

Synchronous Auto-Negotiation Proposal

Howard Baumer
Ali Ghiasi



Contributors and Supporters

Howard Baumer – Broadcom

Ali Ghiasi – Broadcom

Jeff Lynch - IBM

Joe Abler - IBM

Tom Palkert - Xilinx

Brian Brunn - Xilinx

Outline

- **Clause 37 Changes**
 - Figures 37-5 & 37-6
 - Variables and Functions
- Next Pages
- Parallel Detection
- Clause 48 changes
 - Figures 48-2, 48-6 & 48-9
 - Variables and Functions
- Clause 49 changes
 - Figure 49-4
 - Variables and Functions

Figure 37-5 Modification

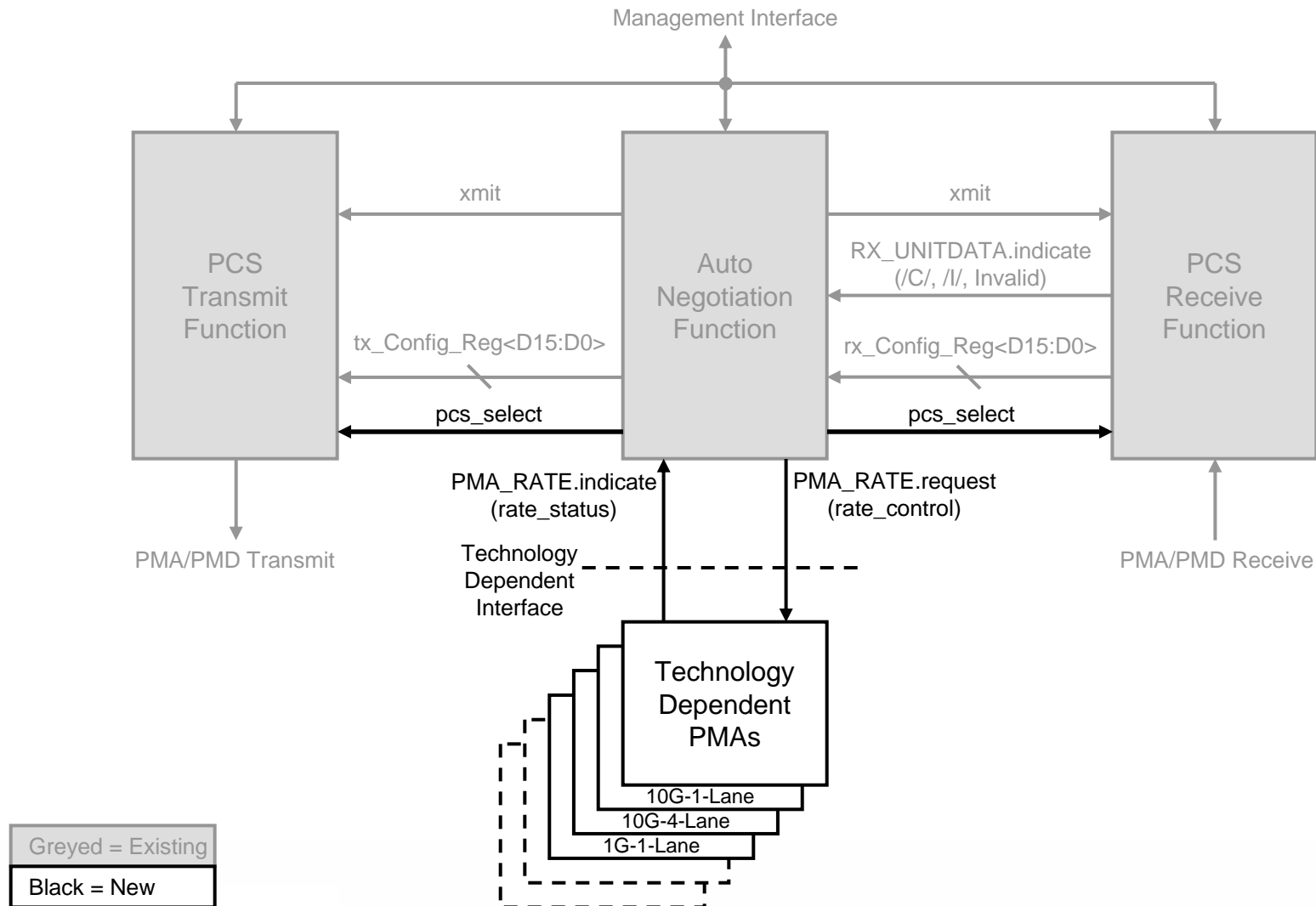


Figure 37-6

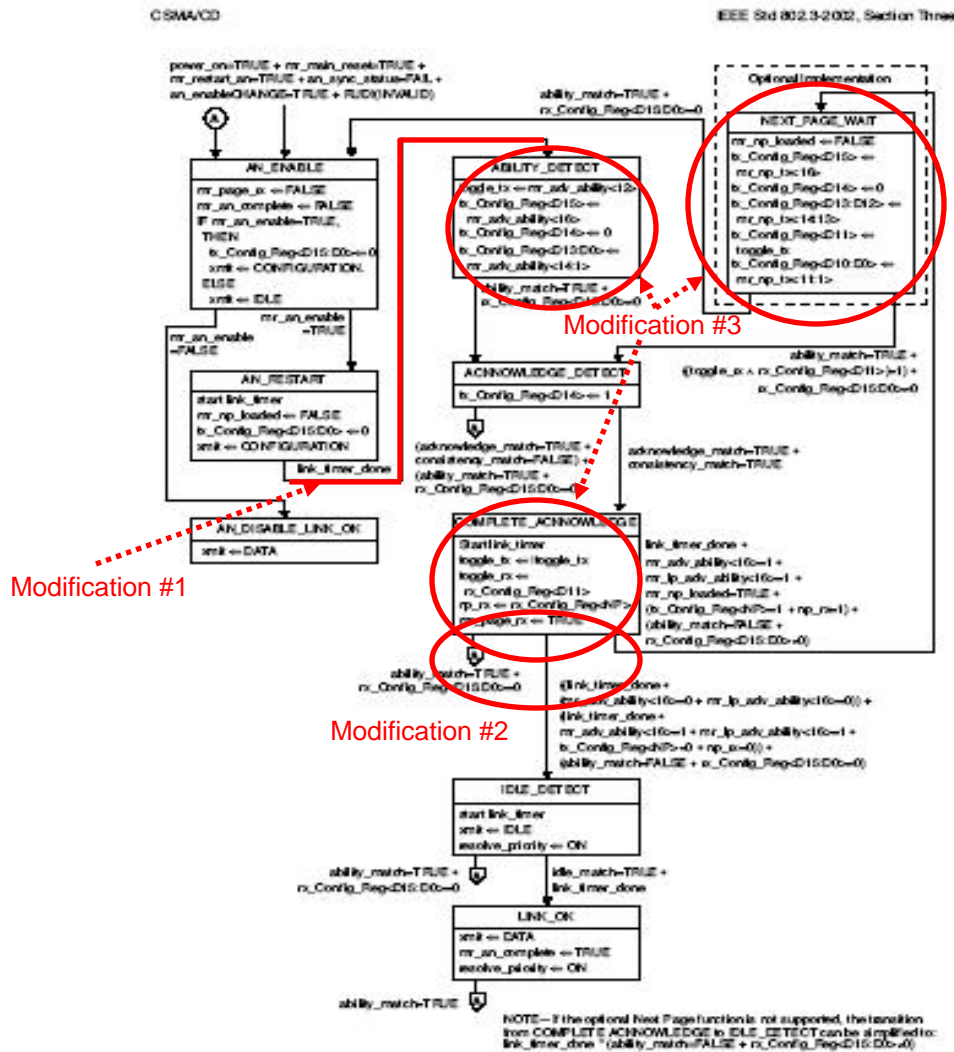


Figure 37-6 — Auto-Negotiation state diagram

Figure 37-6 Modification #1 Parallel Detection

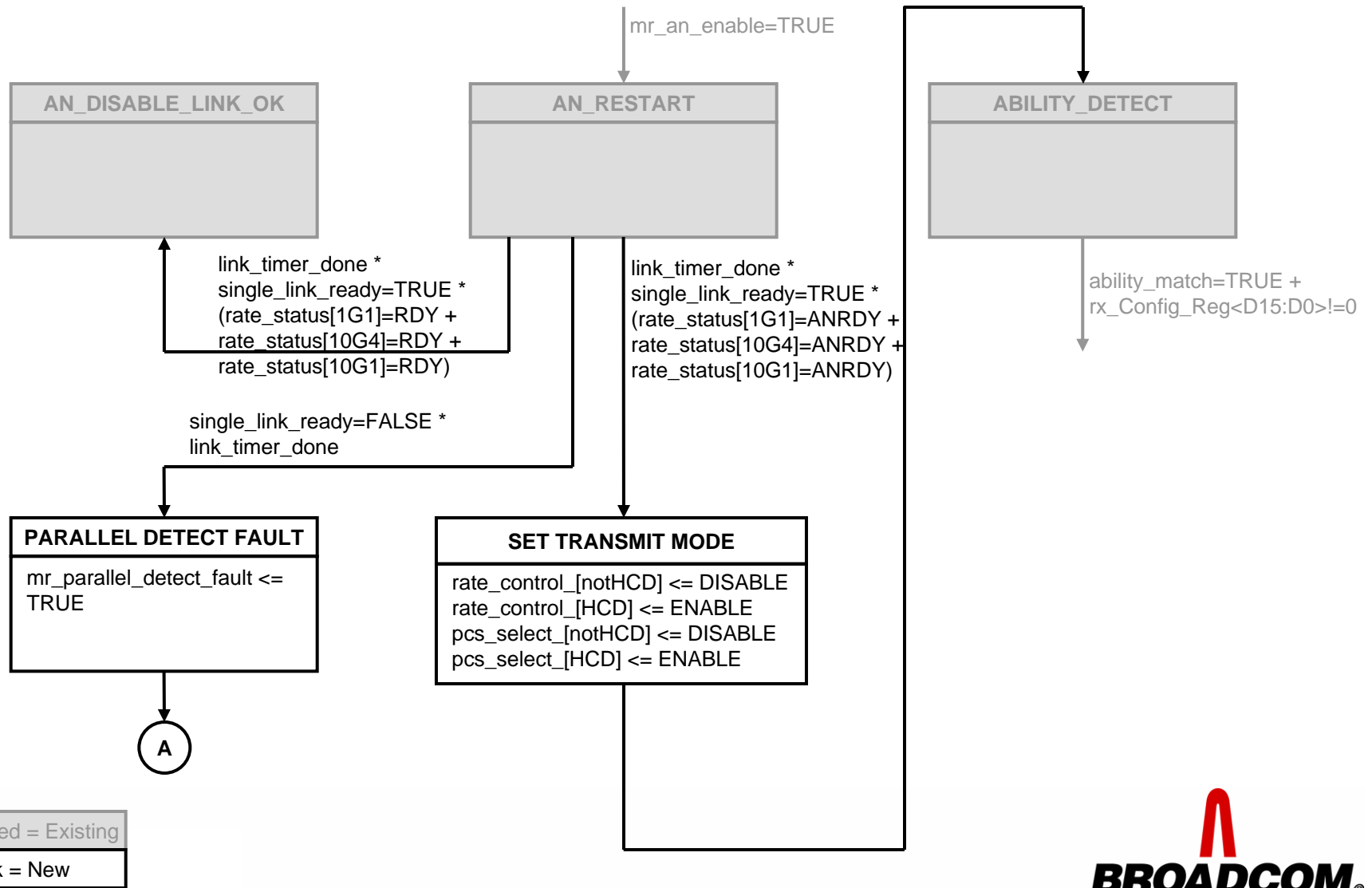
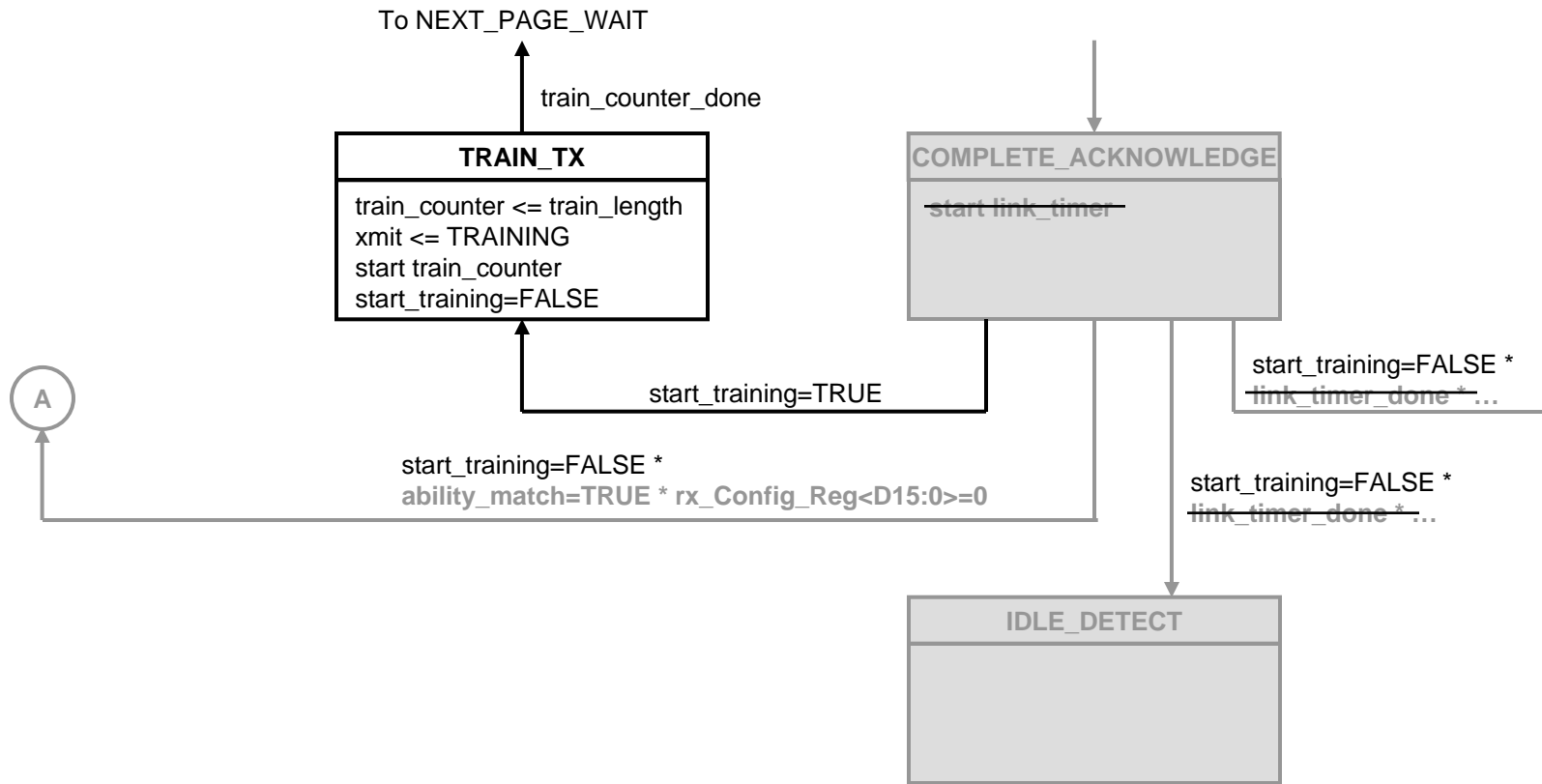


Figure 37-6 Modification #2 Tx Training

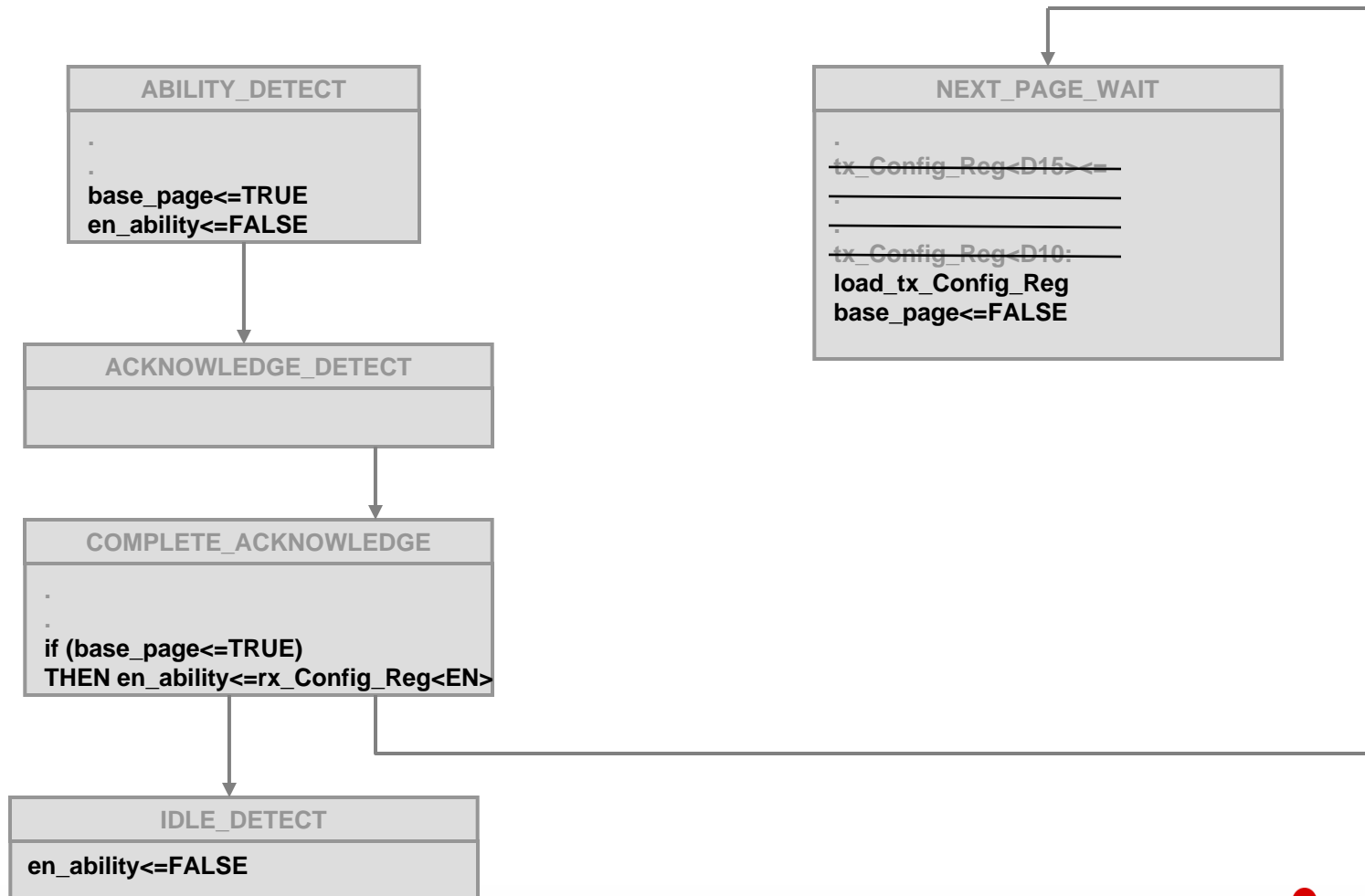
L



Greyed = Existing
Black = New

Figure 37-6 Modification #2

Extended Next Page



Greyed = Existing

Black = New

Clause 37 Technology-Dependent Interface

- **PMA_RATE.indicate:** This primitive is generated by the PMA to indicate the status of the received signal. The purpose of this primitive is to give the Auto-Negotiation function a means of determining the rate of the received signal.
- **PMA_RATE.indicate(rate_status):** The rate_status parameter shall assume one of three values: ANRDY, RDY or FAIL, indicating whether the underlying receive signal is intact, contains /C/ code-groups and is ready to be enabled (ANRDY), intact, does not contain /C/ code-groups and is ready to be enabled (RDY) or not intact (FAIL).
- **PMA_RATE.request:** This primitive is generated by Auto-Negotiation to allow it to enable and disable operation of the PMA.
- **PMA_RATE.request(rate_control):** The rate_control parameter shall assume one of two values: DISABLE or ENABLE. The rate_control=DISABLE mode shall be used by the Auto-Negotiation function to disable PMA processing. The rate_control=ENABLE mode shall be used by Auto-Negotiation to turn control over to a single PMA for all normal processing functions.

Clause 37 Variables, cont.

- tx_Config_Reg expanded from 16 to 48 bits.
- rx_Config_Reg expanded from 16 to 48 bits.
- base_page: Indicates whether a base page or next page set ability_match.
- en_ability: Controls whether extended next pages are enabled or not.
- HCD; represents the single technology-dependent PMA chosen by Auto-Negotiation as the highest common denominator technology through the Priority Resolution or Parallel Detection function.
- notHCD; represents all technology-dependent PMAs not chosen by Auto-Negotiation as the highest common denominator technology through the Priority Resolution or Parallel Detection function.
- 1G1; represents that the 1G-1-Lane PMA is the signal source.
- 10G4; represents that the 10G-4-Lane PMA is the signal source.
- 10G1; represents that the 10G-1-Lane PMA is the signal source.

Clause 37 Variables

- **pcs_select**
 - pcs_select shall assume one of two values: DISABLE or ENABLE. The pcs_select =DISABLE mode shall be used by the Auto-Negotiation function to disable PCS processing. The pcs_select =ENABLE mode shall be used by Auto-Negotiation to turn control over to a single PCS for all normal processing functions.
- **train_length**
 - The length, in UI, of the training sequence to be transmitted / received.
- **train_counter:**
 - Counts down the number of full baud rate training bits.
 - Initial Value is train_length.
- **train_counter_done**
 - goes true when train_counter has counted down to 0.
- **start_training:**
 - Set to True when the “Tx Train Control” Next Page has been sent with RT=1 and a “Tx Train Control” Next Page has been received with RT=0.
 - Set to FALSE by the State Diagram.
 - Default is False

Clause 37 Variables, cont.

- `single_link_ready`
Status indicating that only one the of the following indications is being received:
 1. `rate_status_[1G1] = RDY`
 2. `rate_status_[10G4] = RDY`
 3. `rate_status_[10G1] = RDY`
 4. `rate_status_[1G1] = ANRDY`
 5. `rate_status_[10G4] = ANRDY`
 6. `rate_status_[10G1] = ANRDY`

Values:

FALSE; either zero or more than one of the above six indications are true

TRUE; Exactly one of the above three indications is true

- NOTE—This variable is set by this variable definition; it is not set explicitly in the state diagrams.

Clause 37 Variables, cont.

- xmit: A parameter set by the PCS Auto-Negotiation process to reflect the source of information to the PCS Transmit process and the rate of data transmitted by the PCS Transmit process.

Values (10G1):

- CONFIGURATION: Tx_Config_Reg<D15:D0> information is being sourced from the PCS Auto-Negotiation process. Transmitted data is NRZ with baud rate equal to 1/n of the normal baud rate (n=8 for NRZ & Duo-binary, n=4 for PAM4)
- TRAINING: The transmit training sequence sourced from the PCS. Normal transmit data mode at the normal baud rate.
- DATA: //, sourced from the PCS, is interspersed with packets sourced from the MAC. Normal transmit data mode at the normal baud rate.
- IDLE: // is being sourced from the PCS Auto-Negotiation process. Normal transmit data mode at the normal baud rate.

Clause 37 Variables, cont

- xmit:

Values (10G4):

- CONFIGURATION: Tx_Config_Reg<D15:D0> information is being sourced from the PCS Auto-Negotiation process. Transmitted data is NRZ with baud rate equal to 1/2 of the normal baud rate.
- TRAINING: The transmit training sequence sourced from the PCS. Normal transmit data mode at the normal baud rate.
- DATA: //, sourced from the PCS, is interspersed with packets sourced from the MAC. Normal transmit data mode at the normal baud rate.
- IDLE: // is being sourced from the PCS Auto-Negotiation process. Normal transmit data mode at the normal baud rate.

- xmit:

Values (1G1):

- CONFIGURATION: Tx_Config_Reg<D15:D0> information is being sourced from the PCS Auto-Negotiation process. Normal transmit data mode at the normal baud rate.
- TRAINING: not available for 1G1 PCS.
- DATA: //, sourced from the PCS, is interspersed with packets sourced from the MAC. Normal transmit data mode at the normal baud rate.
- IDLE: // is being sourced from the PCS Auto-Negotiation process. Normal transmit data mode at the normal baud rate.

Clause 37 Functions

- `load_tx_Config_Reg`: This function loads the `tx_Config_Reg` based on the state of `en_ability`.

IF (`en_ability` = FALSE) THEN

- `tx_Config_Reg<D15>` <= `mr_np_tx<16>`
- `tx_Config_Reg<D14>` <= 0
- `tx_Config_Reg<D13:D12>` <= `mr_np_tx<14:13>`
- `tx_Config_Reg<D15>` <= `toggle_tx`
- `tx_Config_Reg<D10:D0>` <= `mr_np_tx<11:1>`

ELSE

- `tx_Config_Reg<D47:D15>` <= `mr_np_tx<48:16>`
- `tx_Config_Reg<D14>` <= 0
- `tx_Config_Reg<D13:D12>` <= `mr_np_tx<14:13>`
- `tx_Config_Reg<D15>` <= `toggle_tx`
- `tx_Config_Reg<D10:D0>` <= `mr_np_tx<11:1>`

Outline

- Clause 37 Changes
 - Figures 37-5 & 37-6
 - Variables and Functions
- Next Pages
- Parallel Detection
- Clause 48 changes
 - Figures 48-2, 48-6 & 48-9
 - Variables and Functions
- Clause 49 changes
 - Figure 49-4
 - Variables and Functions

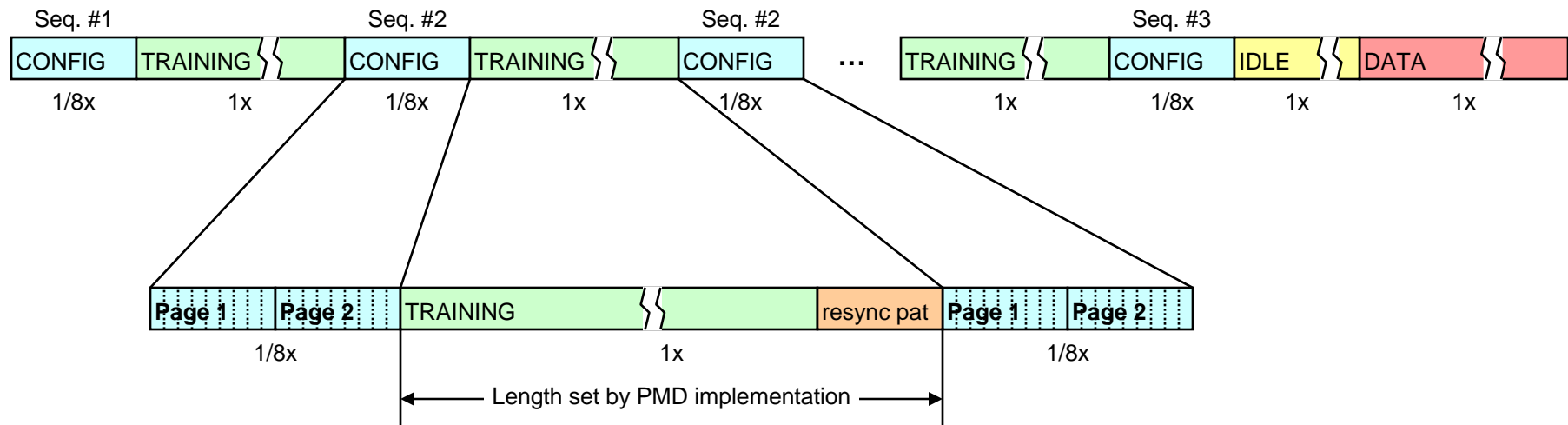
Next Page Modifications

(mandatory for 10G-1-Lane)

- Add “EN” bit to Base Page
- Add EoBP Extended Next Page

Name	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
Base Page	Res	Res	Res	Res	Res	FD	HD	PS1	PS2	Res	Res	EN	RF1	RF2	Ack	NP
EoBP Next Page	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
	0	1	0	1	0	0	0	0	0	0	0	T	Ack2	MP	Ack	NP
	D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31
	TC	PC	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	10G1	10G4	1G1
	D32	D33	D34	D35	D36	D37	D38	D39	D40	D41	D42	D43	D44	D45	D46	D47
	C(-1)	C(0)	C(1)	C(2)	C(3)	C(4)	C(5)	C(6)								
EN	Extend Next Page Capable															
1G1	1G-1-Lane capable															
10G4	10G-4-Lane capable															
10G1	10G-1-Lane capable															
	TC															
	0	0	Send Training Pattern													
	0	1	Training Pattern Follows													
	1	0	Training Done													
	1	1	Reserved													
	PC															
	0	No Tx Precoding														
	1	Turn Duobinary precoding on														
	C(n)		Action													
	0	0	Hold													
	0	1	Decrement													
	1	0	Increment													
	1	1	Hold													
	<ul style="list-style-type: none"> - Taps will increment or decrement until their min or max values are reached and then they hold - Up to 8 coefficients (baseline is 3) - Assume 1 UI between taps 															

Training Sequence Example



Auto-Negotiation sequence 1

Base Page
Next Page, TC=00
Next Page, TC=01

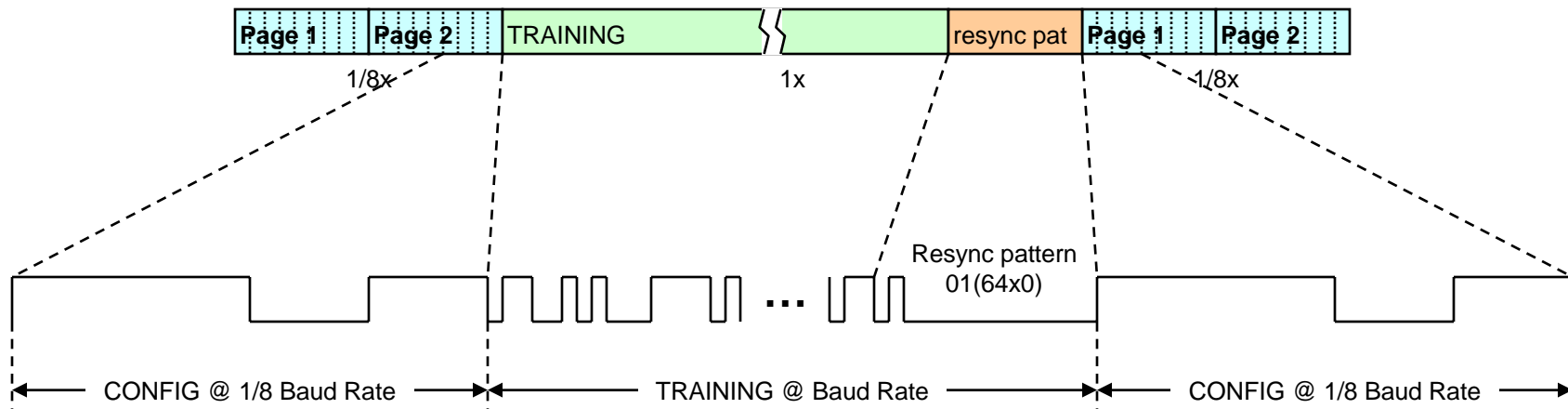
Auto-Negotiation sequence 2

Next Page, TC=00
Next Page, TC=01

Auto-Negotiation sequence 3

Next Page, TC=10

Signaling Method, 10G1 NRZ Example



- All edges are aligned to the baud rate boundaries
- CONFIG signals are 8x repeated full baud bits

Outline

- Clause 37 Changes
 - Figures 37-5 & 37-6
 - Variables and Functions
- Next Pages
- Parallel Detection
- Clause 48 changes
 - Figures 48-2, 48-6 & 48-9
 - Variables and Functions
- Clause 49 changes
 - Figure 49-4
 - Variables and Functions

Proposed Signaling Method

Use normal data signaling & associated PCS

- 1G-1-Lane: 1.25GBd (i.e. 1000BASE-CX 8B/10B signals)
- 10G-4-Lane: 1.5625GBd (i.e. 1/2 10GBASE-X 8B/10B signals)
- 10G-4-Lane: 3.125GBd (i.e. 10GBASE-X 8B/10B signals w/o AN)
- 10G-1-Lane: 1.29GBd, NRZ
 - NRZ & Duo-Binary: 1/8 of the 64B/66B data rate
 - PAM4: 1/4 of the 64B/66B data rate

Priority Resolution

- Data rate priorities: When there is more than one data rate in common between link partners the order of data rate choice is 10G1, 10G4, 1G1.
- Initial data rate: The initial data rate used by a transmitter shall be its lowest capable data rate.

Parallel Detection Requirements

- Needs to differentiate between 1.25G, 1.5625G, 3.125G and 1.29G NRZ signals.

From a reference of 1.25G or 2.5G(2x1.25G) the rate offsets for 1.25G/1.5625G/3.125G/1.29G are 0/250,000/250,000/30,000ppm respectively.

- Needs to detect the presence of valid /C/ or /I/ code groups.
- Return status of rate detected and whether configuration code groups have been detected.

Line Rates: Both Sides Auto-Negotiation

Local Phy	Link Partner	Initial Rate	CONFIG. Rate	TRAINING Rate	IDLE & DATA Rate
		Local / Remote			
1G1	1G1	1.25G / 1.25G	1.25G	-	1.25G
	10G4+1G1	1.25G / 1.25G			
	10G1+1G1	1.25G / 1.25G			
10G4+1G1	10G1+1G1	1.25G / 1.25G	1.25G	-	1.25G
10G4	10G4	1.56G / 1.56G	1.56G	3.125G	3.125G
	10G4+1G1	1.56G / 1.25G			
	10G1+10G4	1.56G / 1.56G			
10G4+1G1	10G4+1G1 10G1+10G4	1.25G / 1.25G 1.25G / 1.56G	1.56G	3.125G	3.125G
10G1	10G1	1.29G / 1.29G	1.29G	10G1	10G1
	10G1+1G1	1.29G / 1.25G			
	10G1+10G4	1.29G / 1.56G			
10G1+10G4	10G1+1G1 10G1+10G4	1.56G / 1.25G 1.56G / 1.56G	1.29G	10G1	10G1
10G1+1G1	10G1+1G1	1.25G / 1.25G	1.29G	10G1	10G1

Line Rates: One Side Auto-Negotiation

Local Phy No A-Neg	Link Partner	Initial Rate Local / Remote	Config. Rate	TRAINING Rate	IDLE & DATA Rate
1G1	1G1 10G4+1G1 10G1+1G1	1.25G / 1.25G 1.25G / 1.25G 1.25G / 1.25G	-	-	1.25G
10G4	10G4 10G4+1G1 10G1+10G4	3.125G / 1.56G 3.125G / 1.25G 3.125G / 1.56G	-	-	3.125G

Outline

- Clause 37 Changes
 - Figures 37-5 & 37-6
 - Variables and Functions
- Next Pages
- Parallel Detection
- Clause 48 changes
 - Figures 48-2, 48-6 & 48-9
 - Variables and Functions
- Clause 49 changes
 - Figure 49-4
 - Variables and Functions
 - Functions

Figure 48-2 Modification

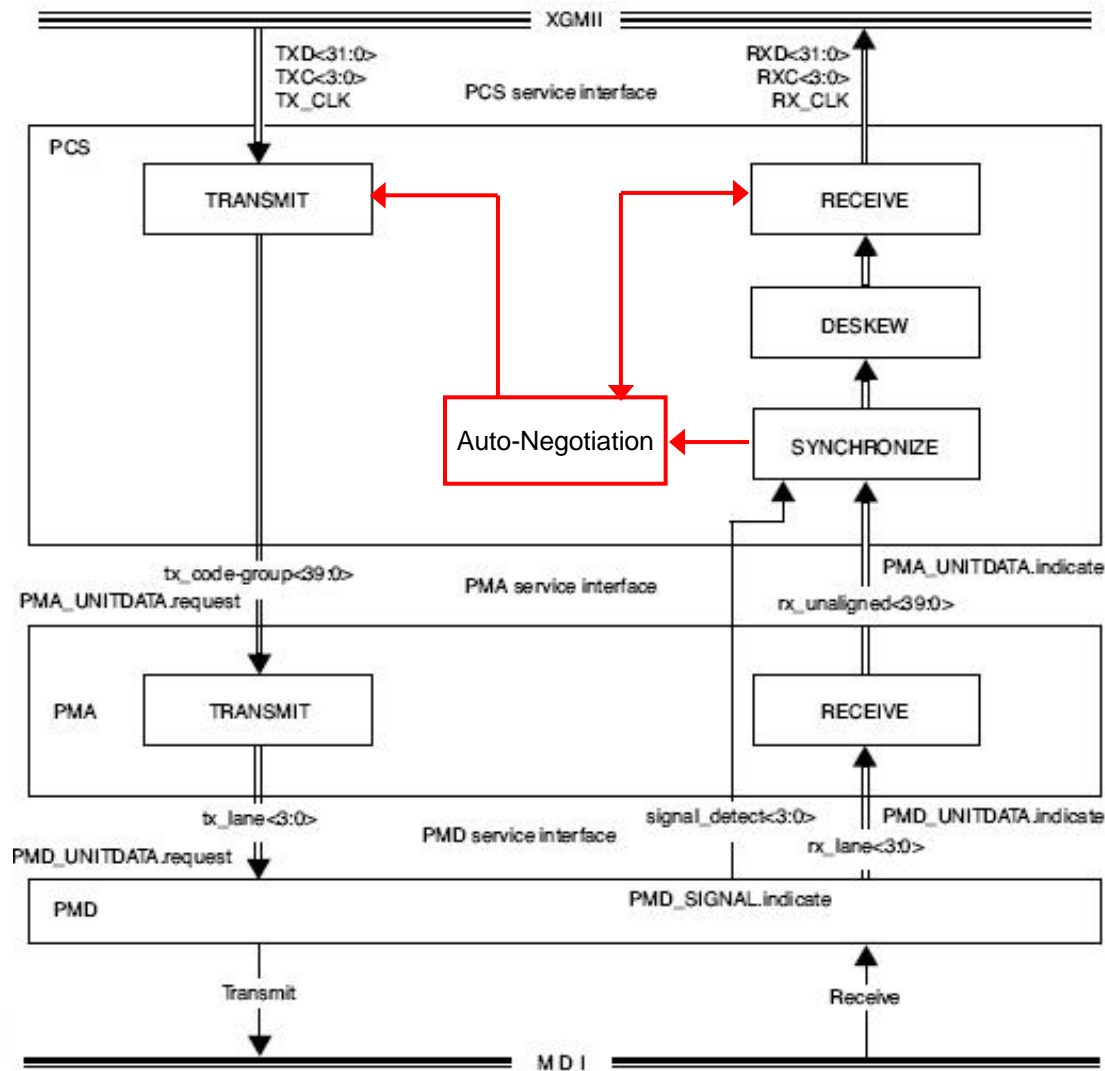


Figure 48-2 – Functional block diagram

Figure 48-6

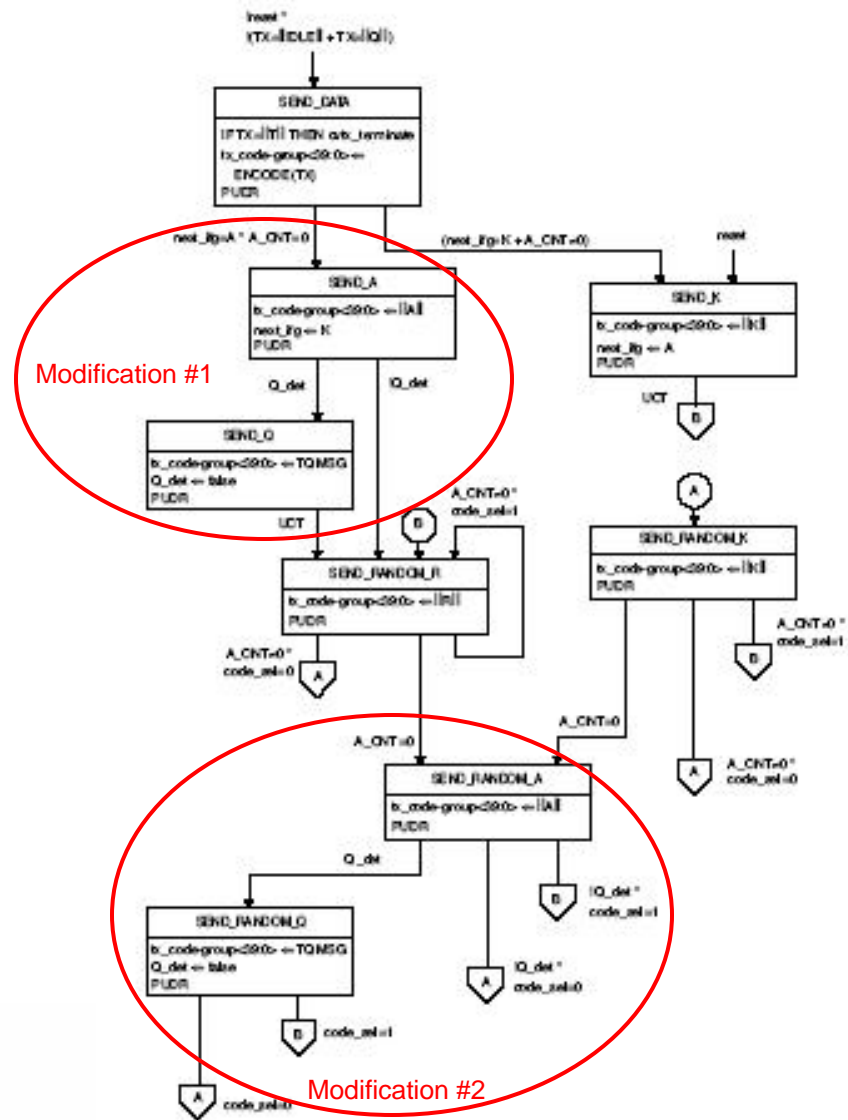


Figure 48-6 Modification #1

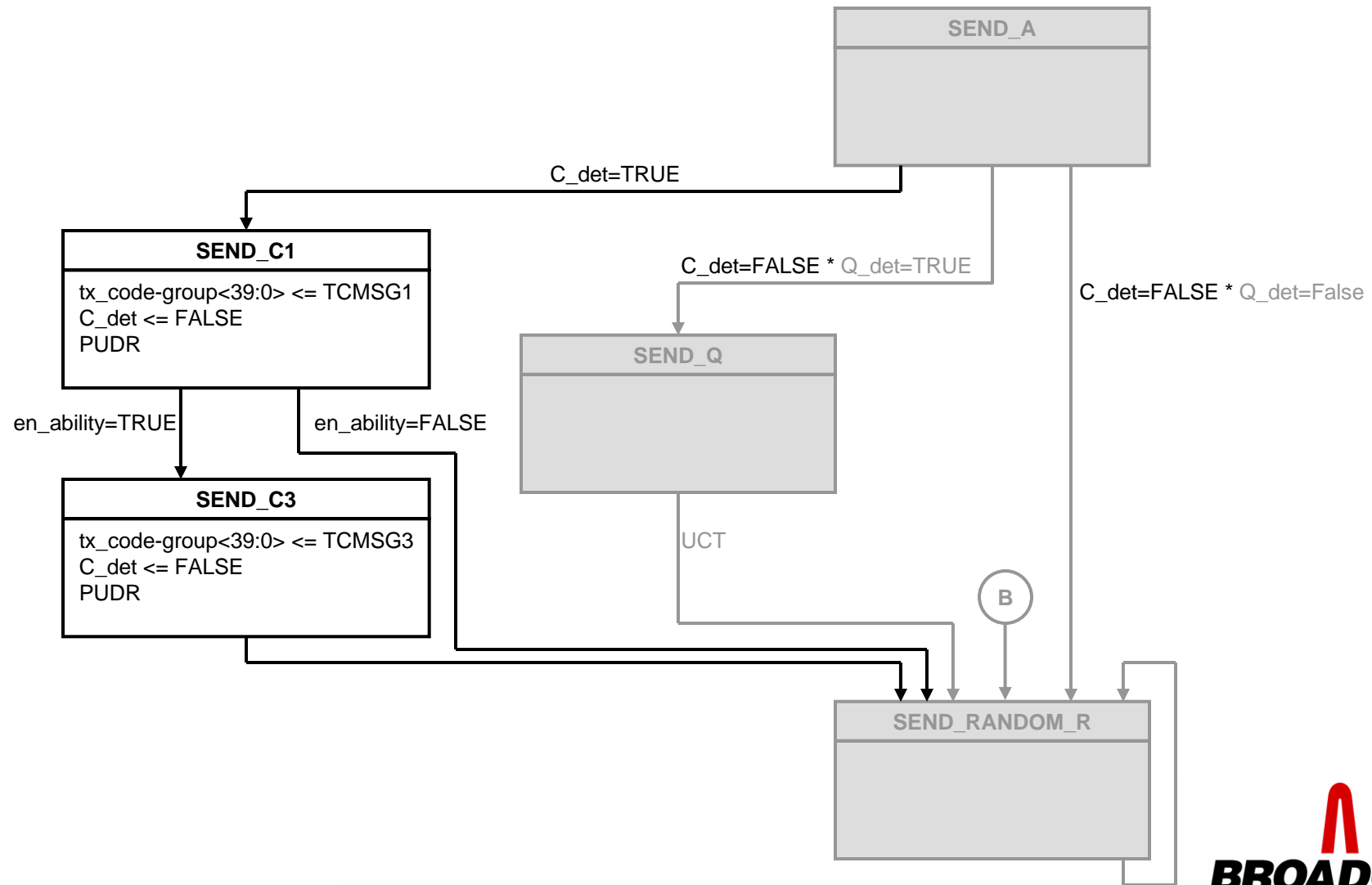


Figure 48-6 Modification #2

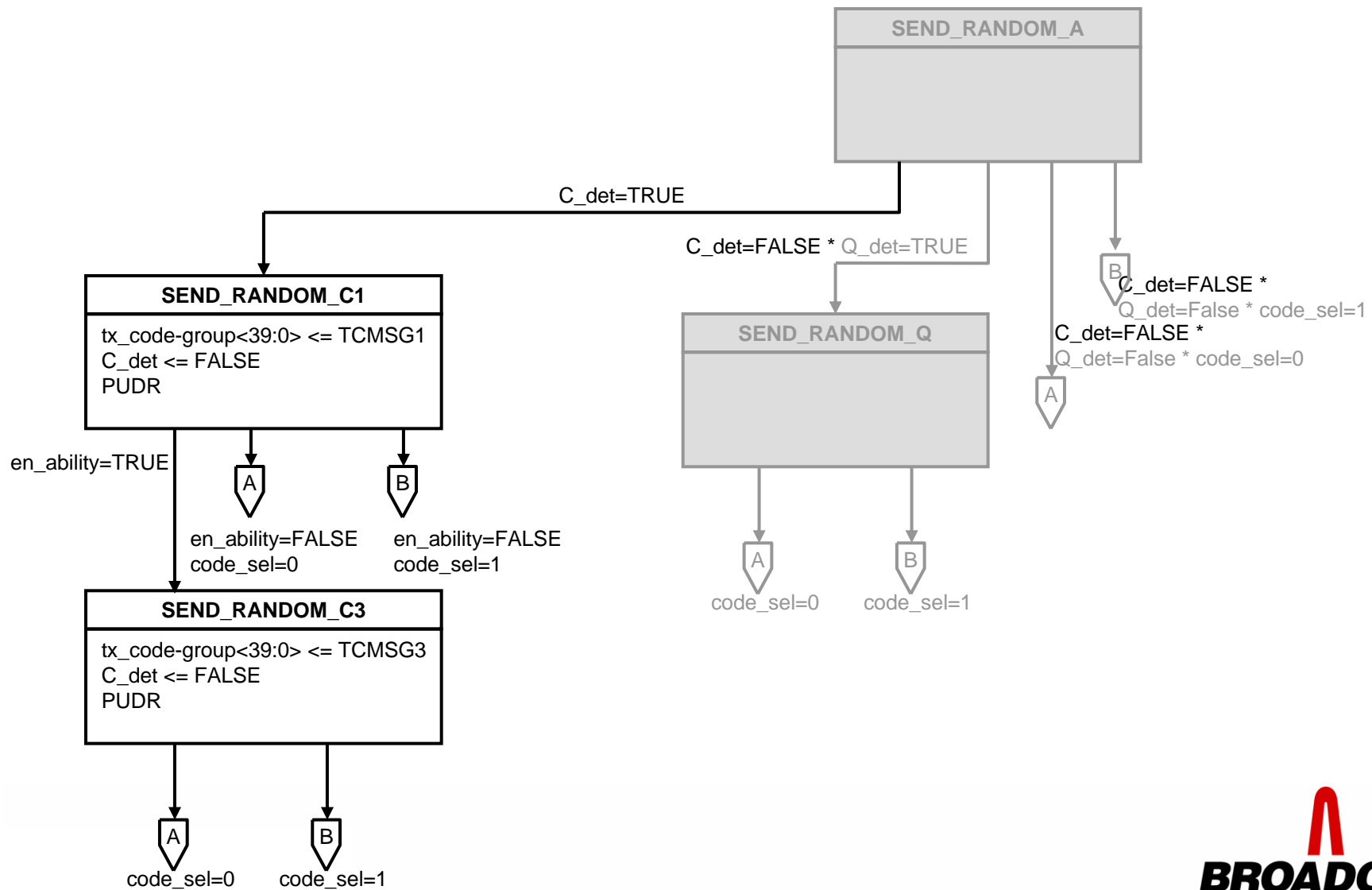


Figure 48-9

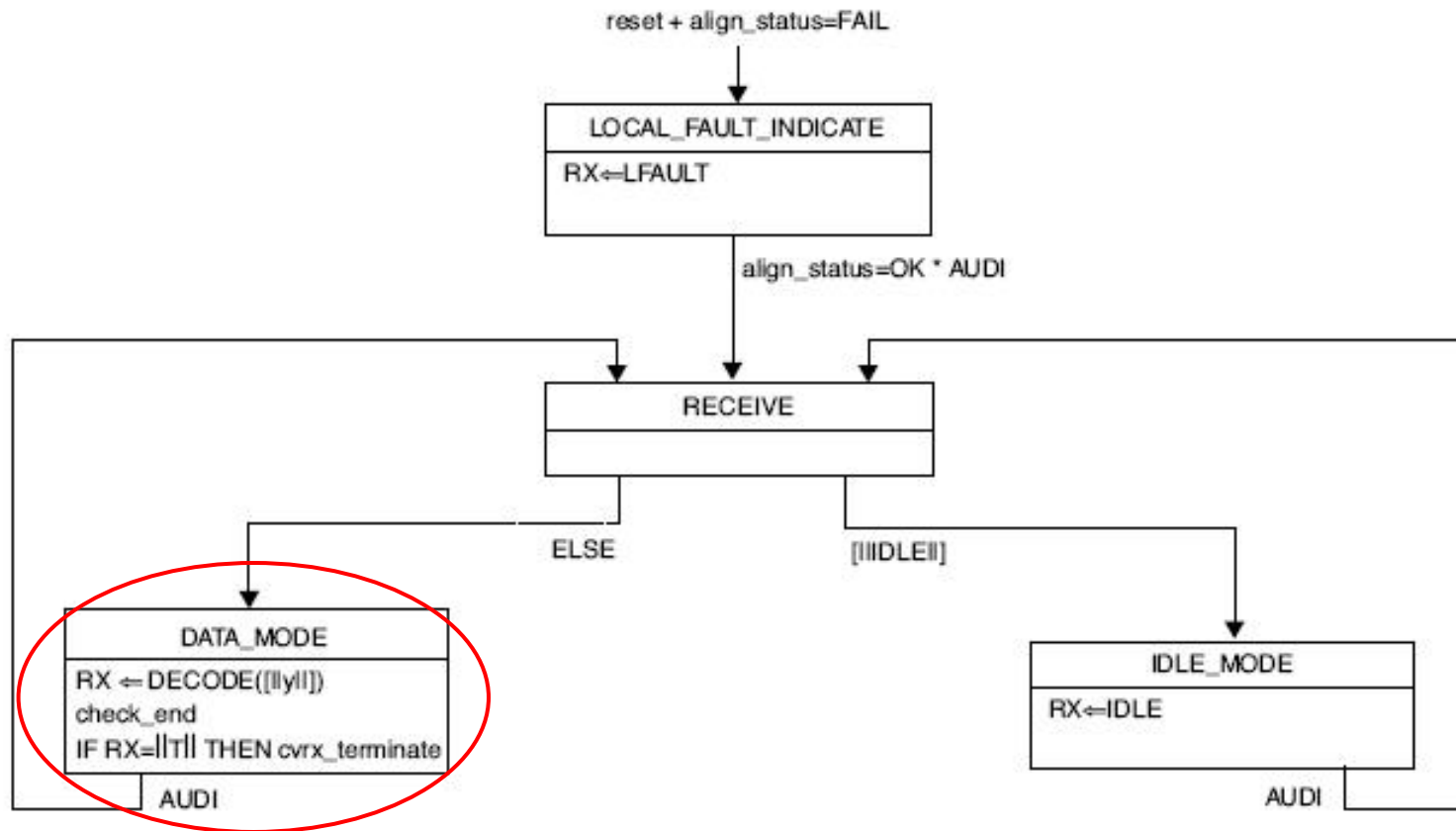


Figure 48-9—PCS receive state diagram

Figure 48-9 Modification

```
DATA_MODE
RX<=DECODE(||y||)
IF RX=|C| THEN
  rx_Config_Reg<=RX<D23:D8>
  RX<=|IDLE|
ELSE
  check_end
  IFRX=|T| THEN crvx_terminate
END
```


Clause 48 Variables

- **TCMSG**
 - A vector of bits representing the configuration page to be transmitted. Used by the PCS Transmit process to load tx_code-group <39:0>.
- **rx_Config_Reg<D15:D0>**
 - A 16-bit array that contains the data bits received from a ||C|| ordered_set as defined in 48.2.4.?. Conveyed by the PCS Receive process to the PCS Auto-Negotiation process. The format of the data bits is context dependent, relative to the state of the Auto-Negotiation function, and is presented in 37.2.1.1 and 37.2.4.3.1. For each element within the array:
Values:
 - ZERO; Data bit is a logical zero.
 - ONE; Data bit is a logical one.
- **tx_Config_Reg<D15:D0>**
 - A 16-bit array that contains the data bits to be transmitted in a ||C|| ordered_set as defined in 48.2.4.?. Conveyed by the PCS Auto-Negotiation process to the PCS Transmit process. The format of the data bits is context dependent, relative to the state of the Auto-Negotiation function, and is presented in 37.2.1.1 and 37.2.4.3.1. For each element within the array:
Values:
 - ZERO; Data bit is a logical zero.
 - ONE; Data bit is a logical one.

Clause 48 Functions

- **C_det**
 - Function to determine the need to transmit configuration ordered_sets. If `xmit=CONFIGURATION` then `C_det` is set to true and `TCMSG` is set to the result of `ENCODE(||C||)`. `C_det` remains true until set to false by the PCS transmit source state diagram. In the event that this function and the state diagram both attempt to modify `C_det`, the setting of `C_det` by this function to true will take priority.
- Configuration ordered_sets:
 - `/C1/ = /K28.5/D21.5/Config_Reg16`
 - `/C3/ = /K28.5/D29.1/Config_Reg48`
 - `Config_Reg16` = Two data code-groups representing the 16 bit `Config_Reg1` value.
 - `Config_Reg48` = Six data code-groups representing the 48 bit `Config_Reg1`, `Config_Reg2`, `Config_Reg3` values.

Outline

- Clause 37 Changes
 - Figures 37-5 & 37-6
 - Variables and Functions
- Next Pages
- Parallel Detection
- Clause 48 changes
 - Figures 48-2, 48-6 & 48-9
 - Variables and Functions
- Clause 49 changes
 - Figure 49-4
 - Variables and Functions

Figure 49-4

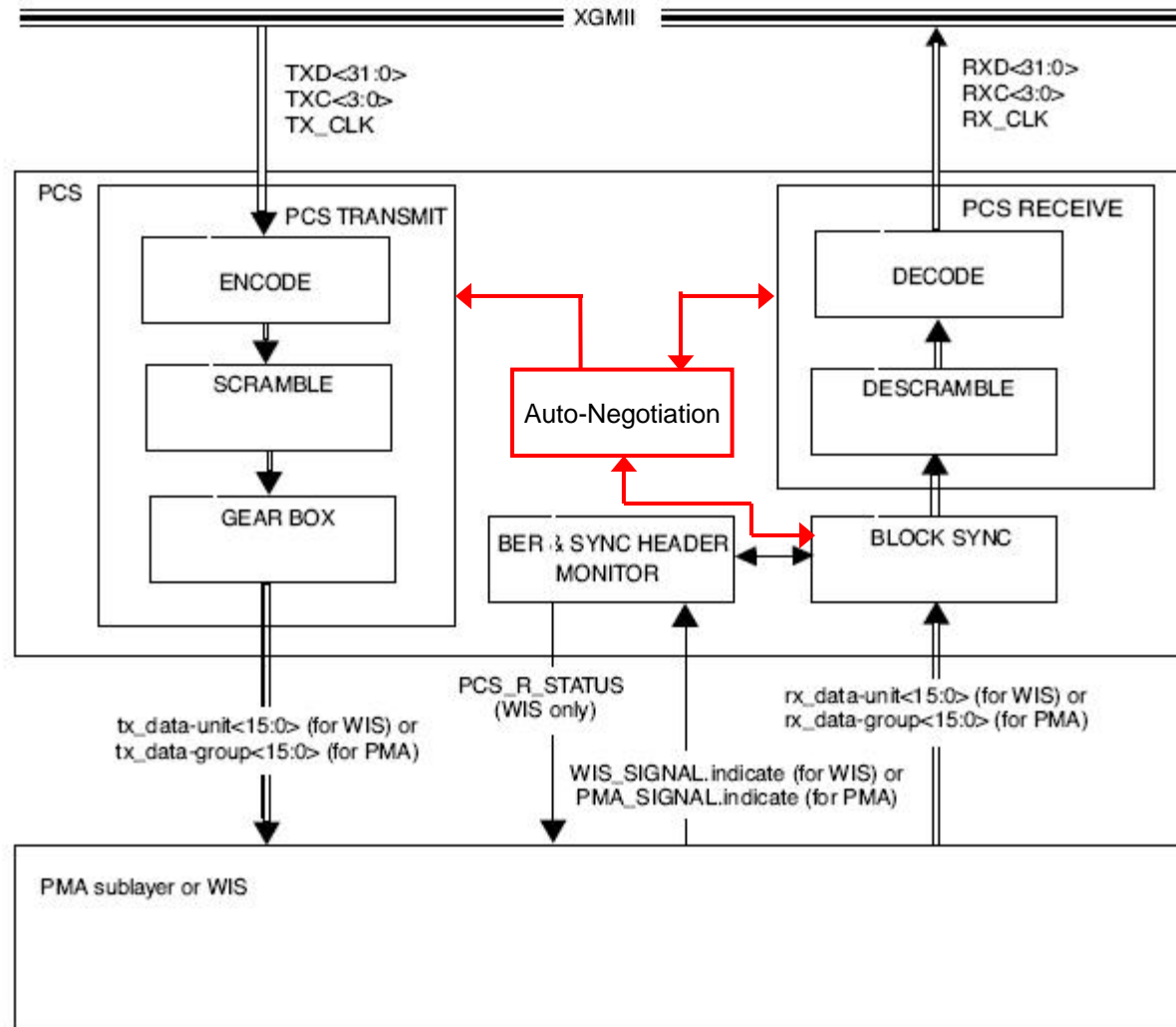


Figure 49-4—Functional block diagram

Clause 49 Variables

- tx_raw

- If xmit=DATA|IDLE|TRAINING then use existing definition
- if xmit=CONFIGURATION then

Vector containing two successive ||C|| transfers. Bits 8, 17, 26 and 35 for the first transfer are placed in tx_raw<0> through tx_raw<3>, respectively. Bits 8, 17, 26 and 35 for the second transfer are placed in tx_raw<4> through tx_raw<7>, respectively. Bits 0 through 7, 9 through 16, 18 through 25 and 27 through 34 for the first transfer are placed in tx_raw<8> through tx_raw<39>, respectively. Bits 0 through 7, 9 through 16, 18 through 25 and 27 through 34 for the first transfer are placed in tx_raw<8> through tx_raw<39>, respectively. for the second transfer are placed in tx_raw<40> through tx_raw<71>, respectively.