AUTO-NEGOTIATION

OVERVIEW

21 MADBURY RD, SUITE 100
DURHAM, NH 03824
+1-603-862-0090
WWW.IOL.UNH.EDU

WRITTEN BY SPENCER JANELLE

TABLE OF CONTENTS

GENERAL OVERVIEW

Auto-Negotiation, also referred to as AutoNeg or ANEG, is a process defined by the IEEE 802.3 Working Group. Ethernet devices use it to resolve the highest possible link between two devices. A simple, non-technical way to envision this process is to determine what language is best to speak with another person. For example, the rate at which everyone can speak a language can be ranked as follows:

1. Spanish
2. French
3. German
4. English

Adding onto this, two people who are trying to have a conversation know the following languages:

Person #1

- English
- French
- Spanish

Person #2

- Spanish
- English
- German

Both people can speak English and Spanish, however, Spanish is the fastest language to speak. Auto-Negotiation is the process of determining that Spanish is the fastest language to have a conversation in, or in networking terms, the highest operating speed.

On the OSI model, Auto-Negotiation is located between the Physical and Data Link layers, as shown in Figure 1 [2]. The process is responsible for determining the link speed between two link partners. Each link partner transmits a “link pulse”, which can be combined to form a “link codeword” [1]. This link codeword is used to advertise which technologies are supported by the device. Each clause in the standard defines how technologies are mapped to the link codeword. [1].

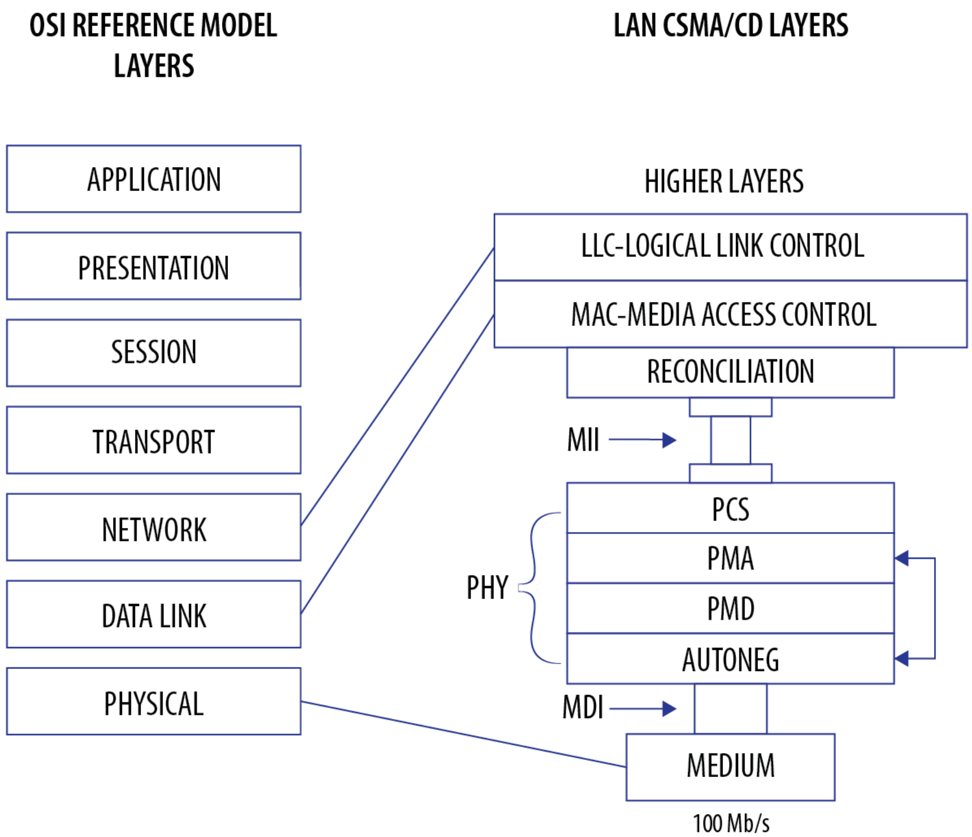


Figure 1. Location of Auto-Negotiation on the OSI Model [2]

Each link partner has their own link codeword to transmit, which are exchanged. The link partners acknowledge the other's link codeword, process what abilities are supported, and transmit the appropriate link signaling.

DETAILED DISCUSSION

APPLICATIONS

In the IEEE 802.3 standard, Auto-Negotiation is present in several clauses and technologies. Of these several technologies, Auto-Negotiation is defined in the following clauses:

- Twisted Pair Auto-Negotiation (IEEE 802.3 Clause 28) [1]
- Fiber Auto-Negotiation (IEEE 802.3 Clause 37) [4]
- Backplane Auto-Negotiation (IEEE 802.3 Clause 73) [3]

- Single Pair Ethernet (IEEE 802.3 Clause 98) [5]

In each of these technologies, a link codeword is defined and characterized differently. In the case of Backplane and Single Pair Ethernet, Differential Manchester Encoding (DME) [3, 5] characterizes the link codewords. This encoding contains 48 data bits that can be used for negotiating an operating mode. For Backplane Ethernet, the possible advertised abilities include 1000BASE-KX, 10GBASE-KX4, 10GBASE-KR, 40GBASE-KR4, 40GBASE-CR4, 100GBASE-CR10, 100G-BASE-KP4, 100GBASE-KR4, 100GBASE-CR4, 25GBASE-KR-S, 25GBASE-CR-S, 25GBASE-KR4, and 25GBASE-CR4 [3]. For Single Pair Ethernet, the technology is new to industry, so only duplex is advertised through DMEs at this moment [5].

Fiber Auto-Negotiation uses “/C/” ordered sets to act as a link codeword [4]. These ordered sets contain 16 bits, which advertise what duplexes the link partner supports, PAUSE mode capabilities for accepting and sending traffic, as well as Remote Fault, a way of signifying that there has been an internal error in the device [4].

Twisted Pair Auto-Negotiation uses Base Pages to act as a link codeword [1]. These Base Pages contain 16 bits, and advertise 10BASE-T half/full duplex, 100BASE-TX half/full duplex, PAUSE mode capabilities, and Remote Fault. In addition to this Base Page functionality, extended advertisement registers can be used to advertise additional speeds and functionality [1]. This functionality is referred to as Next or extended Next Page functionality, which adds to the Auto-Negotiation process. Operating advertisements in these extended registers include 1000BASE-T half/full duplex, 2.5GBASE-T, 5GBASE-T, 10GBASE-T, Energy-Efficient Ethernet, MASTER/SLAVE configuration settings, and Loop Timing configurations [1].

With regards to outlining the Auto-Negotiation process, a focus will be made on Twisted Pair Auto-Negotiation.

Process In Detail

For Twisted Pair Ethernet, the Auto-Negotiation process is fully defined by the IEEE in the 802.3 standard, specifically in Clause 28. In IEEE 802.3 Clause 28, there is a state diagram (Figure 2) that depicts the flow of the Auto-Negotiation process from the moment a device is powered on [6]. These states are defined below:

AUTO-NEGOTIATION ENABLE

When a device powers on, it enters the AUTO-NEGOTIATION ENABLE state. In this state, the device does not transmit anything, and does not accept any transmissions [1]. The device leaves the AUTO-NEGOTIATION ENABLE state when Auto-Negotiation is internally enabled properly and continues into the TRANSMIT DISABLE state [1, 6].

TRANSMIT DISABLE

The TRANSMIT DISABLE state is used as a standardized amount of time a device will disable it's transmitter [1]. It is used when an advertisement is changed via device management, when a device is first powered on, or if there is an error in the Auto-Negotiation process. Once this standardized amount of transmitter silence, also known as break_link_timer is complete, the device will transition into the ABILITY DETECT state [1, 6].

ABILITY DETECT

When a device is not connected to a link partner, it remains in the ABILITY DETECT state, essentially the idle state [1]. From the ABILITY DETECT state, a device can encounter two different scenarios. Either the device receives link signaling at a supported speed from a link partner, which will transition the device into the LINK STATUS CHECK state or it receives link codewords, which will transition the device into the ACKNOWLEDGE DETECT state [1, 6].

LINK STATUS CHECK

Once a device enters the LINK STATUS CHECK state, the highest supported speed has already been determined, because the link partner of the device only supports one speed, as indicated by the sending of link signaling [1]. The LINK STATUS CHECK state verifies that the received link signaling supported speed by the device, as well as that the connection of the two link partners is secure [1]. This occurs when the device transmits link signaling at the same speed as the link partner. If the link partner stops transmitting link signaling, the device will transition to the PARALLEL DETECTION FAULT state [1, 6]. If the link partner continues to send link signaling, it will transition to the FLP LINK GOOD CHECK state [6].

PARALLEL DETECTION FAULT

The purpose of the PARALLEL DETECTION FAULT state is to signify that an error has occurred when the two link partners attempted to link with one link partner only transmitting link signaling [1]. Internal registers signify that a fault occurred so that management understands what happened [1]. The device will then transition back into the ABILITY DETECT state [6].

ACKNOWLEDGE DETECT

When a device receives link codewords, it must then acknowledge that it has received them to the link partner [1]. The device will change the transmitted Base Page to one with the ACKNOWLEDGE BIT set [1]. This bit is only ever set when a device receives link codewords from a link partner. While transmitting a Base Page with the ACKNOWLEDGE BIT set, the device waits to receive link codewords that have the ACKNOWLEDGE BIT set, which will transition the device to the COMPLETE ACKNOWLEDGE state [1, 6]. If the ACKNOWLEDGE BIT set receives no link codewords, the device will enter the TRANSMIT DISABLE state [1].

COMPLETE ACKNOWLEDGE

Once the device receives acknowledgement from the link partner, it then attempts to resolve to the highest supported speed if it does not support Next Page or Extended Next Page functionality. If the device does not support Next Page or Extended Next Page functionality, the device will begin to transmit link signaling at the highest common speed and transition into the FLP LINK GOOD CHECK state [6]. If both link partners support Next Page or Extended Next Page functionality, the device will transmit the first Next Page or Extended Next Page in it's sequence and enter the NEXT PAGE WAIT state [1, 6].

NEXT PAGE WAIT

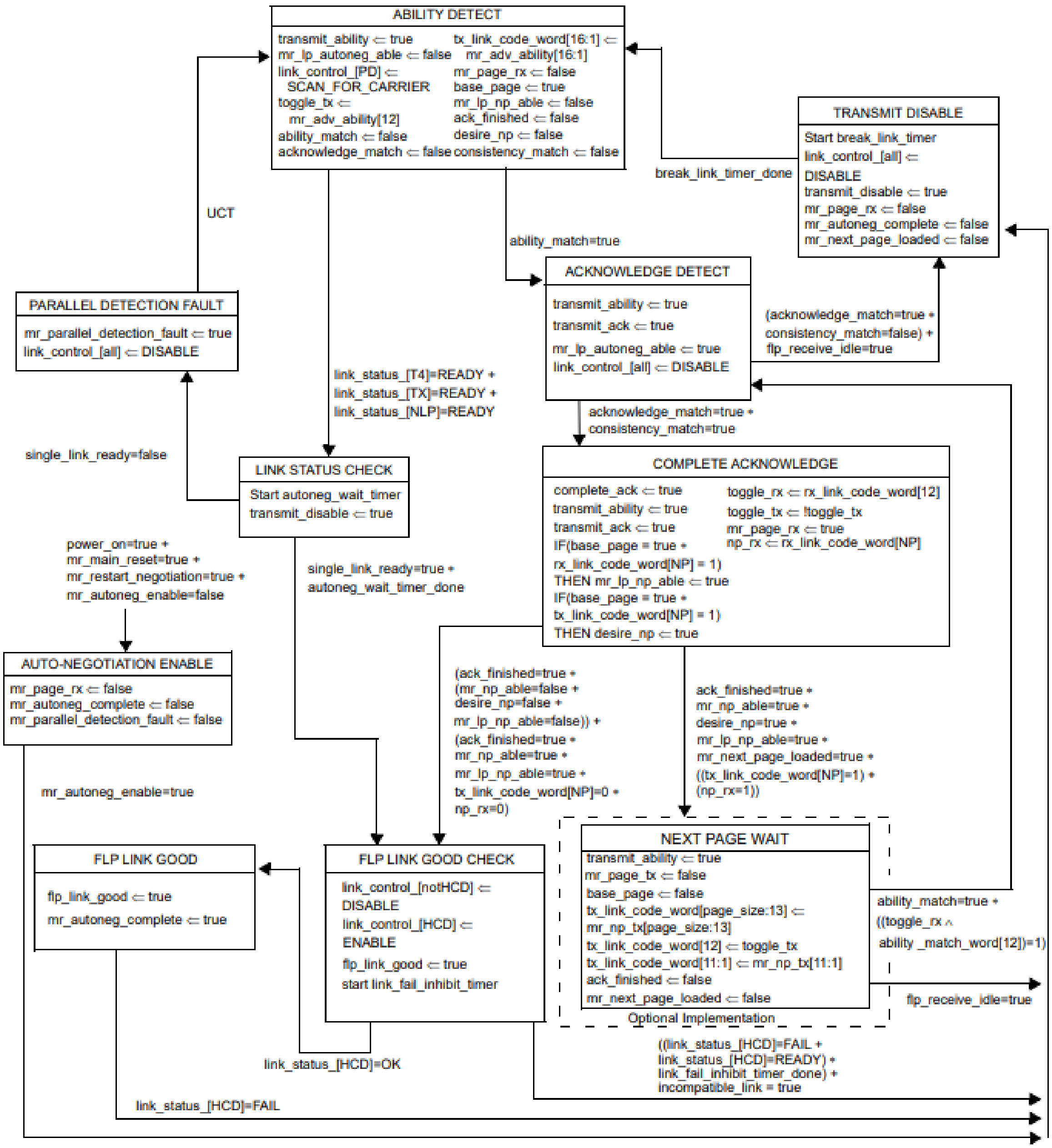
In the NEXT PAGE WAIT state, the device has already transmitted a Next Page or Extended Next Page. Now, the device waits until the link partner transmits a Next Page or Extended Next Page. If the link partner does not, the device transitions into the TRANSMIT DISABLE state [1, 6]. If the device receives a Next Page or Extended Next Page, it then transitions into the ACKNOWLEDGE DETECT state [6]. This cycle continues until both the device and link partner no longer have anyNext Pages or Extended Next Pages to send. Once in the COMPLETE ACKNOWLEDGE state, the device will transmit the highest common speed if there is one advertised in the Next Page or Extended Next Page sequence, or a speed that is commonly advertised in the Base Page of both link partners [1, 6].

FLP LINK GOOD CHECK

FLP LINK GOOD CHECK is the state that determines whether or not Auto-Negotiation completed successfully. At this point, the both link partners will transmit link signaling that has been determined by both devices as the highest common speed [1]. The device will enter the TRANSMIT DISABLE state if there is a link signaling mismatch between the two link partners, if the link signaling provided by the link partner is not the highest common speed determined by the device, or if the link partner no longer provides any link signaling while in FLP LINK GOOD CHECK. If all these conditions remain valid the device will transition into the FLP LINK GOOD state [1, 6].

FLP LINK GOOD

The two link partners have finally negotiated a complete link. The connection can now transition up to the desired functionality between the two link partners. The only transition possible out of the FLP LINK GOOD state is to the TRANSIT DISABLE state, which happens if the connection between the two link partners fails or becomes disconnected [6].



NOTE—The transition from COMPLETE ACKNOWLEDGE to FLP LINK GOOD CHECK can be simplified to "ack_finished=true" if the optional Next Page function is not supported.

NOTE—ability_match, acknowledge_match, single_link_ready, consistency_match, incompatible_link, and page_size are set according to the variable definitions and are not set explicitly in the state diagrams.

Figure 2. Auto-Negotiation Arbitration State Diagram [6]

Purpose

The goal of Auto-Negotiation is to standardize the way that devices can create a link between one another. This eliminates the need to manually configure both link partners to connect with each other. This alone accelerates the bringup time of a system dramatically. The automation of this process also supports interoperability for a massive range of devices. From a speed configuration standpoint, anyone who internalizes a system that implements Auto Negotiation, does not need to understand multiple user interfaces. Additionally, error handling and checking is built into the process itself, which lowers the amount of time required to debug and restore a system from a link level perspective.

Future of Auto-Negotiation

Auto-Negotiation will continue to be a required process whenever any networking media supports multiple speeds, duplexes or operating modes. Therefore, the process will simply need to evolve as new technologies are developed.

As of March 9, 2022, the newest standardized version of Auto-Negotiation has been drafted by the IEEE, 802.3cg [7]. This amendment adds in further support for Clause 98 Auto-Negotiation by differentiating two different modes. These modes are High-Speed Mode and Low-Speed Mode [7]. This will make it so two cable lengths can be used for Single Pair Ethernet to support both automotive and manufacturing implementations.

Works Cited

[1] "IEEE Standard for Ethernet," in IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015) , vol., no., SECTION 2, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[2] "IEEE Standard for Ethernet," in IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015) , vol., no., Figure 28-2-Location of Auto-Negotiation function within the ISO/IEC OSI reference model, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[3] "IEEE Standard for Ethernet," in IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015) , vol., no., SECTION 5, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[4] "IEEE Standard for Ethernet," in IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015) , vol., no., SECTION 3, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[5] "IEEE Standard for Ethernet," in IEEE Std 802.3-2018 (Revision of IEEE Std

802.3-2015) , vol., no., SECTION 7, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[6] "IEEE Standard for Ethernet," in IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015) , vol., no., Figure 28-18-Arbitration State Diagram, 31 Aug. 2018, doi: 10.1109/IEEESTD.2018.8457469.

[7] "IEEE Standard for Ethernet - Amendment 5: Physical Layer Specifications and Management Parameters for 10 Mb/s Operation and Associated Power Delivery over a Single Balanced Pair of Conductors," in IEEE Std 802.3cg-2019 (Amendment to IEEE Std 802.3-2018 as amended by IEEE Std 802.3cb-2018, IEEE Std 802.3bt-2018, IEEE Std 802.3cd-2018, and IEEE Std 802.3cn-2019) , vol., no., pp.1-256, 5 Feb. 2020, doi: 10.1109/IEEESTD.2020.8982251.

About the UNH - IOL

The UNH-IOL is dedicated to being the world's premier data networking resource. Beginning in 1988, the UNH-IOL has built a multimillion dollar testbed and developed testing solutions to help companies efficiently and cost effectively deliver products to market. UNH-IOL's services fit the needs of the market, while maintaining broad, comprehensive interoperability and conformance testing of networking, data, telecommunications, and storage products. The industry relies on UNH-IOL's extensive staff experience, standards-bodies participation, 28,000+ square foot facility, and 100+ students who represent the future of interoperability testing and verification.