98. Auto-Negotiation for Single-lane Media

98.1 Overview

98.1.1 Scope

Clause 98 describes the Auto-Negotiation function used on single-lane media to allow a device to advertise enhanced modes of operation it possesses to a device at the remote end of a link segment and to detect corresponding enhanced operational modes that the other device may be advertising. Annex 98A describes the Selector Field that is used by Auto-Negotiation to identify the type of message being sent.

The objective of the Auto-Negotiation function is to provide the means to exchange information between two devices that share a link segment and to automatically configure both devices to take maximum advantage of their abilities. It has the additional objective of providing a common synchronization time between two devices prior to link training.

Single-lane Auto-Negotiation is performed using differential Manchester encoding (DME) pages. DME provides a DC balanced signal. DME does not add packet or upper layer overhead to the network devices. DME is transferred in a half-duplex manner over the media.

Auto-Negotiation for single-lane media does not test the link segment characteristics.

Single-lane Auto-Negotiation may be used with either differential balanced-pair media or with single-ended unbalanced media.

The function allows the devices at both ends of a link segment to advertise abilities, acknowledge receipt and understanding of the common mode(s) of operation that both devices share, and to reject the use of operational modes that are not shared by both devices. Where more than one common mode exists between the two devices, a mechanism is provided to allow the devices to resolve to a single mode of operation using a predetermined priority resolution function. The Auto-Negotiation function allows the identification of the operational mode of the link partner. Should multiple modes be present, management may select between the various offered modes. How such selection is done is beyond the scope of this standard.

98.1.2 Relationship to the ISO/IEC Open Systems Interconnection (OSI) reference model

The Auto-Negotiation function is provided at the Physical Layer of the ISO/IEC OSI reference model as shown in Figure 98–2. A device that supports multiple modes of operation may advertise its capabilities using the Auto-Negotiation function. The actual transfer of information is observed only at the MDI.

98.2 Functional specifications

The Auto-Negotiation function provides a mechanism to control connection of a single MDI to a single PHY type, where more than one PHY type may exist. A management interface provides control and status of Auto-Negotiation, but the presence of a management agent is not required.

Auto-Negotiation shall provide the following functions (as shown in Figure 98–1):

a) Transmit
b) Receive
c) Half duplex
d) Arbitration

These functions shall comply with the state diagrams from Figure 98–7 through Figure 98–10. The Auto-Negotiation functions shall interact with the technology-dependent PHYs through the Technology Dependent Interface (see 98.4).

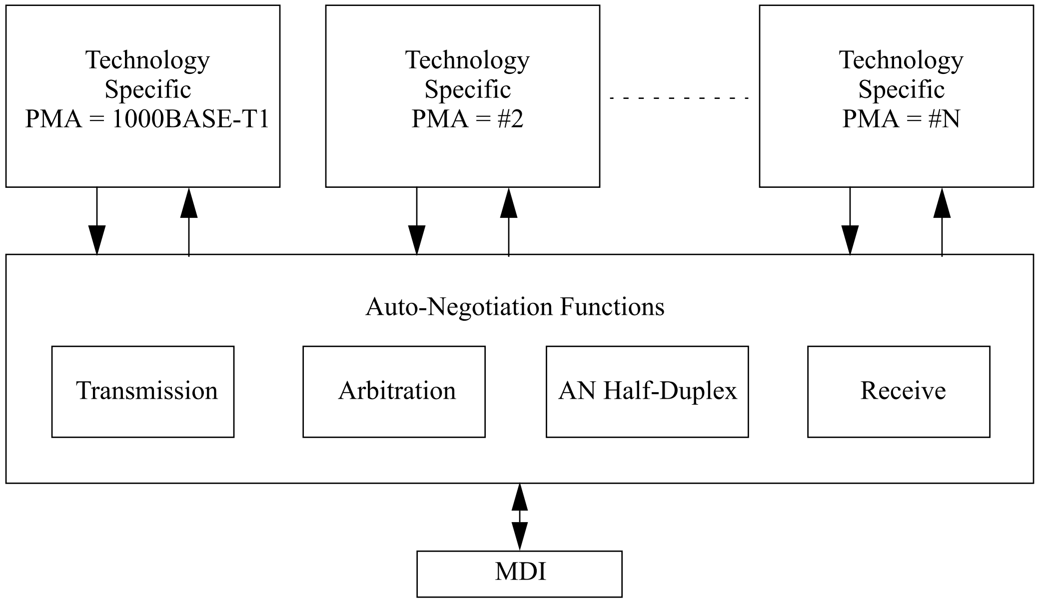


Figure 98–1—High-level model

98.2.1 Transmit function requirements

The Transmit function provides the ability to transmit pages. The first pages exchanged by the local device and its link partner after Power-On, link restart, or renegotiation contain the base link codeword defined in Table 98–2. The local device may modify the link codeword to disable an ability it possesses, but will not transmit an ability it does not possess. This makes possible the distinction between local abilities and advertised abilities so that multi-ability devices may Auto-Negotiate to a mode lower in priority than the highest common ability.

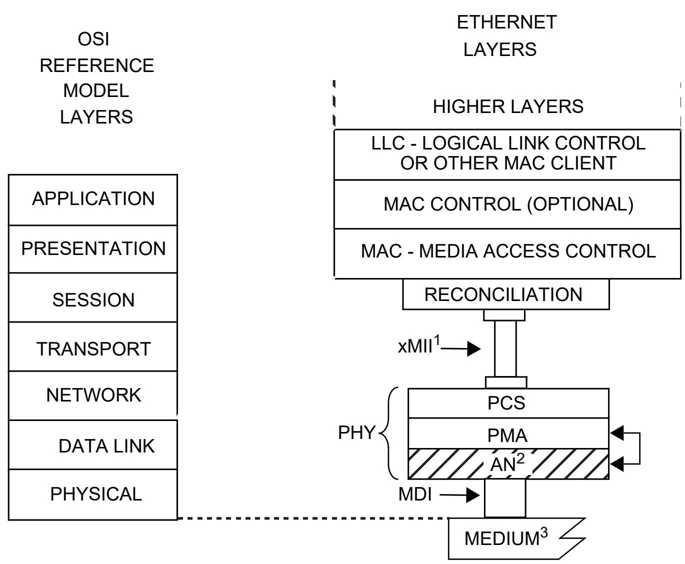
Two different Auto-Negotiation speeds are defined in this subclause. A PHY shall support at least one of these Auto-Negotiation speeds. The two speeds are referred to as *high-speed mode*, or HSM, and *low-speed mode*, or LSM. If Auto-Negotiation is implemented, 1000BASE-T1, 100BASE-T1, and 10BASE-T1S PHYs shall support HSM and may optionally support LSM. For link segments with high insertion loss and those requiring 10BASE-T1L, LSM is provided to enable the full reach capability. If Auto-Negotiation is implemented, 10BASE-T1L PHYs shall support LSM and may optionally support HSM. When performing Auto-Negotiation in high-speed mode, DME pages are transmitted at a nominal rate of 16.667 Mb/s. In low-speed mode, DME pages are transmitted at a nominal rate of 625 kb/s. Subclause 98.5.6 describes the behavior to automatically choose between the different Auto-Negotiation speeds when a PHY supports both.

98.2.1.1 DME transmission

Auto-Negotiation’s method of communication builds upon the encoding mechanism known as differential Manchester encoding (DME). The DME page encodes the data that is used to control the Auto-Negotiation function. DME pages shall not be transmitted when Auto-Negotiation is complete and the highest common denominator PHY has been enabled.

98.2.1.1.1 DME page encoding

A DME page carries a 48-bit Auto-Negotiation page. It consists of 157 evenly spaced transition positions starting from the initial transition from silent to active in the preamble. The page contains a Start Delimiter, the 48-bit page, 16-bit CRC, and an end delimiter (see Figure 98–6). The odd-numbered transition positions represent clock information. The even numbered transition positions represent data information. DME pages are alternately transmitted between the two devices with quiet period separating the DME pages. When the DME page is active, the PHY shall transmit either +1 or –1 level with the voltage levels as specified in 98.2.1.1.4.



MDI = MEDIUM DEPENDENT INTERFACE
xMII = MEDIA INDEPENDENT INTERFACE

PCS = PHYSICAL CODING SUBLAYER
PMA = PHYSICAL MEDIUM ATTACHMENT
PHY = PHYSICAL LAYER DEVICE
AN = AUTO-NEGOTIATION

NOTE 1—xMII is optional.
NOTE 2—Auto-Negotiation is optional. Auto-Negotiation communicates with the PMA sublayer through the PMA service interface messages PMA_LINK.request and PMA_LINK.indication.
NOTE 3—Medium is either a single-lane of differential balanced pair or unbalanced single-ended (e.g., coaxial).

Figure 98–2—Location of Auto-Negotiation function within the ISO/IEC OSI reference model

The first 26 transition positions contain the Start Delimiter, which marks the beginning of the page. The Start Delimiter contains a transition from quiet to active at position 1. For HSM Auto-Negotiation, this transition is followed by transitions at positions 2, 3, 5, 7, 8, 12, 13, 14, 15, 19, 21, 24, 25, 26 and no transitions at the remaining positions. For LSM Auto-Negotiation, this transition is followed by transitions at positions 2, 3, 4, 5, 6, 7, 8, 9, 11, 13, 15, 16, 18, 19, 20, 22, 23, 24, 26 and no transitions at the remaining positions.

The final 2 transition positions contain the ending delimiter, which marks the end of the page. The ending delimiter contains a transition at position 155 and no transitions at the remaining positions. Position 157 contains a transition from active to quiet.

Each of the remaining 64 odd-numbered transition positions between the starting and ending delimiters shall contain a transition. The remaining 64 even-numbered transition positions shall represent data information as follows:

— A transition present in an even-numbered transition position represents a logical one.
— A transition absent from an even-numbered transition position represents a logical zero.

The first 48 of these positions shall carry the data of the Auto-Negotiation page. The final 16 positions carry the 16-bit CRC.

The CRC16 polynomial is $x^{16} + x^{15} + x^2 + 1$. The CRC16 shall produce the same result as the implementation shown in Figure 98–3. The 16 delay elements S0,..., S15, shall be initialized to zero. Afterwards the 48 data bits are used to compute the CRC16 with the switch connected (setting CRCgen). After all the 48 bits have been processed, the switch is disconnected (setting CRCout) and the 16 values stored in the delay elements are transmitted in the order illustrated, first S15, followed by S14, and so on, until the final value S0.

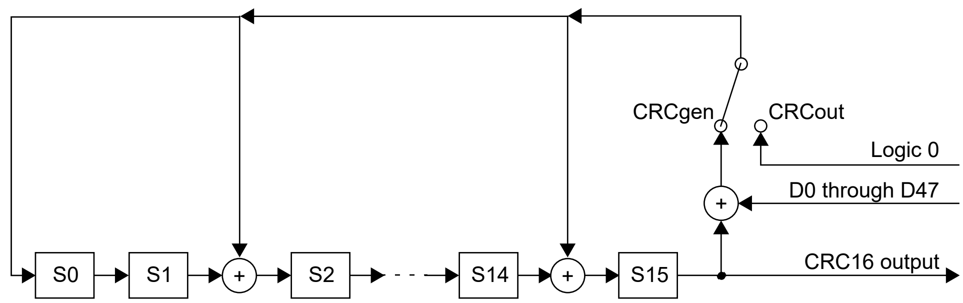


Figure 98–3—CRC16

The polarity at position 0 is randomly determined in an implementation specific manner.

The purpose of randomizing the starting polarity is to remove the spectral peaks that would otherwise occur when sending the same DME page repeatedly. Randomly choosing the starting polarity results in randomly inverting or not inverting the encoded page so that repetitions of the same page no longer produce a periodic signal.

Clock transition positions are differentiated from data transition positions by the spacing between them, as shown in Figure 98–4 and enumerated in Table 98–1.

The encoding of data using DME bits in a DME page is illustrated in Figure 98–4.

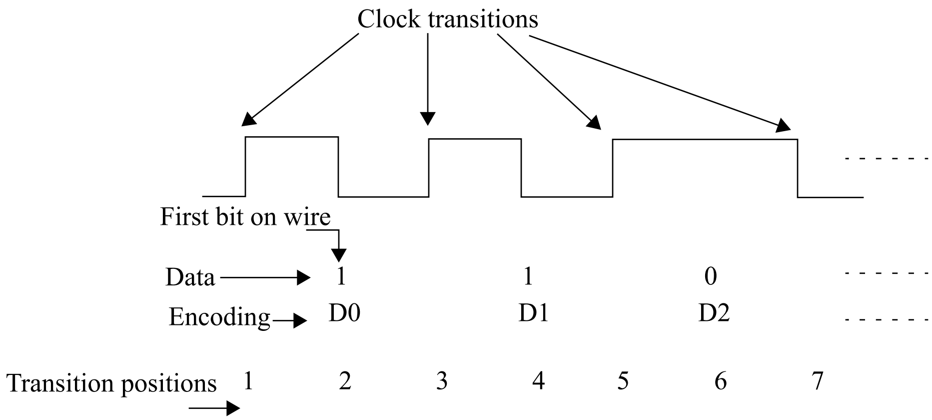


Figure 98–4—Data bit encoding within DME pages

98.2.1.1.2 DME page timing

The timing parameters for DME pages shall be followed as in Table 98–1. The transition positions within a DME page are spaced with a period of T1. T2 is the separation between clock transitions. T3 is the time from a clock transition to a data transition representing a one. When operating in high-speed mode, transitions shall occur within ± 0.8 ns of their ideal positions. When operating in low-speed mode, transitions shall occur within ± 10 ns of their ideal positions.

T5 specifies the duration of a DME page. The minimum number of transitions and maximum number of transitions in a page is represented by T4a. T4b indicates that the start of a DME page begins with a transition from 0 to ±1 and the end of the DME page is a transition from ±1 to 0.

Table 98–1 summarizes the timing parameters. The transition timing parameters are illustrated in Figure 98–5 and Figure 98–6.

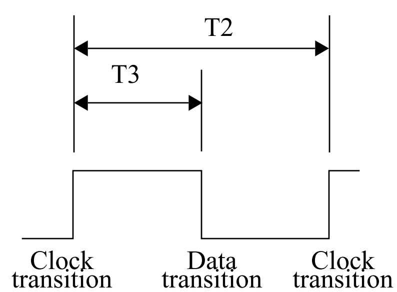


Figure 98–5—DME page transition timing

Table 98–1—DME page timing summary

	Parameters	Mode	Min	Typ	Max	Units
T1	Transmit position spacing (period)	high-speed	29.997	30	30.003	ns
		low-speed	799.96	800	800.04	
T2	Clock transition to clock transition	high-speed	59.8	60	60.2	ns
		low-speed	1590	1600	1610	
T3	Clock transition to data transition (data = 1)	high-speed	29.9	30	30.1	ns
		low-speed	795	800	805	
T4a	+1 to –1 or –1 to +1 transitions in a DME page	high-speed	79	—	143	—
		low-speed	84	—	148	
T4b	0 to ±1 or ±1 to 0 transitions in a DME page	high-speed	2	2	2	—
		low-speed	2	2	2	
T5	DME page width	high-speed	4679	4680	4681	ns
		low-speed	124 793	124 800	124 807	

98.2.1.1.3 DME page Delimiters

The page is preceded by a unique Start Delimiter consisting of a 26 × T1 sequence that includes multiple DME transition violations. For a Start Delimiter starting with a 0 to +1 transition, the bit sequence for high-speed Auto-Negotiation mode is:

+1 –1 +1 +1 –1 –1 +1 –1 –1 –1 –1 +1 –1 +1 –1 –1 –1 –1 +1 +1 –1 –1 –1 +1 –1 +1.

For a Start Delimiter starting with a 0 to +1 transition, the bit sequence for low-speed Auto-Negotiation mode is:

+1 –1 +1 –1 +1 –1 +1 –1 +1 +1 –1 –1 +1 +1 –1 +1 +1 –1 +1 –1 –1 +1 –1 +1 +1 –1.

The DME page ends with an end delimiter that consists of an electrical 0. An example of the delimiters is shown in Figure 98–6.

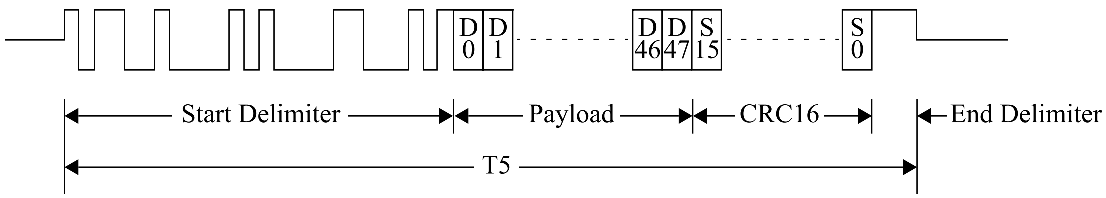


Figure 98–6—DME Page

NOTE—The Start Delimiter may begin with a 0 to +1 or 0 to –1 transition depending upon the DME page starting polarity randomizer.

98.2.1.1.4 Transmitter peak differential output

If used with a differential MDI and medium, when measured with 100 Ω termination, transmit differential signal at MDI shall be within range of 1 V ± 30% peak-to-peak.

If used with a single-ended unbalanced MDI and medium, when measured with 50 Ω termination, transmit signal at MDI shall be within range of 0.5 V ± 30% peak-to-peak.

98.2.1.2 Link codeword encoding

The base link codeword (Base Page) transmitted within a DME page shall convey the encoding shown in Table 98–2. The Auto-Negotiation function supports additional pages using the Next Page function.

Encoding for the link codeword(s) used in the Next Page exchange are defined in 98.2.4.3. In a DME page, D0 shall be the first bit transmitted.

Table 98–2—Link codeword Base Page

D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
S0	S1	S2	S3	S4	E0	E1	E2	E3	E4	C0	C1	M/S	RF	Ack	NP
D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31
T0	T1	T2	T3	T4	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
D32	D33	D34	D35	D36	D37	D38	D39	D40	D41	D42	D43	D44	D45	D46	D47
A11	A12	A13	A14	A15	A16	A17	A18	A19	A20	A21	A22	A23	A24	A25	A26

D[4:0] contains the Selector Field. D[9:5] contains the Echoed Nonce field. D[11:10] contains capability bits to advertise capabilities not related to the PHY. C[1:0] is used to advertise pause capability. D[12] is the force MASTER-SLAVE bit (see 98.2.1.2.5). D[15:13] contains the RF, Ack, and NP bits. The RF, Ack, and NP bits shall function as specified in 98.2.1.2.7, 98.2.1.2.8, and 98.2.1.2.9, respectively. D[20:16] contains the Transmitted Nonce field. D[47:21] contains the Technology Ability Field.

NOTE—Annex K defines optional alternative terminology for “master” and “slave”.

98.2.1.2.1 Selector Field

Selector Field (S[4:0]) is a 5-bit wide field, encoding 32 possible messages. Selector Field encoding definitions are shown in Annex 98A. Combinations not specified are reserved. Reserved combinations of the Selector Field shall not be transmitted.

The Selector Field for IEEE Std 802.3 is shown in Table 98–3.

Table 98–3—Selector Field Encoding

S4	S3	S2	S1	S0	Selector description
0	0	0	0	1	IEEE Std 802.3

98.2.1.2.2 Echoed Nonce Field

Echoed Nonce Field (E[4:0]) is a 5-bit wide field containing the nonce received from the link partner. If the device has not received a DME page with good CRC16, the bits in this field shall contain logical zeros. If the device has received a DME page with good CRC16, the bits in this field shall contain the value received in the Transmitted Nonce Field from the link partner at the same time as the Acknowledge bit is set.

98.2.1.2.3 Transmitted Nonce Field

Transmitted Nonce Field (T[4:0]) is a 5-bit wide field whose lower 4 bits contains a random or pseudo-random number. A new value shall be generated for each entry to the Ability Detect state. The method of generating the nonce is left to the implementer. The lower 4 bits of the transmitted nonce should have a uniform distribution in the range from 0 to $2^4 - 1$. The method used to generate the value should be designed to minimize correlation to the values generated by other devices.

Bit T[4] should be set to 1 if the device prefers or is forced to be MASTER and 0 if the device prefers or is forced to be SLAVE.

If the device has received a DME page with good CRC16 and the link partner has a Transmitted Nonce Field (T[4:0]) that matches the devices generated T[4:0], the device shall invert its T[0] bit and regenerate a new random value for T[3:1] and use that as its new T[4:0] value. Since the DME pages are exchanged in a half-duplex manner, it is possible to swap to a new T[4:0] value prior to transmitting the DME page. One device will always see a DME page with good CRC16 before the other device hence this swapping will guarantee that nonce_match will never be true.

98.2.1.2.4 Technology Ability Field

Technology Ability Field (A[26:0]) is a 27-bit wide field containing information indicating supported technologies specific to the selector field value when used with the Auto-Negotiation Ethernet. These bits are mapped to individual technologies such that abilities are advertised in parallel for a single selector field value. The Technology Ability Field encoding for the IEEE 802.3 selector with Auto-Negotiation Ethernet is described in 98B.3.

Multiple technologies may be advertised in the link codeword. A device shall support the data service ability for a technology it advertises. It is the responsibility of the Arbitration function to determine the common mode of operation shared by a link partner and to resolve multiple common modes.

98.2.1.2.5 Force MASTER-SLAVE

The force MASTER-SLAVE bit, D12, allows a device to force its MASTER-SLAVE configuration. If this bit is set to 0 then the device is in preferred mode, otherwise it is in the forced mode. The MASTER-SLAVE resolution is shown in Table 98–4.

Table 98–4—MASTER-SLAVE Configuration

Local Device		Remote Device		Local Device Resolution	Remote Device Resolution
M/S	T[4]	M/S	T[4]		
0	X	0	X	Device with higher T[4:0] is MASTER, otherwise SLAVE	Device with higher T[4:0] is MASTER, otherwise SLAVE
0	X	1	0	MASTER	SLAVE
0	X	1	1	SLAVE	MASTER
1	0	0	X	SLAVE	MASTER
1	1	0	X	MASTER	SLAVE
1	0	1	0	Configuration Fault	Configuration Fault
1	0	1	1	SLAVE	MASTER
1	1	1	0	MASTER	SLAVE
1	1	1	1	Configuration Fault	Configuration Fault

98.2.1.2.6 Pause Ability

Pause (C0:C1) is encoded in bits D11:D10 of the base link codeword. The 2-bit Pause is encoded as follows:

a) C0 is the same as PAUSE as defined in Annex 28B
b) C1 is the same as ASM_DIR as defined in Annex 28B

The Pause encoding is defined in 28B.2, Table 28B–2. The PAUSE bit indicates that the device is capable of providing the symmetric PAUSE functions as defined in Annex 31B. The ASM_DIR bit indicates that asymmetric PAUSE is supported. The value of the PAUSE bit when the ASM_DIR bit is set indicates the direction the PAUSE frames are supported for flow across the link. Asymmetric PAUSE configuration results in independent enabling of the PAUSE receive and PAUSE transmit functions as defined by Annex 31B. See 28B.3 regarding PAUSE configuration resolution.

98.2.1.2.7 Remote Fault

Remote Fault (RF) is encoded in bit D13 of the base link codeword. The default value is logical zero. The Remote Fault bit provides a standard transport mechanism for the transmission of simple fault information. When the RF bit in the BASE-T1 AN advertisement register (register 7.514.13) is set to logical one, the RF bit in the transmitted base link codeword is set to logical one. When the RF bit in the received base link codeword is set to logical one, the Remote Fault bit in the BASE-T1 AN LP Base Page ability register (register 7.517.13) will be set to logical one, if the management function is present.

98.2.1.2.8 Acknowledge

Acknowledge (Ack) is used by the Auto-Negotiation function to indicate that a device has successfully received its link partner’s link codeword. The Acknowledge Bit is encoded in bit D14 of link codeword. If no Next Page information is to be sent, this bit shall be set to logical one in the link codeword after the reception of at least one DME page with a correct CRC. If Next Page information is to be sent, this bit shall be set to logical one after the device has successfully received at least one DME page with correct CRC, and will remain set until the Next Page information has been loaded into the BASE-T1 AN NEXT PAGE transmit register (Registers 7.520, 7.521, 7.522). In order to save the current received link codeword, it shall be read from the BASE-T1 AN LP NEXT PAGE ability register (register 7.523, 7.524, 7.525) before the Next Page of transmit information is loaded into the BASE-T1 AN NEXT PAGE transmit register. After the COMPLETE ACKNOWLEDGE state has been entered, the link codeword will be transmitted at least three times.

98.2.1.2.9 Next Page

Next Page (NP) is encoded in bit D15 of link codeword. Support of Next Pages is mandatory. If the device does not have any Next Pages to send, the NP bit shall be set to logical zero. If a device wishes to engage in Next Page exchange, it shall set the NP bit to logical one. If a device has no Next Pages to send and its link partner has set the NP bit to logical one, it shall transmit Next Pages with Null message codes and the NP bit set to logical zero while its link partner transmits valid Next Pages. Next page exchanges will occur if either the device or its link partner sets the Next Page bit to logical one. The Next Page function is defined in 98.2.4.3.

98.2.1.3 Transmit Switch function

The Transmit Switch function shall enable the transmit path from a single technology-dependent PHY to the MDI once a highest common denominator choice has been made and Auto-Negotiation has completed. During Auto-Negotiation, the Transmit Switch function shall connect only the DME page generator controlled by the Transmit state diagram to the MDI. When a PHY is connected to the MDI through the Transmit Switch function, the signals at the MDI shall conform to all of the PHY's specifications.

98.2.2 Receive function requirements

The Receive function detects the DME page sequence, decodes the information contained within, and stores the data in rx_link_code_word[64:1]. The receive function incorporates a receive switch to control connection to the various PMAs.

98.2.2.1 DME page reception

To be able to detect the DME bits, the receiver should have the capability to receive DME signals sent with the electrical specifications of the PHY. The DME transmit signal level is specified in 98.2.1.1.4.

98.2.2.2 Receive Switch function

The Receive Switch function shall enable the receive path from the MDI to a single technology-dependent PHY once a highest common denominator choice has been made and Auto-Negotiation has completed.

During Auto-Negotiation, the Receive Switch function shall connect the DME page receiver controlled by the Receive state diagram to the MDI and the Receive Switch function shall also connect the appropriate receivers to the MDI.

98.2.2.3 Link codeword matching

The Receive function shall generate ability_match, acknowledge_match, and consistency_match variables as defined in Arbitration state diagram Figure 98–7.

98.2.3 AN half-duplex function requirements

The AN half-duplex function is defined by Figure 98–10 and ensures that only one of the link partners is transmitting a DME page at each step during the DME page exchange process. The AN half-duplex function uses a blind timer to ensure that the receiver ignores signals reflected from the channel following the end of the device's transmitted DME page. A silent timer is used to ensure that the device does not begin transmitting the DME page until after the link partner has exited from the blind timer. The half-duplex backoff timer resolves concurrent transmissions by using a random wait time to listen for a DME page to arrive from the link partner before the local device transmits a DME page.

98.2.4 Arbitration function requirements

The Arbitration function is defined by Figure 98–7 and ensures proper sequencing of the Auto-Negotiation function using the Transmit function, Receive function, and AN half-duplex function. The Arbitration function enables the Transmit function to advertise and acknowledge abilities. Upon indication of acknowledgment, the Arbitration function determines the highest common denominator using the priority resolution function and enables the appropriate technology-dependent PHY via the Technology Dependent Interface (see 98.4).

98.2.4.1 Renegotiation function

A Renegotiation request from any entity, such as a management agent, shall cause the Arbitration function to disable all technology-dependent PHYs and halt any transmit data and link transition activity until the break_link_timer expires. Consequently, the link partner will go into link fail and normal Auto-Negotiation resumes. The local device shall resume Auto-Negotiation after the break_link_timer has expired by issuing DME pages with the Base Page valid in tx_link_code_word[64:1]. Once Auto-Negotiation has completed, renegotiation will take place if the Highest Common Denominator technology that receives link_control = ENABLE returns link_status = FAIL. To allow the PHY an opportunity to determine link integrity using its own link integrity test function, the link_fail_inhibit_timer qualifies the link_status = FAIL indication such that renegotiation takes place if the link_fail_inhibit_timer has expired and the PHY still indicates link_status = FAIL.

98.2.4.2 Priority Resolution function

Since a local device and a link partner may have multiple common abilities, a mechanism to resolve which mode to configure is required. The mechanism used by Auto-Negotiation is a Priority Resolution function that predefines the hierarchy of supported technologies. The single PHY enabled to connect to the MDI by Auto-Negotiation shall be the technology corresponding to the bit in the Technology Ability Field common to the local device and link partner that has the highest priority as defined in 98B.4 (listed from highest priority to lowest priority).

The common technology is referred to as the *highest common denominator*, or HCD, technology. If the local device receives a Technology Ability Field with a bit set that is reserved, the local device shall ignore that bit for priority resolution. Determination of the HCD technology occurs on entrance to the AN GOOD CHECK state. In the event that there is no common technology, HCD shall have a value of “NULL,” indicating that no PHY receives link_control = ENABLE and link_status[HCD] = FAIL.

98.2.4.3 Next Page function

The Next Page function uses the Auto-Negotiation arbitration mechanisms to allow exchange of Next Pages of information, which may follow the transmission and acknowledgment procedures used for the base link codeword. The Next Page has both Message Code Field and Unformatted Code Fields.

A dual acknowledgment system is used. Acknowledge (Ack) is used to acknowledge receipt of the information; Acknowledge 2 (Ack2) is used to indicate that the receiver is able to act on the information (or perform the task) defined in the message.

The Toggle (T) bit is used to ensure proper synchronization between the local device and the link partner.

Next page exchange occurs after the base link codewords have been exchanged if either end of the link segment set the Next Page bit to logical one indicating that it had at least one Next Page to send. Next page exchange consists of using the normal Auto-Negotiation arbitration process to send Next Page messages.

The Next Page contains two message encodings. The message encodings are defined as follows: message code, which contains predefined 11-bit codes, and unformatted code, which contains 32-bit codes. Multiple Next Pages with appropriate message codes and unformatted codes can be transmitted to send extended messages. Each series of Next Pages shall have a Message code that defines how the Unformatted codes will be interpreted. Any number of Next Pages may be sent in any order; however, it is recommended that the total number of Next Pages sent be kept small to minimize the link startup time.

Next Page transmission ends when both ends of a link segment set their Next Page bits to logical zero, indicating that neither has anything additional to transmit. It is possible for one device to have more pages to transmit than the other device. Once a device has completed transmission of its Next Page information, it shall transmit Next Pages with Null message codes and the NP bit set to logical zero while its link partner continues to transmit valid Next Pages. An Auto-Negotiation able device shall recognize reception of Message Pages with Null message codes as the end of its link partner's Next Page information.

98.2.4.3.1 Next page encodings

The Next Page shall use the encoding shown in Table 98–5 and Table 98–6 for the NP, Ack, MP, Ack2, and T bits. These bits shall function as specified in 98.2.1.2.9, 98.2.1.2.8, 28.2.3.4.5, 28.2.3.4.6, and 28.2.3.4.7 respectively. There are two types of Next Page encodings: message and unformatted. For message Next Pages, the MP bit shall be set to logical one, the 11-bit field D[10:0] shall be encoded as a Message Code Field and D[47:16] shall be encoded as Unformatted Code Field. For Unformatted Next Pages, the MP bit shall be set to logical ZERO: D[10:0] and D[47:16] shall be encoded as the Unformatted Code Field.

Table 98–5—Message Next Page

D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
M0	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	T	Ack2	MP	Ack	NP
D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31
U0	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U11	U12	U13	U14	U15
D32	D33	D34	D35	D36	D37	D38	D39	D40	D41	D42	D43	D44	D45	D46	D47
U16	U17	U18	U19	U20	U21	U22	U23	U24	U25	U26	U27	U28	U29	U30	U31

Table 98–6—Unformatted Next Page

D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
U0	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	T	Ack2	MP	Ack	NP
D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31
U11	U12	U13	U14	U15	U16	U17	U18	U19	U20	U21	U22	U23	U24	U25	U26
D32	D33	D34	D35	D36	D37	D38	D39	D40	D41	D42	D43	D44	D45	D46	D47
U27	U28	U29	U30	U31	U32	U33	U34	U35	U36	U37	U38	U39	U40	U41	U42

98.2.4.3.2 Use of Next Pages

Next page exchange shall commence after the Base Page exchange if either device requests it by setting the NP bit to logical one.

Next page exchange shall continue until neither device on a link has more pages to transmit as indicated by the NP bit. A Next Page with a Null Message Code Field value shall be sent if the device has no other information to transmit.

A message code can carry either a specific message or information that defines how the corresponding unformatted codes should be interpreted.

98.3 State diagram variable to Auto-Negotiation register mapping

The state diagrams of Figure 98–7 to Figure 98–10 generate and accept variables of the form "mr_x," where x is an individual signal name. These variables comprise a management interface to communicate Auto-Negotiation information to and from the management entity. Clause 45 MDIO registers are defined in MMD7 to support Auto-Negotiation. The Clause 45 MDIO electrical interface is optional. Where no physical embodiment of the MDIO exists, provision of an equivalent mechanism to access the information is recommended.

Table 98–7 describes the MDIO register to the state diagrams variable mapping.

Table 98–7—State diagram variable to Single lane-media Auto-Negotiation MDIO register mapping

State diagram variable	Description / MDIO register mapping
mr_adv_ability[48:1]	{7.516.15:0, 7.515.15:0, 7.514.15:0} BASE-T1 AN advertisement registers
mr_autoneg_complete	7.513.5 Auto-Negotiation complete
mr_autoneg_enable	7.512.12 Auto-Negotiation enable
mr_lp_adv_ability[48:1]	For Base Page: {7.519.15:0, 7.518.15:0, 7.517.15:0} BASE-T1 AN LP Base Page ability registers For Next Page(s): {7.525.15:0, 7.524.15:0, 7.523.15:0} BASE-T1 AN LP NEXT PAGE ability register
mr_main_reset	7.512.15 AN reset
mr_next_page_loaded	Set on write to BASE-T1 AN NEXT PAGE transmit register; cleared by Arbitration state diagram
mr_np_tx[48:1]	{7.522.15:0, 7.521.15:0, 7.520.15:0} BASE-T1 AN NEXT PAGE transmit register
mr_page_rx	7.513.6 Page received
mr_restart_negotiation	7.512.9 Restart Auto-Negotiation
set to 1	7.513.3 Auto-Negotiation ability

98.4 Technology-Dependent Interface

The Technology-Dependent Interface is the communication mechanism between each technology's PMA and the Auto-Negotiation function. Auto-Negotiation can support multiple technologies, all of which need not be implemented in a given device. Each of these technologies may utilize its own technology-dependent link integrity test function.

98.4.1 PMA_LINK.indication

This primitive is generated by the PMA to indicate the status of the underlying medium. The purpose of this primitive is to give the Auto-Negotiation function a means of determining the validity of received code elements.

98.4.1.1 Semantics of the service primitive

PMA_LINK.indication(link_status)

The link_status parameter shall assume one of two values: OK or FAIL, indicating whether the underlying receive channel is intact and enabled (OK) or not intact (FAIL).

98.4.1.2 When generated

A technology-dependent PMA generates this primitive to indicate a change in the value of link_status.

98.4.1.3 Effect of receipt

The effect of receipt of this primitive shall be governed by the state diagram of Figure 98–7.

98.4.2 PMA_LINK.request

This primitive is generated by Auto-Negotiation to allow it to enable and disable operation of the PMA.

98.4.2.1 Semantics of the service primitive

PMA_LINK.request(link_control)

The link_control parameter shall assume one of two values: DISABLE or ENABLE.

The link_control=DISABLE mode shall be used by the Auto-Negotiation function to disable PMA processing.

The link_control=ENABLE mode shall be used by Auto-Negotiation to turn control over to a single PMA for all normal processing functions.

98.4.2.2 When generated

The Auto-Negotiation function shall generate this primitive to indicate to the PHY how to respond, in accordance with the state diagram of Figure 98–7. Upon power-on or reset, if the Auto-Negotiation function is enabled (mr_autoneg_enable=true) the PMA_LINK.request(DISABLE) message shall be issued to all technology-dependent PMAs.

98.4.2.3 Effect of receipt

This primitive affects operation of the underlying PMA.

98.5 Detailed functions and state diagrams

The notation used in state diagrams follows the conventions in Clause 28. Variables in a state diagram with default values evaluate to the variable default in each state where the variable value is not explicitly set.

Auto-Negotiation shall implement the Transmit state diagram, Receive state diagram, half-duplex state diagram, and Arbitration state diagram. Additional requirements to these state diagrams are made in the respective functional requirements sections. Options to these state diagrams clearly stated as such in the functional requirements sections or state diagrams shall be allowed. In the case of any ambiguity between stated requirements and the state diagrams, the state diagrams shall take precedence.

98.5.1 State diagram variables

A variable with "_[x]" appended to the end of the variable name indicates a variable or set of variables as defined by "x". "x" may be as follows:

- all; represents all specific technology-dependent PMAs supported in the local device.
- HCD; represents the single technology-dependent PMA chosen by Auto-Negotiation as the highest common denominator technology through the Priority Resolution.
- notHCD; represents all technology-dependent PMAs not chosen by Auto-Negotiation as the highest common denominator technology through the Priority Resolution.
- 1GigT1; represents that the 1000BASE-T1 PMA is the signal source.
- 2.5GigT1; represents that the 2.5GBASE-T1 PMA is the signal source.
- 5GigT1; represents that the 5GBASE-T1 PMA is the signal source.
- 10GigT1; represents that the 10GBASE-T1 PMA is the signal source.
- 25GigT1; represents that the 25GBASE-T1 PMA is the signal source.

Variables with [48:1] appended to the end of the variable name indicate arrays that can be directly mapped to 48-bit registers. For these variables, "[x]" indexes an element or set of elements in the array, where "[x]" may be as follows:

a) Any integer
b) Any range of integers
c) Any variable that takes on integer values
d) NP; represents the index of the Next Page bit
e) ACK; represents the index of the Acknowledge bit
f) RF; represents the index of the Remote Fault bit

Variables of the form "mr_x", where x is a label, comprise a management interface that is intended to be connected to the Management function. However, an implementation-specific management interface may provide the control and status function of these bits. The mapping between state diagram variables and Auto-Negotiation MDIO registers is shown in Table 98–7.

ability_match
Indicates that at least one link codeword with good CRC16 was received.
Values:
false: at least one link codeword with good CRC16 has not been received (default)
true: at least one link codeword with good CRC16 has been received

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

ability_match_word [48:1]
A 48-bit array that is loaded upon transition to Acknowledge Detect state with the value of the link codeword that caused ability_match = true for that transition. For each element in the array transmitted.
Values:
ZERO: data bit is logical zero
ONE: data bit is logical one

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

ack_finished
Status indicating that the final remaining_ack_cnt link codewords with the Ack bit set have been transmitted.
Values:
false: more link codewords with the Ack bit set to logical one is transmitted
true: all remaining link codewords with the Ack bit set to logical one have been transmitted

ack_nonce_match
Indicates whether the echoed nonce received from the link partner matches the transmitted nonce field sent by the local device. The echoed nonce value from the DME page that caused acknowledge_match to be set is used for this test.
Values:
false: link partner echoed nonce does not equal local device transmitted nonce
true: link partner echoed nonce equals local device transmitted nonce

acknowledge_match
Indicates that at least one link codeword with the Acknowledge bit set and with good CRC16 was received. The link codeword that initially set the ability_match variable should not be used to set this variable.
Values:
false: at least one link codeword with the Acknowledge bit set and with good CRC16 has not been received (default)
true: at least one link codeword with the Acknowledge bit set and with good CRC16 has been received

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

an_link_good
Indicates that Auto-Negotiation has completed.
Values:
false: negotiation is in progress (default)
true: negotiation is complete, forcing the Transmit and Receive functions to IDLE

an_receive_idle
Indicates that the Receive state diagram is in the IDLE state.
Values:
false: the Receive state diagram is not in the IDLE state (default)
true: the Receive state diagram is in the IDLE state

ANSP
This variable contains the type of the selected Auto-Negotiation speed.
Values:
HSM: high-speed mode
LSM: low-speed mode

base_page
Status indicating that the page currently being transmitted by Auto-Negotiation is the initial link codeword encoding used to communicate the device’s abilities.
Values:
false: a page other than base link codeword is being transmitted
true: the base link codeword is being transmitted

code_sel

A random or pseudo-random value uniformly distributed. A new value is generated each time the variable code_sel is used.

Values:

ZERO: a zero has been assigned
ONE: a one has been assigned

complete_ack

Controls the counting of transmitted link codewords that have their Acknowledge bit set.

Values:

false: transmitted link codewords with the Acknowledge bit set are not counted (default)
true: transmitted link codewords with the Acknowledge bit set are counted

consistency_match

Indicates that the ability_match_word is the same as the link codeword that caused acknowledge_match to be set.

Values:

false: the link codeword that caused ability_match to be set is not the same as the link codeword that caused acknowledge_match to be set, ignoring the Acknowledge bit value and the echoed nonce value
true: the link codeword that caused ability_match to be set is the same as the link codeword that caused acknowledge_match to be set, ignoring the Acknowledge bit value and the echoed nonce value

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

detect_mv_end

Status indicating that the receiver has detected the end delimiter.

Values:

false: set to false after any Receive State Diagram state transition (default)
true: end delimiter has been detected

detect_mv_start

Status indicating that the receiver has detected a Start Delimiter as defined in 98.2.1.1.1.

Values:

false: set to false after any Receive State Diagram state transition (default)
true: Start Delimiter has been detected

detect_transition

Status indicating that the receiver has detected a transition.

Values:

false: set to false after any Receive State Diagram state transition (default)
true: set to true when a transition is received

incompatible_link

Parameter used following Priority Resolution to indicate the resolved link is incompatible with the local device settings. A device’s ability to set this variable to true is optional.

Values:

false: a compatible link exists between the local device and link partner (default)
true: optional indication that Priority Resolution has determined no highest common denominator exists following the most recent negotiation

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

link_control_[x]
Controls the connection of each PMD to the MDI. When all PMD transmitters are isolated from the MDI, the AN transmitter is connected to the MDI.
Values:
DISABLE: isolates the PMD from the MDI
ENABLE: connects the PMD (both transmit and receive) to the MDI

link_status_[x]
The link_status parameter set by PMA Link Monitor and passed to the PCS via the PMA_LINK.indication primitive. This variable takes the values of OK or FAIL.

mr_autoneg_complete
Status indicating whether Auto-Negotiation has completed or not.
Values:
false: Auto-Negotiation has not completed
true: Auto-Negotiation has completed

mr_autoneg_enable
Controls the enabling and disabling of the Auto-Negotiation function.
Values:
false: Auto-Negotiation is disabled
true: Auto-Negotiation is enabled

mr_adv_ability[48:1]
A 48-bit array that contains the Advertised Abilities link codeword. For each element within the array:
Values:
ZERO: data bit is logical zero
ONE: data bit is logical one

mr_lp_adv_ability[48:1]
A 48-bit array that contains the link partner's Advertised Abilities link codeword. For each element within the array:
Values:
ZERO: data bit is logical zero
ONE: data bit is logical one

mr_main_reset
Controls the resetting of the Auto-Negotiation state diagrams.
Values:
false: do not reset the Auto-Negotiation state diagrams
true: reset the Auto-Negotiation state diagrams

mr_next_page_loaded
Status indicating whether a new page has been loaded into the BASE-T1 AN NEXT PAGE transmit register (see 45.2.7.24).
Values:
false: a New Page has not been loaded
true: a New Page has been loaded

mr_np_tx[48:1]
A 48-bit array that contains the new Next Page to transmit. For each element within the array:
Values:

ZERO: data bit is logical zero
ONE: data bit is logical one

mr_page_rx
Status indicating whether a New Page has been received. A New Page has been successfully received when acknowledge_match = true and consistency_match = true and the link codeword has been written to mr_lp_adv_ability[48:1].
Values:
false: a New Page has not been received
true: a New Page has been received

mr_restart_negotiation
Controls the entrance to the TRANSMIT DISABLE state to break the link before Auto-Negotiation is allowed to renegotiate via management control.
Values:
false: renegotiation is not taking place
true: renegotiation is started

multispeed_autoneg_reset
See 98.5.6.1.

nonce_match
Indicates whether the transmitted nonce received from the link partner matches the transmitted nonce field sent by the local device.
Values:
false: link partner transmitted nonce does not equal local device transmitted nonce
true: link partner transmitted nonce equals local device transmitted nonce

np_rx
Flag to hold the value of rx_link_code_word[NP] upon entry to the COMPLETE ACKNOWLEDGE state. This value is associated with the value of rx_link_code_word[NP] when acknowledge_match was last set.
Values:
ZERO: local device np_rx bit equals a logical zero
ONE: local device np_rx bit equals a logical one

page_polarity
Starting polarity of the page.
Values:
ZERO: starting polarity of page is negative
ONE: starting polarity of page is positive

power_on
Condition that is true until such time as the power supply for the device that contains the Auto-Negotiation state diagrams has reached the operating region or the device has low-power mode set via 1000BASE-T1 PMA control register bit 1.2304.11 or via 10BASE-T1L PMA control register bit 1.2294.11.
Values:
false: the device is completely powered (default)
true: the device has not been completely powered

receive_blind
Controls whether the receiver should ignore activity on the line.
Values:

true: ignore received DME transitions
false: accept received DME transitions

receive_DME_active
Status indicating whether or not a DME page reception is in progress.
Values:
true: DME page reception in progress
false: DME page reception completed

rx_link_code_word[64:1]
A 64-bit array that contains the data bits to be received from a DME page. For each element within the array:
Values:
ZERO: data bit is a logical zero
ONE: data bit is a logical one

rx_nonce[4:0]
A 5-bit array that contains the transmitted nonce received from the DME page that caused ability_match = true. For each element within the array:
Values:
ZERO: data bit is a logical zero
ONE: data bit is a logical one

TD_AUTONEG
Controls the signal sent by Auto-Negotiation on the TD_AUTONEG circuit.
Values:
disable: transmission of Auto-Negotiation signals is disabled
idle: Auto-Negotiation maintains the current signal level on the MDI
mv_end_delimiter: Auto-Negotiation causes the transmission of the end delimiter on the MDI
mv_start_delimiter: Auto-Negotiation causes the transmission of the Start Delimiter on the MDI as defined in 98.2.1.1.1
transition: Auto-Negotiation causes a transition in the level on the MDI

toggle_rx
Flag to keep track of the state of the link partner's Toggle bit.
Values:
ZERO: link partner's Toggle bit equals logical zero
ONE: link partner's Toggle bit equals logical one

toggle_tx
Flag to keep track of the state of the local device's Toggle bit.
Values:
ZERO: local device's Toggle bit equals logical zero
ONE: local device's Toggle bit equals logical one

transmit_ability
Controls the transmission of the link codeword containing tx_link_code_word[64:1].
Values:
false: any transmission of tx_link_code_word[64:1] is halted (default)
true: the transmit state diagram begins sending tx_link_code_word[64:1]

transmit_ack
Controls the setting of the Acknowledge bit in the tx_link_code_word[64:1] to be transmitted.

Values:

false: sets the Acknowledge bit in the transmitted tx_link_code_word[64:1] to a logical zero (default)

true: sets the Acknowledge bit in the transmitted tx_link_code_word[64:1] to a logical one

transmit_disable

Controls the transmission of tx_link_code_word[64:1].

Values:

false: tx_link_code_word[64:1] transmission is allowed (default)

true: tx_link_code_word[64:1]transmission is halted

transmit_DME_done

Status indicating the DME page transmission completed.

Values:

true: DME page transmission completed

false: DME page transmission in progress

transmit_DME_wait

Control indication whether a DME page can be transmitted.

Values:

true: pause DME page transmission

false: continue DME page transmission

transmit_mv_end_done

Status indicating that the transmission of the end delimiter has completed.

Values:

false: transmission of the end delimiter is in progress

true: transmission of the end delimiter has completed

transmit_mv_start_done

Status indicating that the transmission of the Start Delimiter defined in 98.2.1.1.1 has been completed.

Values:

false: transmission of the Start Delimiter is in progress

true: transmission of the Start Delimiter has been completed

tx_link_code_word[64:1]

A 64-bit array that contains the data bits to be transmitted in an DME page. tx_link_code_word[48:1] contains the Auto-Negotiation page to be transmitted. tx_link_code_word[64:49] contains the CRC16. This array may be loaded from mr_adv_ability or mr_np_tx. For each element within the array:

Values:

ZERO: data bit is logical zero

ONE: data bit is logical one.

98.5.2 State diagram timers

All timers operate in the manner described in 40.4.5.2.

When operating in high-speed mode, the following timer value definitions shall apply:

backoff_timer_[HSM]

Timer for the random amount of time to wait for a page to arrive from the link partner before

transmitting a page. The timer shall expire according to the formula below after being started.
If T[4] bit is 1, the timer duration is (6805 ns to 6925 ns) + (random integer from 0 to 15) × (2120 ns to 2240 ns).
If T[4] bit is 0, the timer duration is (7895 ns to 8015 ns) + (random integer from 0 to 15) × (2120 ns to 2240 ns).
A new random integer from 0 to 15 inclusive is generated every time the backoff_timer_[HSM] is started. The random value should be uniformly distributed.

blind_timer_[HSM]
Timer for the amount of time to blind the receiver after end of transmission to prevent the device from seeing its own echo. The timer shall expire 2000 ns to 2120 ns after being started.

break_link_timer_[HSM]
Timer for the amount of time to wait in TRANSMIT DISABLE to assure that the link partner will exit from either ACKNOWLEDGE DETECT or NEXT PAGE WAIT; effect on the link partner in other states is not defined. The timer shall expire 300 µs to 305 µs after being started.

clock_detect_max_timer_[HSM]
Timer for the maximum time between detection of differential Manchester clock transitions. The clock_detect_max_timer_[HSM] shall expire 63 ns to 75 ns after being started or restarted.

clock_detect_min_timer_[HSM]
Timer for the minimum time between detection of differential Manchester clock transitions. The clock_detect_min_timer_[HSM] shall expire 45 ns to 57 ns after being started or restarted.

data_detect_max_timer_[HSM]
Timer for the maximum time between a clock transition and the following data transition. This timer is used in conjunction with the data_detect_min_timer_[HSM] to detect whether the data bit between two clock transitions is a logical zero or a logical one. The data_detect_max_timer_[HSM] shall expire 33 ns to 45 ns from the last clock transition.

data_detect_min_timer_[HSM]
Timer for the minimum time between a clock transition and the following data transition. This timer is used in conjunction with the data_detect_max_timer_[HSM] to detect whether the data bit between two clock transitions is a logical zero or a logical one. The data_detect_min_timer_[HSM] shall expire 15 ns to 27 ns from the last clock transition.

interval_timer_[HSM]
Timer for the separation of a transmitted clock pulse from a data bit. The interval_timer_[HSM] shall expire 30 ns ± 0.01% from each clock pulse and data bit.

page_test_max_timer_[HSM]
Timer for the maximum time between detection of start and end delimiters. The page_test_max_timer_[HSM] shall expire 4800 ns to 4920 ns after being started or restarted.

receive_DME_timer_[HSM]
Timer for the maximum amount of time to receive a complete page before timeout. The timer shall expire 6805 ns to 6925 ns after being started.

rx_wait_timer_[HSM]
Timer for the maximum time between detection of DME pages. This timer is used to detect

whether the link partner is transmitting DME pages. The rx_wait_timer_[HSM] shall expire 15 µs to 17 µs after being started or restarted.

silent_timer_[HSM]
: Timer for the amount of time to wait after receiving a page before transmitting a page. The timer shall expire 2120 ns to 2240 ns after being started.#

When operating in low-speed mode, the following timer value definitions shall apply:

backoff_timer_[LSM]
: Timer for the random amount of time to wait for a page to arrive from the link partner before transmitting a page. The timer shall expire according to the formula below after being started.
If T[4] bit is 1, the timer duration is (156 300 ns to 159 500 ns) + (random integer from 0 to 15) × (31 400 ns to 34 600 ns).
If T[4] bit is 0, the timer duration is (172 800 ns to 176 000 ns) + (random integer from 0 to 15) × (31 400 ns to 34 600 ns).
A new random integer from 0 to 15 inclusive is generated every time the backoff_timer_[LSM] is started. The random value should be uniformly distributed.

blind_timer_[LSM]
: Timer for the amount of time to blind the receiver after end of transmission to prevent the device from seeing its own echo. The timer shall expire 28 200 ns to 31 400 ns after being started.

break_link_timer_[LSM]
: Timer for the amount of time to wait in TRANSMIT DISABLE to assure that the link partner will exit from either ACKNOWLEDGE DETECT or NEXT PAGE WAIT; effect on the link partner in other states is not defined. The timer shall expire 8000 µs to 8133 µs after being started.

clock_detect_max_timer_[LSM]
: Timer for the maximum time between detection of differential Manchester clock transitions. The clock_detect_max_timer_[LSM] shall expire 1680 ns to 2000 ns after being started or restarted.

clock_detect_min_timer_[LSM]
: Timer for the minimum time between detection of differential Manchester clock transitions. The clock_detect_min_timer_[LSM] shall expire 1200 ns to 1520 ns after being started or restarted.

data_detect_max_timer_[LSM]
: Timer for the maximum time between a clock transition and the following data transition. This timer is used in conjunction with the data_detect_min_timer_[LSM] to detect whether the data bit between two clock transitions is a logical zero or a logical one. The data_detect_max_timer_[LSM] shall expire 880 ns to 1200 ns from the last clock transition.

data_detect_min_timer_[LSM]
: Timer for the minimum time between a clock transition and the following data transition. This timer is used in conjunction with the data_detect_max_timer_[LSM] to detect whether the data bit between two clock transitions is a logical zero or a logical one. The data_detect_min_timer_[LSM] shall expire 400 ns to 720 ns from the last clock transition.

interval_timer_[LSM]
: Timer for the separation of a transmitted clock pulse from a data bit. The interval_timer_[LSM] shall expire 800 ns ± 0.005% from each clock pulse and data bit.

page_test_max_timer_[LSM]
: Timer for the maximum time between detection of start and end delimiters. The

page_test_max_timer_[LSM] shall expire 128 000 ns to 131 200 ns after being started or restarted.

receive_DME_timer_[LSM]
: Timer for the maximum amount of time to receive a complete page before timeout. The timer shall expire 156 300 ns to 159 500 ns after being started.

rx_wait_timer_[LSM]
: Timer for the maximum time between detection of DME pages. This timer is used to detect whether the link partner is transmitting DME pages. The rx_wait_timer_[LSM] shall expire 330 µs to 370 µs after being started or restarted.

silent_timer_[LSM]
: Timer for the amount of time to wait after receiving a page before transmitting a page. The timer shall expire 31 400 ns to 34 600 ns after being started.

Depending on the selected PHY type, done by Auto-Negotiation, the following timer values shall be used:

link_fail_inhibit_timer_[HCD]
: Timer for qualifying a link_status=FAIL indication or a link_status=OK indication when a specific technology link is first being established. A link will be considered “failed” only if the link_fail_inhibit_timer_[HCD] has expired and the link has still not gone into the link_status=OK state. The expiration time of the link_fail_inhibit_timer_[HCD] shall be dependent on the selected PHY type. For all PHY types, except 10BASE-T1L and 10BASE-T1S, this timer shall expire 97 ms to 98 ms after entering the AN GOOD CHECK state. For a 10BASE-T1L PHY, this timer shall expire 3030 ms to 3090 ms after entering the AN GOOD CHECK state. For a 10BASE-T1S PHY, this timer shall expire 400 ms to 405 ms after entering the AN GOOD CHECK state.

NOTE—The link_fail_inhibit_timer_[HCD] expiration value is greater than the time required for the link partner to complete Auto-Negotiation after the local device has completed Auto-Negotiation plus the time required for the specific technology to enter the link_status=OK state.

98.5.3 State diagram counters

remaining_ack_cnt
: A counter that may take on integer values from 0 to 3. The number of additional link codewords with the Acknowledge Bit set to logical one to be sent to ensure that the link partner receives the acknowledgment.
: Values:
  - not_done: positive integers between 0 and 2 inclusive
  - done: positive integer 3
  - init: counter is reset to zero

rx_bit_cnt
: A counter that may take on integer values from 0 to 64. This counter is used to keep a count of data bits received from a DME page and to ensure that when erroneous extra transitions are received, the first 48 bits are kept while the next 16 bits are used for CRC16 check and any additional bits are ignored. When this counter reaches 64, enough data bits have been received. This counter does not increment beyond 64 and does not return to 0 until it is reinitialized.

tx_bit_cnt
: A counter that may take on integer values from 1 to 64. This counter is used to keep a count of data bits sent within a DME page. When this counter reaches 64, all data bits have been sent.

98.5.4 Function

CRC16(x[48:1])

Returns the output of the CRC16 generator described in 98.2.1.1.1 after processing the 48-bit input x.

98.5.5 State diagrams

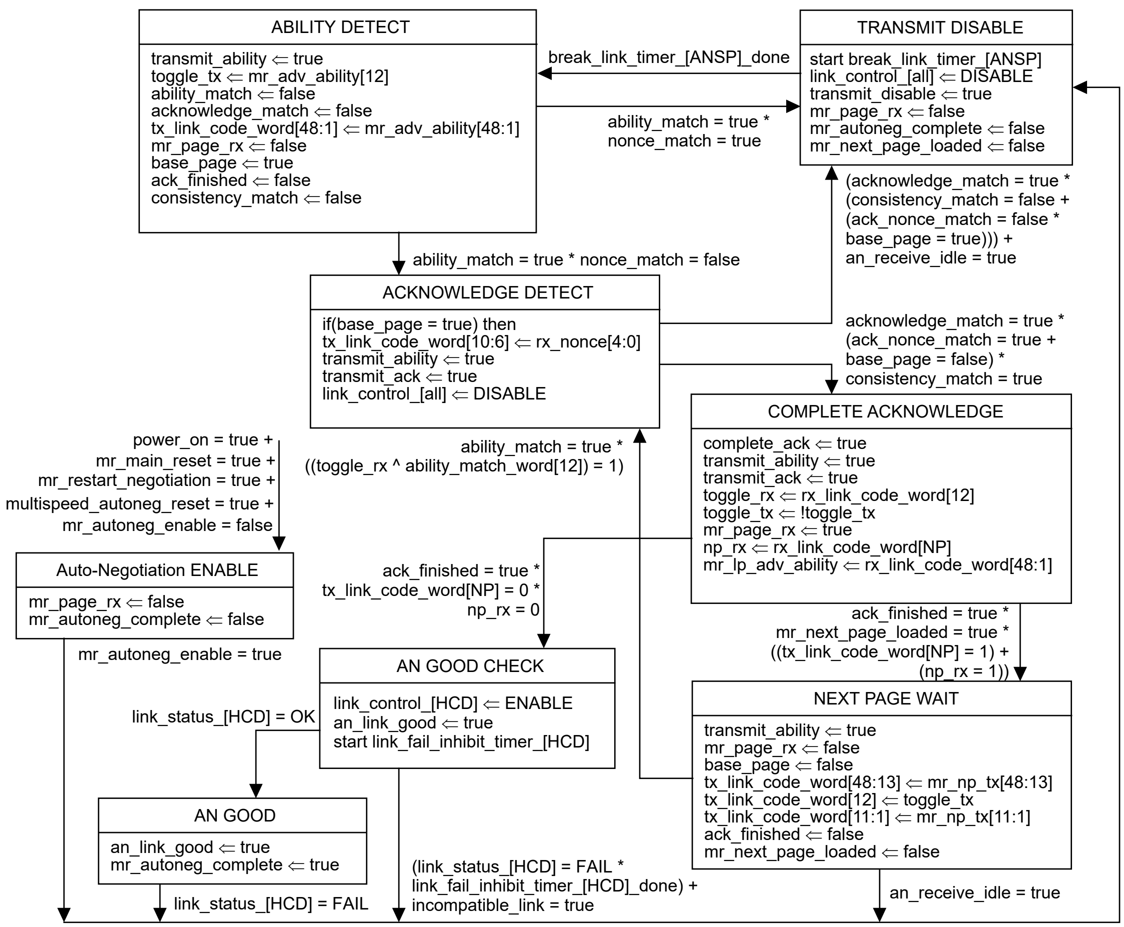


Figure 98–7—Arbitration state diagram

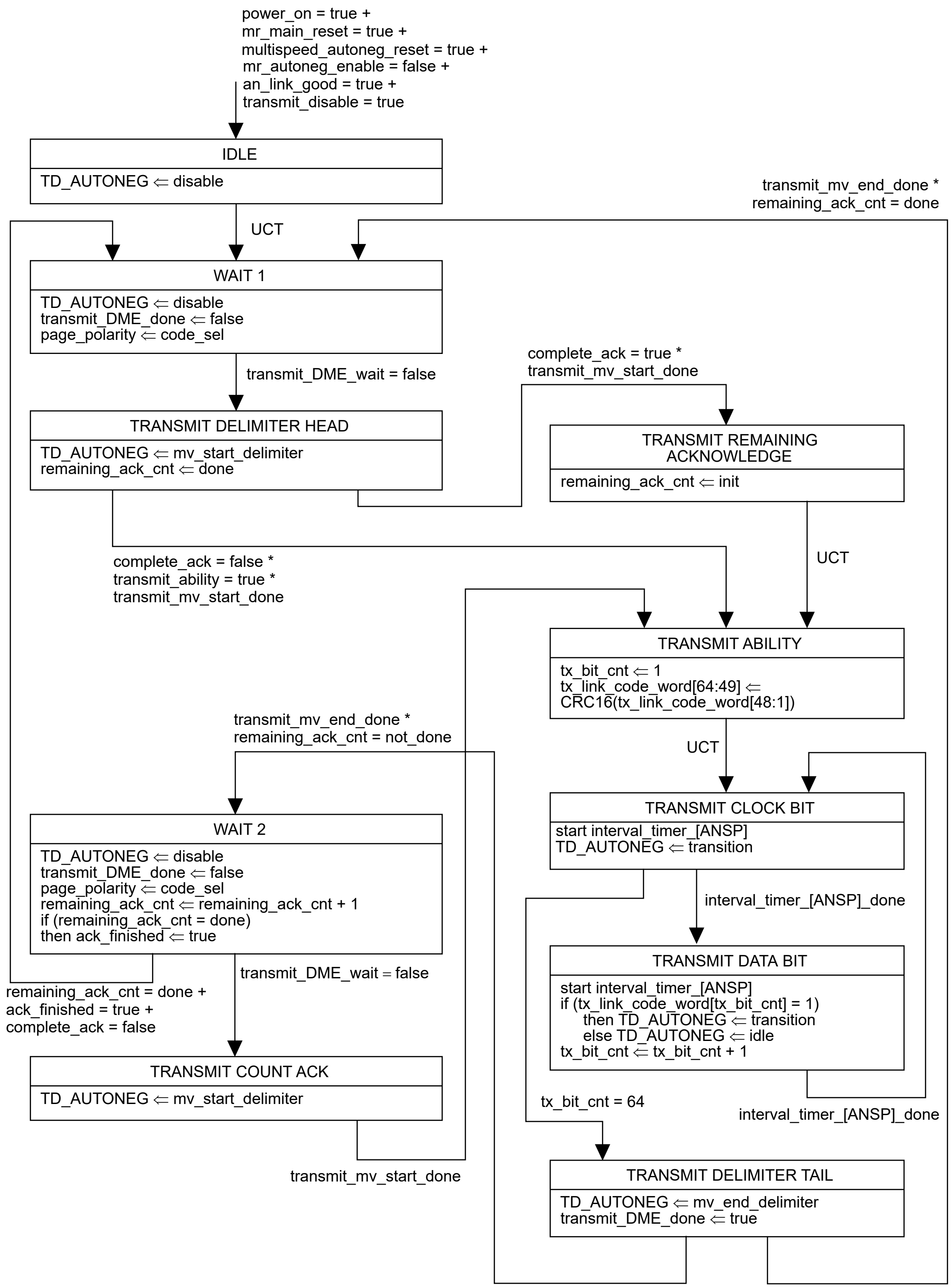


Figure 98–8—Transmit state diagram

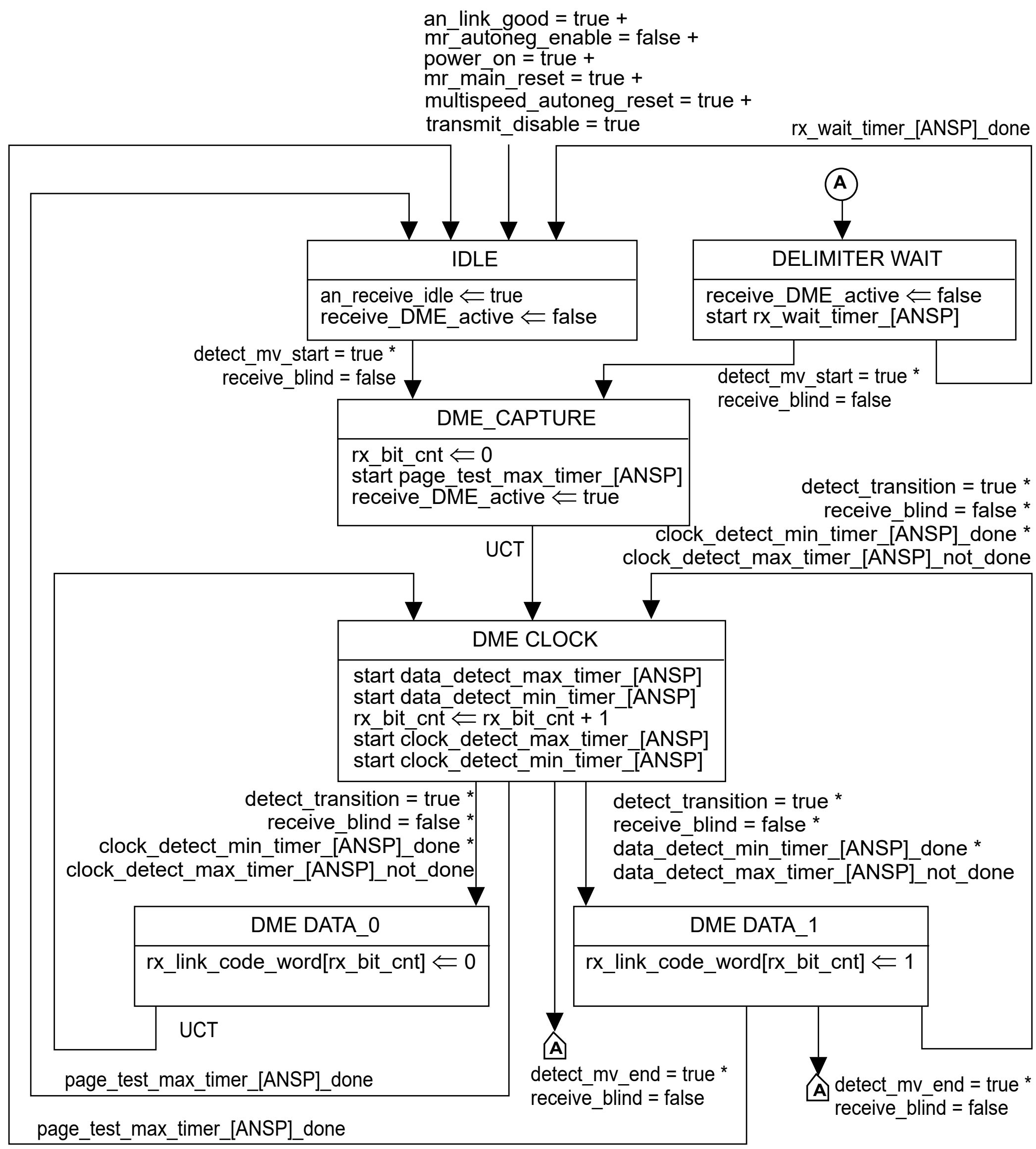


Figure 98–9—Receive state diagram

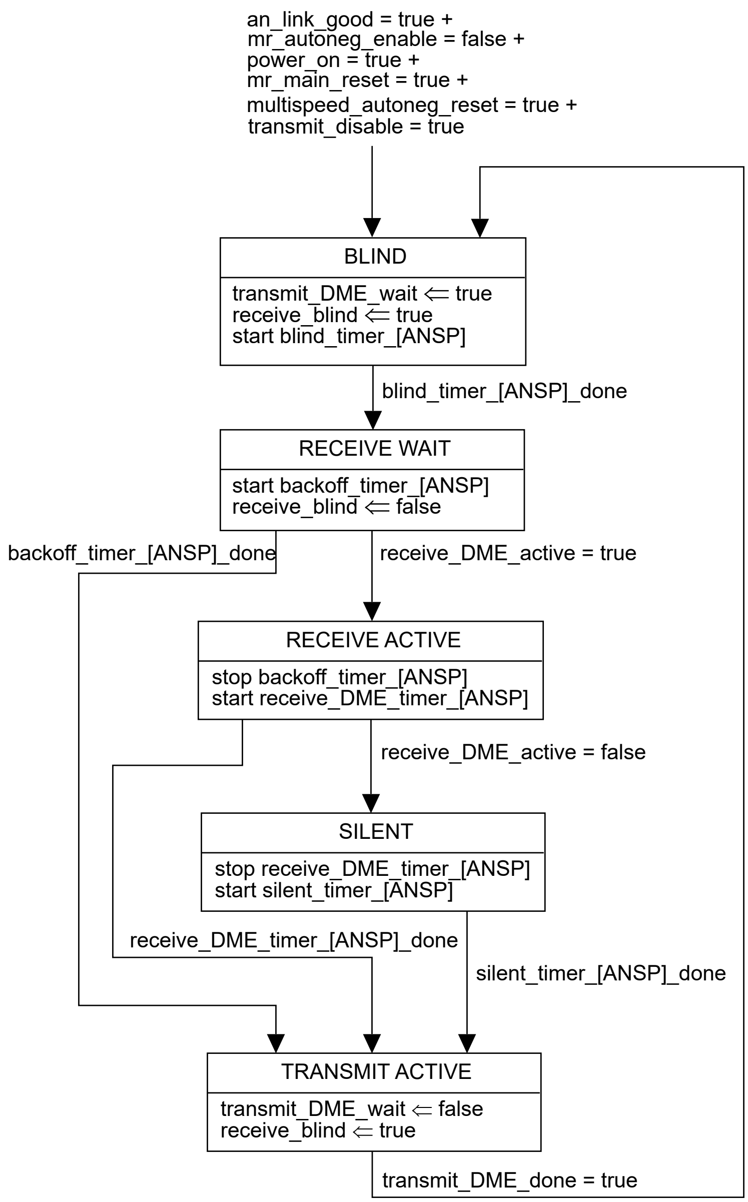


Figure 98–10—Half-duplex state diagram

98.5.6 High-speed and low-speed Auto-Negotiation modes

A PHY supporting two different Auto-Negotiation speeds, as described in 98.2.1.1.2, shall implement the behavior shown in Figure 98–11. Figure 98–11 determines the mode used for the timers in Figure 98–7, Figure 98–8, Figure 98–9, Figure 98–10, and Figure 98–11 through the variable ANSP and synchronizes them through the variable multispeed_autoneg_reset.

A PHY supporting only one Auto-Negotiation speed shall implement the behavior as shown in Figure 98–7, Figure 98–8, Figure 98–9, and Figure 98–10, using the associated timer values for high-speed mode (HSM) or low-speed mode (LSM) Auto-Negotiation as described in 98.5.2.

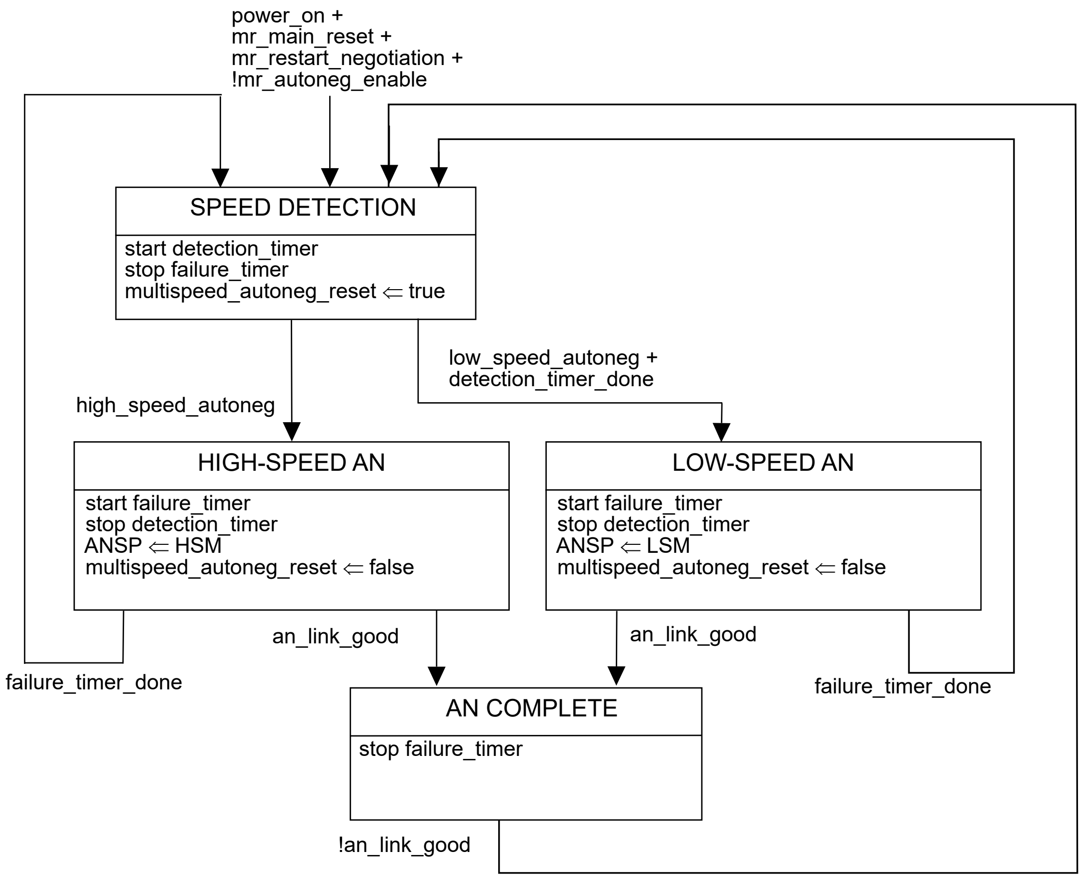


Figure 98–11—Auto-Negotiation—high-speed mode and low-speed mode selection

98.5.6.1 Variables

an_link_good
See 98.5.1.

ANSP
See 98.5.1.

mr_autoneg_enable
See 98.5.1.

mr_main_reset
See 98.5.1.

mr_restart_negotiation
See 98.5.1.

multispeed_autoneg_reset
If two different Auto-Negotiation speeds are implemented and this variable is set to true by the state diagram in Figure 98–11, then the state diagrams in Figure 98–7, Figure 98–8, Figure 98–9, and Figure 98–10 are restarted. If only single speed Auto-Negotiation is implemented, then this variable remains set to false.

Values: true: Auto-Negotiation state diagrams are restarted
false: Auto-Negotiation state diagrams are in normal operation

power_on

See 98.5.1.

98.5.6.2 Functions

high_speed_autoneg

This function returns true if at least the last 12 received DME pulses are within the allowed range for the high-speed Auto-Negotiation communication (15 ns to 135 ns pulse width) including the violations of the DME encoding within the start delimiter; otherwise, this function returns false.
Values: true or false

low_speed_autoneg

This function returns true if at least the last 12 received DME pulses are within the allowed range for the low-speed Auto-Negotiation communication (400 ns to 2000 ns pulse width) including the violations of the DME encoding within the start delimiter; otherwise, this function returns false.
Values: true or false

98.5.6.3 Timers

All timers operate in the manner described in 40.4.5.2.

detection_timer

This timer limits the maximum time for detection of Auto-Negotiation frames sent by the far end PHY, before starting to send its own Auto-Negotiation frames at low-speed. This timer is not automatically restarted after expiration. A new random integer from 0 to 15 inclusive is generated every time the detection_timer is started. The random value should be uniformly distributed.
Timer value: (10 ms ± 0.1 ms) + (random integer from 0 to 15) × (0.5 ms ± 0.05 ms)

failure_timer

This timer limits the maximum time for the underlying Auto-Negotiation state diagrams to complete the Auto-Negotiation process before restarting the Auto-Negotiation process. This timer is not automatically restarted after expiration.
Timer value: 250 ms ± 1 ms

98.6 Protocol implementation conformance statement (PICS) proforma for Clause 98, Auto-Negotiation for Single-lane Media[175]

98.6.1 Introduction

The supplier of a protocol implementation that is claimed to conform to Clause 98, Auto-Negotiation for Single-lane Media, shall complete the following protocol implementation conformance statement (PICS) proforma. A detailed description of the symbols used in the PICS proforma, along with instructions for completing the PICS proforma, can be found in Clause 21.

98.6.2 Identification

98.6.2.1 Implementation identification

Supplier	
Contact point for inquiries about the PICS	
Implementation Name(s) and Version(s)	
Other information necessary for full identification—e.g., name(s) and version(s) for machines and/or operating systems; System Name(s)	
NOTE 1—Only the first three items are required for all implementations; other information may be completed as appropriate in meeting the requirements for the identification. NOTE 2—The terms Name and Version should be interpreted appropriately to correspond with a supplier's terminology (e.g., Type, Series, Model).	

98.6.2.2 Protocol summary

Identification of protocol standard	IEEE Std 802.3-202x, Clause 98, Auto-Negotiation for Single-lane Media
Identification of amendments and corrigenda to this PICS proforma that have been completed as part of this PICS	
Have any Exception items been required? No [ ] Yes [ ] (See Clause 21; the answer Yes means that the implementation does not conform to IEEE Std 802.3-202x.)	

Date of Statement	

[175] *Copyright release for PICS proformas:* Users of this standard may freely reproduce the PICS proforma in this subclause so that it can be used for its intended purpose and may further publish the completed PICS.

98.6.3 Major capabilities/options

Item	Feature	Subclause	Value/Comment	Status	Support
*ANSM	Auto-Negotiation Speed Mode	98.5.6		O	Yes [ ] No [ ]
*HSM	High-Speed Mode	98.5.2, 98.5.6		O	Yes [ ] No [ ]
*LSM	Low-Speed Mode	98.5.2, 98.5.6		O	Yes [ ] No [ ]
*10T1L	10BASE-T1L PHY type	98.5.2		O	Yes [ ] No [ ]
*10T1S	10BASE-T1S PHY type	98.5.2		O	Yes [ ] No [ ]
*DIFF	Differential balanced-pair media	98.1.1		O	Yes [ ] No [ ]
*UNB	Unbalanced single-ended media	98.1.1		O	Yes [ ] No [ ]

98.6.4 General

Item	Feature	Subclause	Value/Comment	Status	Support
G1	Single-lane media Auto-Negotiation function	98.2	Provide Auto-Negotiation transmit, Auto-Negotiation receive, Auto-Negotiation half-duplex, and Auto-Negotiation arbitration	M	Yes [ ]
G2	Auto-Negotiation functions	98.2		M	Yes [ ]
G3	PHY support for High-Speed Mode	98.2.1, 98.5.6		ANSM: O.1	Yes [ ] No [ ] N/A [ ]
G4	PHY support for Low-Speed Mode	98.2.1, 98.5.2, 98.5.6		ANSM: O.1	Yes [ ] No [ ] N/A [ ]

98.6.5 DME transmission

Item	Feature	Subclause	Value/Comment	Status	Support
DME1	DME pages	98.2.1.1	Not be transmitted when Auto-Negotiation is complete and the highest common denominator PHY has been enabled.	M	Yes [ ]
DME2	DME page encoding levels	98.2.1.1.1	Transmit either a +1 or –1 level	M	Yes [ ]
DME3	DME Page 64 odd-numbered transmission positions	98.2.1.1.1	Contain a transition	M	Yes [ ]

Item	Feature	Subclause	Value/Comment	Status	Support
DME4	DME Page 64 even-numbered transmission positions	98.2.1.1.1	Represent data	M	Yes [ ]
DME5	DME Page first 48 even-numbered positions	98.2.1.1.1	Carry the data of the Auto-Negotiation page	M	Yes [ ]
DME6	CRC16	98.2.1.1.1	Comply with Figure 98–3 and initialize 16 delay elements to zero	M	Yes [ ]
DME7	Timing parameters for DME pages	98.2.1.1.2	Table 98–1	M	Yes [ ]
DME8	DME page transitions in high-speed mode	98.2.1.1.2	Occur within ± 0.8 ns of their ideal position	HSM: M	Yes [ ] N/A [ ]
DME9	DME page transitions in low-speed mode	98.2.1.1.2	Occur within ± 10 ns of their ideal position	LSM: M	Yes [ ] N/A [ ]
DME10	Transmit differential signal at the MDI	98.2.1.1.4	Within range of 1 V ± 30% peak-to-peak when measured in reference to a 100 Ω termination	DIFF: M	Yes [ ]
DME11	Transmit signal at the MDI	98.2.1.1.4	Within range of 0.5 V ± 30% peak-to-peak when measured in reference to a 50 Ω termination	UNB: M	Yes [ ]

98.6.6 Link codeword encoding

Item	Feature	Subclause	Value/Comment	Status	Support
DT1	Base Page transmitted within a DME page	98.2.1.2	Convey the encoding shown in Table 98–2	M	Yes [ ]
DT2	Encoding for link codeword(s) used inside a Next Page exchange	98.2.1.2	In a DME page, D0 is the first bit transmitted	M	Yes [ ]
DT3	RF, ACK, NP bits	98.2.1.2	Function as specified in 98.2.1.2.7, 98.2.1.2.8, and 98.2.1.2.9, respectively	M	Yes [ ]
DT4	Reserved combinations of the Selector Field	98.2.1.2.1	Not be transmitted	M	Yes [ ]
DT5	Echoes Nonce Field bad	98.2.1.2.2	Contain logical zeros when the device does not receive a DME page with good CRC16	M	Yes [ ]
DT6	Echoed Nonce Field good	98.2.1.2.2	Contain the value received in the Transmitted Nonce Field from the link partner when the device has received a DME page with good CRC16	M	Yes [ ]
DT7	Generate a new Transmitted Nonce Field value for each entry to the Ability Detect state	98.2.1.2.3		M	Yes [ ]
DT8	Matching T[4:0]	98.2.1.2.3	Invert T[0] bit and regenerate a new random value for T[3:1]	M	Yes [ ]

Item	Feature	Subclause	Value/Comment	Status	Support
DT9	Support the data service ability for a technology it advertises	98.2.1.2.4		M	Yes [ ]
DT10	Acknowledge bit with no Next Page	98.2.1.2.8	Set to 1 after the reception of at least one DME page with correct CRC	M	Yes [ ]
DT11	Acknowledge bit with Next Page	98.2.1.2.8	Set to 1 after the device successfully receives reception of at least one DME page with correct CRC and remain set until Next Page information has been loaded	M	Yes [ ]
DT12	No Next Page to send	98.2.1.2.9	Set Next Page to 0	M	Yes [ ]
DT13	Next Page to send	98.2.1.2.9	Set Next Page to 1	M	Yes [ ]
DT14	No Next Page to send and link partner Next Page is set to 1	98.2.1.2.9	Transmit Next Pages with Null message codes and the Next Page set to 0	M	Yes [ ]
DT15	Transmit Switch function	98.2.1.3	Enable the transmit path from a single technology-dependent PHY to the MDI once Auto-Negotiation has completed	M	Yes [ ]
DT16	Transmit Switch function during Auto-Negotiation	98.2.1.3	Connect only the DME page generator controlled by the Transmit state diagram to the MDI	M	Yes [ ]
DT17	Signal at the MDI	98.2.1.3	Conform to all the PHY's specifications when a PHY is connected to the MDI through the Transmit Switch function	M	Yes [ ]

98.6.7 Arbitration function requirements

Item	Feature	Subclause	Value/Comment	Status	Support
AF1	Regeneration request	98.2.4.1	Cause Arbitration function to disable all technology-dependent PHYs and halt any transmit data and link transition activity until the break_link_timer expires	M	Yes [ ]
AF2	break_link_timer expires	98.2.4.1	Resume Auto-Negotiation	M	Yes [ ]
AF3	Priority Resolution function	98.2.4.2	Highest priority is assigned to the single PHY enabled to connect to the MDI by Auto-Negotiation	M	Yes [ ]
AF4	Receipt of a Technology Ability Field with a bit that is reserved	98.2.4.2	Ignore the bit for priority resolution	M	Yes [ ]

Item	Feature	Subclause	Value/Comment	Status	Support
AF5	No common technology	98.2.4.2	Value "NULL" is assigned to HCD	M	Yes [ ]
AF6	Next Pages message encodings	98.2.4.3	Have a Message code that defines how the Unformatted code will be interpreted for each series of Next Pages	M	Yes [ ]
AF7	Completed transmission of Next Page information	98.2.4.3	Transmit Next Pages with Null message codes and the NP bit set to 0	M	Yes [ ]
AF8	Receipt of Message Pages with Null message codes	98.2.4.3	End of its link partner's Next Page information	M	Yes [ ]
AF9	Next Page encoding	98.2.4.3.1	Shown in Table 98–5 and Table 98–6	M	Yes [ ]
AF10	NP, Ack, MP, Ack2 and T bits	98.2.4.3.1	Function as specified in 98.2.1.2.9, 98.2.1.2.8, 28.2.3.4.5, 28.2.3.4.6, and 28.2.3.4.7, respectively	M	Yes [ ]
AF11	Message Next Pages	98.2.4.3.1	MP bit set to 1, D[10:0] encoded as a Message Code Field, and D[47:16] encoded as Unformatted Code Field	M	Yes [ ]
AF12	Unformatted Next Pages	98.2.4.3.1	MP bit set to 0, D[10:0] and D[47:16] encoded as the Unformatted Code Field	M	Yes [ ]
AF13	Next Page exchange	98.2.4.3.1	Continue until neither device on a link has more pages to transmit	M	Yes [ ]
AF14	Next Page with Null Message Code Field value	98.2.4.3.1	Sent if the device has no other information to transmit	M	Yes [ ]

98.6.8 Service primitives

Item	Feature	Subclause	Value/Comment	Status	Support
TDI1	link_status parameter	98.4.1.1	OK, FAIL	MDIO:M	Yes [ ]
TDI2	Receipt of link_status primitive	98.4.1.3	Comply with Figure 98–7	M	Yes [ ]
TDI3	link_control parameter	98.4.2.1	DISABLE, ENABLE	M	Yes [ ]
TDI4	link_control=DISABLE	98.4.2.1	Used by Auto-Negotiation to disable PMA processing	M	Yes [ ]
TDI5	link_control=ENABLE	98.4.2.1	Used by Auto-Negotiation to turn control over to a single PMA for all normal processing functions	M	Yes [ ]

Item	Feature	Subclause	Value/Comment	Status	Support
TDI6	Generation of link_control primitive	98.4.2.2	Generated by Auto-Negotiation function	M	Yes [ ]
TDI7	Auto-Negotiation enabled upon power-on or reset	98.4.2.2	Issue PMA_LINK.request(DISABLE) message to all technology-dependent PMAs	M	Yes [ ]

98.6.9 State diagram and variable definitions

Item	Feature	Subclause	Value/Comment	Status	Support
SD1	Auto-Negotiation	98.5.5	Implement Figure 98–7, Figure 98–8, and Figure 98–9	M	Yes [ ]
SD2	State diagrams	98.5.5	State diagrams take precedence when ambiguity exists between state requirements and the state diagram	M	Yes [ ]
SD3	backoff_timer_[HSM]	98.5.2	Expire according to the formula described in 98.5.2	HSM:M	Yes [ ] N/A [ ]
SD4	backoff_timer_[LSM]	98.5.2	Expire according to the formula described in 98.5.2	LSM:M	Yes [ ] N/A [ ]
SD5	blind_timer_[HSM]	98.5.2	Expire 2000 ns to 2120 ns after being started	HSM:M	Yes [ ] N/A [ ]
SD6	blind_timer_[LSM]	98.5.2	Expire 28 200 ns to 31 400 ns after being started	LSM:M	Yes [ ] N/A [ ]
SD7	break_link_timer_[HSM]	98.5.2	Expire 300 µs to 305 µs after being started	HSM:M	Yes [ ] N/A [ ]
SD8	break_link_timer_[LSM]	98.5.2	Expires 8000 µs to 8133 µs after being started	LSM:M	Yes [ ] N/A [ ]
SD9	clock_detect_max_timer_[HSM]	98.5.2	Expire 63 ns to 75 ns after being started or restarted	HSM:M	Yes [ ] N/A [ ]
SD10	clock_detect_max_timer_[LSM]	98.5.2	Expires 1680 ns to 2000 ns after being started or restarted	LSM:M	Yes [ ] N/A [ ]
SD11	clock_detect_min_timer_[HSM]	98.5.2	Expire 45 ns to 57 ns after being started or restarted	HSM:M	Yes [ ] N/A [ ]
SD12	clock_detect_min_timer_[LSM]	98.5.2	Expires 1200 ns to 1520 ns after being started or restarted	LSM:M	Yes [ ] N/A [ ]
SD13	data_detect_max_timer_[HSM]	98.5.2	Expire 33 ns to 45 ns from the last clock transition	HSM:M	Yes [ ] N/A [ ]
SD14	data_detect_max_timer_[LSM]	98.5.2	Expires 880 ns to 1200 ns from the last clock transition	LSM:M	Yes [ ] N/A [ ]
SD15	data_detect_min_timer_[HSM]	98.5.2	Expire 15 ns to 27 ns from the last clock transition	HSM:M	Yes [ ] N/A [ ]
SD16	data_detect_min_timer_[LSM]	98.5.2	Expires 400 ns to 720 ns from the last clock transition	LSM:M	Yes [ ] N/A [ ]

Item	Feature	Subclause	Value/Comment	Status	Support
SD17	interval_timer_[HSM]	98.5.2	Expire 30 ns ± 0.01% from each clock pulse and data bit	HSM:M	Yes [ ] N/A [ ]
SD18	interval_timer_[LSM]	98.5.2	Expires 800 ns ± 0.005% from each clock pulse and data bit	LSM:M	Yes [ ] N/A [ ]
SD19	link_fail_inhibit_timer_[HCD]	98.5.2	Expire 97 ms to 98 ms after entering the AN GOOD CHECK state	!10T1L: M	Yes [ ] N/A [ ]
SD20	link_fail_inhibit_timer_[HCD] for 10BASE-T1L PHY	98.5.2	Expires 3030 ms to 3090 ms after entering the AN LINK GOOD CHECK state	10T1L: M	Yes [ ] N/A [ ]
SD21	link_fail_inhibit_timer_[HCD] for 10BASE-T1S PHY	98.5.2	Expires 400 ms to 405 ms after entering the AN LINK GOOD CHECK state	10T1S: M	Yes [ ] N/A [ ]
SD22	page_test_max_timer_[HSM]	98.5.2	Expire 4800 ns to 4920 ns after being started or restarted	HSM:M	Yes [ ] N/A [ ]
SD23	page_test_max_timer_[LSM]	98.5.2	Expires 128 000 ns to 131 200 ns after being started or restarted	LSM:M	Yes [ ] N/A [ ]
SD24	receive_DME_timer_[HSM]	98.5.2	Expire 6805 ns to 6925 ns after being started	HSM:M	Yes [ ] N/A [ ]
SD25	receive_DME_timer_[LSM]	98.5.2	Expires 156 300 ns to 159 500 ns after being started	LSM:M	Yes [ ] N/A [ ]
SD26	rx_wait_timer_[HSM]	98.5.2	Expire 15 µs to 17 µs after being started or restarted	HSM:M	Yes [ ] N/A [ ]
SD27	rx_wait_timer_[LSM]	98.5.2	Expires 330 µs to 370 µs after being started or restarted	LSM:M	Yes [ ] N/A [ ]
SD28	silent_timer_[HSM]	98.5.2	Expire 2120 ns to 2240 ns after being started	HSM:M	Yes [ ] N/A [ ]
SD29	silent_timer_[LSM]	98.5.2	Expires 31 400 ns to 34 600 ns after being started	LSM:M	Yes [ ] N/A [ ]

98.6.10 High-speed and low-speed Auto-Negotiation modes

Item	Feature	Subclause	Value/Comment	Status	Support
SM1	Supports two Auto-Negotiation speeds	98.5.6	Implements the state diagram in Figure 98–11	ANSM: M	Yes [ ] N/A [ ]
SM2	Supports only high-speed mode	98.5.6	Implements Figure 98–7, Figure 98–8, Figure 98–9, and Figure 98–10 using the timer values for high-speed mode	!LSM:M	Yes [ ] N/A [ ]
SM3	Supports only low-speed mode	98.5.6	Implements Figure 98–7, Figure 98–8, Figure 98–9, and Figure 98–10 using the timer values for low-speed mode	!HSM:M	Yes [ ] N/A [ ]