### 64.3.3 MPCP Theory of operation

#### 64.3.3.1 MPCP Gate Process

(1) OLT

- Normal case

At the OLT, the WAIT state is the initial state of the Gate Process. In this state, the Gate Process waits for the MA_CONTROL.request primitive from the client or the Discovery Process. The MA_CONTROL.request primitive contains the contents of GATE message to be sent. When the Gate Process receives the MA_CONTROL.request primitive, it transits to the SEND GATE state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the GATE message and starts the periodic_timer. Then, it transits to the WAIT state. This is the normal case.

- Expiration of periodic_timer

The Gate Process uses the periodic_timer to send the GATE messages at certain intervals. The Gate Process starts this timer whenever it issues the OMP.request primitive indicating the GATE message. At the expiration of periodic timer in the WAIT state, the Gate Process transits to the PERIODIC TRANSMISSION state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the GATE message with no grant and starts the periodic_timer. Then, it transits to the WAIT state.

- Completion of Discovery procedure

At the completion of Discovery procedure, the Gate Process transits to the PERIODIC TRANSMISSION state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the GATE message with no grant and starts the periodic_timer. Then, it transits to the WAIT state.

(2) ONU

The ONU maintains two states in the Gate Process. One is the Programming state, another is the Activation state.

Programming state

The WAIT state is the initial state of the Programming state. In this state, the Gate Process waits for the OMP.indication primitive from the OMP Parser. The OMP.indication primitive contains the contents of GATE message received from the OLT. When the Gate Process receives the OMP.indication primitive, it transits to the INCOMING GRANT state. In this state, it delineates maximum four grants from the primitive, issues the MA_CONTROL.indication primitive to inform the client of grants and adds these grants to the list of grants. In this list, grants which will be treated in the future are listed. If the specified start time of the delineated grant is the past, this grant is ignored. Then, it transits to the WAIT state.

Activation state

- Normal case

The SORT state is the initial state of the Activation state. In this state, it gets the grant with the smallest start_time value from the list and begins to treat this grant. If the length of the grant is enough, it transits to the SET START TIMER state. In this state, the Gate Process starts the grant_start_timer to detect the beginning of grant. After this timer expires, it transits to the TURN LASER ON state. In this state, it makes the laser on, removes the grant from the list and starts the IDLE_timer to detect the end of the IDLE period. After this timer expires, it transits to the START

TX state. In this state, it enables the transmission of frames, issues the MA_CONTROL.indication primitive to inform the client of the effective grant and starts the grant_window_timer to detect the end of effective grant. After this timer expires, it transits to the STOP TX state. In this state, it makes laser off, disables the transmission of frames and issues the MA_CONTROL.indication primitive to inform the client of the end of effective grant. Then it transits to the SORT state and begins to treat the next grant in the list.

- Too short length of grant

If the length of the grant is too short in the SORT state, the Gate Process transits to the REMOVE LIST state. In this state, it removes this grant from the list and transits to the SORT state.

- No grant

While the list of grants is empty in the SORT state, the Gate Process stays in the SORT state.

#### 64.3.3.2 MPCP Discovery Process

(1) OLT

The OLT maintains three states in the Discovery Process. They are the Window Setup state, the Process Request state and the Final Registration state.

Window Setup state

The IDLE state is the initial state of the Window Setup state. In this state, the Discovery Process waits for the MA_CONTROL.request primitive from the client. The MA_CONTROL.request primitive contains the discovery grant and the length of discovery window. When the Discovery Process receives the MA_CONTROL.request primitive, it transits to the SEND REGISTER WINDOW state. In this state, it issues the MA_CONTROL.request primitive to the Gate Process to send the Discovery GATE message and starts the wait_for_window timer to detect the beginning of the discovery window. After this timer expires, it transits to the INSIDE REGISTER WINDOW state. In this state, it starts the register_window_timer to detect the end of the discovery window. After this timer expires, it transits to the IDLE state.

Process Request state

The IDLE state is the initial state of the Process Request state. At the beginning of the discovery window, the Discovery Process transits to the ACCEPT REGISTER REQUEST state. When it receives the OMP.indication primitive indicating the REGISTER_REQ message in this state, it transits to the SIGNAL state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of the receipt of the REGISTER_REQ message. Then, it transits to the ACCEPT REGISTER REQUEST state. At the end of the discovery window, it transits to the IDLE state.

Final Registration state

- Normal registration

The ILDE state is the initial state of the Final Registration state. When the Discovery Process receives the MA_CONTROL.request primitive indicating the REGISTER message in this state, it transits to the REGISTER state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the REGISTER message with the success flag. Then, it transits to the WAIT for REGISTER_ACK state. In this state, it starts the ONU_timer to monitor the REGISTER_ACK message from the ONU. If it receives the OMP.indication primitive indicating the REGISTER_ACK message from the OMP Parser before the expiration of the ONU_timer, it transits to the

COMPLETED DISCOVERY state. In the COMPLETE DISCOVERY state, it checks the flag in the REGISTER_ACK message. If the flag indicates the success, the Discovery Process transits to the REGISTERED state. In the REGISTERED state, it issues the MA_CONTROL.indication primitive to inform the client of the success of registration.

- Rejection of requested registration

When the Discovery Process rejects the requested registration, it issues the OMP.request primitive to the OMP Multiplexer to send the REGISTER message with the nack flag. Then, it transits to the IDLE state.

- Expiration of ONU_timer

If the ONU_timer expires in the WAIT for REGISTER_ACK state, the Discovery Process transits to the DEREGISTER state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of the failure of registration and issues the OMP.request primitive to send the REGISTER message with the de-register flag. Then, it transits to the IDLE state.

- REGISTER_ACK with failure flag

If the flag in the REGISTER_ACK message indicates the failure, the Discovery Process transits to the DEREGISTER state. The sequential behavior is same as the expiration of ONU_timer case.

- De-registration from ONU

When the Discovery Process receives the MA_CONTROL.request primitive requesting de-registration in the REGISTERED state, it transits to the DEREGISTER state. The sequential behavior is same as the expiration of ONU_timer case.

- Expiration of omp_timer

When the omp_timer expires in the REGISTERED state, the Discovery Process transits to the DEREGISTER state. The sequential behavior is same as the expiration of ONU_timer case.

- De-registration from OLT

When the Discovery Process receives the OMP.indication primitive indicating the REGISTER_REQ message with the de-register flag, it transits to the DEREGISTER state. The sequential behavior is same as the expiration of ONU_timer case.

- Re-registration

When the Discovery Process receives the MA_CONTROL.request primitive requesting the re-registration, it issues the OMP.request primitive to the OMP Multiplexer to send the REGISTER message with the re-registration flag. Then, it transits to the WAIT for REGISTER_ACK state. The sequential behavior is same as the normal registration case.

(2) ONU

The ONU maintains two states in the Discovery Process. One is the Window Setup state, another is the Process state.

Window Setup state

- Broadcast DA

The WAIT state is the initial state of the Window Setup state. In this state, the Discovery Process disables the transmission of frames and makes the laser off. When the Discovery Process receives the OMP.indication primitive from the OMP Parser indicating the GATE message with the discovery flag, it transits to the CHECH UNICAST state. In this state, it checks the DA. If the DA is not the unicast address, it transits to the WAIT FOR WINDOW state. In

this state, it starts the wait_for_window_timer to detect the beginning of discovery grant. After this timer expires, it transits to the RANDOM WAIT state. In the RANDOM WAIT state, it starts the random_delay_timer to perform the random delay process. After this timer expires, it transits to the TURN LASER ON state. In this state, it makes the laser on and starts the IDLE_timer to detect the beginning of the effective grant. After this timer expires, it transits to the INSIDE REGISTER WINDOW state. In this state, it enables the transmission of frames and starts the grant_window_timer to detect the end of the effective grant. After this timer expires, it transits to the WAIT state.

- Unicast DA

If the DA in the OMP.indication primitive is the unicast address, Discovery Process transits to the WAIT for WINDOW UNICAST state. In this state, it starts the wait_for_window_timer to detect the beginning of discovery grant. After this timer expires, it transits to the TURN LASER ON state. The sequential behavior is same as the broadcast DA case.

(Note) “MA_CONTROL.request()” just below the WAIT state is replaced with “OMP.indication()”.

Process state

- Normal registration

The WAIT state is the initial state of the Process state. When the Discovery Process receives the MA_CONTROL.request primitive requesting the registration from the client, it transits to the REGISTERING state. At the beginning of the effective grant, it transits to the REGISTER_REQ state. In this state, it issues the OMP.request primitive indicating the REGISTER_REQ message to the OMP Multiplexer and starts the wait_for_register_msg_timer to monitor the REGISTER message from the OLT. If it receives the OMP.indication primitive indicating the REGISTER message with the success flag before the expiration of the wait_for_register_msg_timer, it transits to the ACK state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of the acceptance of registration and issues the OMP.request primitive indicating the REGISTER_ACK message with the success flag to the OMP Multiplexer. Then, it transits to the REGISTERED state.

- Rejection of requested registration

In the REGISTER_REQ state, if the Discovery Process receives the OMP.indication primitive indicating the REGISTER message with the nack flag before the expiration of the wait_for_register_msg_timer, it transits to the NACK state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of the rejection of registration and issues the OMP.request primitive indicating the REGISTER_ACK message with the failure flag to the OMP Multiplexer. Then, it transits to the WAIT state.

- Expiration of wait_for_register_msg_timer

If the wait_for_register_msg_timer expires in the REGISTER_REQ state, the Discovery Process transits to the TIMEOUT state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of the retry of registration. Then, it transits to the REGISTERING state.

- Re-registration

When the Discovery Process receives the OMP.indication primitive indicating the REGISTER message with the re-registration flag in the REGISTERED state, it transits to the ACK state. The sequential behavior is same as the normal registration case.

- De-registration from OLT

When the Discovery Process receives the OMP.indication primitive indicating the REGISTER message with the

de-registration flag in the REGISTERED state, it transits to the REMOTE DEREGISTER state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of de-registration. Then, it transits to the WAIT state.

- De-registration from ONU

When the Discovery Process receives the MA_CONTROL.request primitive requesting de-registration from the client in the REGISTERED state, it transits to the LOCAL DEREGISTER state. In this state, it issues the OMP.request primitive indicating the REGISTER_REQ message with the de-registration flag to the OMP Multiplexer and issues the MA_CONTROL.indication primitive to inform the client of de-registration. Then, it transits to the WAIT state.

- Expiration of omp_timer

At the expiration of omp_timer, the Discovery Process transits to the LOCAL DEREGISTER state. The sequential behavior is same as de-registration from ONU case.

#### 64.3.3.3 Discovery Message Handshake

The OLT starts the discovery procedure by sending the GATE message with the discovery flag. This message is called as Discovery GATE and will be frequently sent to discover additional unregistered ONUs. The Discovery GATE message contains only one grant which may have long length to register multiple ONUs. Additionally, this message contains the AGC settling time and the CDR lock time. After sending the Discovery GATE message, the OLT opens the discovery window based on the assigned grant. When an unregistered ONU receives the Discovery GATE message, it invokes a timer at the beginning of grant. This timer means the random delay which is used to avoid the collision of registrations from multiple ONUs. After this timer expires, the ONU sends the REGISTER_REQUEST message. This message contains the pending grants. If the OLT receives the REGISTER_REQUEST message without collision during the discovery window, it selects one of unassigned LLIDs, assigns the selected LLID to the ONU which sent the REGISTER_REQUEST message, and sends the REGISTER message. This message contains the Assigned port, the AGC settling time, the CDR lock time and the echo of pending grants. The selected LLID is included in the Assigned port field. Additionally, the OLT sends the GATE message without the discovery flag to solicit the acknowledge message from the ONU. This message is called as Normal GATE, and the LLID is indicated in the MAC preamble. If the ONU receives the REGISTER message, it recognizes that the logical link has been correctly established. The LLID indicated in the Assigned port field in the REGISTER message is assigned to this logical link. After that, if the ONU receives the Normal GATE message, it sends the REGISTER_ACK message using the specified grant. The LLID is indicated in the MAC preamble. When the OLT receives the REGISTER_ACK message, it completes the discovery procedure and begins to send the Normal GATE messages to solicit the data transmission.

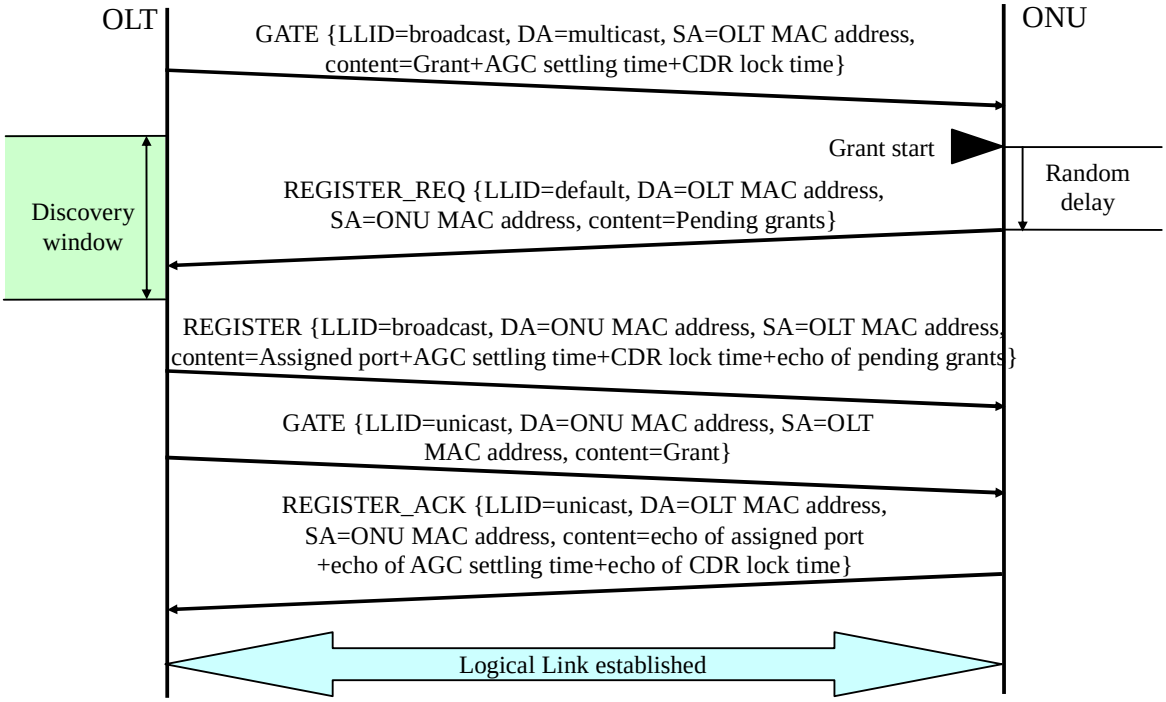


**Figure 64-xx – Discovery Message Handshake**

#### 64.3.3.4 Ranging and Timing Process

The OLT has a 32 bits counter. This counter is incremented every 16 bit transmission. When the OLT transmits MPCPDUs, it maps the counter value in the timestamp field.

The ONU also has a 32 bit counter. This counter is also incremented every 16 bit transmission. Additionally, this counter is used to set the value of timestamp field whenever the ONU receives MPCPDUs. When the ONU transmits MPCPDUs, it maps the counter value minus the processing delay in the timestamp field.

The OLT measures the round trip time of ONU whenever it receives MPCPDUs. The RTT is equal to the difference between the counter value and the value in the timestamp field. Since the ONU sets the counter value minus the processing delay in the timestamp filed, the processing delay is absorbed in the RTT at the OLT.

The calculated RTT is notified to the client via the MA_CONTROL.indication primitive. The client can use this RTT for the ranging process.

| It is better that the timing chart that is shown in slide 3 of hirth_p2mp_1_0103.pdf is inserted here. |
|---|

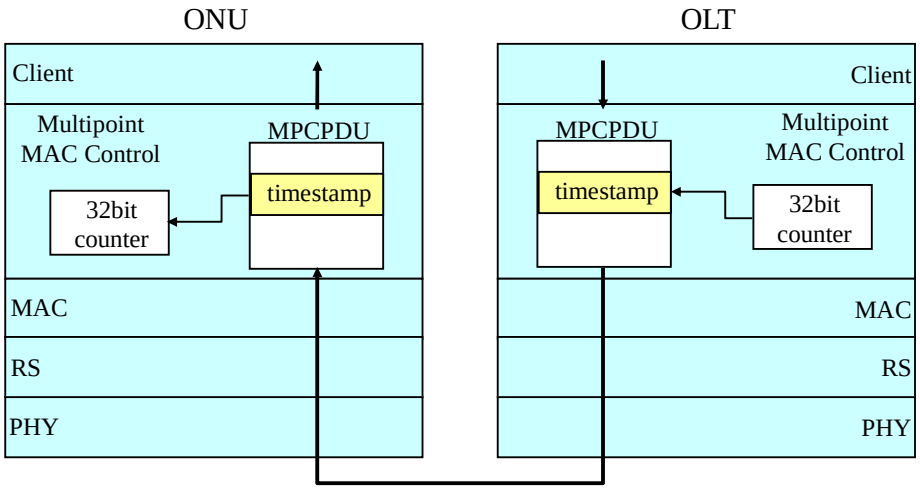


**Figure 64-xx – MPCPDU from OLT to ONU**

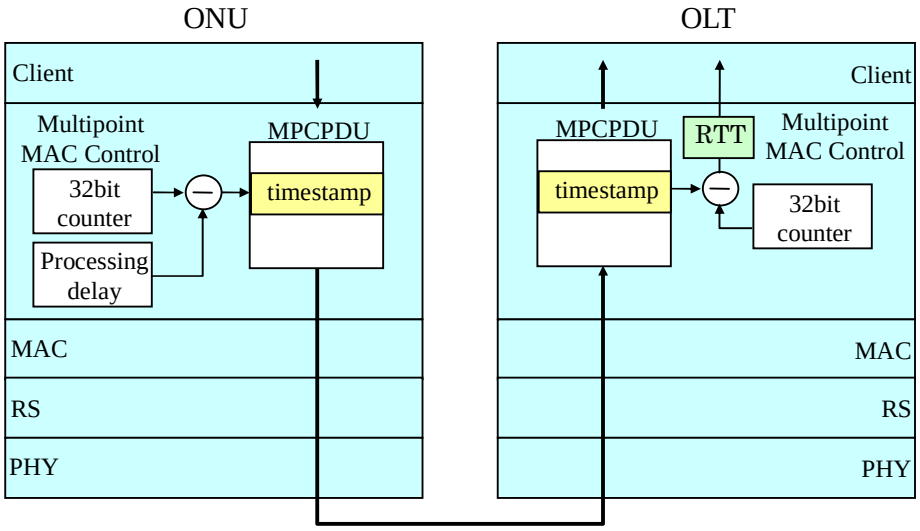


**Figure 64-xx – MPCPDU from ONU to OLT**

### 64.3.3.5 MPCP Report Process

(1) OLT

At the OLT, the WAIT state is the initial state of the Report Process. In this state, the Report Process waits for the OMP.indication primitive from the OMP Parser. The OMP.indication primitive contains the contents of REPORT message received from the ONU. When the Report Process receives the OMP.indication primitive, it transits to the RECEIVE REPORT state. In this state, it issues the MA_CONTROL.indication primitive to inform the client of reports information and the calculated RTT. Then, it transits to the WAIT state.

(2) ONU

- Normal case

At the ONU, the WAIT state is the initial state of the Report Process. In this state, the Report Process waits for the

MA_CONTROL.request primitive from the client. The MA_CONTROL.request primitive contains the contents of REPORT message to be sent. When the Report Process receives the MA_CONTROL.request primitive, it transits to the SEND REPORT state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the REPORT message and starts the periodic_timer. Then, it transits to the WAIT state.

- Expiration of periodic_timer

The Report Process uses the periodic_timer to send the REPORT messages at certain intervals. The Report Process starts this timer whenever it issues the OMP.request primitive indicating the REPORT message. At the expiration of periodic_timer in the WAIT state, the Report Process transits to the PERIODIC TRANSMISSION state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the REPORT message that contains no queue report and starts the periodic_timer. Then, it transits to the WAIT state.

- Completion of Discovery procedure

At the completion of Discovery procedure, the Report process transits to the PERIODIC TRANSMISSION state. In this state, it issues the OMP.request primitive to the OMP Multiplexer to send the REPORT message that contains no queue report and starts the periodic_timer. Then, it transits to the WAIT state.