

Clause 28 & 40 Auto-Negotiation

An introduction to the
Auto-Negotiation process

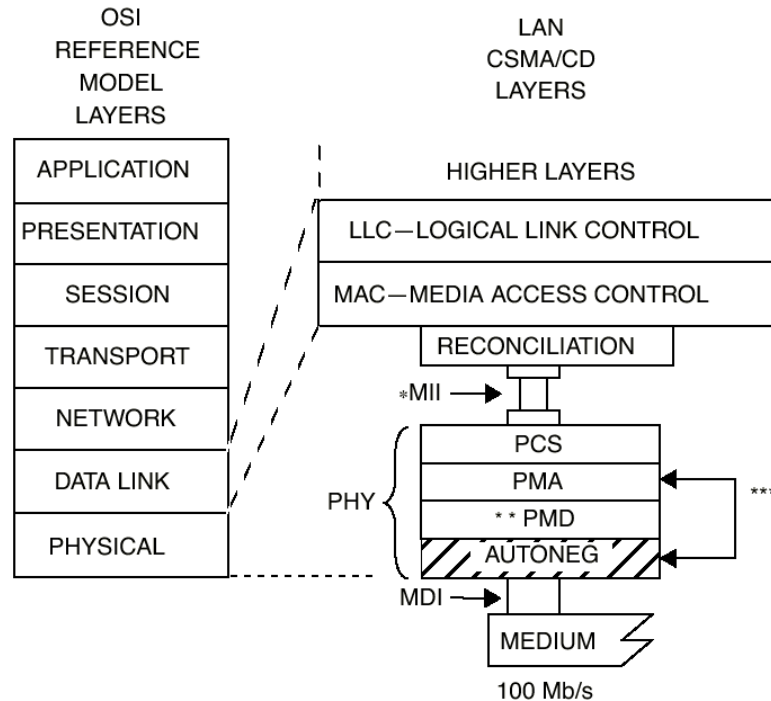


Presentation Overview:

- Auto-Negotiation in the OSI model
- An Introduction to Auto-Negotiation
- Interface with the MAC
- Interface with the Medium
- Auto-Negotiation Functions



The OSI Stack



MDI = MEDIUM DEPENDENT INTERFACE
MII = MEDIA INDEPENDENT INTERFACE
AUTONEG = AUTO-NEGOTIATION

PCS = PHYSICAL CODING SUBLAYER
PMA = PHYSICAL MEDIUM ATTACHMENT
PHY = PHYSICAL LAYER DEVICE
PMD = PHYSICAL MEDIUM DEPENDENT

* MII is optional for 10 Mb/s DTEs and for 100 Mb/s systems and is not specified for 1 Mb/s systems.

** PMD is specified for 100BASE-X only; 100BASE-T4 does not use this layer.

*** AUTONEG communicates with the PMA sublayer through the PMA service interface messages PMA_LINK.request and PMA_LINK.indicate.



Why use Auto-Negotiation (a brief introduction to Auto-Negotiation)

- Section Objectives
 - To explain why Auto-Negotiation was developed and for what purpose it is used.
 - Different Speeds and Duplexes
 - How devices worked before Auto-Negotiation
 - Taking the User out of the equation



Speed, Duplex, and all the rest

- Ethernet Devices run at many speeds and duplexes, each with their own signaling type
 - 10BASE-T Half and Full Duplex: Manchester encoding over 2 pair cat3
 - 100BASE-TX Half and Full Duplex: MLT-3 over 2 pair cat5
 - 100BASE-T4 : 8B6T over 4 pair cat3
 - 100BASE-T2 : 2dPAM-5 over 2 pair cat3
 - 1000BASE-T Half and Full Duplex: 4dPAM-5 over 4 pair cat 5 cable



Before Auto-Negotiation

- Single Speed Networks
 - Plug two devices together and they work, you might see some dropped packets if there is a duplex mismatch
- Now More than One speed
 - The need to configure both speed and duplex
 - Mismatches stop all traffic



The User and Configuration

- Why should someone have to manually configure every link in a network?
 - Some switches have 48 ports, that's 96 ports to configure in just one switched network!!
 - Auto-Negotiation allows the devices to auto-magically link at the best speed shared between the two. This means less work for the user and fewer conflicts due to speed/duplex mismatches.



What does Auto-Negotiation do (introduction continued)

- Section Objectives
 - To explain what Auto-Negotiation does
 - Management Information Exchange
 - What an “Operational Mode” is
 - How Auto-Negotiation works for the user



Operational Mode

- It allows the link partners to exchange some management information, this information pertains to the operational modes of each device.
- What is an operational mode?
 - Speed (10BASE-T / 100BASE-TX / 1000BASE-T / 10GBASE-T / ...)
 - Duplex (Full Duplex / Half Duplex)
 - Pause Mode (Synchronous / Asynchronous)
 - Etc



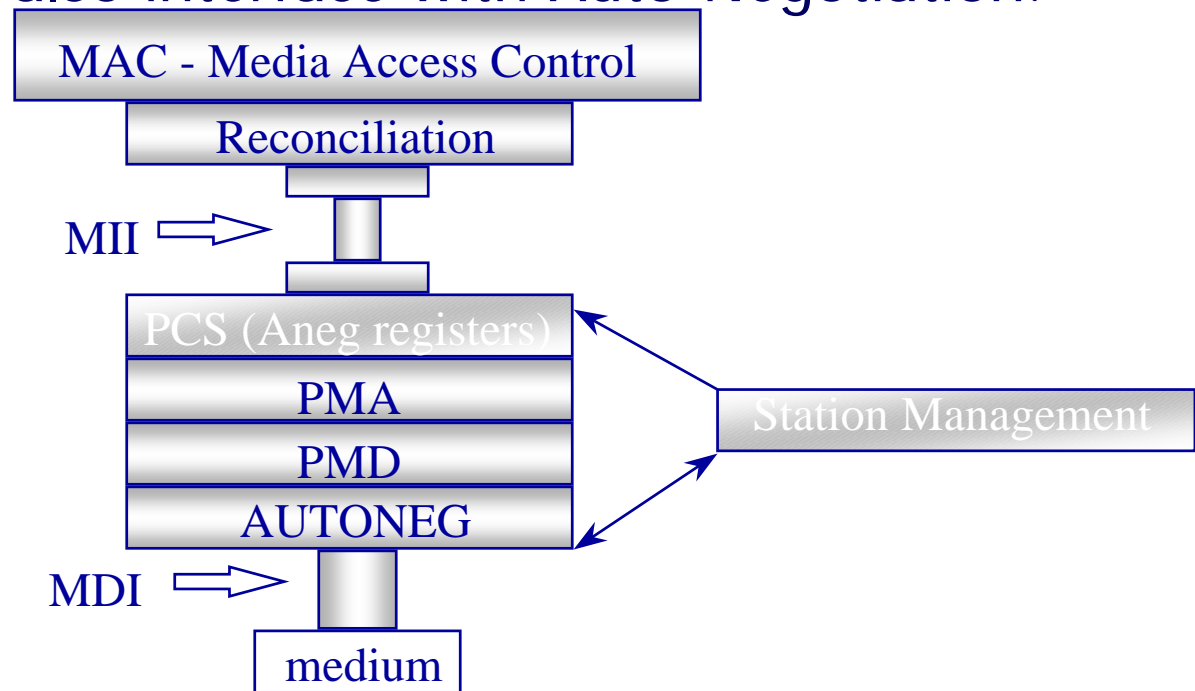
Auto-Negotiation working for you

- Takes the operational modes of the local device, and receives the operational modes from the link partner and determines the highest shared operational mode to link at.
- Works with 10BASE-T link pulses. Signaling that is independent of the speed that each device wants to link at.



Interface with the MAC

- While Auto-Negotiation takes place before any PMD or PMA is activated, the management control registers lie higher up in the PHY. This allows communication with the MAC through the Reconciliation layer. Other Physical Sublayers can also interface with Auto-Negotiation.



Management

- The PHY management controls the Auto-Negotiation process
 - A series of Registers store information on the operation modes possessed by a device
 - These Registers also allow for a PHY to turn on and off PMAs as well as restarting the Auto-Negotiation process.



Management Functions

- Register Set

Register	Name	Description
0	Control	
1	Status	
2,3	PHY identifier	
4	Auto-Negotiation Advertisement	
5	Auto-Negotiation Link Partner Base Page Ability	
6	Auto-Negotiation Expansion	
7	Auto-Negotiation Next Page Transmit	
8	Auto-Negotiation Link Partner Received Next Page	
9	MASTER-SLAVE control register	
10	MASTER-SLAVE status register	
11-14	Reserved	
15	Extended Status	
16-31	Vendor Specific	



Management Functions

- Control Register

Bit(s)	Name	Description	R/W
0.15	Reset	1 = PHY reset 0 = Normal Ops	R/W
0.14	Loopback	1 = Enable Loopback 0 = Disable	R/W
0.13	Speed Selection (LSB)	0.6 0.13 1 1 = Reserved 1 0 = 1000Mb/s 0 1 = 100Mb/s 0 0 = 10Mb/s	R/W
0.12	Auto-Neg Enable	1 = Enable 0 = Disable	R/W
0.11	Power Down	1 = Power Down 0 = Normal Ops	R/W
0.1	Isolate	1 = Isolate PHY 0 = Normal Ops	R/W
0.9	Restart Auto-Neg	1 = Restart 0 = Normal Operation	R/W
0.8	Duplex Mode	1 = Full Duplex 0 = Half Duplex	R/W
0.7	Collision Test	1 = Enable Col Test 0 = Disable	R/W
0.6	Speed Selection (MSB)	0.6 0.13 1 1 = Reserved 1 0 = 1000Mb/s 0 1 = 100Mb/s 0 0 = 10Mb/s	R/W
0.5:0	Reserved		R/W



Management Functions

- Status Register

Bit(s)	Name	Description	R/W
1.15	100Base-T4	1 = if PHY able 0 = otherwise	RO
1.14	100Base-X Full Duplex	1 = if PHY able 0 = otherwise	RO
1.13	100Base-X Half Duplex	1 = if PHY able 0 = otherwise	RO
1.12	10Mb/s Full Duplex	1 = if PHY able 0 = otherwise	RO
1.11	10Mb/s Half Duplex	1 = if PHY able 0 = otherwise	RO
1.1	100Base-T2 Full Duplex	1 = if PHY able 0 = otherwise	RO
1.9	100Base-T2 Half Duplex	1 = if PHY able 0 = otherwise	RO
1.8	Extended Status	1 = Extended Status in Reg 15	RO
1.7	Reserved	Ignore when read	RO
1.6	MF Preamble Suppress	1 = Will accept Man frames w/o Preamble 0 = Will not	RO
1.5	Auto-Negotiation	1 = Aneg Complete 0 = Otherwise	RO
1.4	Remote Fault	1 = Remote Fault Detected	RO
1.3	Auto-Negotiation Ability	1 = PHY Able to perform Aneg	RO
1.2	Link Status	1 = Link is up 0 = Link is down	RO
1.1	Jabber Detect	1 = Jabber Detected 0 = Not Detected	RO
1	Extended Capability	1 = Extended Register 0 = Normal	RO



Management Functions

- Extended Status Register

Bit(s)	Name	Description	R/W
15.15	1000BASE-X Full Duplex	1 = if PHY able 0 = otherwise	RO
15.14	1000BASE-X Half Duplex	1 = if PHY able 0 = otherwise	RO
15.13	1000BASE-T Full Duplex	1 = if PHY able 0 = otherwise	RO
15.12	1000BASE-T Half Duplex	1 = if PHY able 0 = otherwise	RO
15.11:0	Reserved	Ignore when read	RO



Management Functions

- Auto-Negotiation Advertisement Register

Bit(s)	Name	Description	R/W
4.15	Next Page	Desire to Transmit additional pages	R/W
4.14	Acknowledge	Link Partner Page seen indication	RO
4.13	Remote Fault	Remote Fault Indication	R/W
4.12	Reserved		
4.11	Asynchronous Pause	1 = Supported 0 = Not Supported	
4.10	PAUSE	0 = Supported 0 = Not Supported	RO
4.9	100BASE-T4	1 = Supported 0 = Not Supported	R/W
4.8	100 Mb/s Full Duplex	1 = Supported 0 = Not Supported	R/W
4.7	100 Mb/s Half Duplex	1 = Supported 0 = Not Supported	R/W
4.6	10 Mb/s Full Duplex	1 = Supported 0 = Not Supported	R/W
4.5	10 Mb/s Half Duplex	1 = Supported 0 = Not Supported	R/W
4.4:0	Selector Field		RO



Management Functions

- Link Partner Base Page Ability

Bit(s)	Name	Description	R/W
5.15	Next Page	Desire to Transmit additional pages	R/W
5.14	Acknowledge	Link Partner Page seen indication	RO
5.13	Remote Fault	Remote Fault Indication	R/W
5.12	Reserved		
5.11	Asynchronous Pause	1 = Supported 0 = Not Supported	
5.10	PAUSE	0 = Supported 0 = Not Supported	RO
5.9	100BASE-T4	1 = Supported 0 = Not Supported	R/W
5.8	100 Mb/s Full Duplex	1 = Supported 0 = Not Supported	R/W
5.7	100 Mb/s Half Duplex	1 = Supported 0 = Not Supported	R/W
5.6	10 Mb/s Full Duplex	1 = Supported 0 = Not Supported	R/W
5.5	10 Mb/s Half Duplex	1 = Supported 0 = Not Supported	R/W
5.4:0	Selector Field		RO



Management Functions

- Auto-Negotiation Expansion Register

Bit(s)	Name	Description	R/W
6.15:5	Reserved		RO
6.4	Parallel Detection Fault	1 = Fault Has been Detected 0 = Not	RO/LH
6.3	Link Partner NP Able	1 = Is Able 0 = Not Able	RO
6.2	Next Page Able	1 = Is Able 0 = Not Able	RO
6.1	Page Received	1 = A New Page has been received	RO/LH
6	Link Partner A-neg Able	1 = Is Able 0 = Not Able	RO



Management Functions

- Auto-Negotiation Next Page Transmit Register

Bit(s)	Name	Description	R/W
7.15	Next Page	More Pages to Transmit	R/W
7.14	Acknowledge	Link partner page seen	RO
7.13	Message Page	1 = Message Page 0 = Unformatted Page	R/W
7.12	Acknowledge 2		R/W
7.11	Toggle	Page Identifier	RO
7.10:0	Message/Unformatted Data	Data to be transmitted	R/W



Management Functions

- Link Partner Ability Next Page Register

Bit(s)	Name	Description	R/W
8.15	Next Page	More Pages to Transmit	R/W
8.14	Acknowledge	Link partner page seen	RO
8.13	Message Page	1 = Message Page 0 = Unformatted Page	R/W
8.12	Acknowledge 2		R/W
8.11	Toggle	Page Identifier	RO
8.10:0	Message/Unformatted Data	Data to be transmitted	R/W



Management Functions

- Master/Slave Control Register

Bit(s)	Name	Description	R/W
9.15-13	Test Mode bits	1 = if PHY able 0 = otherwise	R/W
9.12	MASTER-SLAVE Manual Config Enable	1 = Enable Manual Config	R/W
9.11	MASTER-SLAVE Config Value	1 = Manual Master 0 = Manual Slave	R/W
9.10	Port Type	1 = Multiport 0 = Single Port	R/W
9.9	1000BASE-T Full Duplex	Advertise if = 1	R/W
9.8	1000BASE-T Half Duplex	Advertise if = 1	R/W
9.7:0	Reserved	Write as 0, Ignore when read	R/W



Management Functions

- Master/Slave Status Register

Bit(s)	Name	Description	R/W
10.15	Master Slave Configuration Fault	1 = Master Slave configuration fault detected 0 = No Fault detected	RO
10.14	Master Slave Configuration Resolution	1 = local PHY resolved to Master 0 = Local Phy resolved to Slave	RO
10.13	Local Reciever Staus	1 = local receiver status ok	RO
10.12	Remote Reciever Staus	1 = remotereceiver status ok	RO
10.11	LP 1000T Full Duplex	1 = Link partner supports 1000T full duplex	RO
10.10	LP 1000T Half Duplex	1 = Link partner supports 1000T half duplex	RO
10.9:8	Reserved	Reserved	RO
10.7:0	Idle Error Count	Cumulative count of errors detected while receiving idle	RO



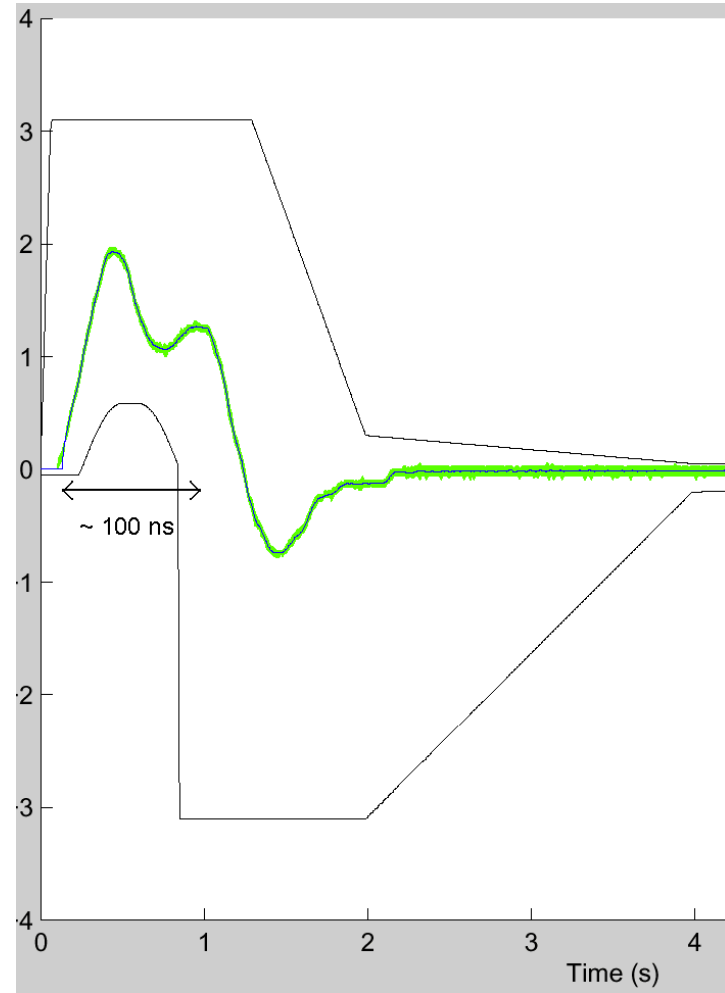
Interface with the Medium

- Auto-Negotiation uses signaling similar to that found in 10BASE-T (the lowest common denominator).
 - This allows older 10BASE-T to not be adversely affected by an Auto-Negotiating device.
 - Simple, only uses pulses.



NLPS

- This is a 10BASE-T LTP (link test pulse)
- It exactly matches an NLP (normal link pulse) sent by an Auto-Negotiating device

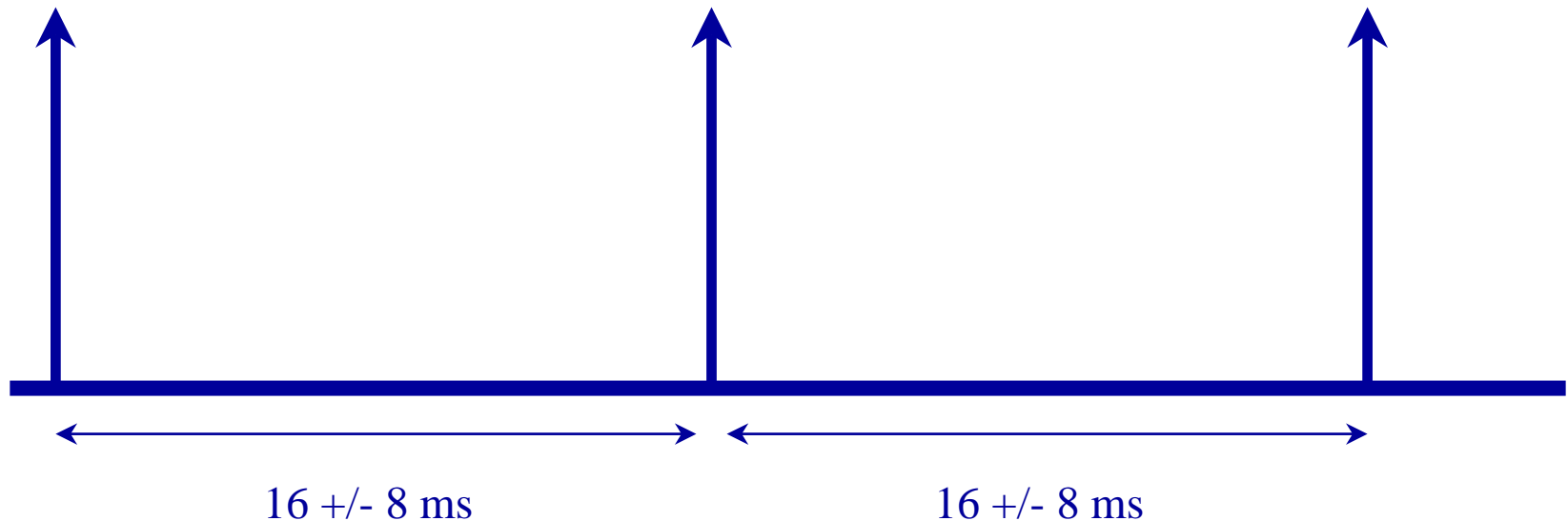


Clause 28 & 40 Auto-Negotiation



NLP Spacing

- 10 BASE-T LTPs are sent about every 16 ms

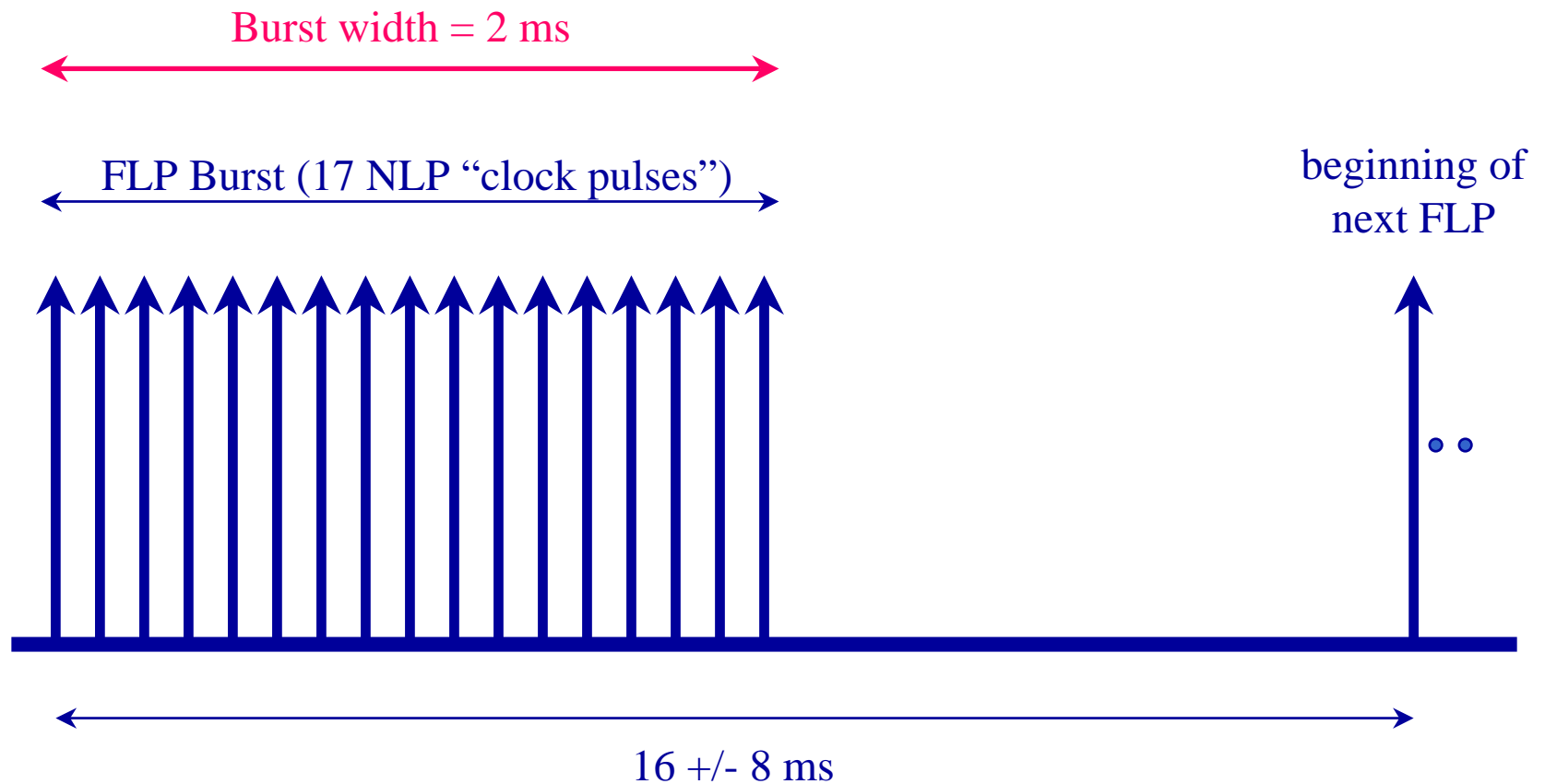


FLPs (Auto-Negotiation)

- FLPs are quick bursts of NLPs
- While a 10BASE-T device will not link with an FLP, because the NLPs are spaced too closely together it allows the 10BASE-T device to see that it is receiving NLPs and will establish a link once the Auto-Negotiating device parallel detects to the 10BASE-T device.

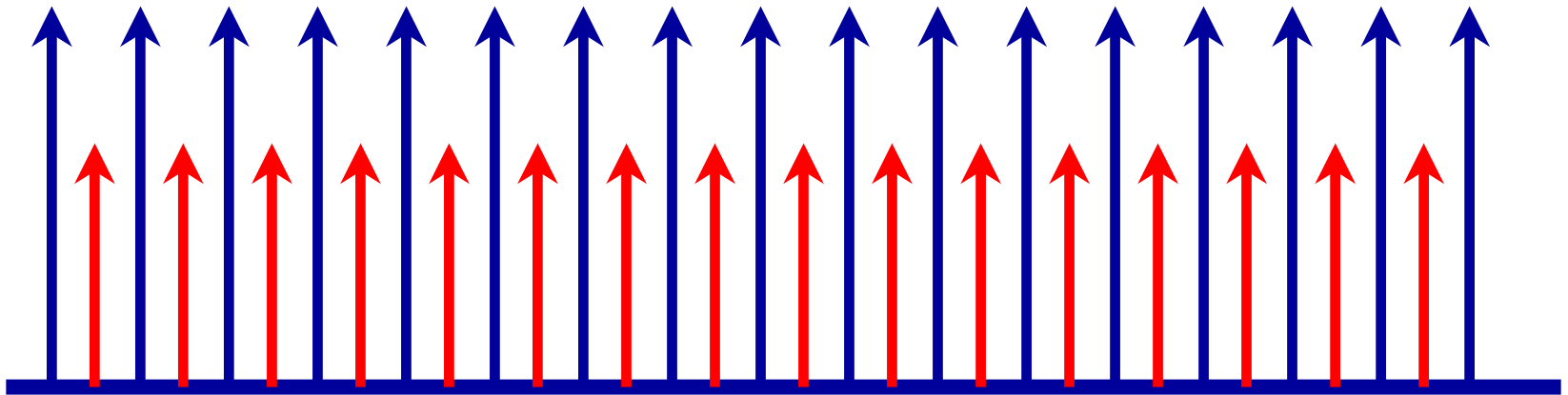


FLP Spacing

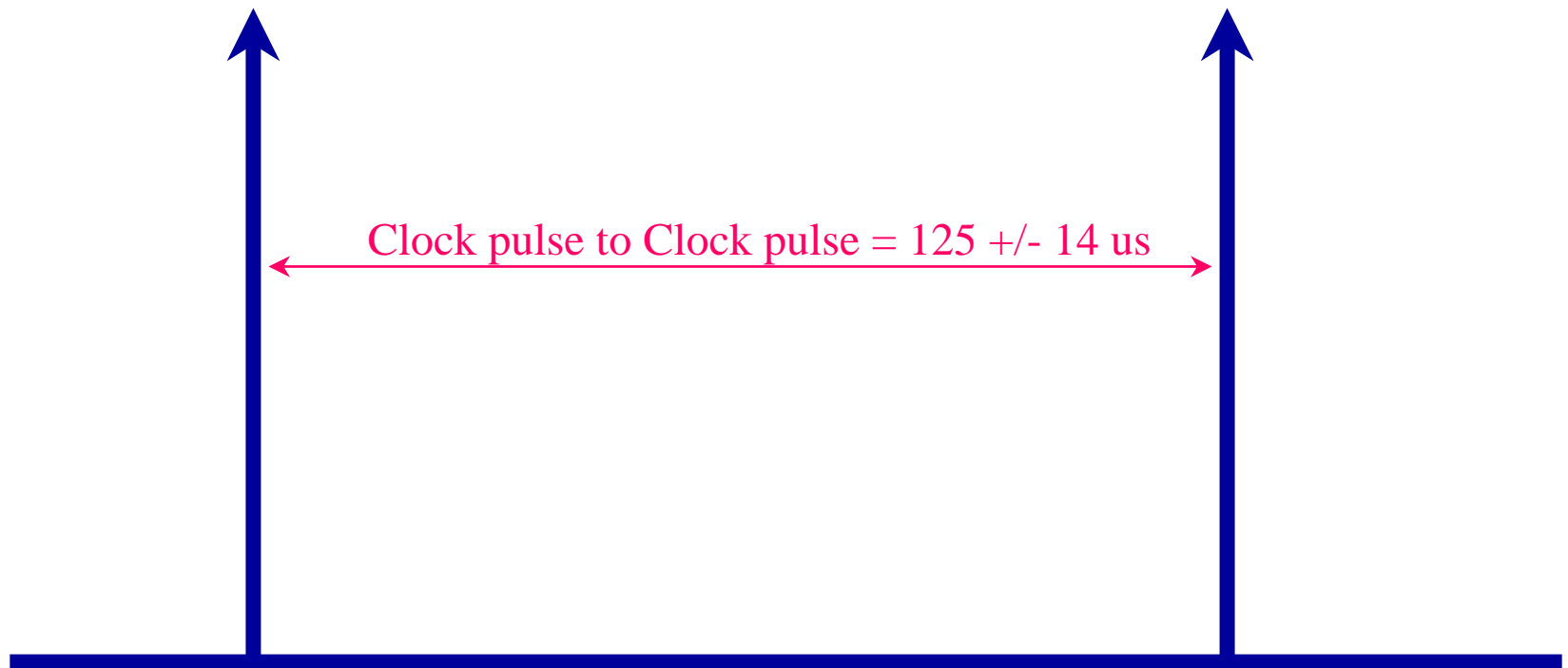


Studying the FLP

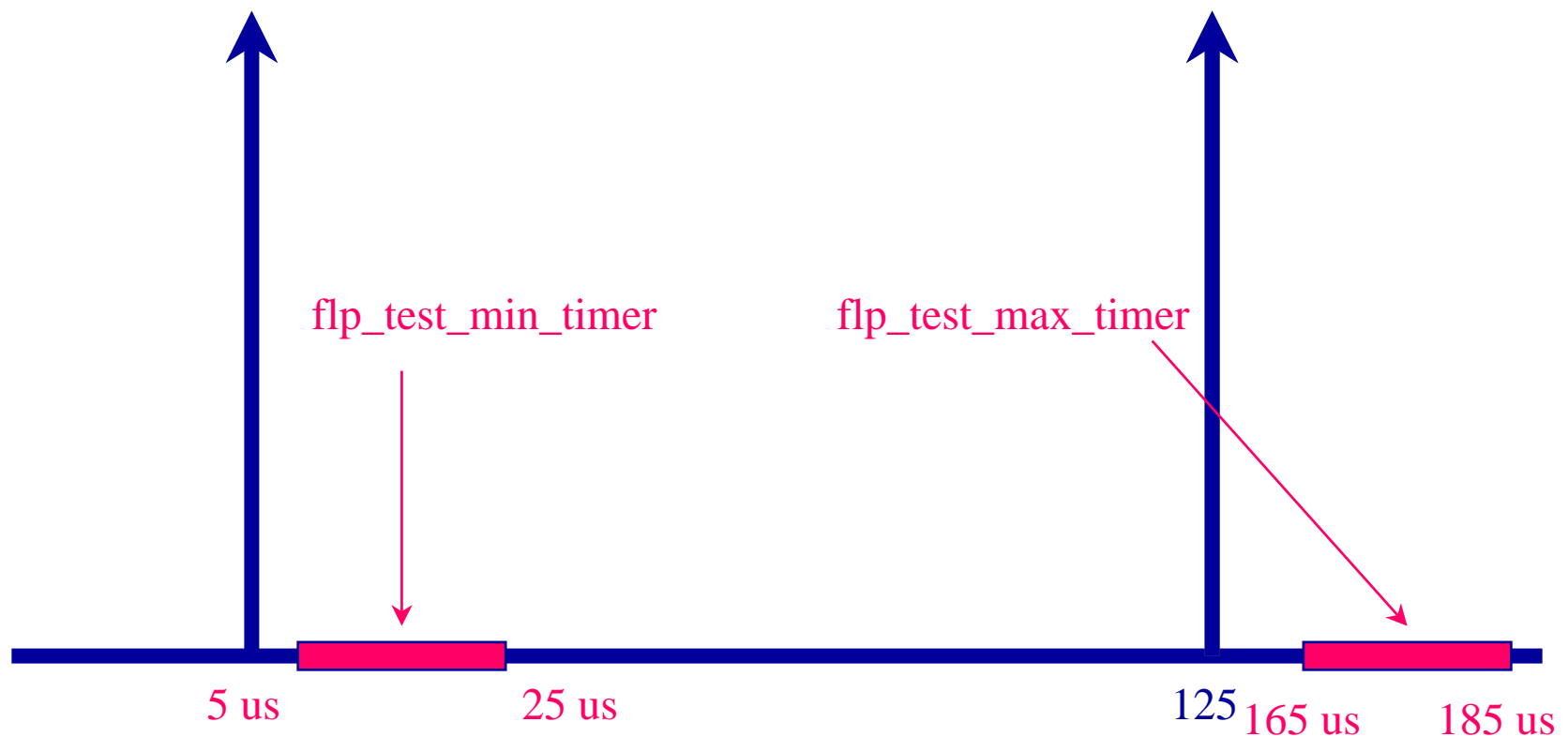
- FLPs consist of 17 Clock pulses (blue)
- Data pulses can be “stuffed” in between clock pulses (red)
- If a data pulse is found, it represents a logical “1” if no data pulse is found, it represents a logical “0”



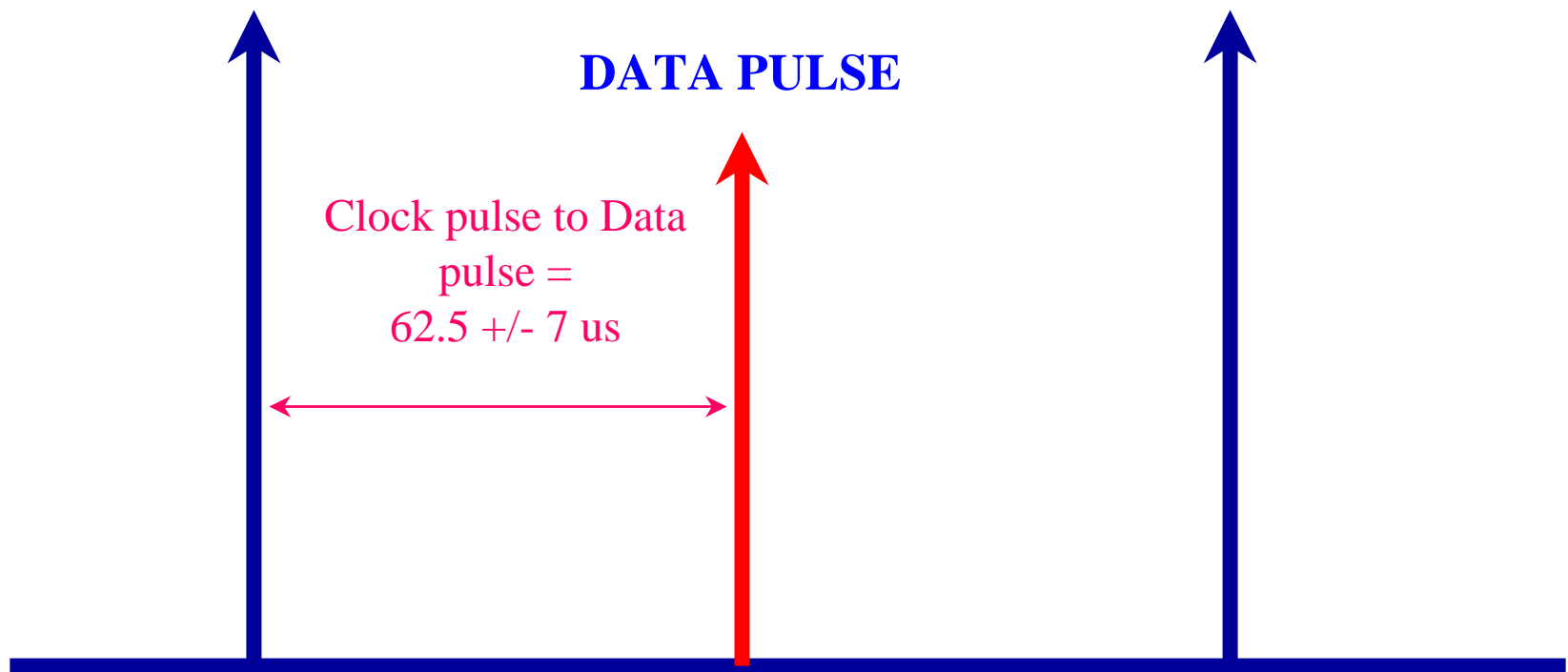
Studying the FLP



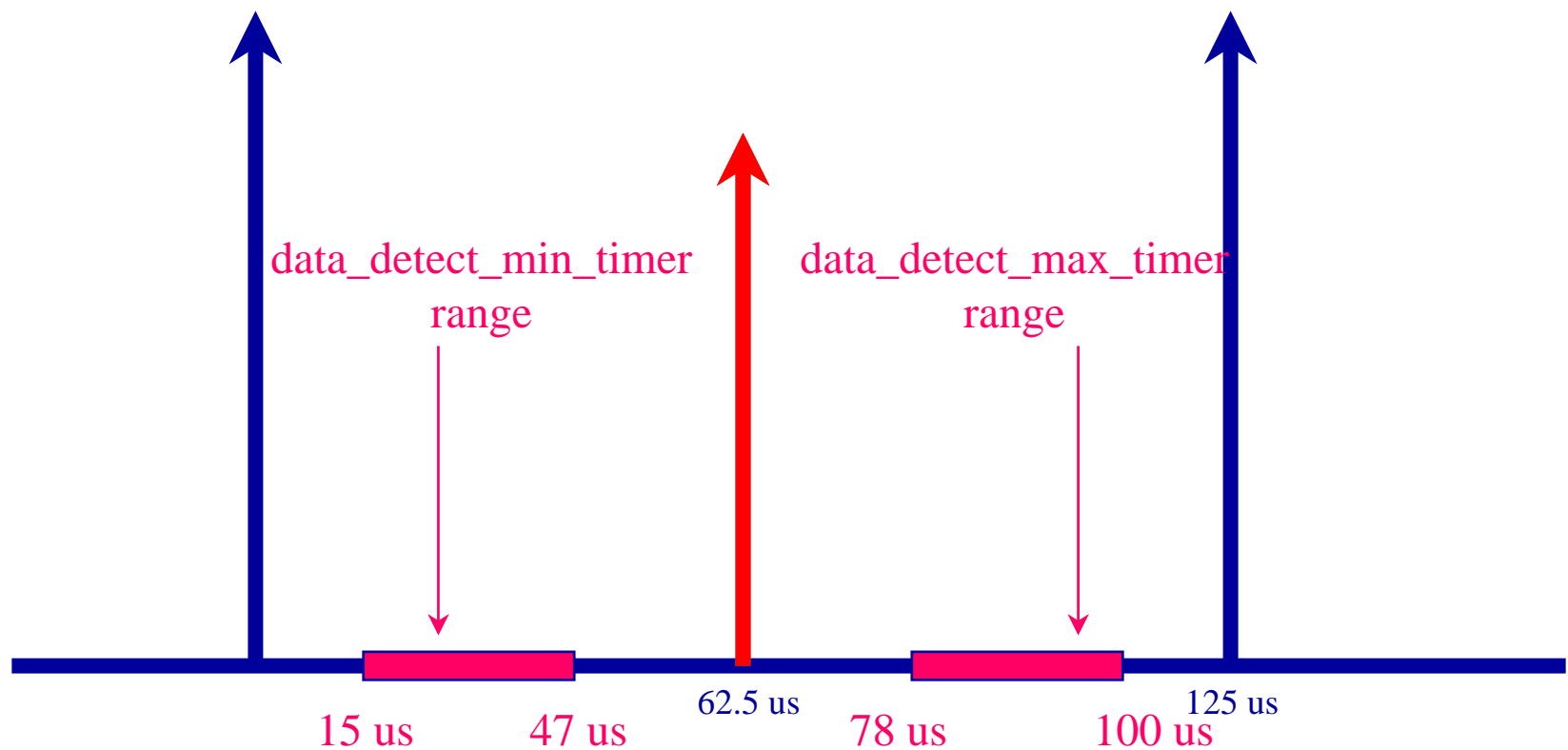
Studying the FLP



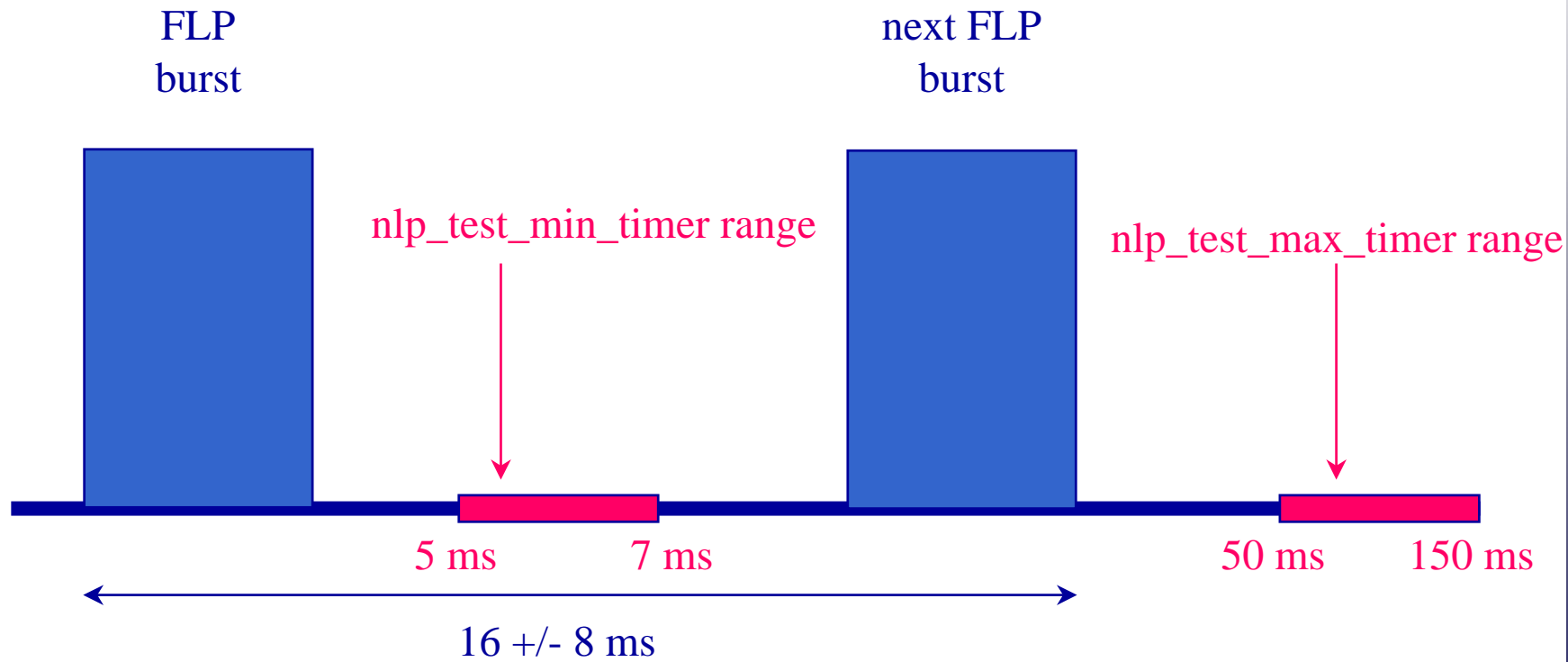
Studying the FLP



Studying the FLP



Studying the FLP



Auto-Negotiation Functionality

How Auto-Negotiation works

- Section Objectives
 - To understand how information is passed between link partners
 - To look at the structure of the data
 - To fully understand the arbitration protocol
 - To examine the State Machines and Flow charts to better understand how the process works



Remember its all about sharing

- Auto-Negotiation's purpose is to share management information before a link is established.
- To facilitate this exchange, this information is passed in 16 bit blocks at a time, each block is called a "Code Word" or "FLP Burst" or "Page".



The “Page”

- There are three kinds of pages that are exchanged during Auto-Negotiation
 - The Base Page
 - **Always** the First Page to be transmitted
 - Contains information about 10BASE-T, 100BASE-TX, 100BASE-T4, and PAUSE abilities
 - This page represents the beginning of the Auto-Negotiation Process



Pages and Pages of Info

- Next Pages (These Pages come after the base page)
 - Message Page
 - Each message page contains a code number
 - The standard defined what each code number means
 - For example, a message code 8 means that 2 unformatted pages are coming and that they will hold information on 1000BASE-T abilities
 - Unformatted Page
 - Contains 11 bits of information
 - Contents of these pages can be defined by previous message pages



“Bitwise” look at pages

- This is a bit by bit look at the three types of pages
- As you can see each is the same 16 bits long, but have slightly different configurations
- Blue indicates the Base Page, and Yellow indicates Next Pages

Page Type	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9	Bit 10	Bit 11	Bit 12	Bit 13	Bit 14	Bit 15
Base	Selector Field bits (0 - 4)					10BASE-T Half Duplex	10BASE-T Full Duplex	100BASE-TX Half Duplex	100BASE-TX Full Duplex	100BASE-T4	PAUSE	Asynchronous PAUSE	Not Defined	Remote Fault	Acknowledge	Next Page
Message	Message Code bits (0 - 10)											Toggle	Ack2	Message Page	Acknowledge	Next Page
Unformatted	Data bits (0 - 10)											Toggle	Ack2	Message Page	Acknowledge	Next Page



“Bitwise” look at pages

Page Type	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9	Bit 10	Bit 11	Bit 12	Bit 13	Bit 14	Bit 15
Base	Selector Field bits (0 - 4)					10BASE-T Half Duplex	10BASE-T Full Duplex	100BASE-TX Half Duplex	100BASE-TX Full Duplex	100BASE-T4	PAUSE	Asynchronous PAUSE	Not Defined	Remote Fault	Acknowledge	Next Page
Message	Message Code bits (0 - 10)											Toggle	Ack2	Message Page	Acknowledge	Next Page
Unformatted	Data bits (0 - 10)											Toggle	Ack2	Message Page	Acknowledge	Next Page

- Some of the bits are special
 - Acknowledge bit
 - Next Page Bit
 - Toggle bit
 - Message Page



“Bitwise” look at pages

- Some of the bits are special
 - Acknowledge bit
 - Lets the Link Partner know that you received his last page
 - Next Page Bit
 - Lets the LP know you want to send more pages after this one
 - Toggle bit
 - Allows the LP to tell the difference between successive pages
 - Message Page
 - Tells the LP that this page is contains a message code, so he knows what is coming next.



More than just the Base Page

- A next page exchange allows more information to be transmitted
 - This allows for 1000BASE-T, 10GBASE-T(coming soon), and other new technologies.
- A next page exchange works through the use of the next page bit (the last bit in every page)
- When this bit is set, it indicates that the link partner wants to send additional pages



The Next Page Exchange process

- If the next page bit is set, after the complete acknowledge state has been finished during the base page exchange, the link partners send the next page instead of link signaling.
- Then the same acknowledge process as in the base page exchange follows
- Pages can be differentiated by the toggle bit (each successive page alternates the toggle bit)
- If one device runs out of pages to send, then it can send a "Null Message" page(code #1) until the link partner has finished sending all of its pages

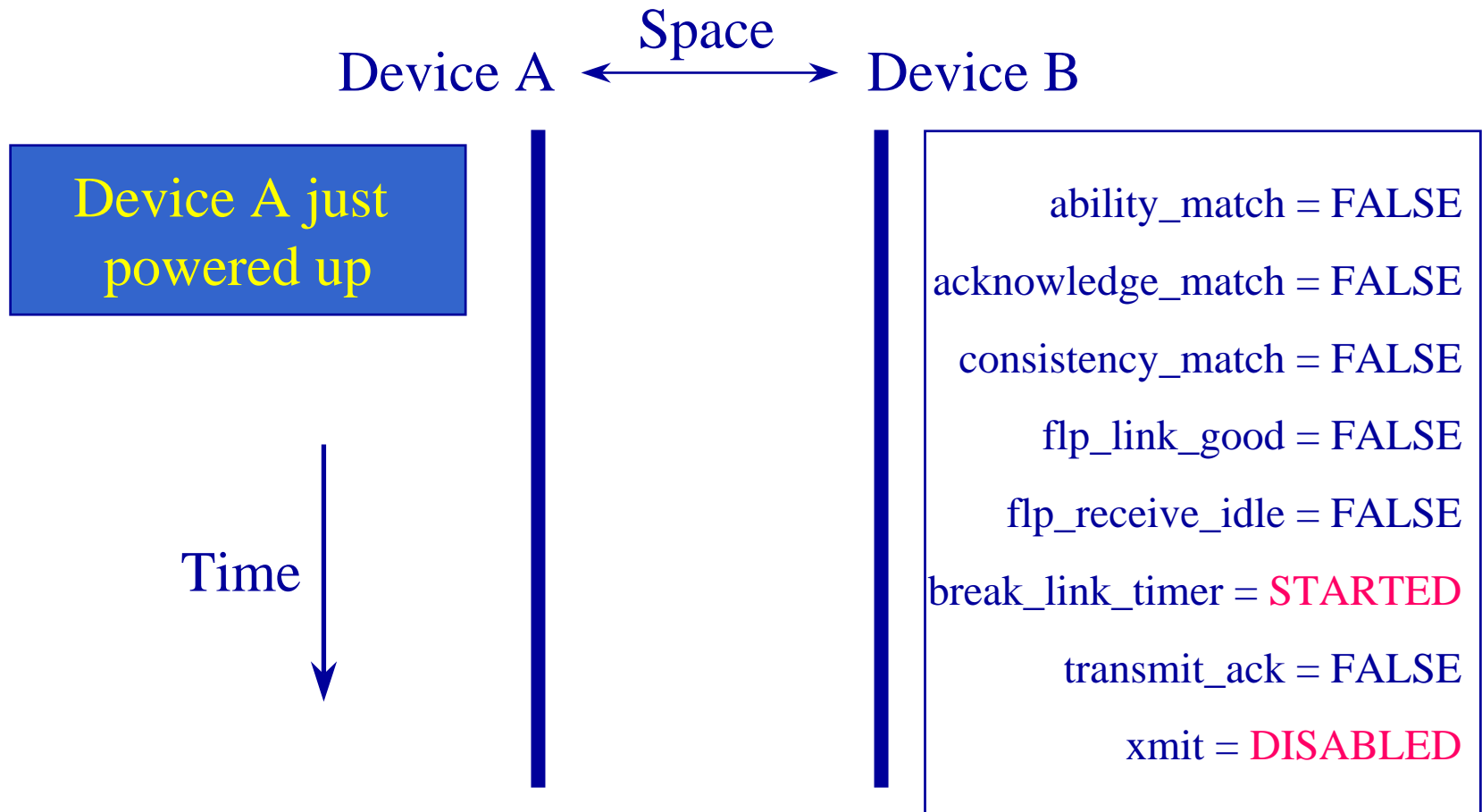


Typical Next Page exchanges

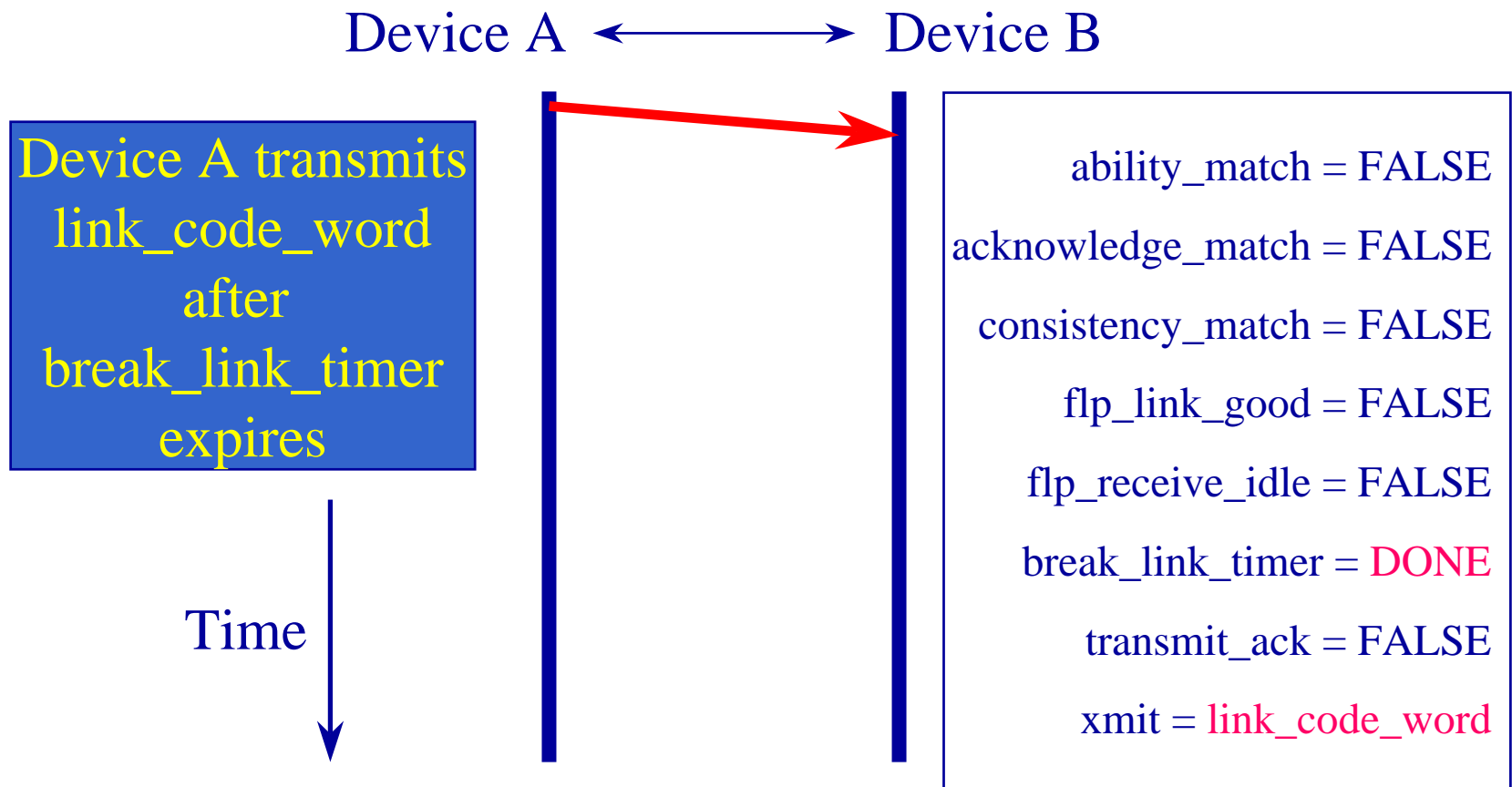
- A message code 8 page, followed by 2 unformatted pages.
 - This code stands for a 1000BASE-T abilities exchange.
- A message code 7 page, followed by 1 unformatted page
 - This code stands for a 100BASE-T2 exchange
- A message code 4 page, followed by 1 unformatted page
 - This code indicates that the unformatted page will contain information about a Remote Fault condition



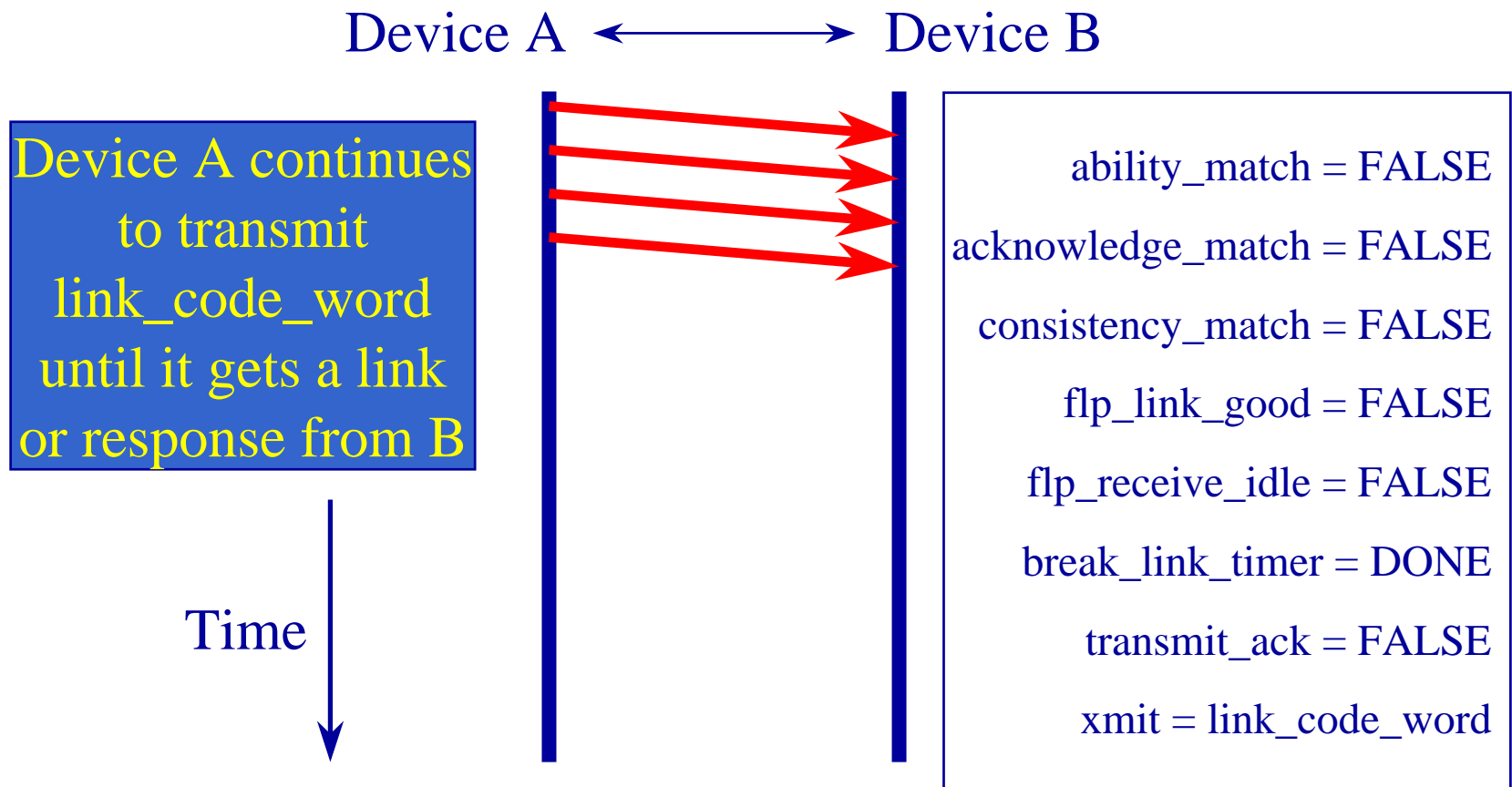
The Arbitration Protocol



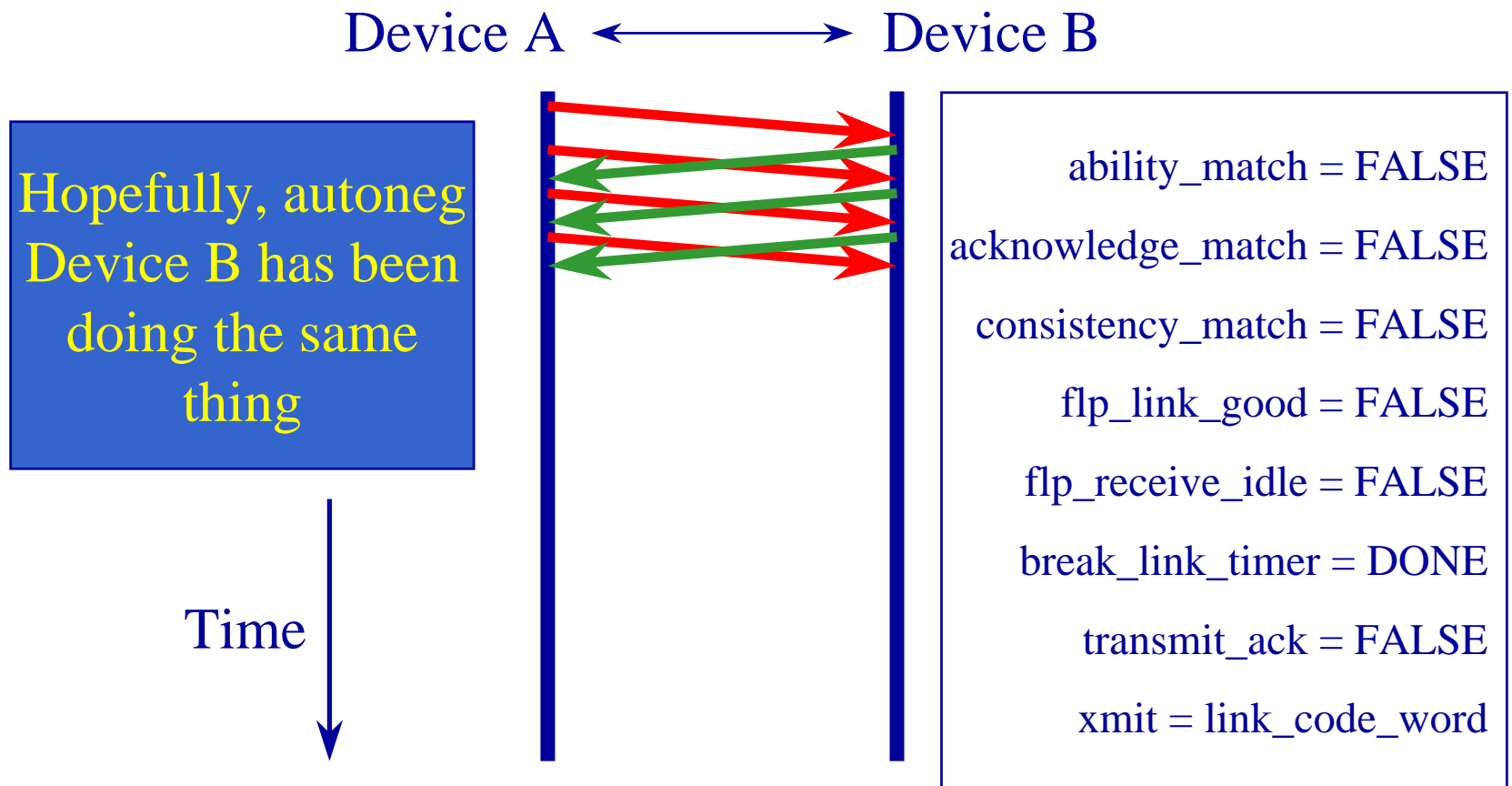
The Arbitration Protocol



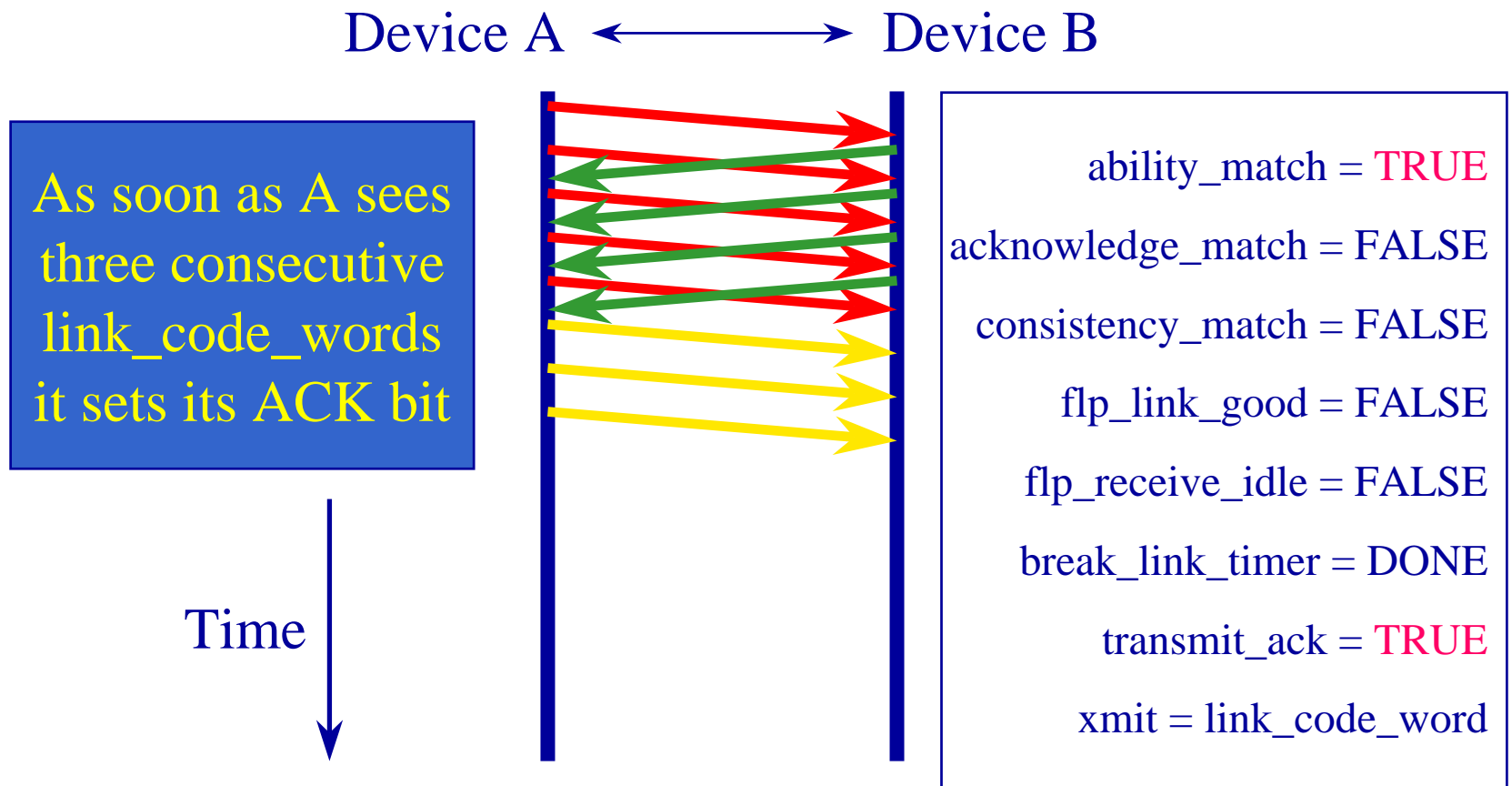
The Arbitration Protocol



The Arbitration Protocol

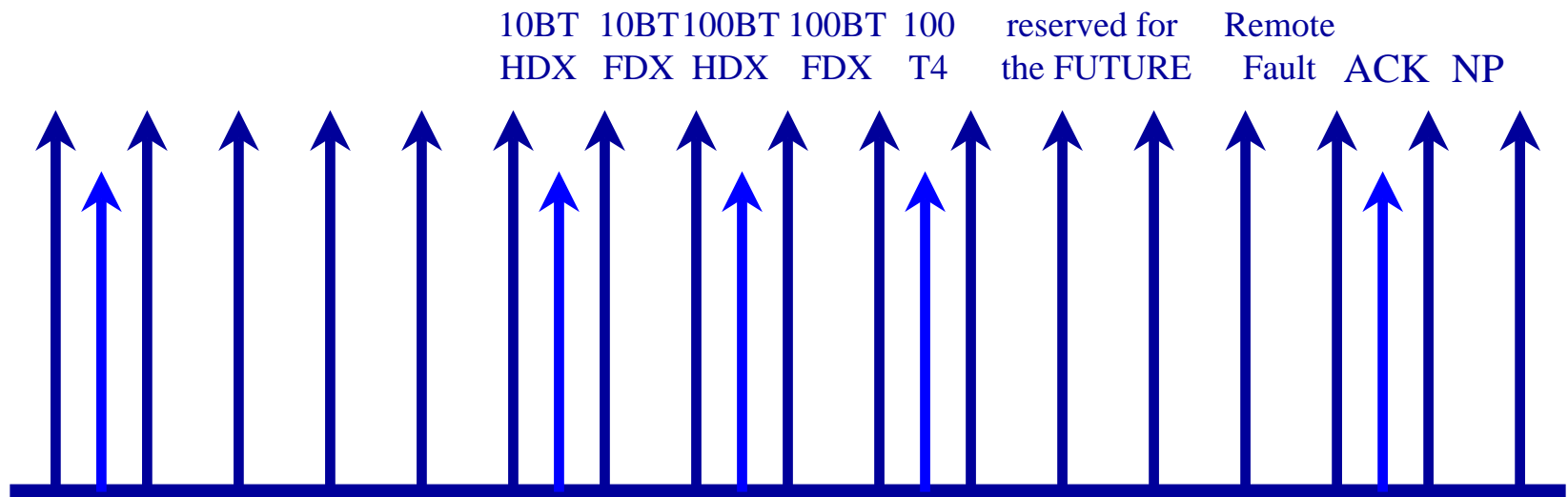


The Arbitration Protocol

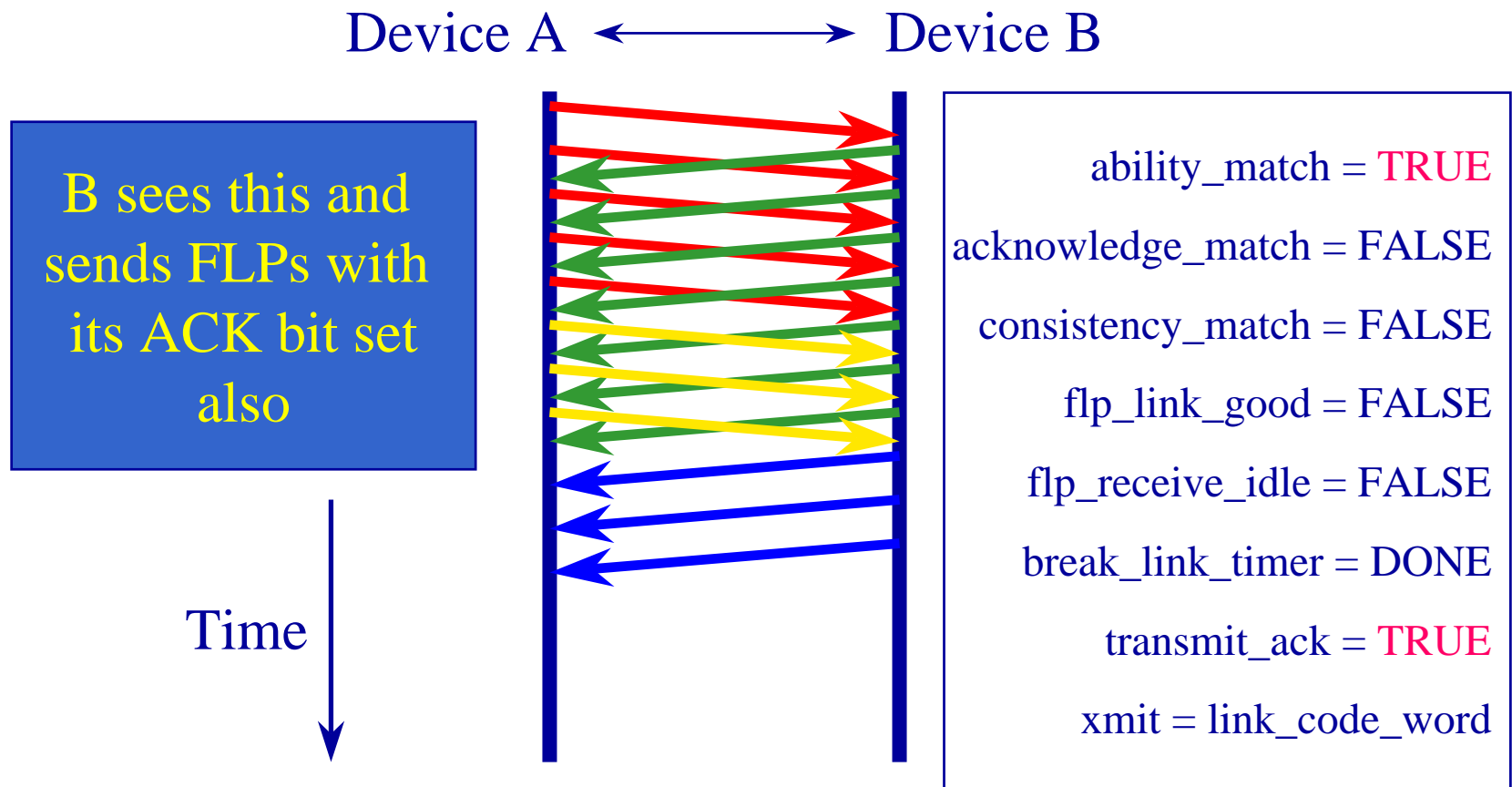


FLP review

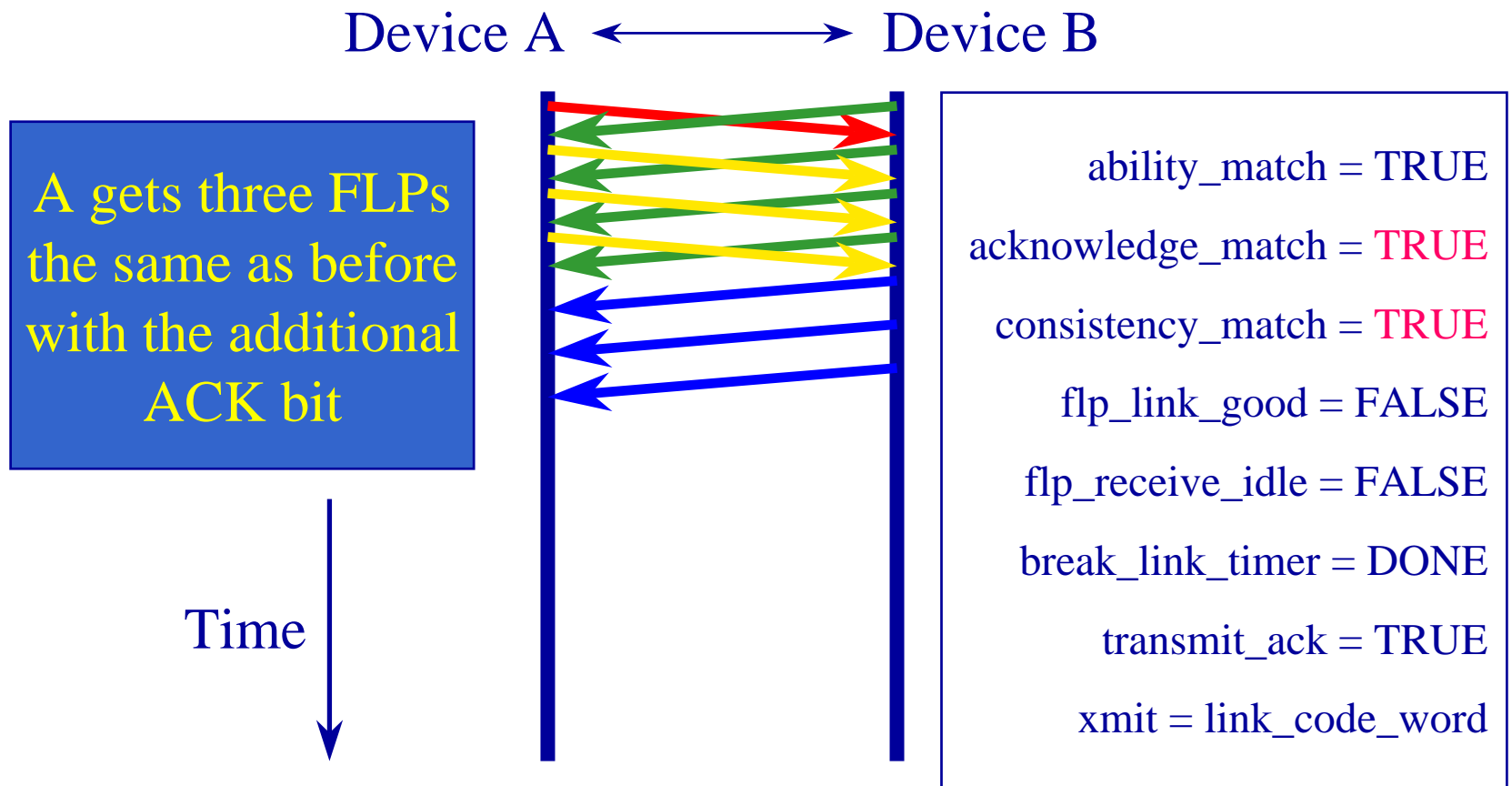
← Selector Field → ← Technology Ability Field →



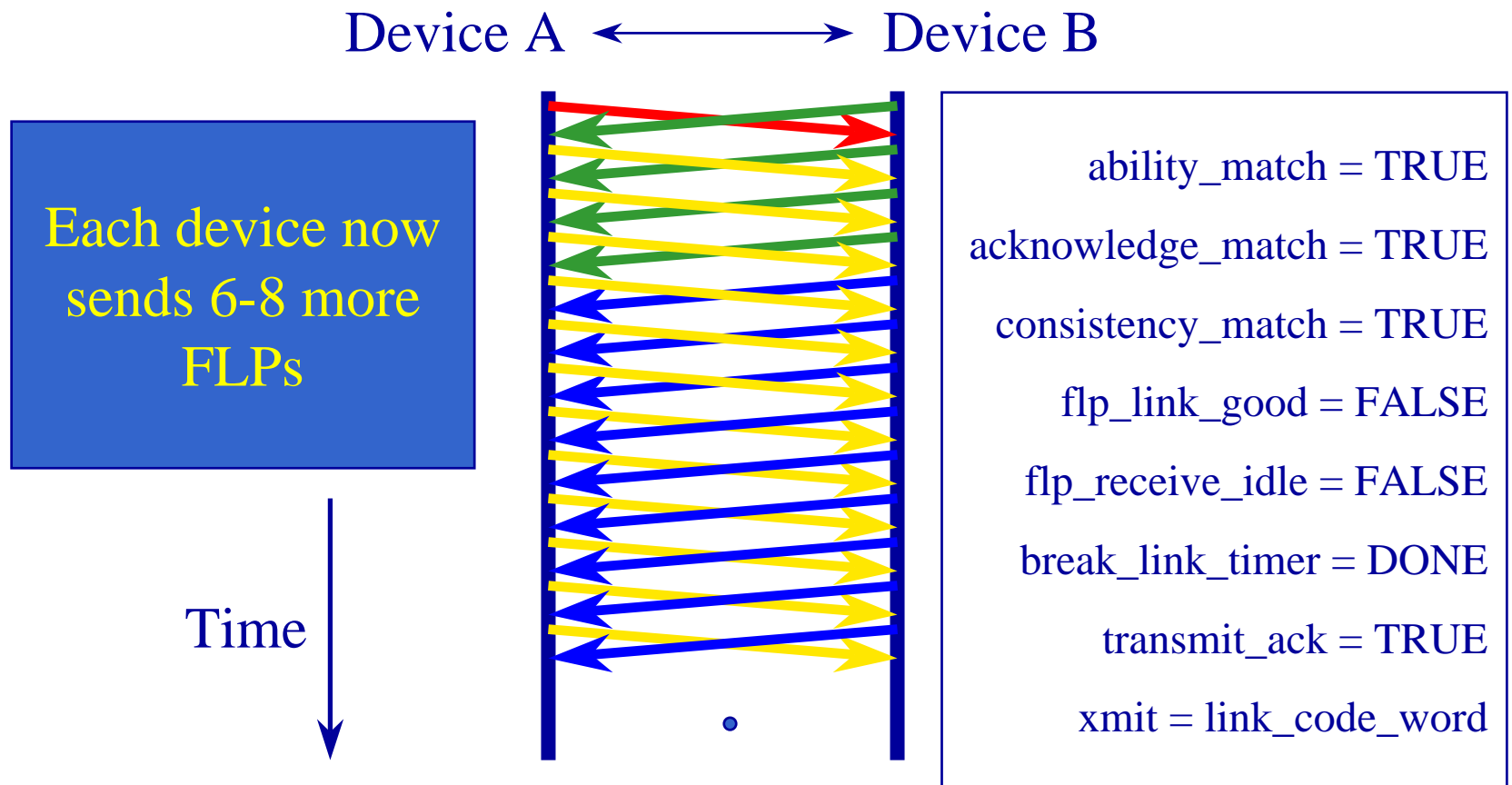
The Arbitration Protocol



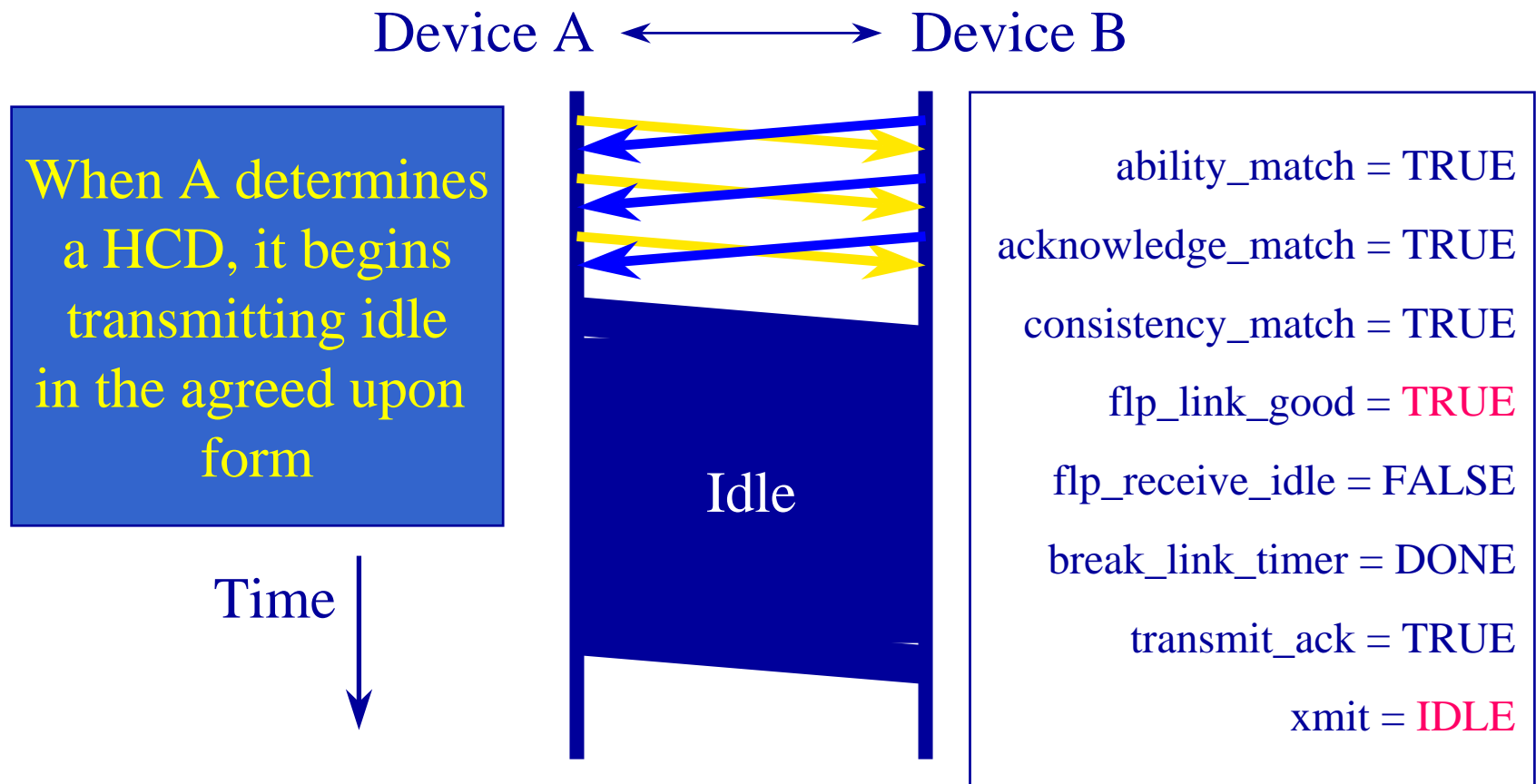
The Arbitration Protocol



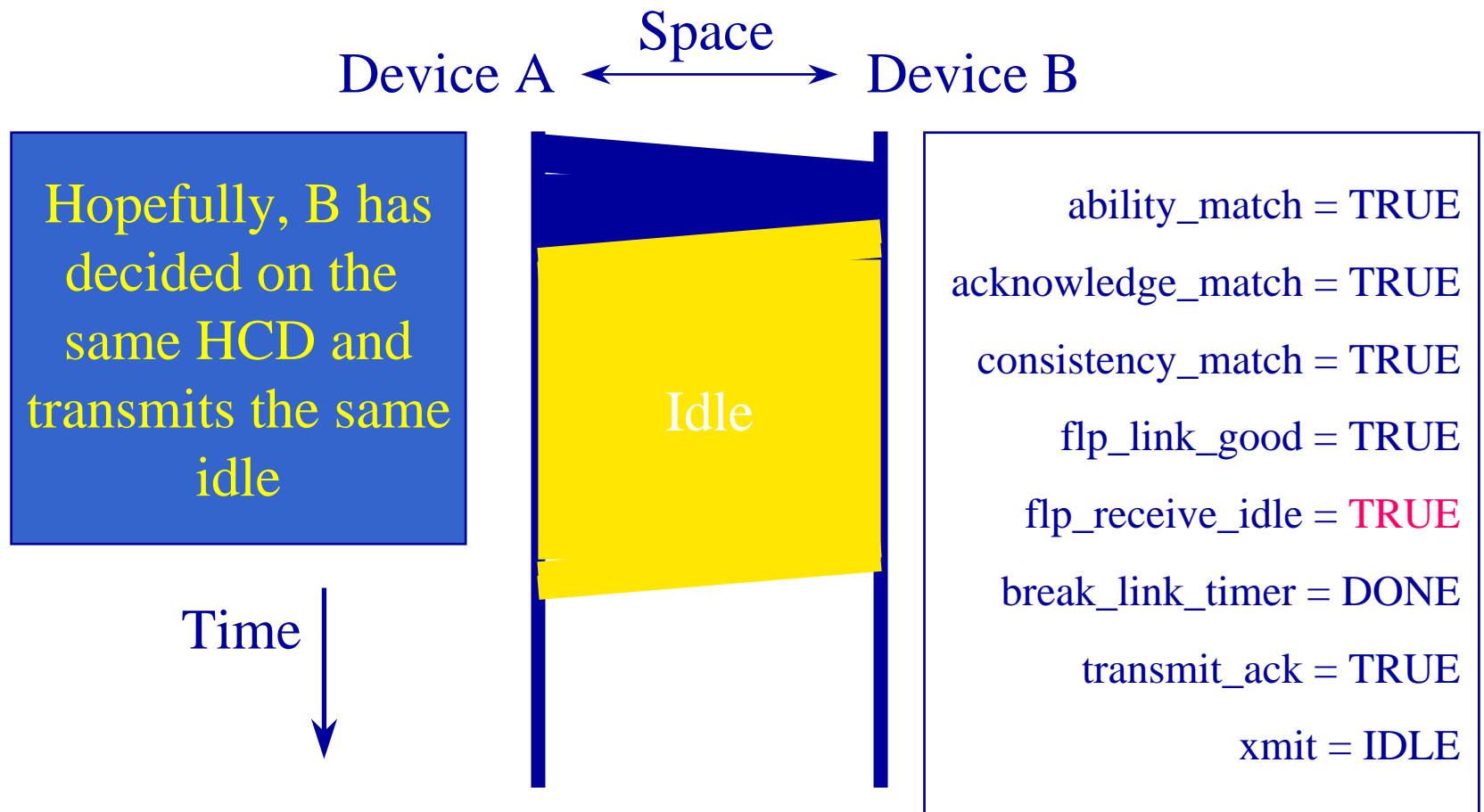
The Arbitration Protocol



The Arbitration Protocol



The Arbitration Protocol

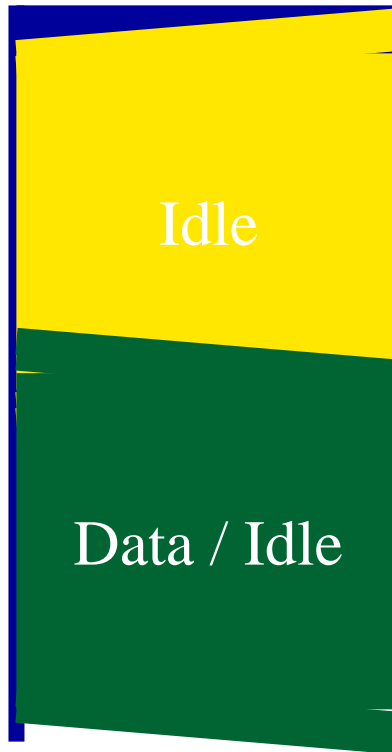


The Arbitration Protocol

Device A \longleftrightarrow Device B

Once
link_fail_inhibit_timer
is done, xmit is set to
DATA, allowing the
data codes to be
sent or received

Time



ability_match = TRUE
acknowledge_match = TRUE
consistency_match = TRUE
flp_link_good = TRUE
flp_receive_idle = TRUE
break_link_timer = DONE
transmit_ack = TRUE
xmit = DATA



Transmission of pages

- Flow of Auto-Negotiation
 - 1) Base Page Transmission
 - 2) Transmit a Message Page or if there are no next pages to send goto 4
 - 3) Transmit an the number of unformatted pages as described by the message page: if more next pages goto 2
 - 4) Transmit appropriate link signaling



16 bits to Hex

- To make it easier to specify a device's abilities, each page can be encoded to hexadecimal
- This makes discussing a device's abilities much easier
- Instead of saying what each device can support, you can just give the hex value of the page.



A page in Hex

- Here the mapping of 01E1 and 41E1 is illustrated

Selector Field Bits				Technology Ability Field Bits				rsvd	RF	Ack	NP
---------------------	--	--	--	-------------------------------	--	--	--	------	----	-----	----

LSB

MSB

D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15

1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1 E 1 0

LSB

MSB

D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15

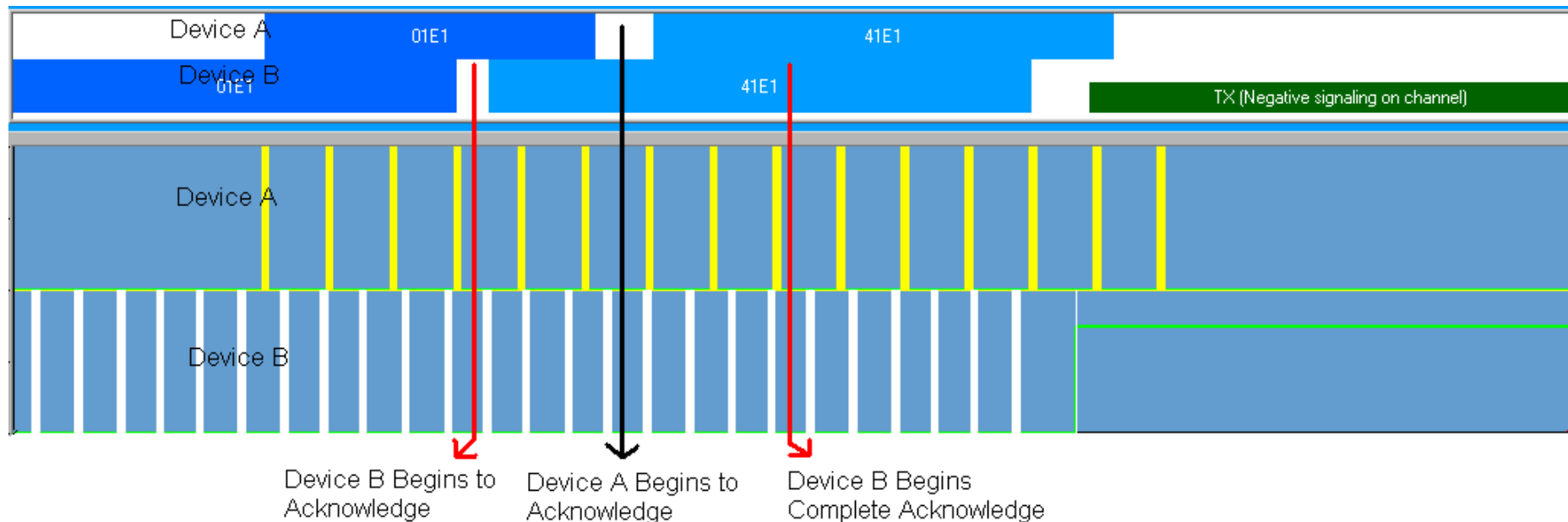
1	0	0	0	0	1	1	1	1	0	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1 E 1 4



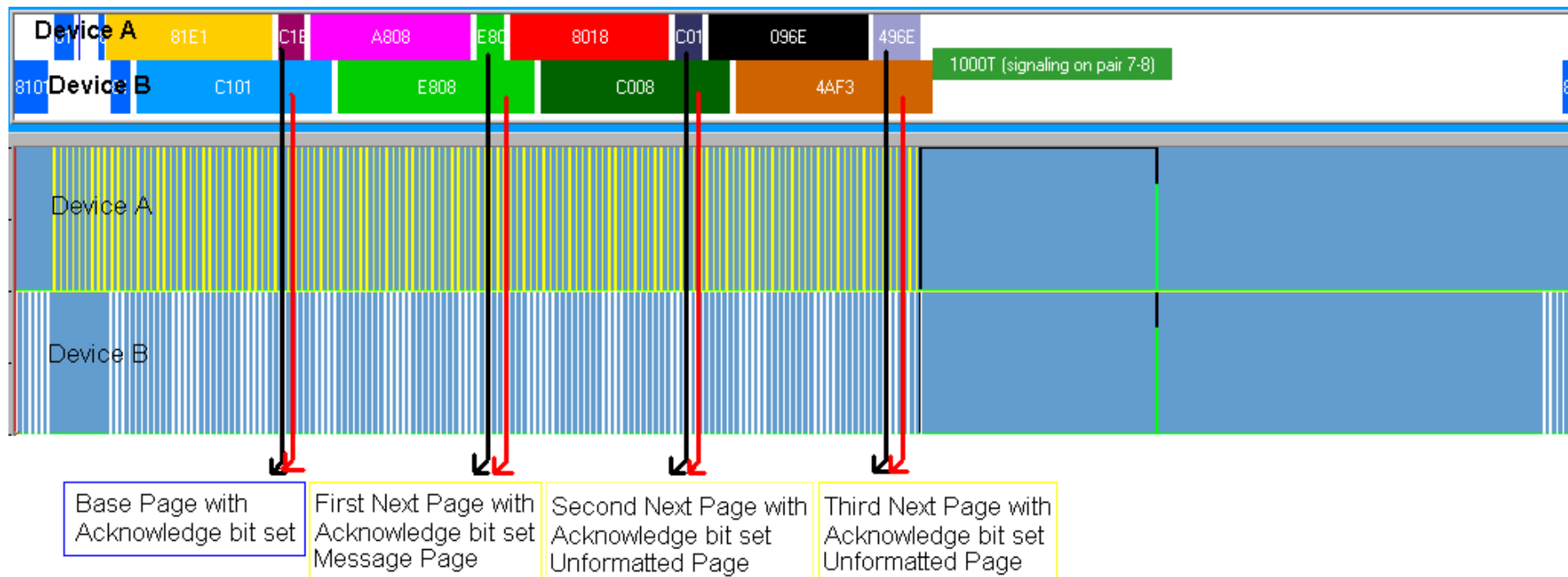
A Simple Exchange

- To link partners want to make a simple exchange to determine the optimal operational mode and link



More complicated Exchange

- This more complicated exchange does the following:
 - Sends 1000BASE-T abilities and Determines Master/Slave status of a 1000BASE-T link



Exchange Wrap up

- Devices exchange 1 page at a time, and use the acknowledge bit to indicate that a page has been received.
- More than a single page can be exchanged through the use of the Next Page bit and Toggle bit.
- After the exchange is complete both link partners send out signaling according to the best shared operational mode



The total Aneg package (Time to wake back up)

- Flow of Auto-Negotiation
 - Why is auto-negotiation set up the way it is? (The 2 islands example)
 - The real Clause 28 State Diagrams



The two islands

- Lets look at an example.
 - Two men are stranded on two different islands. Each one is very lonely and wants to talk to someone.
 - So one day the first man “Dave” carves a note into a piece of bark and throws it into the ocean.
 - To his great surprise and excitement many days later a piece of bark floats onto his beach, but when he picks it up, the note he finds is in a different language.



The two islands

- Dave isn't about to give up, so he sits down and tries to figure out a way to communicate, first he decides they need to speak the same language, but he doesn't know what language(s) his friend speaks.
- While he is thinking a second piece of bark floats onto the beach. He looks at it and finds that is identical to the first one, so he still can't read it, but he realizes that the man he is talking to is very smart.
- By throwing more than one piece of bark with the same note, it increased that chance that he would receive it



The two islands

- So now Dave starts to carve a new note. He writes down each of the languages he can speak, and if he speaks it fluently or not. Then he copies it many times and throws them into the ocean.
- Days later, he starts to get pieces of bark back. His friend had gotten the idea, and all of these pieces of bark listed the languages his friend could speak.
- So he circled a language they could both speak fluently and threw them back into the ocean.



The two islands

- The next piece of bark he received was a hello note from his new friend “Lenny” in a language that he could read, and from then on they were great pen pals.

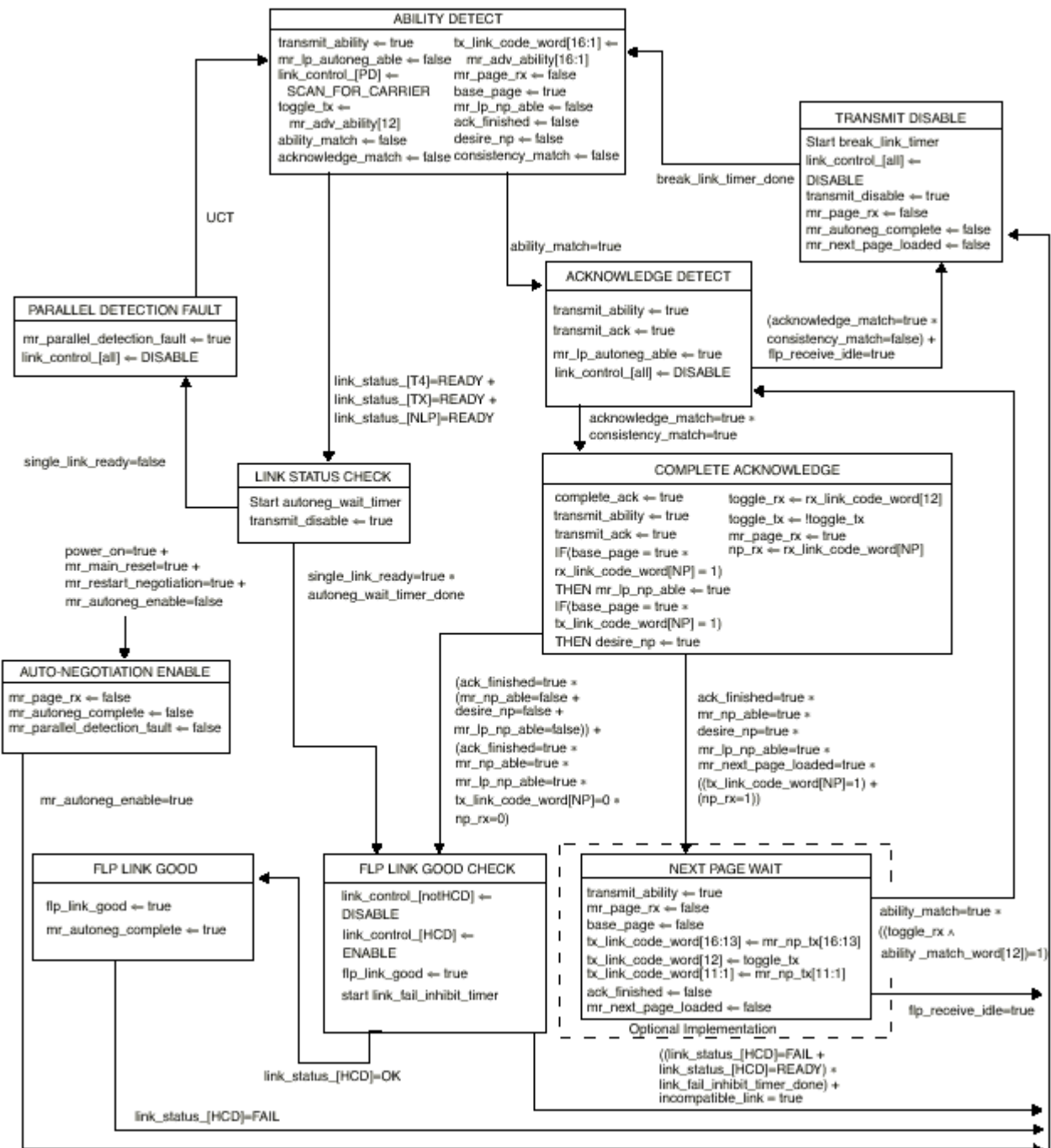


The Real State Machines

- The actual Auto-Negotiation state machines work on the same principle as the two island guys.
- It also takes into account several special events that could happen.

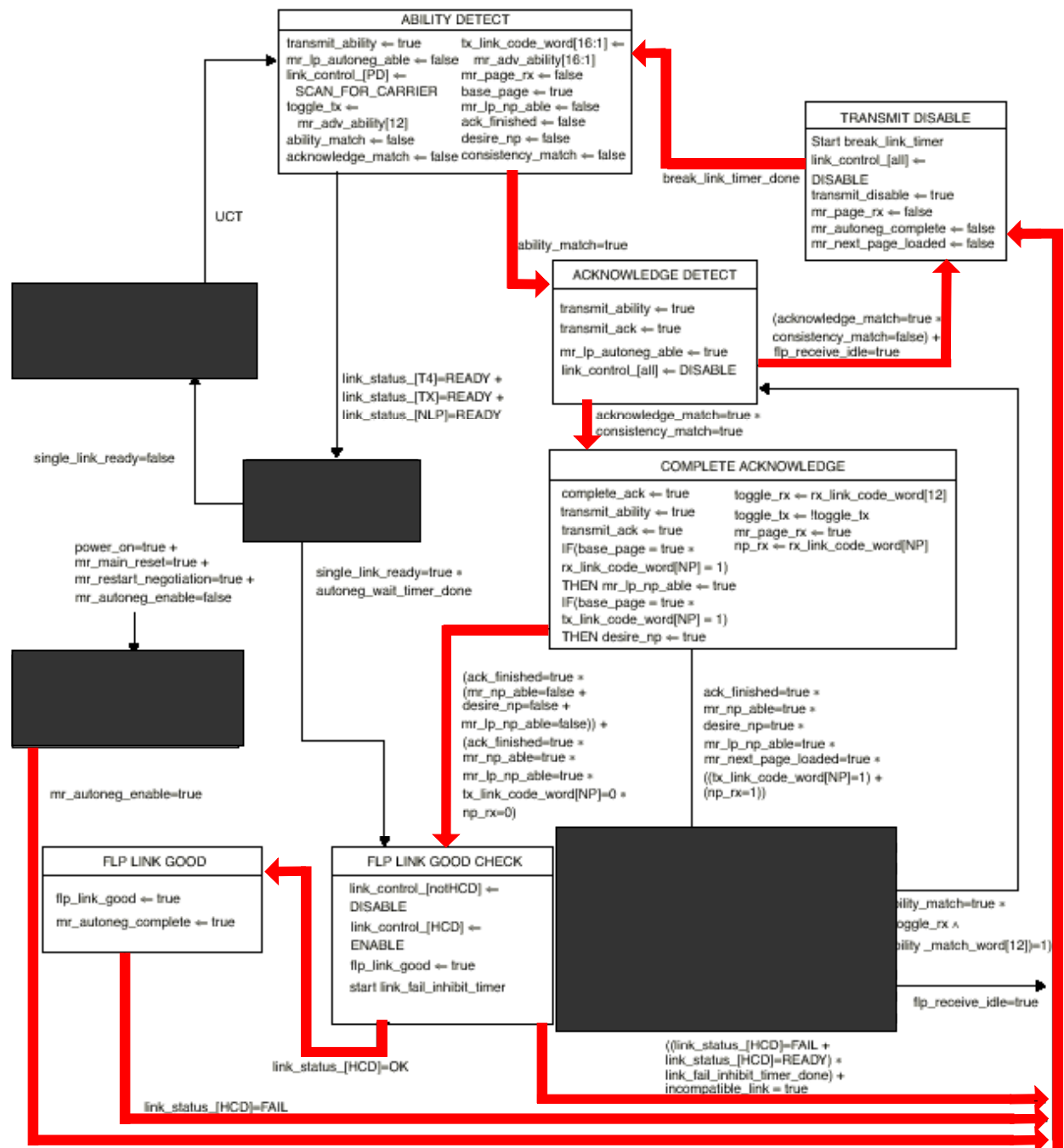


The full ANEG arbitration state Diagram



The full ANEG arbitration state Diagram

ANEG w/o Next Page or Parallel Detection



Ability Detect: The starting point of the Auto-Negotiation sequence. The device is sending out FLP's, advertising that it is a ANEG device, as well as what abilities it has.

Exit Paths:

Link_Status_Check (Parallel detection)

Acknowledge Detect (ANEG)

Ability_Match=True: The device has received 3 identical FLP's, ignoring the ACK bit.

Ability_Match=False:
The device has not received 3 identical FLP's, ignoring the ACK bit.

Ability_Match=True

ACKNOWLEDGE DETECT

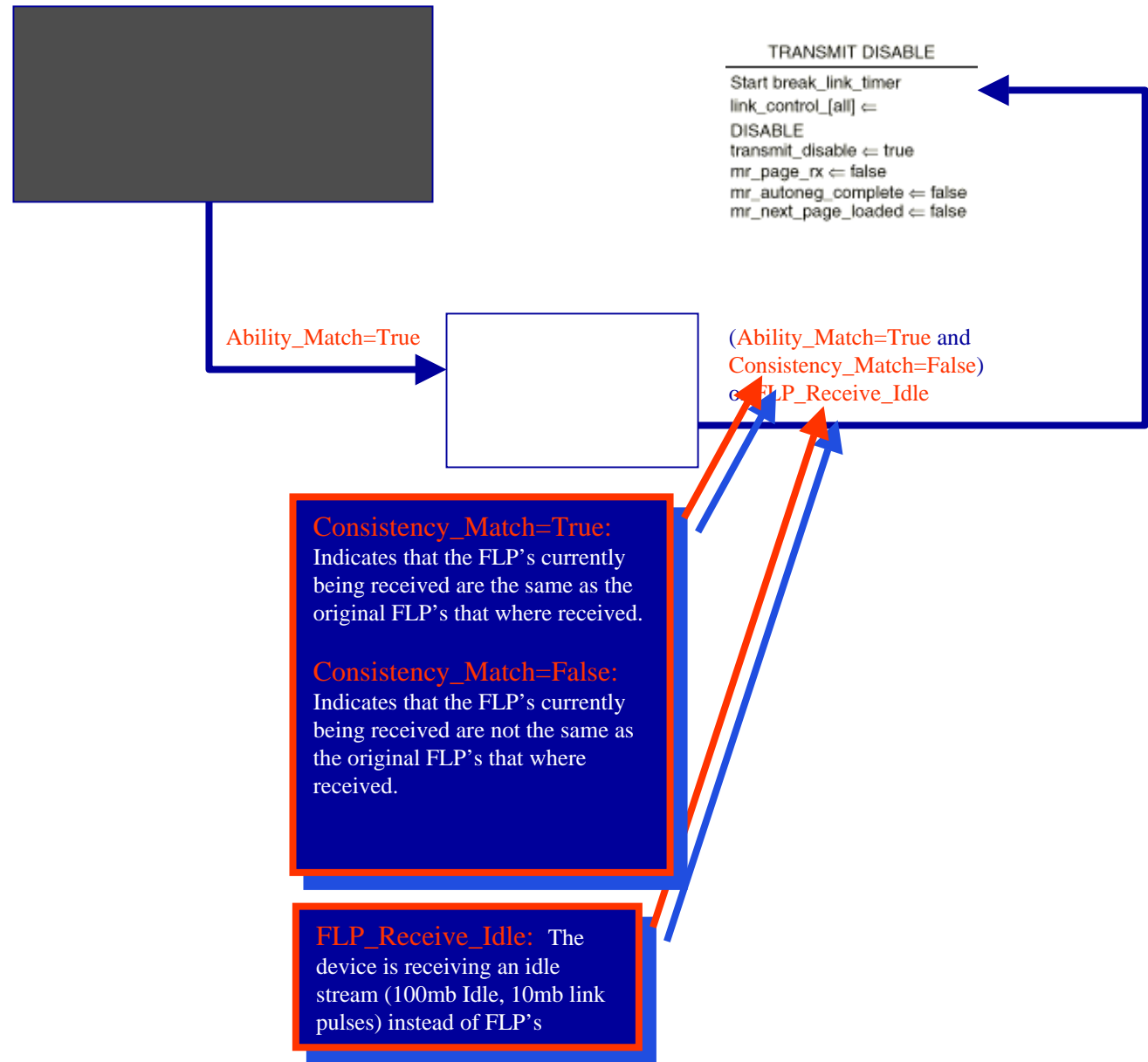
transmit_ability \leftarrow true
transmit_ack \leftarrow true
mr_lp_autoneg_able \leftarrow true
link_control_[all] \leftarrow DISABLE

Acknowledge Detect: The device has received 3 identical FLP's

Exit Paths:

Transmit_Disable (some sort of ANEG error)

Complete Acknowledge (ANEG "Handshake")



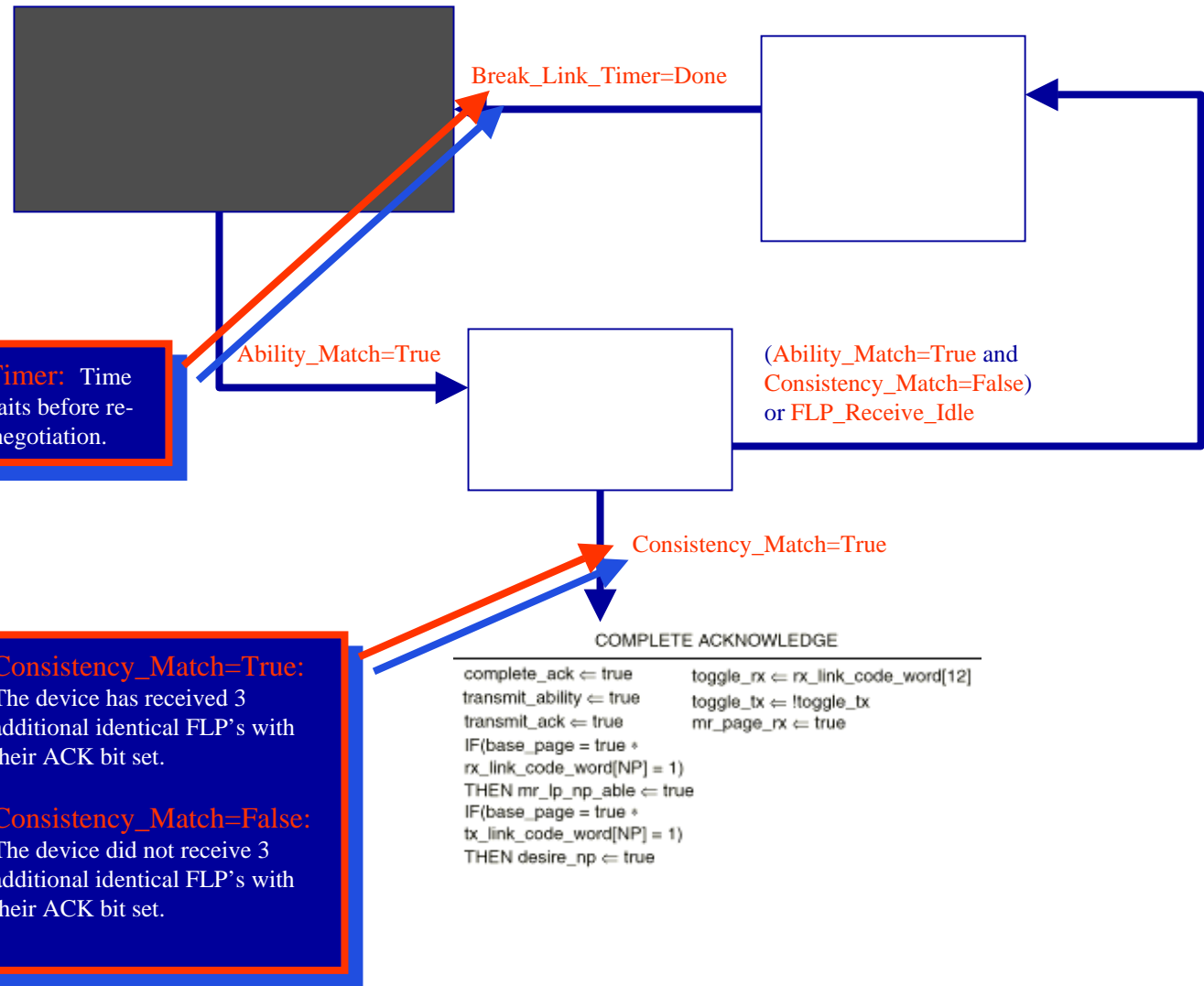
Transmit Disable: Something “wrong” has happened - break link and try again.

Exit Paths:
Ability Detect (restart auto-negotiation)

Break_Link_Timer: Time that the device waits before re-attempting auto-negotiation.

Consistency_Match=True:
The device has received 3 additional identical FLP's with their ACK bit set.

Consistency_Match=False:
The device did not receive 3 additional identical FLP's with their ACK bit set.

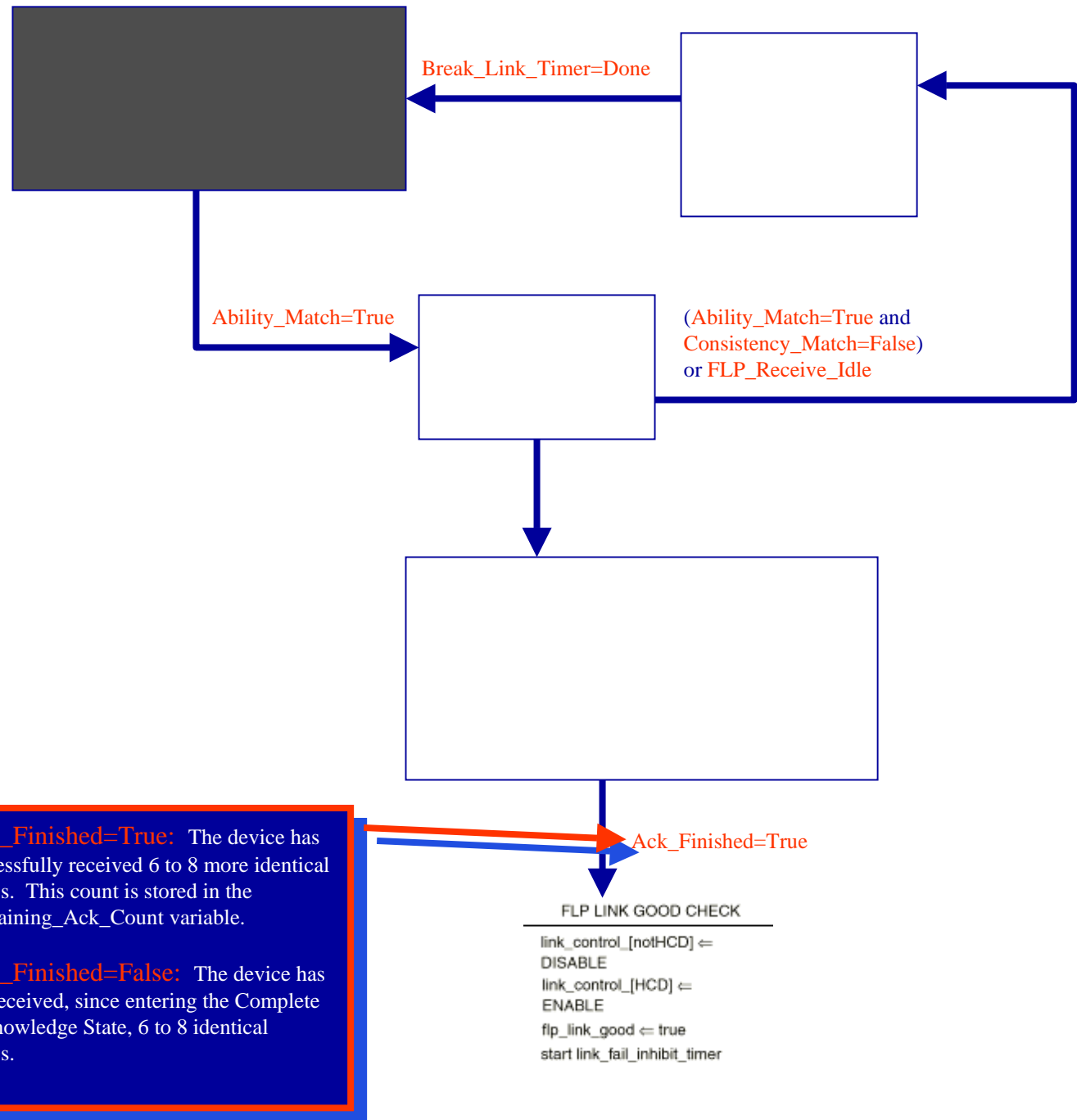


Complete Acknowledge: Point where either Next Page information is exchanged, or more FLP's are exchanged before continuing

Exit Paths:

FLP Link Good Check - choose HCD link, and then validate it

Next Page Wait - Next Page Stuff

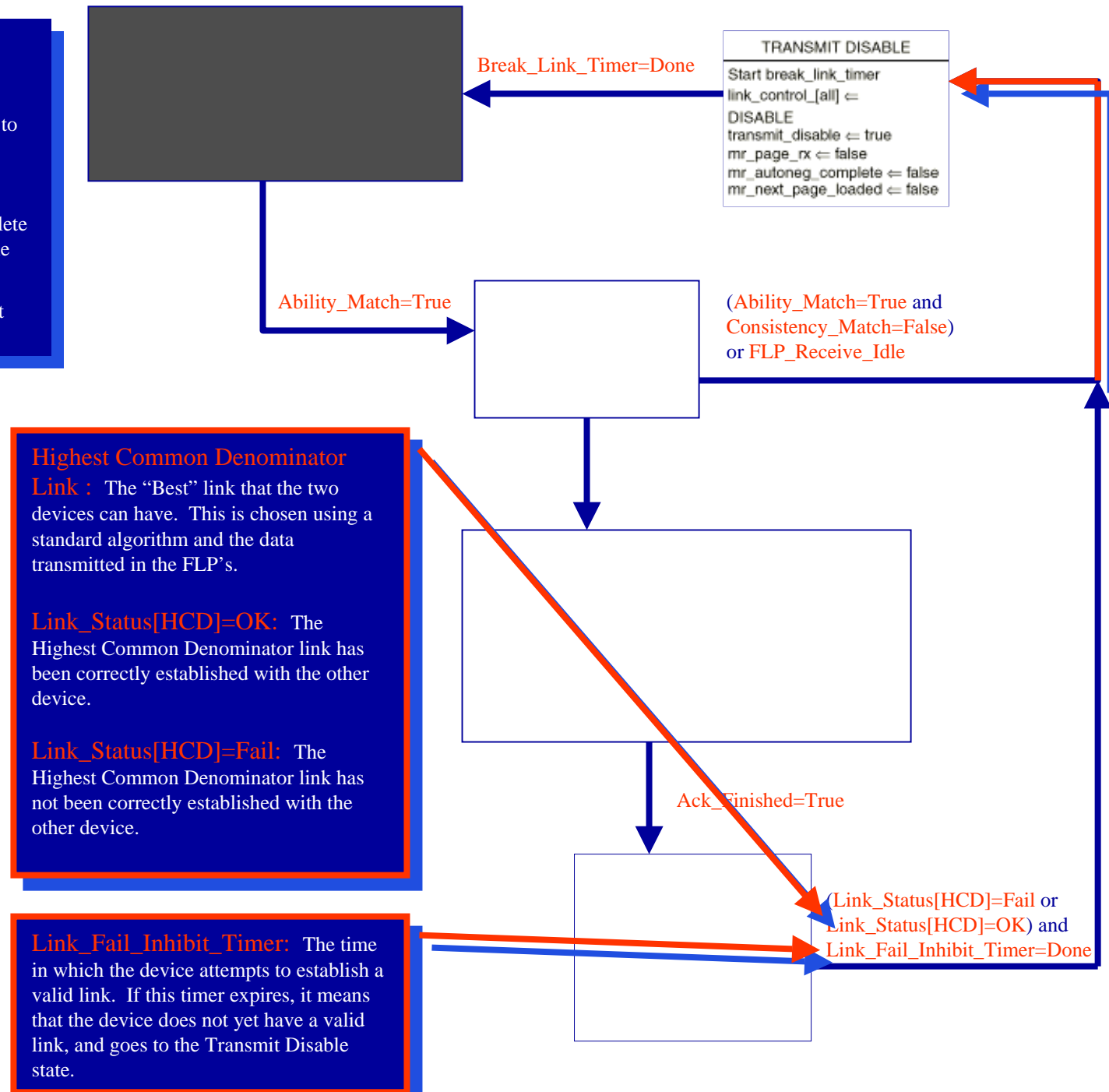


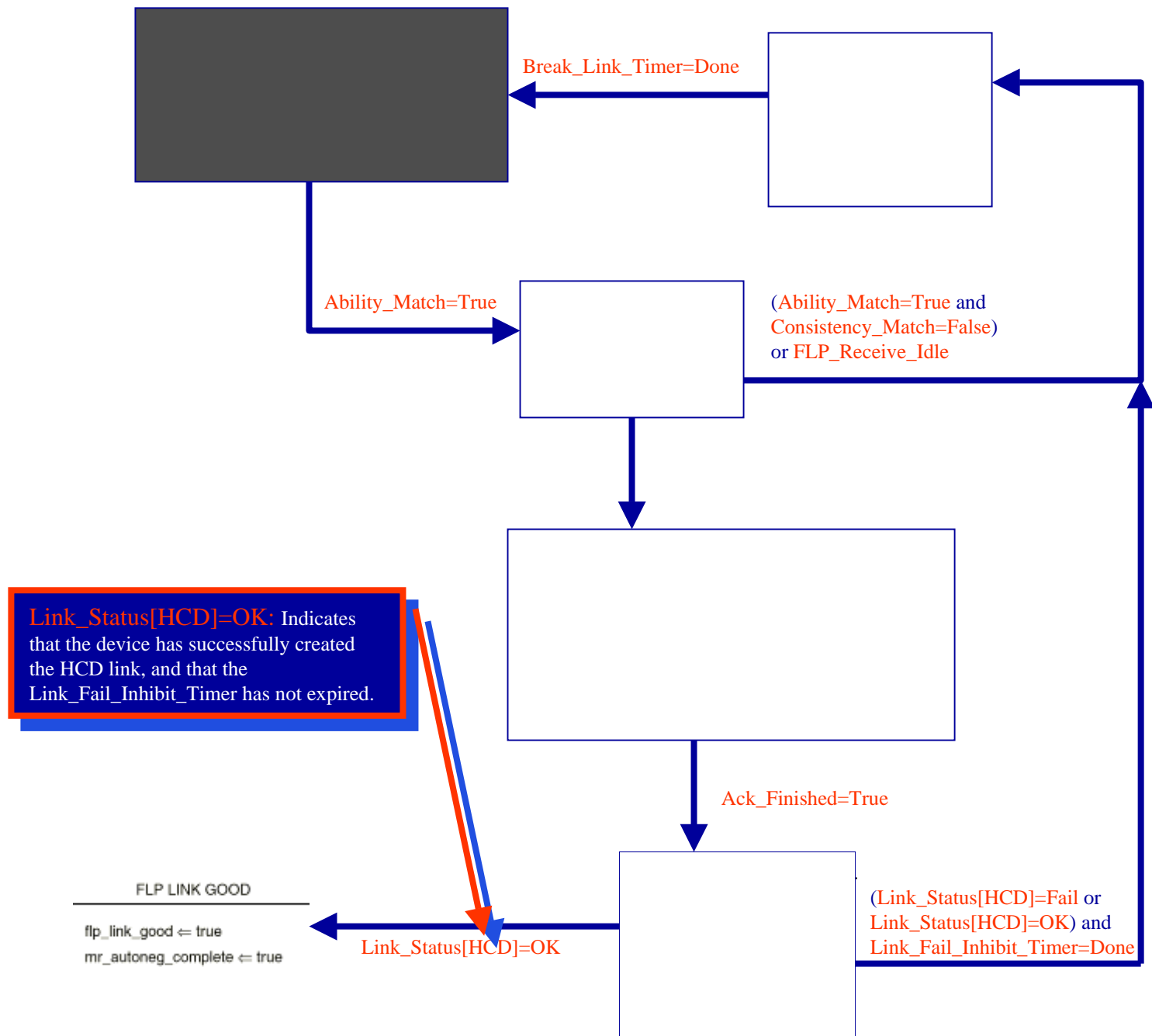
FLP Link Good Check: The device enables the Highest Common Denominator Technology, and waits to ensure that it gets a good link.

Exit Paths:

FLP Link Good - Successfully complete the Auto-Negotiation process with the HCD link.

Transmit Disable - HCD link was not successful - break link and try again.

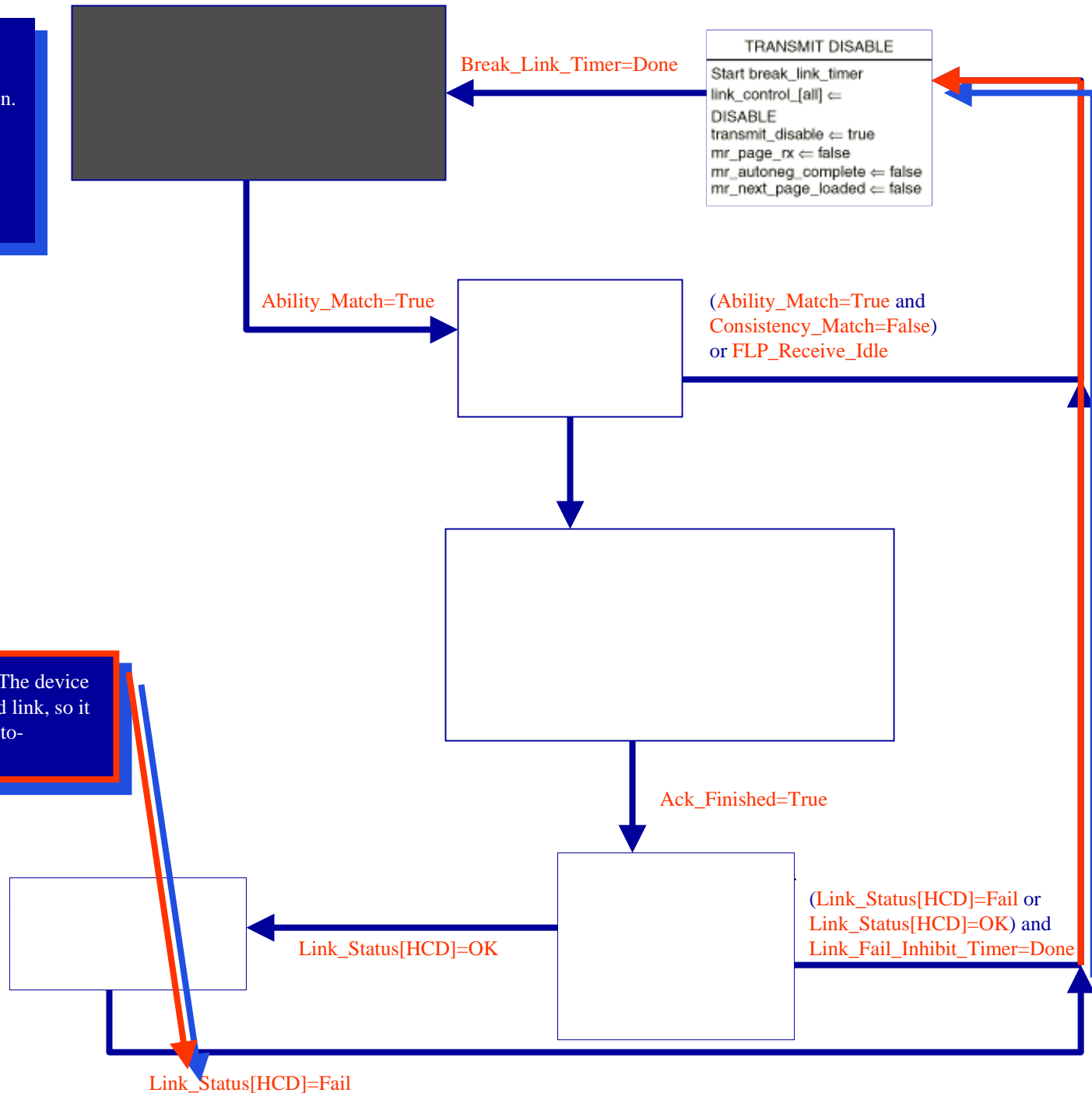




FLP Link Good: The device has successfully completed auto-negotiation.

Exit Paths:

Transmit Disable - HCD link has been broken, restart the auto-negotiation process.

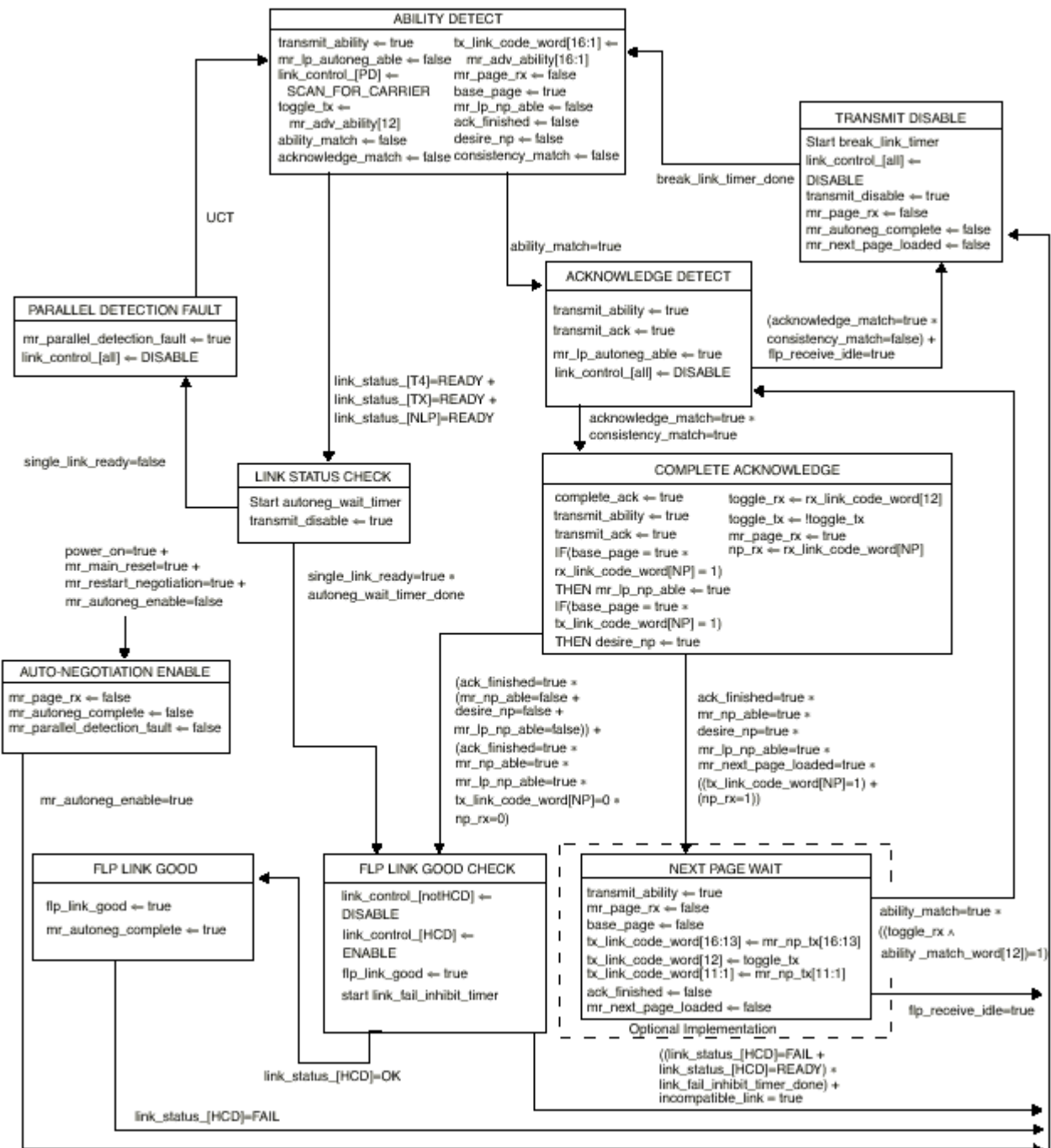


Parallel Detection

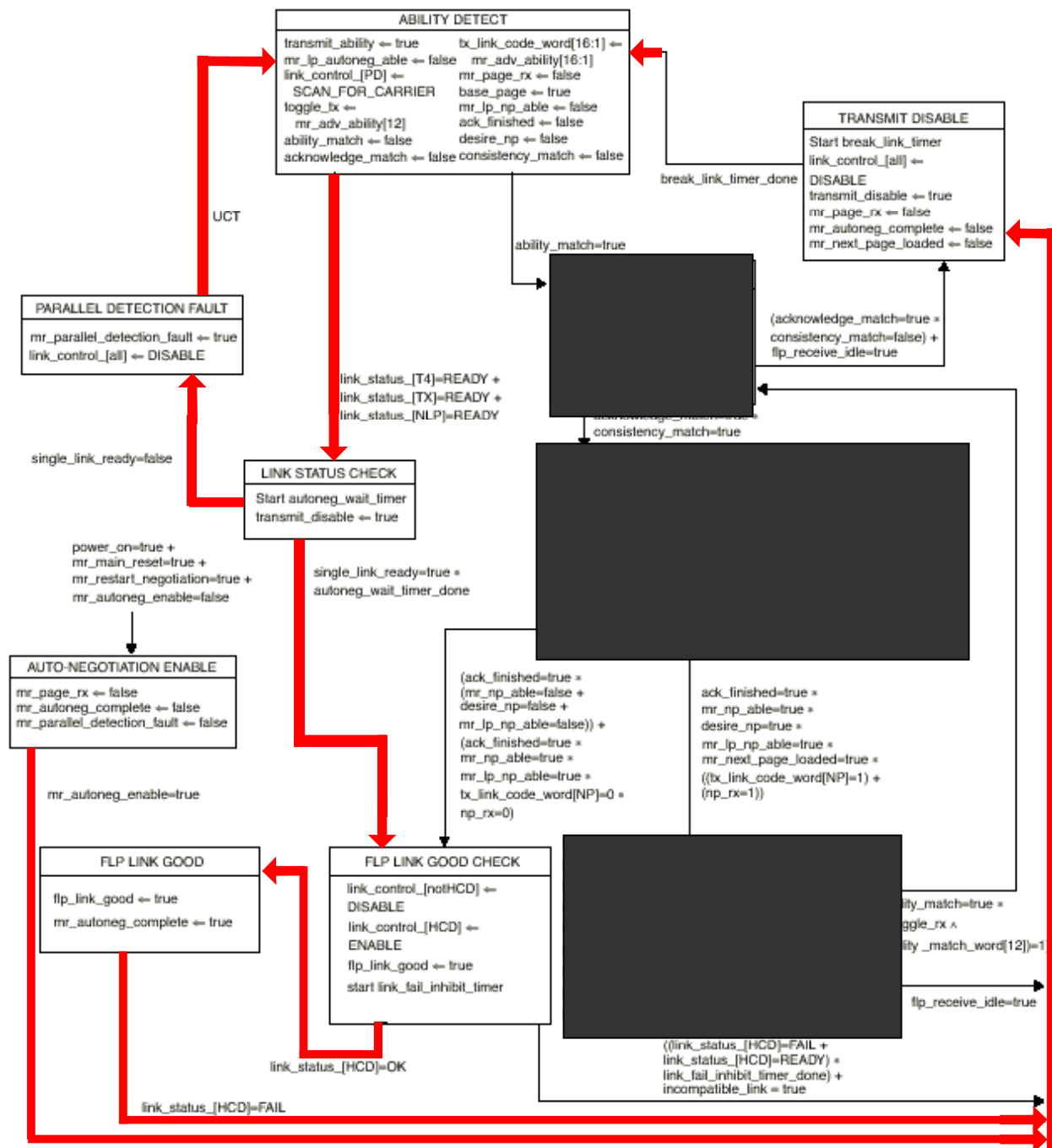
- What if Dave understood the first note he got back?

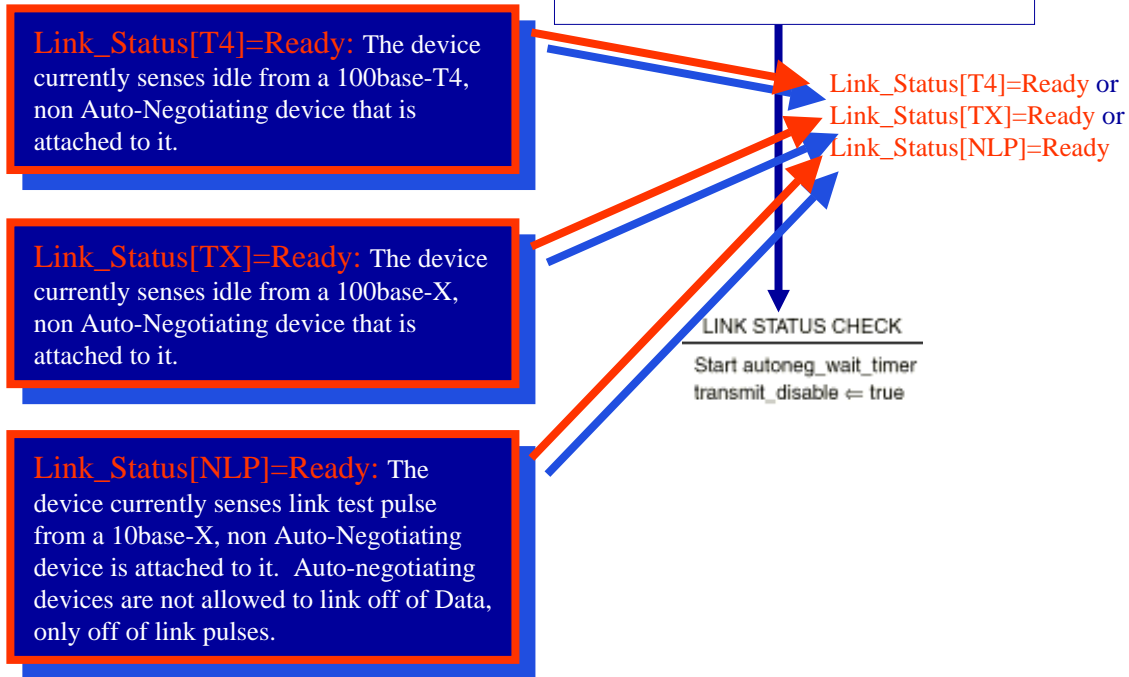


The full ANEG arbitration state Diagram



ANEG - the Parallel Detection way



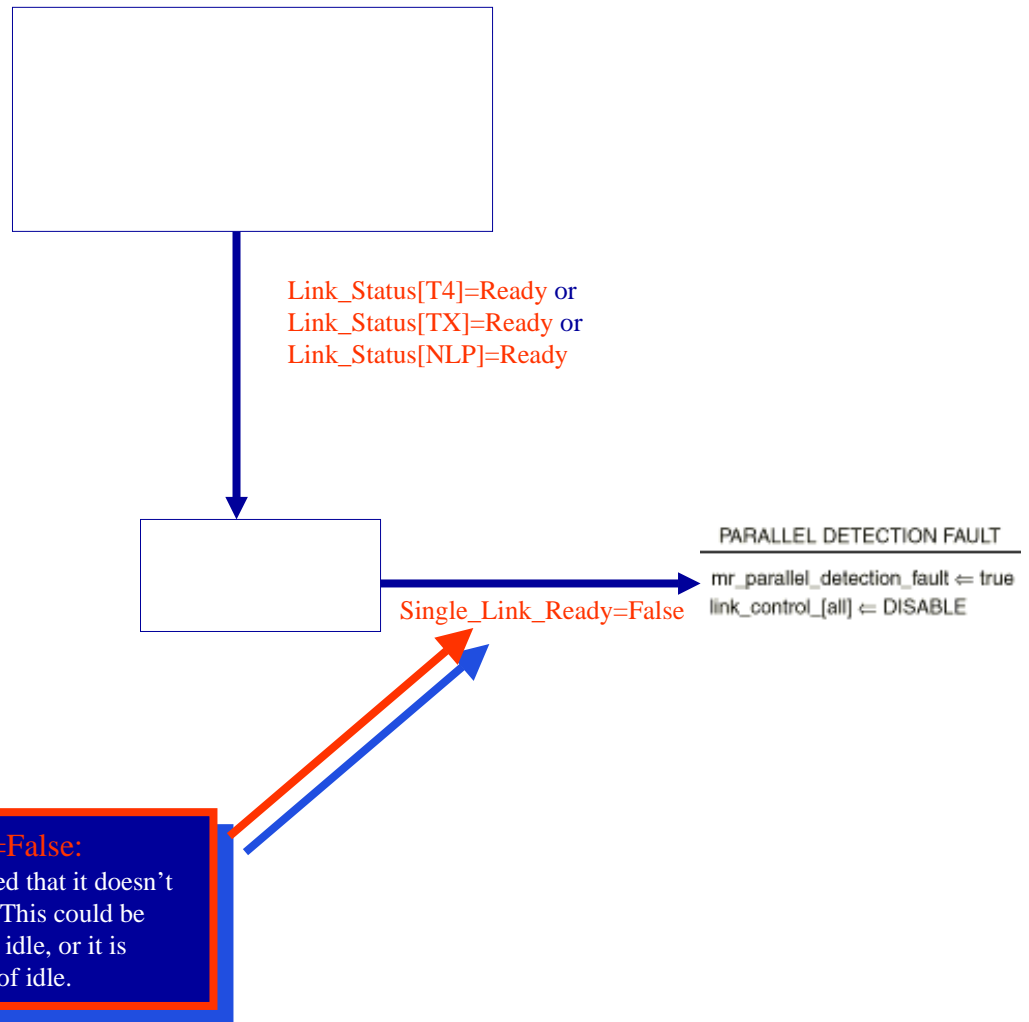


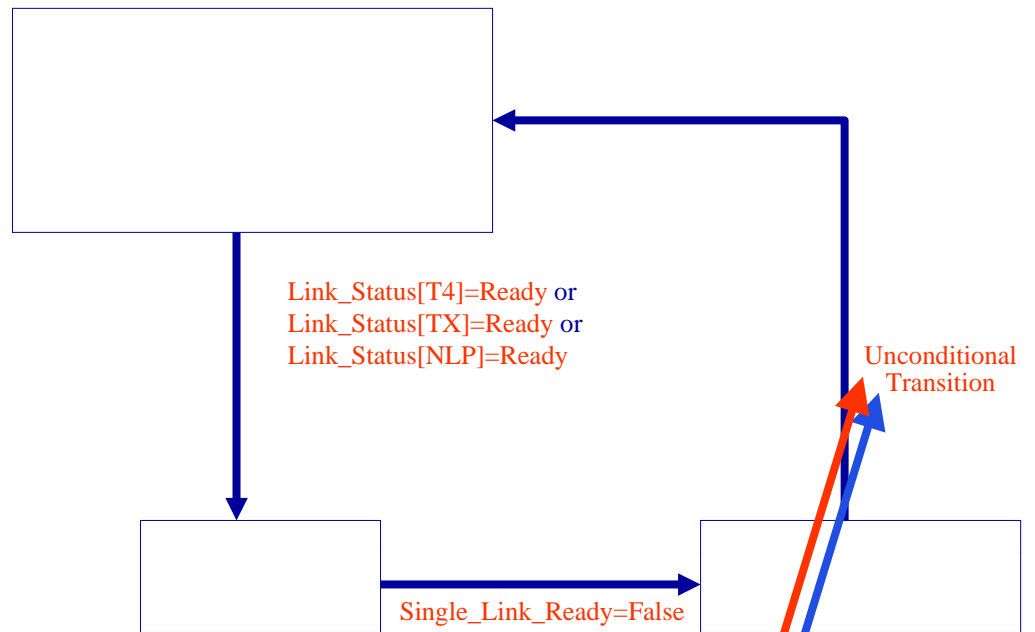
Link Status Check: The device has detected idle from a non-autoneg device attached to it, and is checking to make sure that it can establish a valid link.

Exit Paths:

Parallel Detection Fault - The device has encountered a problem in the parallel detection process, and is going to restart auto-negotiation.

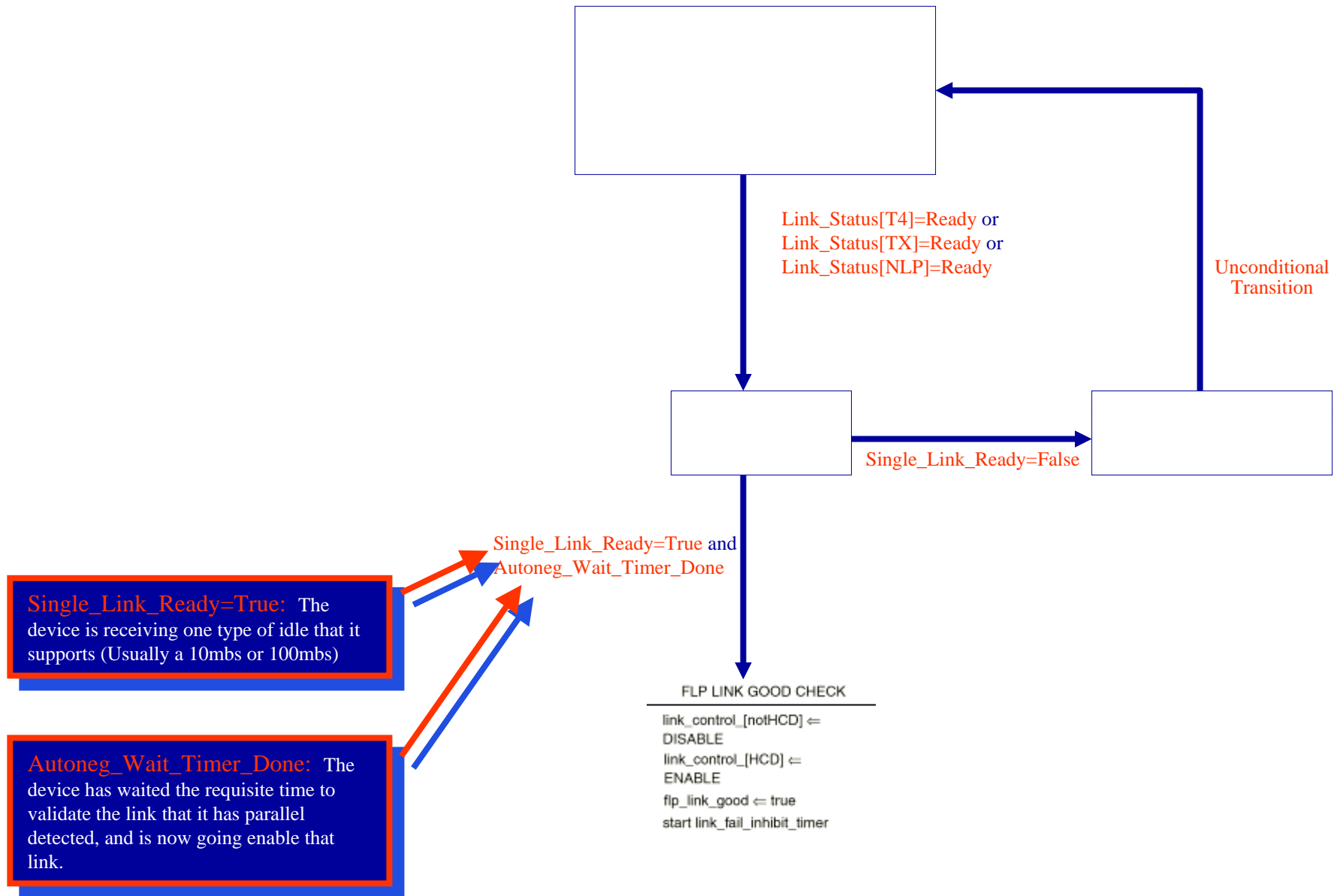
FLP Link Good Check - Enable link control for the Highest Common Denominator link, and then make sure that the link is valid.





Unconditional Transition:

Once the device has determined that `Single_Link_Ready=False`, it immediately disables all link(s) it may have, and then unconditionally goes back to the Ability Detect State.



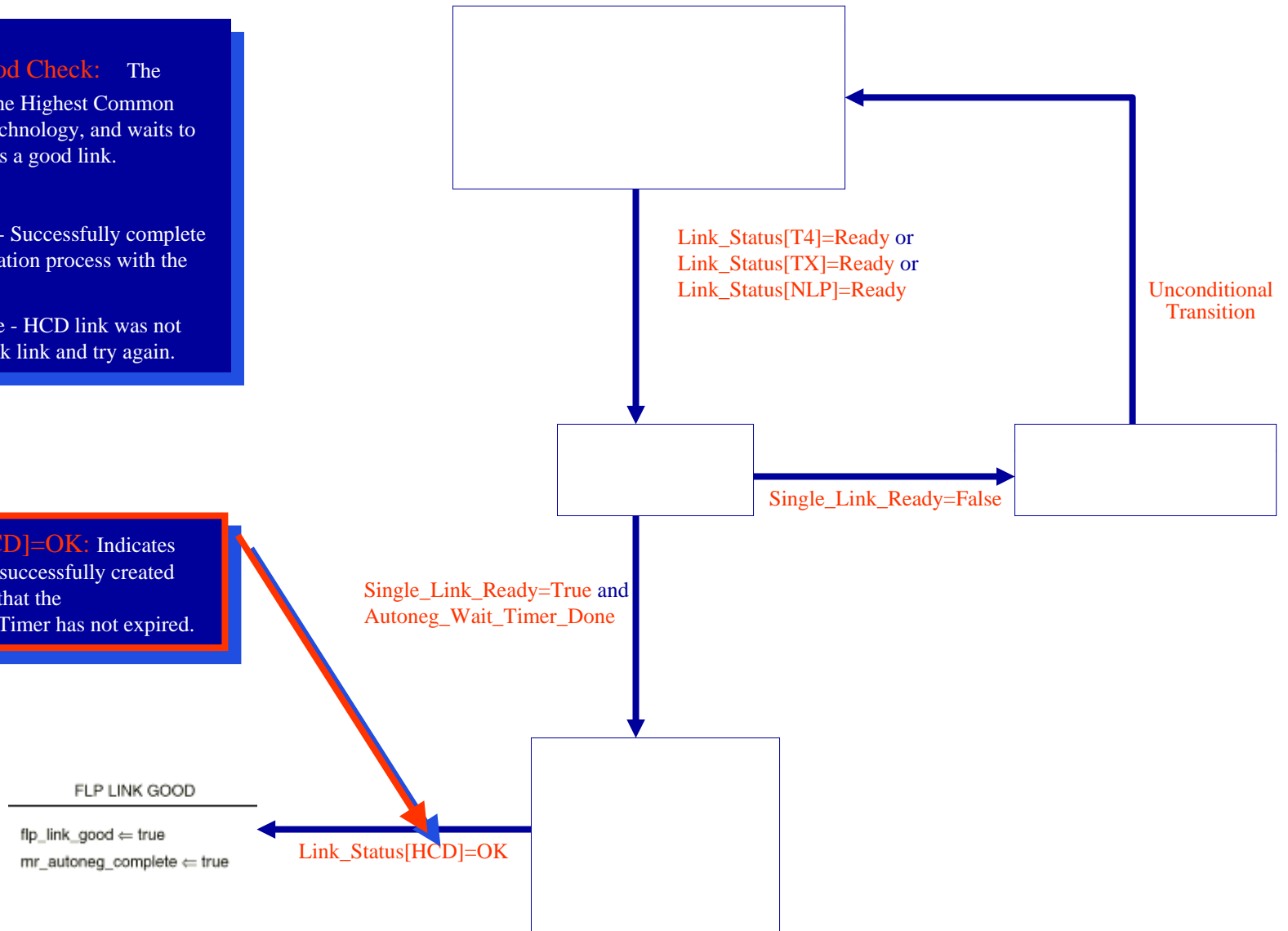
FLP Link Good Check: The device enables the Highest Common Denominator Technology, and waits to ensure that it gets a good link.

Exit Paths:

FLP Link Good - Successfully complete the Auto-Negotiation process with the HCD link.

Transmit Disable - HCD link was not successful - break link and try again.

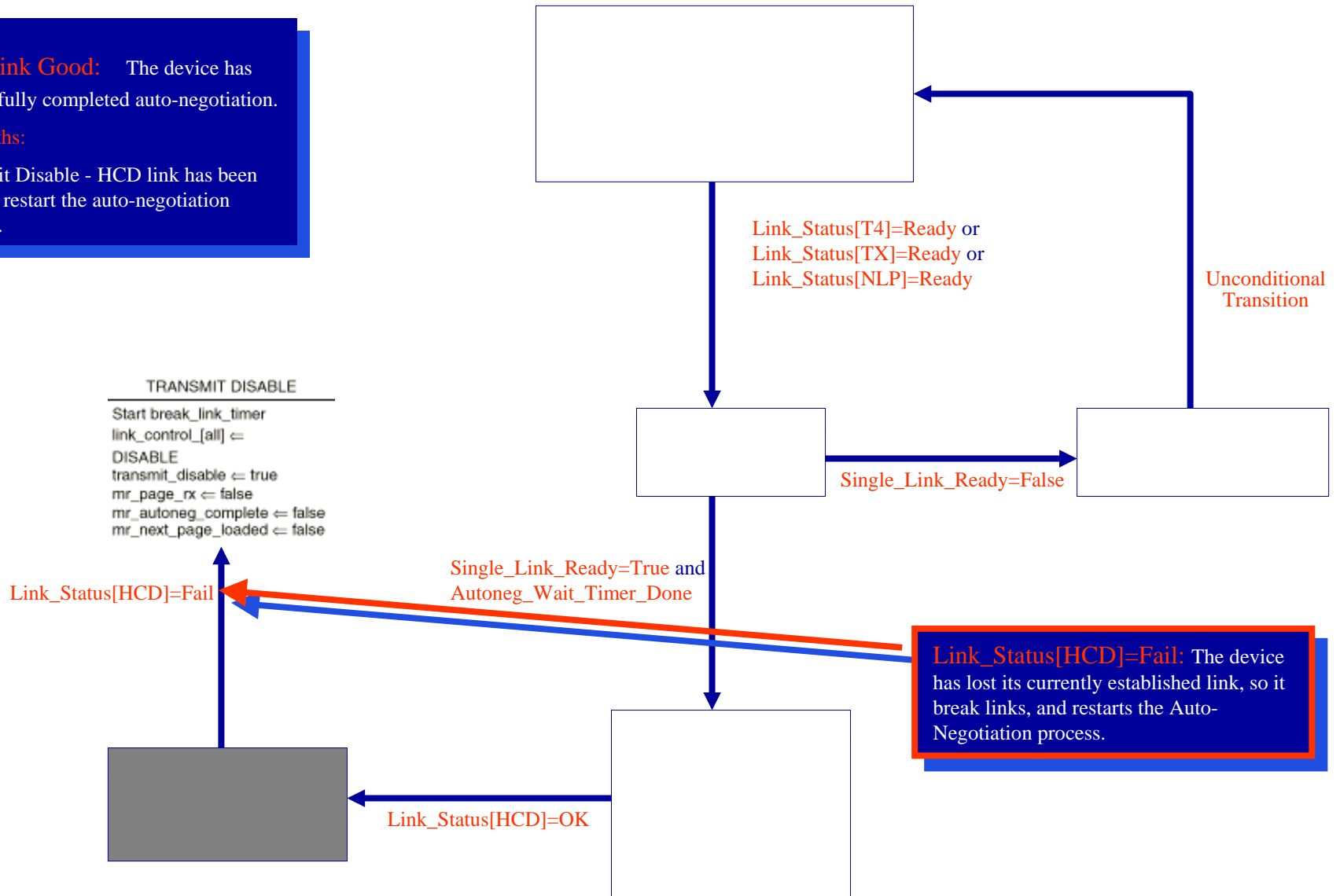
Link_Status[HCD]=OK: Indicates that the device has successfully created the HCD link, and that the Link_Fail_Inhibit_Timer has not expired.



FLP Link Good: The device has successfully completed auto-negotiation.

Exit Paths:

Transmit Disable - HCD link has been broken, restart the auto-negotiation process.

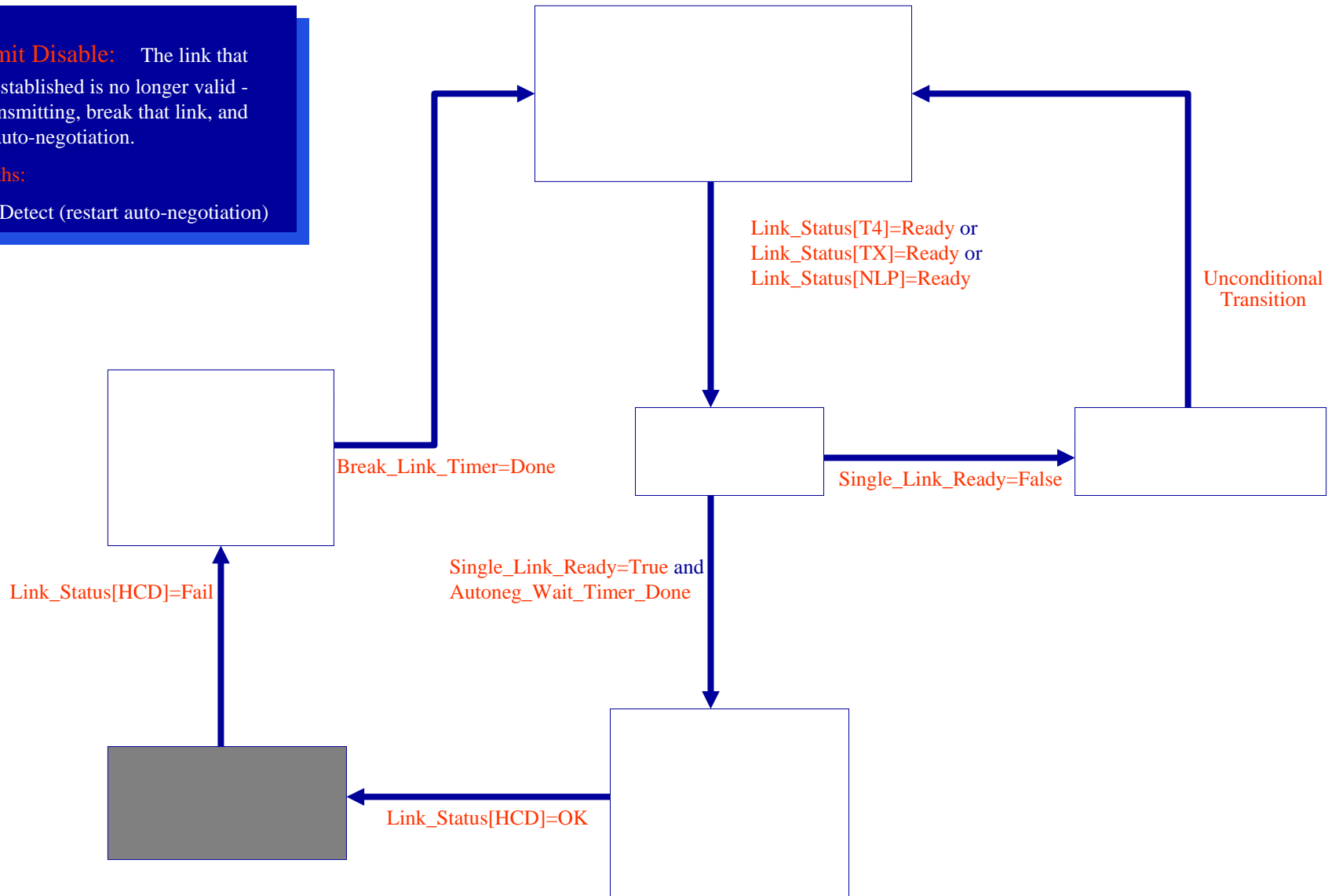


Link_Status[HCD]=Fail: The device has lost its currently established link, so it break links, and restarts the Auto-Negotiation process.

Transmit Disable: The link that we've established is no longer valid - stop transmitting, break that link, and restart auto-negotiation.

Exit Paths:

Ability Detect (restart auto-negotiation)

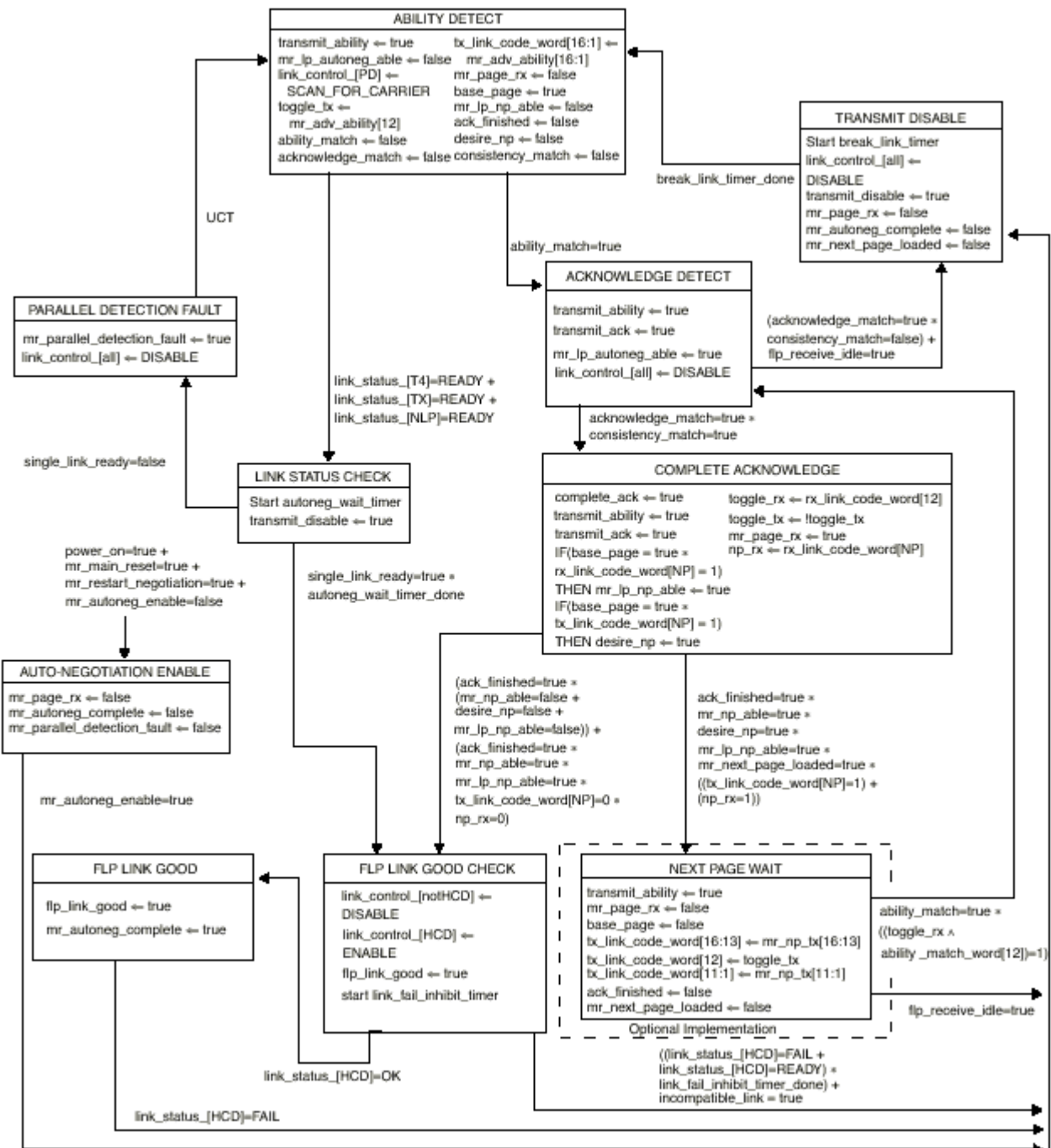


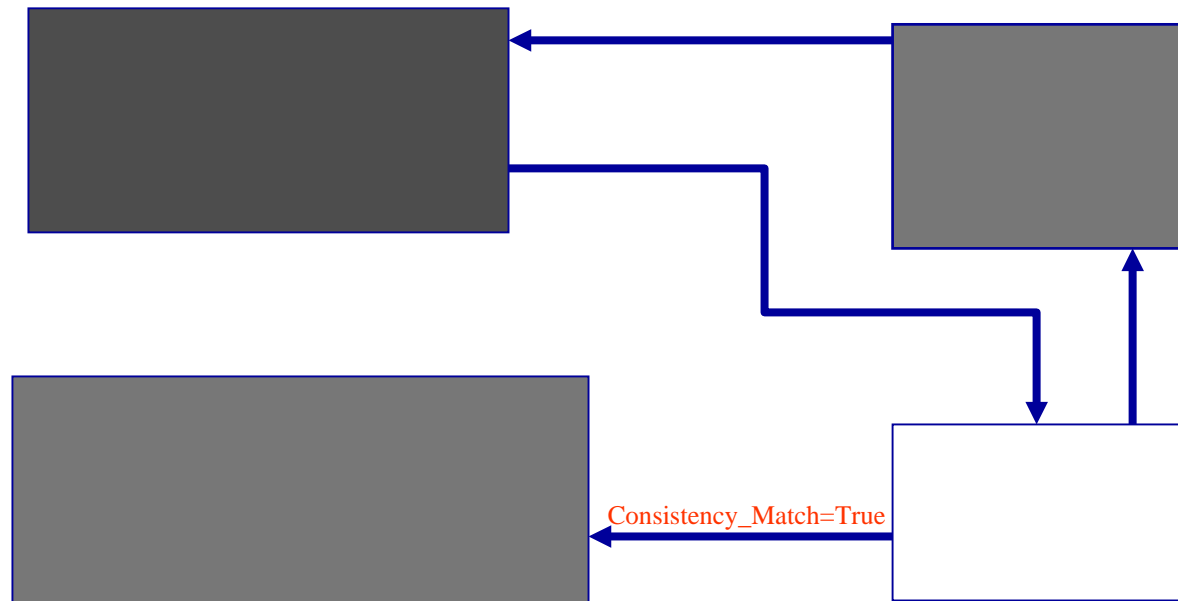
More than just language

- What if a piece of bark wasn't enough space?
(Dave is quite the linguist)



The full ANEG arbitration state Diagram





TX_Link_Code[NP]=1 : This device has its next page data bit set

RX_Link_Code[NP]=1 : The device that this device is attached to has its next page data bit set

Desire_NP=True and
Mr_np_able=True and
Mr_lp_np_able=True and
(TX_link_code[NP]=1 or RX_link_code[NP]=1)

Desire_NP=True : This device and the device it is attached to desire Next_page exchange

MR_NP_Able : This device is Next-Page capable

MR_LP_NP_Able : The device that is currently attached to this one is Next-Page capable

NEXT PAGE WAIT

```

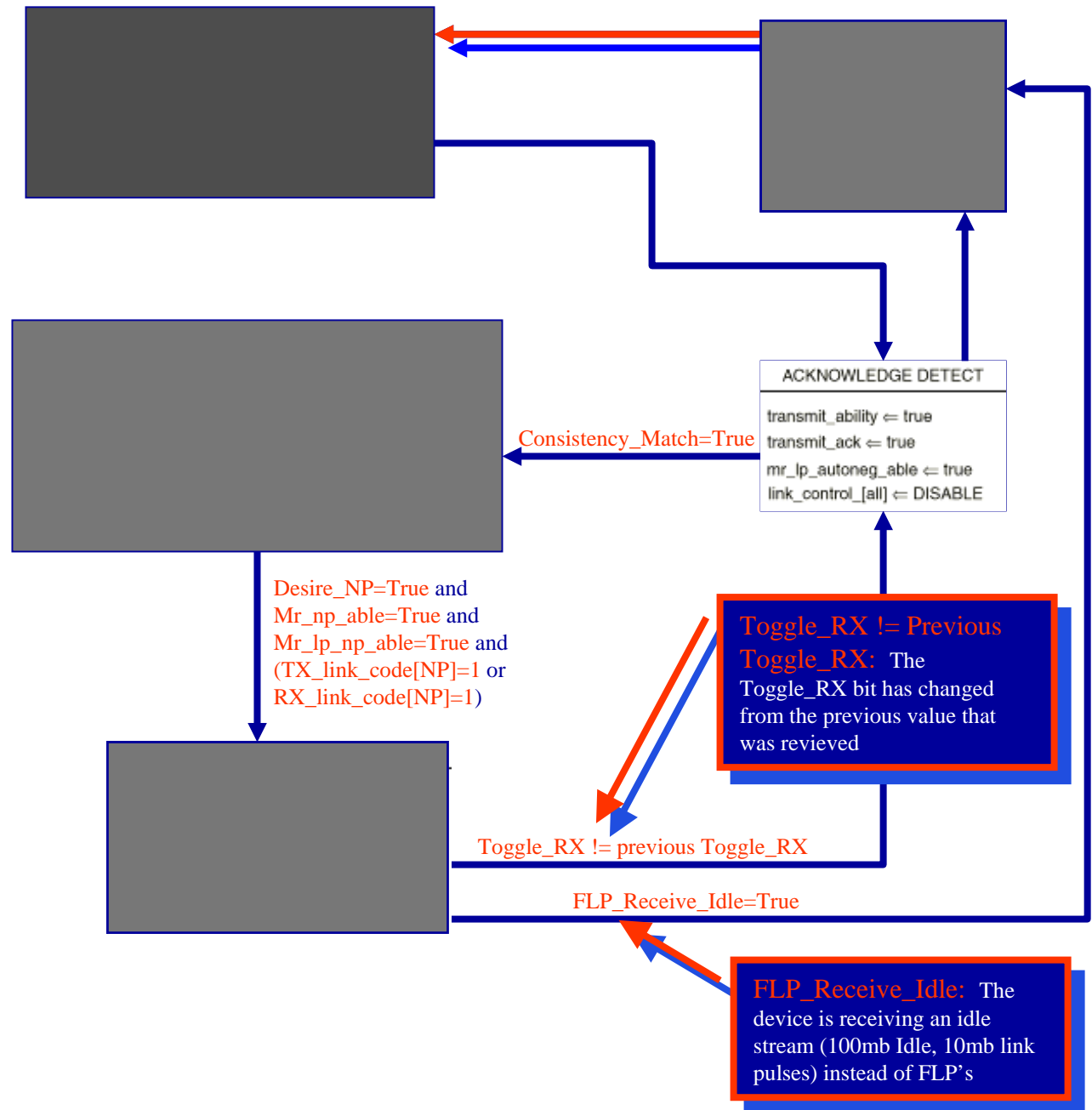
transmit_ability <= true
mr_page_rx <= false
base_page <= false
tx_link_code_word[16:13] <= mr_np_tx[16:13]
tx_link_code_word[12] <= toggle_tx
tx_link_code_word[11:1] <= mr_np_lx[11:1]
ack_finished <= false
mr_next_page_loaded <= false
  
```

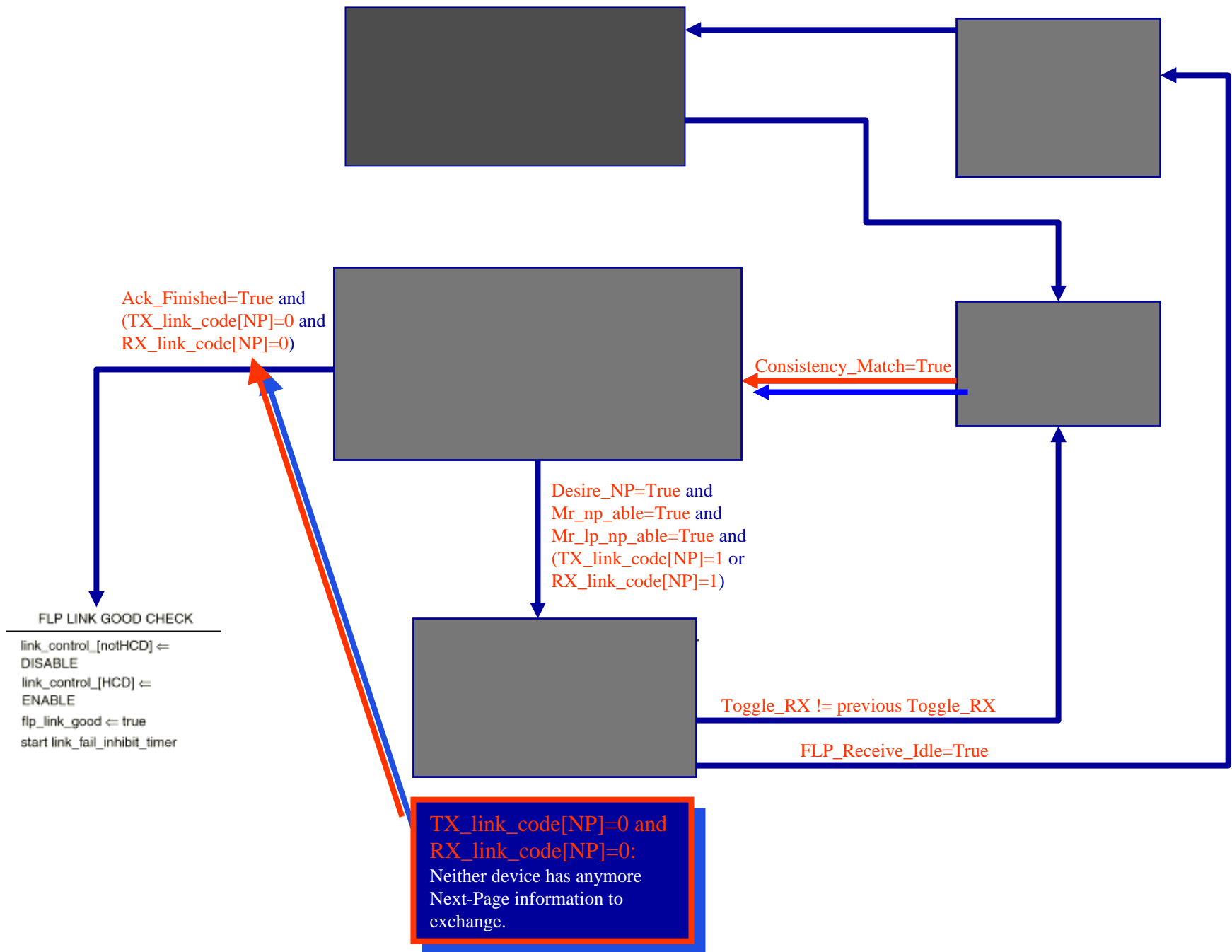
Next Page Wait: Both devices are aware that they are Next-Page able, and this device will now start sending its next page, toggling the toggle bit. The device will wait in this state until the Toggle_RX value changes.

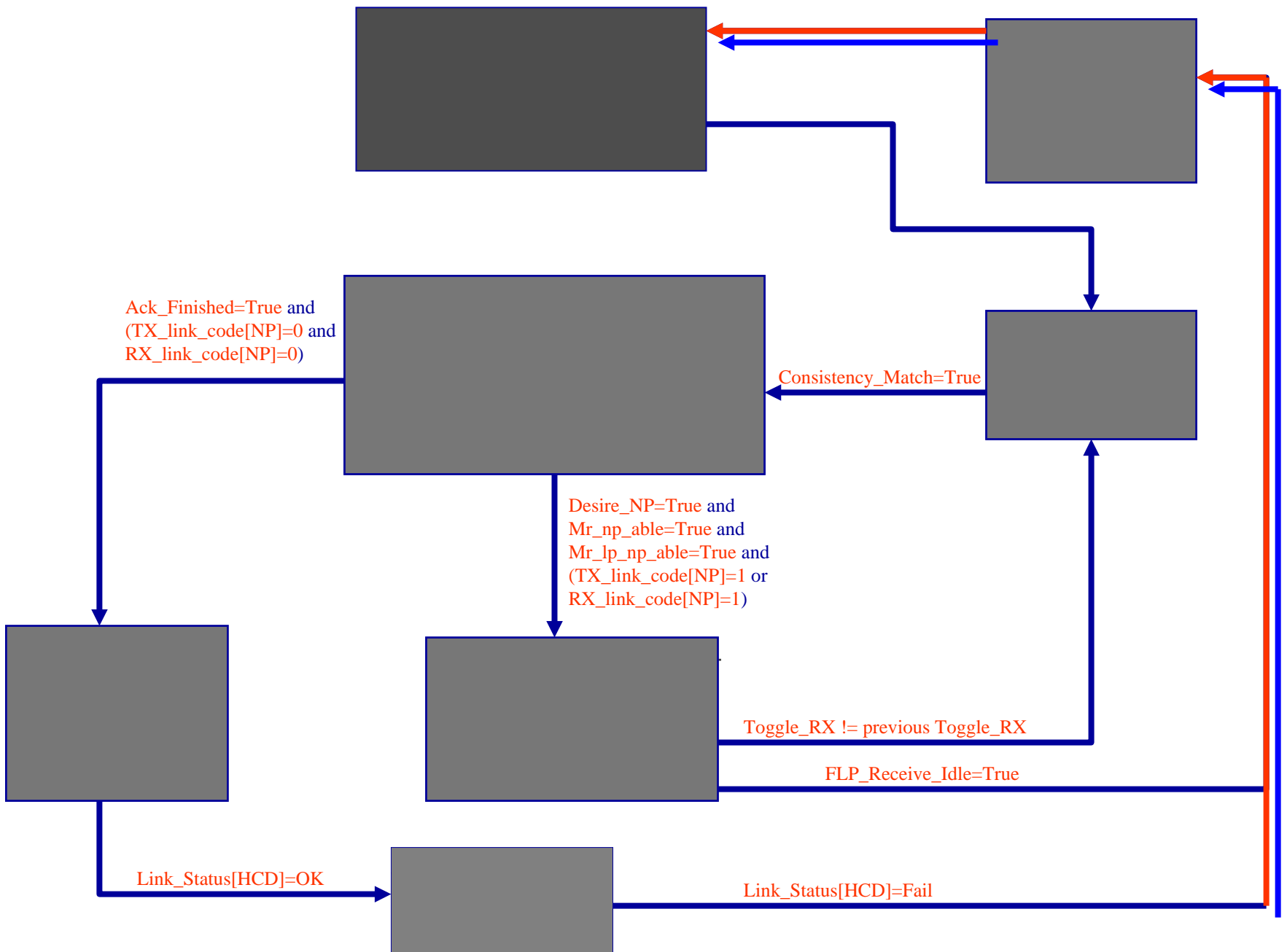
Exit Paths:

Transmit Disable: An error occurred during Next-Page exchange, break link and restart Auto-Negotiation

Acknowledge Detect: We received a Next-Page FLP with the toggle bit correctly set.







What can go wrong?

- Some common Auto-Negotiation Problems.
 - Incompatible Speeds
 - Manual Configurations
 - Bad management



In Summary

- Auto-Negotiation allows devices to link together without the user at the best possible operational mode by exchanging management information
- Uses 10Base-T compatible signaling (for legacy support)
- Uses a series of Pages, which allows for an unlimited amount of information to be exchanged.
- Almost never gets the wrong link.



Additional resources

- Internetworking with TCP/IP - 4th Ed., Douglas E. Comer
- IEEE 802.3 Clauses 28 & 40
- Anyone who tests Auto-Negotiation at the Lab



References

- IEEE 802.3 Clauses 28 & 40
- For correct citation format for virtually any source see:
<http://www.reference.unh.edu/bib.html>
- A special thank you to:
 - Bob Noseworthy
 - Jake Odell

