

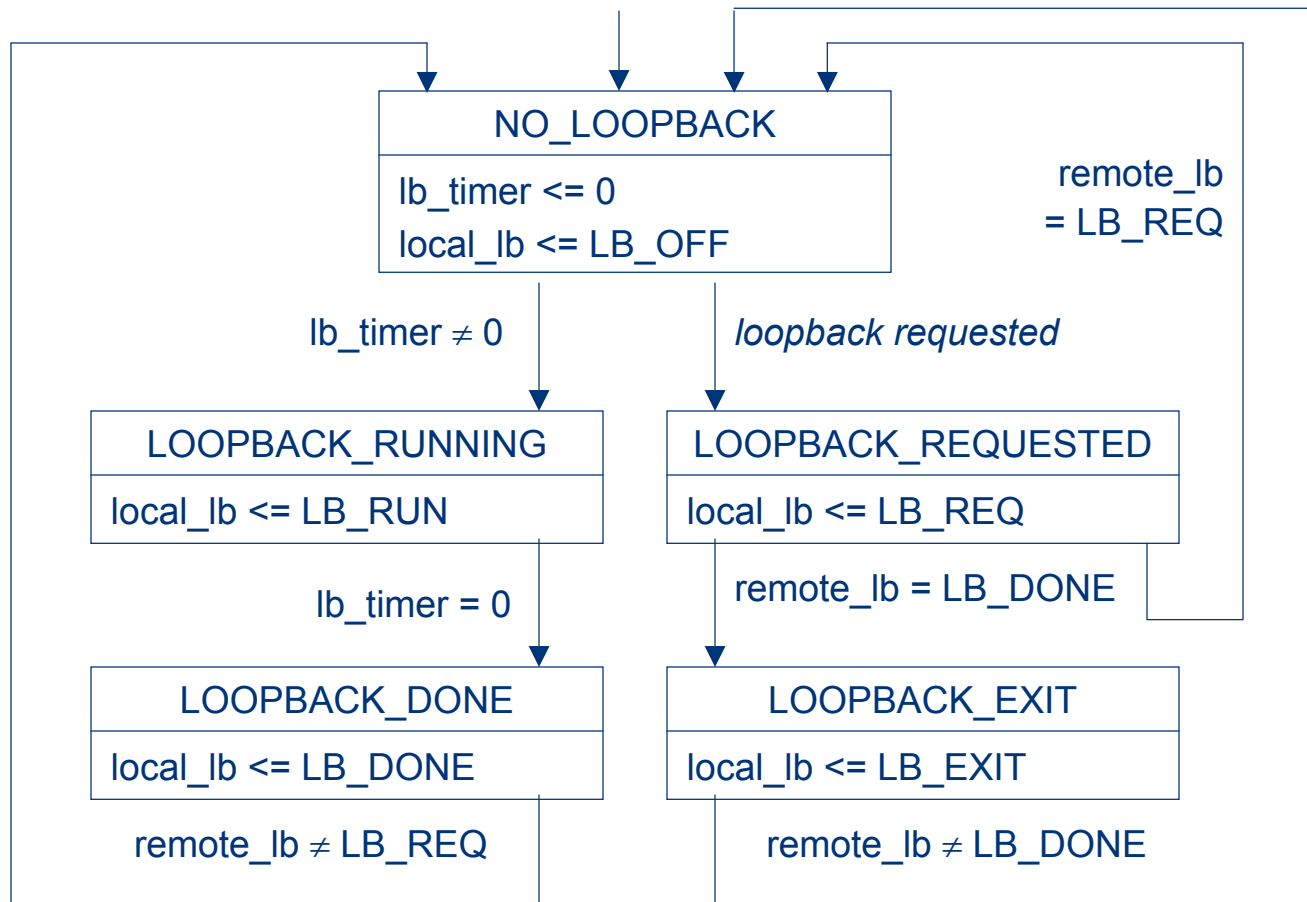
Loopback

Loopback state diagram

- **Following slides show**
 - **Loopback state diagram**
 - **Parser action during each state**
 - **Observations about two approaches**
 - **Suggested pros/cons about each**
 - **Recommendation**
 - **Proposed change to Parser state diagram**

[D1.2] Fig 55-7 Loopback state diagram

BEGIN + oam_enable=FALSE + lost_link_timer_done
 + satisfied=FALSE + link_status=FALSE +
 remote_stable=UNSTABLE



[D1.2] Fig 55-7 Loopback state diagram

BEGIN + oam_enable=FALSE + lost_link_timer_done
 + satisfied=FALSE + link_status=FALSE +
 remote_stable=UNSTABLE

Remote parser

Local parser

forward

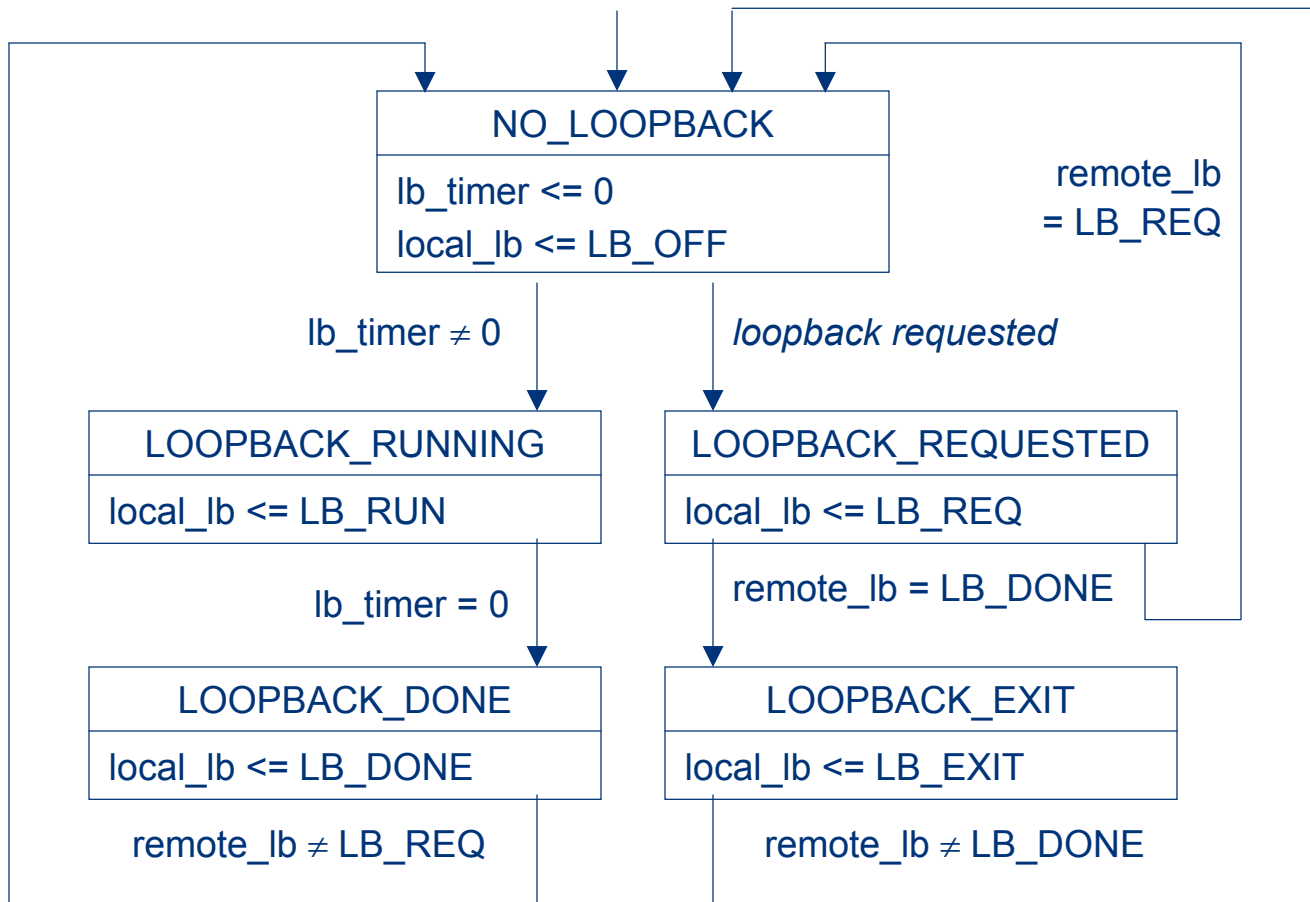
forward

loopback

discard

discard

discard



Loopback observations

- **At least two approaches exist to implement/specify loopback**
 - **Have the OAM Client pass a parameter called “enable_remote_loopback” to OAM entity. OAM entity then manages the sending of Loopback Control OAMPDUs, the lb_timer, (near) simultaneous loopback requests, contain the state diagram etc.**
 - **Have the OAM Client contain the process to manage loopback and merely pass one parameter to the OAM entity: “parser_action” (forward, loopback, discard)**

(1) “enable_remote_loopback”

■ Pros:

- State diagram specified in OAM sublayer
- Simple interface from OAM Client
 - Parameter to OAM : enable_remote_loopback
 - As long as enable_remote_loopback is TRUE, OAM entity keeps remote device in loopback mode
 - Parameter from OAM: loopback_over
 - When enable_remote_loopback returns to FALSE, OAM entity disables remote loopback and notifies OAM Client

■ Cons:

- OAM client would then need to **transmit** and **receive** Loopback Control OAMPDUs (currently only Information OAMPDUs are transmitted by OAM entity, all OAMPDUs are passed to OAM Client for parsing)
- Synchronization between two state machines is complex

(2) “*parser_action*”

■ Pros:

- Simple to specify for OAM sublayer
 - Parameter from OAM Client: *parser_action*
 - Has three values forward, discard, loopback
- OAM Client needs shadow state diagram for (1) anyway, additional burden is minimal

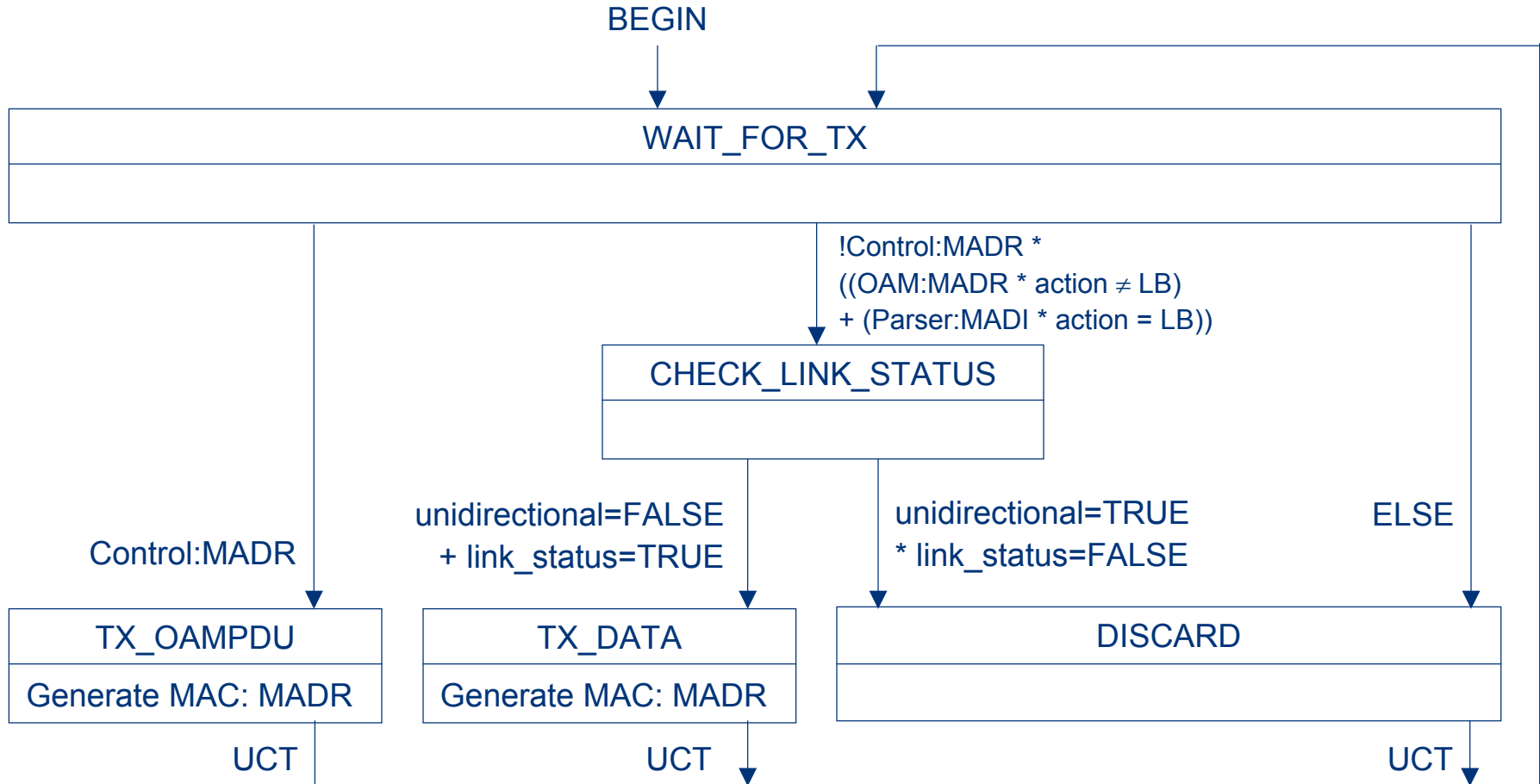
■ Cons:

- Potential for OAM Client implementation differences

Recommendation

- **Move loopback process to OAM Client**
 - **OAM sublayer only performs Discovery**
 - **OAM sublayer only transmits Information OAMPDUs**
 - **All other OAMPDUs are transmitted by OAM Client**
 - **All OAMPDUs are passed to OAM Client for parsing**
 - **Parser state diagram amended (see next slide)**

[Proposed] Fig 55-5 Multiplexer



Tx OAMPDU given precedence whenever

Tx non-OAMPDUs when

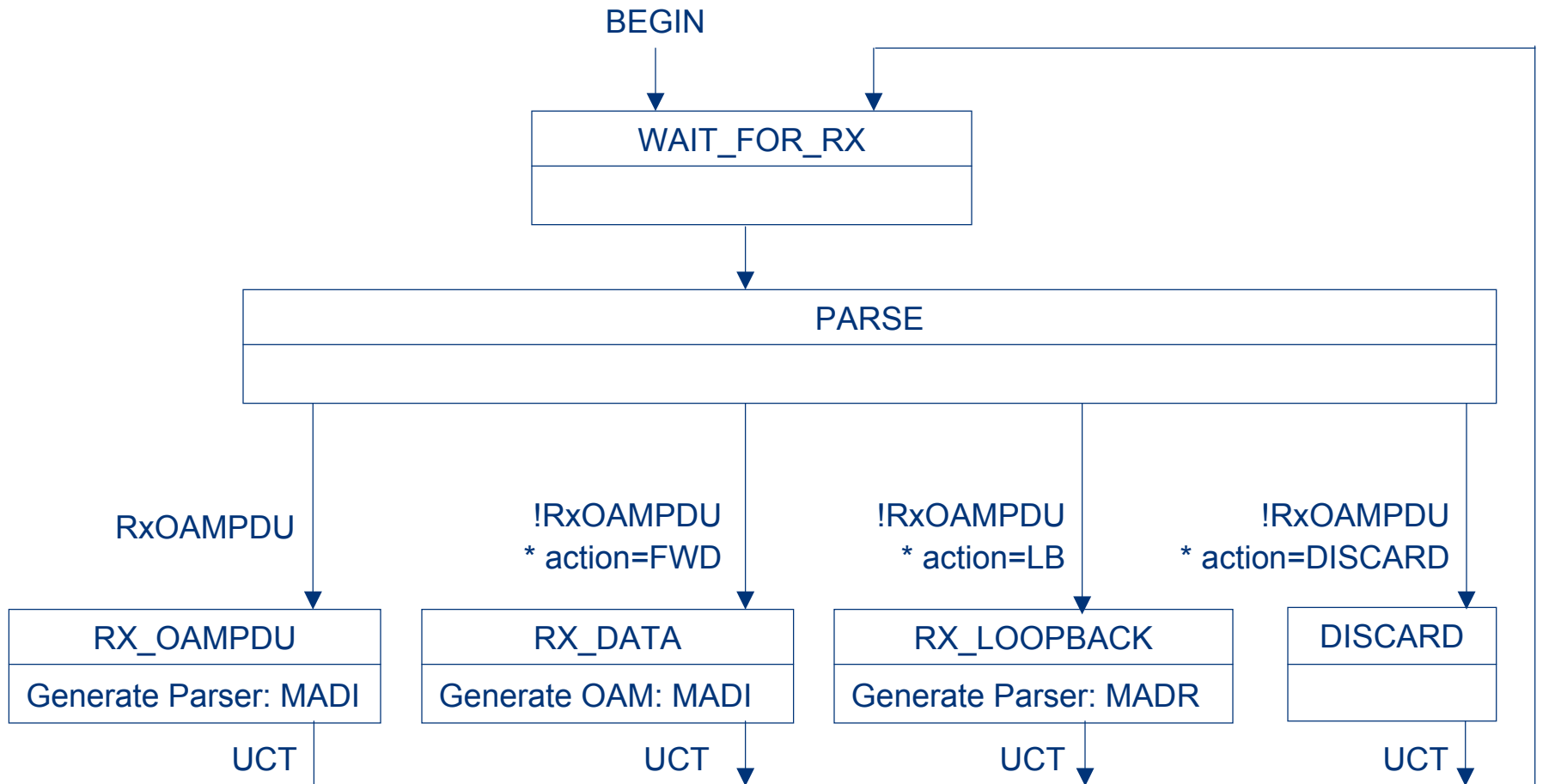
Discard non-OAMPDUs when !link & uni

Discard

Multiplexer Table

OAMPDU	Action	Uni	Link	Multiplexer Action
1	x	x	x	Transmit OAMPDU from OAM Client/Control block
0	LB	x	x	Transmit non-OAMPDU from Parser
0	!LB	x	1	Transmit non-OAMPDU from MAC Client, as long as Link=OK
0	!LB	0	x	Transmit non-OAMPDU from MAC Client, as long as unidirectional=FALSE
0	!LB	1	0	Discard non-OAMPDU from MAC Client when link!=OK and when unidirectional=TRUE
				Discard

[Proposed] Fig 55-6 Parser



All OAMPDUs are **passed** to the OAM Client

OAM Client indicates when to **forward** non-OAMPDUs to MAC Client

OAM Client indicates when to **loopback** non-OAMPDUs to MAC Client

OAM Client indicates when to **discard** non-OAMPDUs

Steps for entering loopback

- 0) local MAC client stops sending data frames
- 0') local client changes action to discard
- 1) local client sends LC w/ non-zero lb_time
- 2) remote client receives #1
- 3) within 1 sec of #2
 - a) remote client sets action to lb
 - b) remote client sends [non-zero, lb]
- 4) local client receives #3b (rec)
- 5) local MAC client begins loopback test (rec)
- 6) loopback test

Steps for continuing #1

- 0) test in progress
- 1) local client sends LC w/ non-zero lb_time
- 2) remote client receives #1
- 3) within 1 sec of #2
 - a) remote client sets action to lb
 - b) **remote client sends [non-zero, lb]**
- 4) local client receives #3b
- 5) test continues

Steps for continuing #2

- 0) test in progress
- 1) remote lb_timer expires
- 2) within 1 sec of #1
 - A) remote client sets action to discard
 - B) remote sends [0,discard]
- 3) local client sends LC w/ non-zero lb_time
- 4) local client receives #2B
- 5) remote client receives #3
- 6) within 1 sec of #5
 - a) remote client sets action to lb
 - b) remote client sends [non-zero, lb]
- 7) local client receives #6a
- 8) test continues

Steps for exiting loopback #1

- 8) local MAC client stops sending test frames
- 9) local OAM client checks counters
 - Checks tx and rx
 - If not equal checks OAM discard
- 10) local OAM client sends LC w/lb=0
- 11) remote OAM client rx #10
- 12) within 1 sec of #11
 - A) remote sends [zero, forward]
 - B) remote client sets action to forward
- 13) local client receives #12A
- 14) local client changes action to forward
- 15) local MAC client resumes data frames

Steps for exiting loopback #2

- 8) remote lb_timer expires
- 9) within 1 sec of #8
 - A) remote client sets action to discard
 - B) remote sends [0,discard]
- 10) local client receives #9B
- 11) local MAC client stops sending test frames
- 12) local OAM client checks counters
- 13) local OAM client sends LC w/lb=0
- 14) within 1 sec of #13
 - A) remote sends [0,forward]
 - B) remote client sets action to forward
- 15) local client receives #14A
- 16) local client changes action to forward
- 17) local MAC client resumes data frames

Edits

■ OAM sublayer

- Parser: add action parameter from OAM client
- Mux: add action parameter from OAM client
- LC OAMPDU definition
 - Data field structure
- Add state information in OAM_Information TLV (found in Information OAMPDUs) [lb_timer, action]

■ OAM Client

■ PICS entries

- Remote: Within 1s of lb_timer being loaded or timing out, send Information OAMPDU with updated state information

OAM Client