_hdr_mac802_16**
- **Mac802_16MHeaderClass class_hdr_mac802_16mgmt**
- **Mac802_16MapHeaderClass class_hdr_mac802_16map**
- **Mac802_16BsClass class_mac_802_16bs**
- **Mac802_16BS** * **global_compare_obj**

5.3.1 Detailed Description

This file contains the BS object class method implementations.

5.3.2 Define Documentation

5.3.2.1 #define DUMPBSMAP 0

5.3.2.2 #define FILLWITHCONTOPS 0

5.3.2.3 #define FLOW_ID 0

5.3.2.4 #define SHORT_STATS 1

5.3.2.5 #define SS_ID 0

5.3.3 Function Documentation

5.3.3.1 int compare_priority (const void * *a*, const void * *b*)

5.3.4 Variable Documentation

5.3.4.1 Mac802_16BS* bs_arr[NUM_802_16_LANS]

5.3.4.2 Mac802_16HeaderClass class_hdr_mac802_16 [static]

Mac802_16HeaderClass (p. 65) class

5.3.4.3 Mac802_16EHeaderClass class_hdr_mac802_16extd [static]

Mac802_16EHeaderClass (p. 64) class

5.3.4.4 Mac802_16MapHeaderClass class_hdr_mac802_16map [static]

Mac802_16MHeaderClass (p. 67) class

5.3.4.5 Mac802_16MHeaderClass class_hdr_mac802_16mgmt [static]

Mac802_16MHeaderClass (p. 67) class

5.3.4.6 Mac802_16BsClass class_mac_802_16bs [static]

TCL Hooks for the simulator

5.3.4.7 Mac802_16BS* global_compare_obj

5.3.4.8 int lan_num

5.4 ns-2.28/mac/mac-802_16-FSM.cc File Reference

```
#include "mac-802_16.h"
```

Defines

- #define **ACK_SUPPRESSION** 0
- #define **DUMPSSMAP** 0
- #define **DUMPSSID** 0
- #define **US_RATE_CONTROL** 1

5.4.1 Detailed Description

This file contains implementation of state machines for the three types of service-flows

5.4.2 Define Documentation

5.4.2.1 #define ACK_SUPPRESSION 0

5.4.2.2 #define DUMPSSID 0

5.4.2.3 #define DUMPSSMAP 0

5.4.2.4 #define US_RATE_CONTROL 1

5.5 ns-2.28/mac/mac-802_16-ss.cc File Reference

```
#include "mac-802_16.h"
#include "random.h"
#include <stdio.h>
#include "ping.h"
```

Classes

- class **Mac802_16SSClass**

Defines

- #define **US_RATE_CONTROL** 1
- #define **SHORT_STATS** 1
- #define **SET_RX_STATE**(x)
- #define **SET_TX_STATE**(x)

Variables

- int **lan_num**
- **Mac802_16BS** * **bs_arr** [NUM_802_16_LANS]
- **Mac802_16SSClass class_mac_802_16ss**

5.5.1 Detailed Description

This file contains the SS object class method implementations

5.5.2 Define Documentation

5.5.2.1 #define SET_RX_STATE(x)

Value:

```
{                                       \
        rx_state_ = (x);                        \
}
```

5.5.2.2 #define SET_TX_STATE(x)

Value:

```
{                                               \
        tx_state_ = (x);                                \
}
```

5.5.2.3 #define SHORT_STATS 1

5.5.2.4 #define US_RATE_CONTROL 1

5.5.3 Variable Documentation

5.5.3.1 Mac802_16BS* bs_arr[NUM_802_16_LANS]

5.5.3.2 Mac802_16SSClass class_mac_802_16ss [static]

TCL Hooks for the simulator

5.5.3.3 int lan_num

5.6 ns-2.28/mac/mac-802_16-timers.cc File Reference

```
#include "mac-802_16.h"
#include "mac-802_16-timers.h"
```

5.6.1 Detailed Description

This file contains c++ methods for implementing various timer classes

5.7 ns-2.28/mac/mac-802_16-timers.h File Reference

Classes

- class **Mac802_16Timer**
- class **RxPkt802_16Timer**
- class **TxPkt802_16Timer**
- class **BsTxPkt802_16Timer**
- class **Map802_16Timer**
- class **BsUcd802_16Timer**
- class **BsRng802_16Timer**
- class **BsSync802_16Timer**
- class **BsToken802_16Timer**
- class **BsServiceFlowSendTimer**
- class **BsUplinkSchedWindowTimer**
- class **SSRng802_16Timer**
- class **SSServiceFlowSendTimer**
- class **SSServiceFlowRequestTimer**

5.7.1 Detailed Description

This file contains Timer class definitions.

5.8 ns-2.28/mac/mac-802_16.h File Reference

```
#include "hdr-802_16.h"
```

Classes

- class **Mac802_16**
- class **Mac802_16BS**
- class **Mac802_16SS**

Defines

- #define **SET_RX_STATE**(x)
- #define **SET_TX_STATE**(x)
- #define **UP_HIGH_QUEUE** 0
- #define **UP_INTERM_QUEUE** 1
- #define **UP_LOW_QUEUE** 2
- #define **DOWN_UGS_QUEUE** 3
- #define **DOWN_RT_QUEUE** 4
- #define **DOWN_ERT_QUEUE** 5
- #define **DOWN_NRT_QUEUE** 6
- #define **DOWN_BE_QUEUE** 7

5.8.1 Detailed Description

This file contains SS and BS class definitions. Both of these classes are derived from an abstract base class

5.8.2 Define Documentation

5.8.2.1 #define DOWN_BE_QUEUE 7

5.8.2.2 #define DOWN_ERT_QUEUE 5

5.8.2.3 #define DOWN_NRT_QUEUE 6

5.8.2.4 #define DOWN_RT_QUEUE 4

5.8.2.5 #define DOWN_UGS_QUEUE 3

5.8.2.6 #define SET_RX_STATE(x)

Value:

```
{                                       \
    rx_state_ = (x);                    \
  }
```

5.8.2.7 #define SET_TX_STATE(x)

Value:

```
{                                                   \
    tx_state_ = (x);                                \
  }
```

5.8.2.8 #define UP_HIGH_QUEUE 0

5.8.2.9 #define UP_INTERM_QUEUE 1

5.8.2.10 #define UP_LOW_QUEUE 2

Index