



# IEEE 802.3az Energy Efficient Ethernet: Build Greener Networks

## Energy Efficient Ethernet

Ethernet is the most ubiquitous networking interface in the world; virtually all network traffic passes over multiple Ethernet links. However, the majority of Ethernet links spend much of the time idle, waiting between packets of data, but consuming power at a near constant level. It is estimated that network devices and network interfaces comprise more than 10% of total IT power usage: tens of TWhr per year<sup>1</sup>. Energy Efficient Ethernet (EEE) provides a mechanism and a standard for reducing this energy usage without reducing the vital function that these network interfaces perform.

## Executive Summary

This paper seeks to answer three questions about Energy Efficient Ethernet (EEE):

- What are the benefits of EEE? What is the technology and how does it work?
  - How do people use EEE and how can it be deployed in a network?

Cisco<sup>®</sup>, the worldwide leader in networking technologies; and Intel, the world's largest manufacturer of semiconductor chips and processors and the leading supplier of Gigabit Ethernet controllers and Gigabit and 10 Gigabit adapters, contributed towards the definition of this standard, and collaboratively developed and tested this technology for interoperability to help ensure easy adoption in the enterprise market. Through this partnership, Cisco is introducing IEEE 802.3az, a new standard for EEE on the Cisco Catalyst<sup>®</sup> 4500E Switches, the most widely deployed modular access platform in the industry. Intel supports EEE across its Gigabit Ethernet product lines for both client and server platforms.

## Background

As electronics and information technology have become ubiquitous in our lives there has been a growing focus on the energy usage of these devices. It is unimportant whether this stems from concern over greenhouse gas emissions contributing to climate change; energy shortages due to dwindling oil reserves; or a need to reduce the burden on expensive and fragile energy infrastructures. The programs to reduce IT energy consumption have initially concentrated on the areas of highest energy usage: computers and consumer devices. However, networking equipment has been identified as consuming as much as 10% of all IT energy, so it is logical to consider how networking energy consumption can be reduced without adversely affecting the critical functionality that networking performs.

As part of this movement towards networking energy efficiency, the IEEE entertained a "Call for Interest"<sup>2</sup> in November 2006 that led to the formation of the Energy Efficient Ethernet project (IEEE 802.3az). Over the course of four years, the project task force considered many proposals for changes to the Ethernet standard that would allow efficiency improvements. It was agreed that the project would deal with the mainstream "BASE-T" interfaces (i.e. 10BASE-T; 100BASE-TX; 1000BASE-T; and 10GBASE-T) that operate over twisted pair wiring. These interface types comprise the vast majority of Ethernet deployments, especially at the edge of networks where the opportunities for energy savings are maximal.

---

<sup>1</sup> Roth, Goldstein & Kleinman, 2001, **Energy Consumption by Office and Telecommunications Equipment in Commercial Buildings**, Lanzisera, Nordman, Brown, 2010, **Data Network Equipment Energy Use and Savings Potential in Buildings**. Kawamoto, Koomey, Nordman, Brown, Piette, Ting, Meier, 2002. **Electricity Used by Office Equipment and Network Equipment in the U.S.**

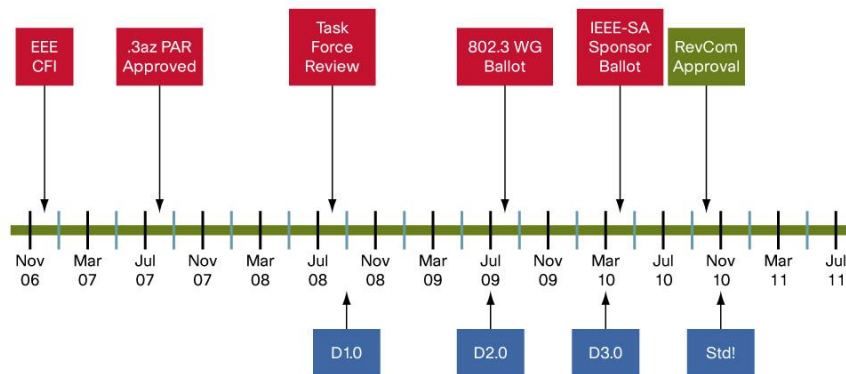
<sup>2</sup> [http://www.ieee802.org/3/cfi/1106\\_1/EEE-CFI.pdf](http://www.ieee802.org/3/cfi/1106_1/EEE-CFI.pdf)

The standard also covers Backplane Ethernet interfaces used in blade servers (as well as within proprietary systems) because the amount of change required for those interfaces was considered minor.

During the standards development process, a lot of attention was given to backwards compatibility. The new standard must be deployable in networks where the majority of equipment uses legacy interfaces and must also seamlessly support the very wide range of applications that already run on these networks. It was accepted that interfaces complying with the new standard might not save energy when connecting with older devices as long as the existing functions are fully supported. This allows incremental upgrades for networks to increasingly benefit from EEE as the proportion of EEE equipment increases.

The standard also recognizes that some network applications may allow larger amounts of traffic disturbance and includes a negotiation mechanism to take advantage of such environments to increase the depth of energy savings. After many rounds of review to help ensure that compatibility and robustness were not compromised, the standard was finally published in November 2010 and is now available from the IEEE 802 website. Figure 1 below describes the four-year process in detail.

**Figure 1:** Timeline - the four-year process



## Generations of Development and Expectations

EEE represents the beginning of a change of opinion in networking architecture. Previously it had been acceptable that networking devices, like the communications on the links themselves, continue to use energy at the same rate, regardless of the level of usage. The standard for EEE defines the signaling necessary for energy savings during periods where no data is sent on the interface, but does not define how the energy is saved, nor mandate a level of savings. This approach allows for a staged rollout of systems with minimal changes that are compatible with future developments that extend the energy savings.

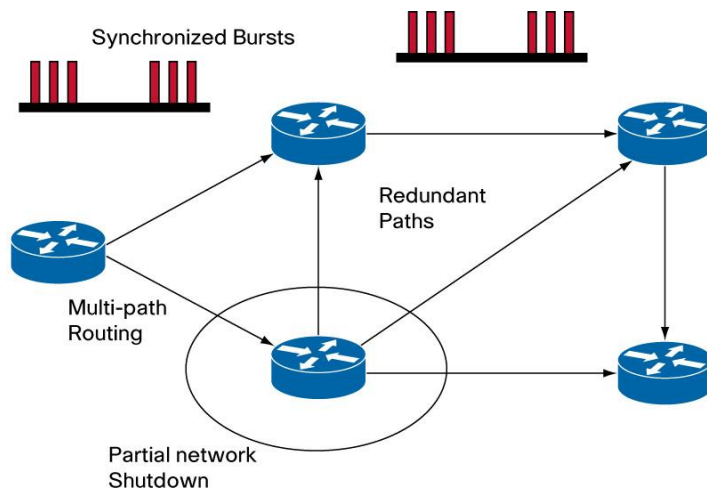
It should be expected that early implementations of the standard save relatively small amounts of energy (comparing idle energy to full rate usage). However, these systems will be compatible with later products that may save much greater proportions of their energy use. The early systems may use a simple application of static logic design in the physical layer devices (PHYs) to save energy when data is not present. PHYs typically consume between 20 to 40 percent of the system power, and the static design methods allow savings of up to 50 percent of the PHY power. Therefore the expected system-level savings may be in the range of five to 20 percent.

Later generations of networking systems will use more aggressive energy savings techniques, such as power islands or voltage scaling. These methods can be applied to all of the system silicon, extending the range of energy savings. However, such aggressive techniques require significant new architecture design and will necessarily follow much later. With these power savings, an individual networking system may be able to save as much as 80 percent of its worst case energy use in certain situations.

Devices at the edge of the network may be able to use other techniques in order to minimize their energy. In particular, edge devices may be able to enter a deep sleep state while maintaining a network link to help ensure security and to wake in response to a network request. In such a deep sleep, the edge device may require a longer wake time, which can be negotiated using the link layer protocol defined in the standard. It is important that edge-facing networking systems support the wake time negotiation and can support extended wake times to enable future developments in edge device design that massively reduce the energy footprint of the networked devices.

Finally, there will be developments in network architecture that can utilize energy-efficient control plane solutions to help ensure that multiple interconnected EEE networking systems can operate in a way that minimizes the total energy use. The co-ordination of control policies will prevent situations where individual devices aiming to optimize their own energy usage result in an overall network that is sub-optimal. This area is currently under intense study, so it will take some time before the network-level solutions can be realized, although, once more, it is important that early systems have the management and negotiation abilities that will enable them to participate. Figure 2 below shows a potential control plane driven energy efficient solution.

**Figure 2:** Network Level Conceptions

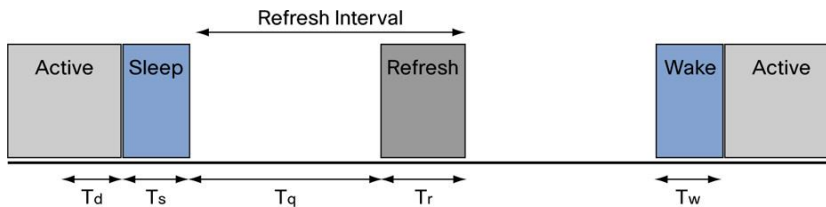


## The Core of EEE - Low Power Idle

The fundamental idea of EEE is that the communication link should need to consume power only when real data is being sent. Most wireline communications protocols developed since the 1990s have used continuous transmission - consuming power whether or not data was being sent. The reasoning behind this was that the link must be maintained with full bandwidth signaling so that it is ready to support data transmission at all times. In order to save energy during times where there is a gap in the data stream, EEE uses a signaling protocol that allows a transmitter to indicate that there is a gap in the data and that the link can go idle. The signaling protocol is also used to indicate that the link needs to resume after a pre-defined delay.

The EEE protocol uses a signal that is a modification of the normal idle that is transmitted between data packets. This signal is termed low power idle (LPI). The transmitter sends LPI in place of idle to indicate that the link can go to sleep. After sending LPI for a period ( $T_s$  = time to sleep), the transmitter can stop signaling altogether so that the link becomes quiescent. Periodically, the transmitter sends some signals so that the link does not remain quiescent for too long without a refresh. Finally, when the transmitter wishes to resume the fully functional link, it sends normal idle signals. After a pre-determined time ( $T_w$  = time to wake) the link is active and data can be sent. Figure 3 below describes the different EEE states pictorially

**Figure 3:** LPI diagram



The EEE protocol allows the link to be re-awakened at any time; there is no minimum or maximum sleep interval. This allows EEE to function effectively in the presence of unpredictable traffic. The default wake time is defined for each type of PHY and is generally aimed to be similar to the time taken to transmit a maximum length packet at the particular link speed. For example, the wake time for 1000BASE-T is 16.5 $\mu$ S - roughly the same time that it takes to transmit a 2000 byte Ethernet frame.

## Maintaining the Link

The refresh signal that is sent periodically while the link is idle is important for multiple reasons. First, it serves the same purpose as the link pulse in traditional Ethernet. The heartbeat of the refresh signal helps ensure that both partners know that the link is present and allows for immediate notification following a disconnection. The frequency of the refresh, which is typically greater than 100Hz, prevents any situation where one link partner can be disconnected and another inserted without causing a link fail event. This maintains compatibility with security mechanisms that rely on continuous connectivity and require notification when a link is broken.

The maintenance of the link through refresh signals also allows higher layer applications to understand that the link is continuously present so that network stability is preserved. Changing the power level must not cause connectivity interruptions that would result in link flap, network reconfiguration, or client association changes.

Second, the refresh signal can be used to test the channel and create an opportunity for the receiver to adapt to changes in the channel characteristics. For high speed links, this is vital to support the rapid transition back to the full speed data transfer without sacrificing data integrity. The specific makeup of the refresh signal is designed for each PHY type to assist the adaptation for the medium supported.

## Low Power Idle, and Not Speed Changes

Early proposals in the 802.3az task force were focused on developing a faster link speed change protocol; faster than the existing auto-negotiation protocol, which can take several seconds to affect a speed change. This would have allowed products to save power by transitioning quickly from a high speed, higher power link, such as 1000BASE-T, or 10GBASE-T, to a low speed, lower power mode such as 10BASE-T, when the higher bandwidth was not needed. After considering more proposals, however, the task force agreed that the LPI mechanism described previously could achieve similar or better power saving benefits while avoiding disruption to the network and management tools that would have occurred with a link speed change.

Because of the early focus on speed changes to save power, it is still a common misperception that EEE uses speed changes to achieve power savings. This is not the case, however. When a 1000BASE-T link transitions to LPI mode, it is still a 1000BASE-T link.

## Deeper Sleep Negotiation

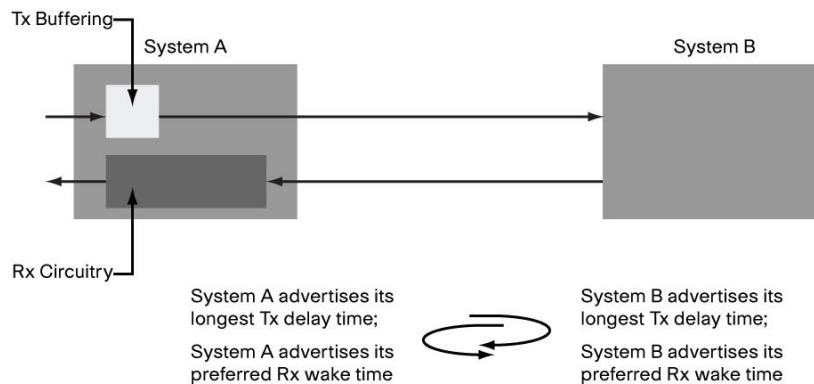
An additional feature of the EEE standard is the ability to re-negotiate the wake time. As mentioned earlier, the default wake time is defined for each type of PHY and is generally aimed to be approximately the same as the time taken to transmit a maximum length packet for the type of PHY in use. In some cases, equipment may be designed to go into a deeper sleep that requires a longer time to wake up and be ready for fully functional operation. Some examples of this include PC or server network interface cards (NICs) that can allow the entire system to go to sleep and be woken up by network activity; or systems with large memories that remove the power from their memory systems in order to save the most power.

To accommodate the deeper sleep requirement, it is necessary to negotiate. A receiver may have multiple levels of sleep that require different wake times. Similarly, a transmitter may have a limit to the depth of buffering that it can support to guarantee that it does not have to discard packets that arrive while it is waiting for the receiver to be ready. The negotiation takes place with a packet interchange using the link-layer discovery protocol (LLDP) defined in IEEE 802.1AB. This standard is already widely supported by networking equipment, so it does not place much extra burden on the system to add wake time negotiation.

The negotiation matches the wake time of the required sleep mode of the receiver with the maximum delay that the transmitter can tolerate in order to get the optimal energy savings for the required performance. This can be negotiated separately for each direction so that the wake times may be asymmetric. It can also be renegotiated at any time if conditions or control policies require it to change. Sophisticated networking systems may support re-negotiation based on policy driven from network energy management systems such as Cisco EnergyWise, as well as balancing the use of shared buffers between multiple ports to allow edge devices to conserve the most energy.

Figure 4 below outlines how Negotiation works.

**Figure 4:** Negotiation



## Deploying EEE

Each PHY that supports EEE advertises its capability through auto-negotiation when it is first connected to a link. If the link partner PHY does not advertise EEE capability, then the link operates in legacy mode and is identical in all ways to a link between two peers prior to the definition of EEE. This means that network administrators can start to roll out EEE-capable systems without fear that it will cause problems to an existing network. As more EEE-capable systems are added to the network - either as networking devices or as end-points - more and more links will start to benefit from the energy savings of EEE.

Early systems may only support the default wake time operation and unsophisticated EEE policies. This will limit the potential energy savings during the first phase of deployment. However, as these systems are upgraded and more advanced systems deployed, they can use the wake time negotiation just described to allow deeper sleep. They can also support multiple EEE policies that can match the sleep and wake behavior to the requirements of the key applications. The net result of this is that users can expect further improvements in efficiency and reductions in energy use as the EEE deployment in the network matures.

## Energy-Efficient Service Delivery




From its inception, EEE was defined so that higher layer services should run over EEE links without any noticeable degradation. The definition of EEE assumes that the link can enter the power-saving state while maintaining the link integrity and guaranteeing that the fully functional link is still available. For this reason, the higher layer functions do not need to take account of the state changes between data and LPI. Higher layer functions may use the management information available from the EEE function and may control the re-negotiation of wake times.

Most EEE interfaces default to a wake time that is similar to the delay of a maximum length packet at the target link speed. For this reason, the impact to applications is minimal as a similar delay is introduced by store and forward switching and by normal Ethernet packet operation. Applications that require controlled maximum latency in LAN environments (such as IP telephony or telepresence) can operate without difficulty over hundreds of EEE links using the default wake times. If this type of application is required, then the network operator should pay attention to the use of deeper sleep operation (i.e. longer wake time negotiation) in significant parts of the network.

Any application that was designed to operate across the Internet or a WAN can be assumed to be tolerant of very high latency (less than one millisecond). These applications can tolerate deep sleep negotiation for EEE without difficulty.

Some applications are extremely sensitive to latency. For example, high performance computing (HPC) may be very sensitive to the latency experienced by inter-processor communications or synchronization traffic. Similarly, some financial trading applications may require latency that is much lower than a maximum packet delay, requiring the use of cut-through switching. Network-wide energy control applications such as Cisco EnergyWise, can disable EEE functions while these sensitive applications are running and allow energy savings for the times when those applications are dormant. Also, in a highly optimized data center, it may be possible to discern some performance degradation on large file transfers according to the particular EEE profile in operation. Again, network-wide control applications can adjust the profile for peak and off-peak hours to mitigate this effect.

### Applications Over EEE

<b>Video delivery - typically over User Datagram Protocol (UDP)</b>	Wake times are much smaller than frame buffering	
<b>IP video telephony or telepresence</b>	Latency tolerance: approximately 1 - 10 ms built in (default EEE invisible)	
<b>Large data or file transfers</b>	May be visible for ultra optimal direct current (policy-dependent)	
<b>Any Internet application</b>	Built for high latency; allows deep sleep	
<b>HPC - interprocessor communication</b>	May be sensitive to microsecond delays	

## EEE at the Edge

The strongest justification for EEE came from the typical utilization of edge devices. Even during peak work hours, most client computers use their network connections with infrequent bursts. The normal EEE operation is well suited to this behavior. During off-peak times the client devices may use sleep or hibernate modes. At this time the network interface can be completely inactive, although more often it is expected to stay awake so that the client stays connected to network services and is able to be woken on demand by remote request.

The requirement that the client must stay connected to network services can restrict the ability of the device to go into a low power states. This problem can be solved by delegating the connectivity maintenance to a subsystem in the network interface or even to a remote device. This delegation is referred to as network proxy and is being defined.<sup>3</sup> When network proxy is used with EEE, the remote proxy device is able to help ensure that the client link remains connected without keeping the network interface at full power.

The edge device can also use the wake time negotiation function when it enters a low power state. When a packet is sent to the client from a remote device in order to reactivate it (e.g. wake-on-LAN), the negotiation gives the client sufficient wake time to come out of its low power state in order to process the incoming packet. This can improve the performance and reliability of services that rely on this type of remote wake function.

Prior to the EEE standard, it had become common practice for personal computers and servers to save power during sleep states by re-negotiating the Ethernet link to a lower speed. For example, a link normally operating at one Gigabit per second (1000BASE-T) could be downshifted to 10 megabits per second or 100 megabits per second when the PC enters a sleep state, then re-negotiated back to one Gigabit per second when the PC wakes. Although the power consumption difference between these link speeds is small compared to the fully on operating power of a PC, which can run at 50-100 watts or more, it is a significant power savings for a sleeping PC, which uses between one and five watts. With EEE capability, PCs and other “sleep oriented” devices can achieve power savings similar to the speed downshift while allowing a much faster transition back to the active state. That faster link wake time will improve the user’s experience by allowing a faster transition for the computer from the sleep to active and connected state.

## Managing EEE in Next-Generation Networks

The standard for EEE defines how LPI is communicated between systems and how the interface transitions into and out of its low power mode. However, the standard does not define why LPI should be communicated at any specific time. Each system must use its own policy to decide when to transition to sleep and wake. Some examples of these policies are:

- Simplest algorithm: when the transmit buffer is empty, wait a short time and then communicate LPI; when a packet arrives to send, re-activate the link
- Buffer-and-burst: when the transmit buffer is empty, communicate LPI; when a packet arrives, wait until a large enough burst arrives or until a timer expires and then re-activate the link
- Synchronized bursts: de-activate and re-activate the link according to a timer so that systems behavior is optimized to reduce backplane bandwidth or to avoid multiple interfaces being active simultaneously
- Application aware policy: monitor the transport or higher layer communication to understand whether the link can be de-activated or whether more packets should be expected shortly

Given the novelty of the standard, it can be expected that yet more policies may evolve according to various requirements. It is important that devices that support EEE can communicate their available policies to a network energy management system, such as Cisco EnergyWise, and that the policies can be controlled and coordinated for optimal network-wide behavior. Early EEE systems may not support many policies. Nevertheless they should interact with the network energy management system for future compatibility.

## Cisco Catalyst 4500E Switches and EEE

EEE is being introduced on the Cisco Catalyst 4500E Switches, Cisco’s leading campus access switching platform. This section discusses power savings that were achieved from actual lab test.

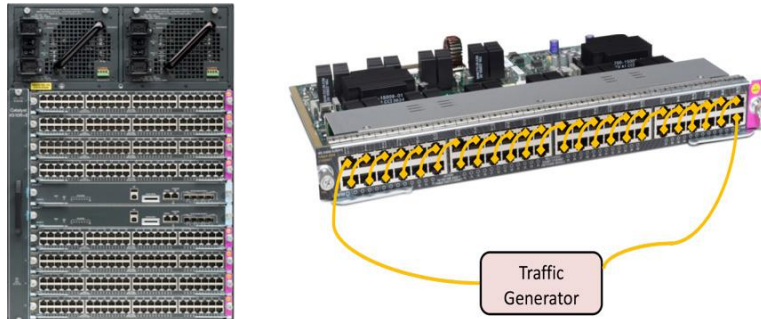
---

<sup>3</sup> <http://www.ecma-international.org/memento/TC38-TG4.htm>

A fully loaded Cisco Catalyst 4500E system with 384 1000Base-T ports was used to EEE testing. In the test setup, ports 1 and 384 are connected to a traffic generator sending traffic in bursts with very low link utilization to mimic the actual traffic profile of a desktop/laptop. The packet generator is tuned to have an inter-packet gap of 100 milliseconds with the bursts happening at the end of every 100 milliseconds. Each burst comprises of 100,000 64-byte packets.

All adjacent ports are looped in a snake fashion such that traffic entering port 1 is switched to port 2 which is externally looped back into port 3 and so on. This helps ensure that all ports see the same burst traffic from the traffic generator. Figure 5 shows how the traffic looping is performed on a single line card in the system:

**Figure 5:** Traffic Looping



The CLI output below shows the power consumed by the 384-port system before EEE is enabled. It can be seen that the instantaneous power consumed by the system is 892 watts.

```
4510_Sup7E#show power mod
```

Mod	Model	Watts Used of System Power(12V)				
		budgeted	instantaneous	peak	out of reset	in reset
1	WS-X4748-UPOE+E	75	59	59	75	35
2	WS-X4748-UPOE+E	75	62	44	75	35
3	WS-X4748-NGPOE+E	75	61	62	75	35
4	WS-X4748-NGPOE+E	75	61	62	75	35
5	WS-X45-SUP7-E	260	202	205	260	100
6	WS-X45-SUP7-E	225	199	203	225	225
7	WS-X4748-NGPOE+E	75	64	64	75	35
8	WS-X4748-NGPOE+E	75	62	62	75	35
9	WS-X4748-UPOE+E	75	59	59	75	35
10	WS-X4748-NGPOE+E	75	63	63	75	35
--	Fan Tray	255	--	--	--	--
Total		1340	<b>892</b>	883	1085	6053

The exact setup is used to measure the instantaneous power of the system after EEE is enabled. As can be seen from the output, the power drops to 751 watts.

```
4510_Sup7E#show power mod
```

Mod	Model	Watts Used of System Power(12V)				
		budgeted	instantaneous	peak	out of reset	in reset
1	WS-X4748-UPOE+E	75	42	59	75	35
2	WS-X4748-UPOE+E	75	44	44	75	35
3	WS-X4748-NGPOE+E	75	44	62	75	35
4	WS-X4748-NGPOE+E	75	43	62	75	35
5	WS-X45-SUP7-E	260	202	205	260	100
6	WS-X45-SUP7-E	225	199	203	225	225
7	WS-X4748-NGPOE+E	75	45	64	75	35
8	WS-X4748-NGPOE+E	75	45	62	75	35
9	WS-X4748-UPOE+E	75	42	59	75	35
10	WS-X4748-NGPOE+E	75	45	63	75	35
--	Fan Tray	255	--	--	--	--
Total		1340	<b>751</b>	883	1085	605

There is a 141 watt reduction in power consumed by the 384-port system. Even though the above setup has 384 ports, there are only 191 EEE-enabled links since adjacent ports are connected to each other and form a single link. Moreover, the traffic generator itself is not EEE-capable. Therefore, it can be concluded that the average power saved per EEE link is 0.74 watts.



EEE is supported on Cisco Catalyst 4500 Supervisor 7-E, 7L-E or later based Cisco Catalyst 4500E Switches with EEE-capable line cards. Table 1 below shows EEE compatibility on Cisco Catalyst 4500E platform.

**Table 1:** EEE compatibility on Cisco Catalyst 4500E Platform

Chassis	Supervisor	Line Card
WS-C4503-E	WS-X45-SUP7-E	WS-X4748-UPOE+E
WS-C4506-E	WS-X45-SUP7L-E	WS-X4748-RJ45-E
WS-C4507R+E		
WS-C4510R+E		

## For More Information

For more information, please refer to the [Cisco Catalyst 4500E Switch data sheet](#).

## Intel and EEE

Intel® Ethernet products for client and server platforms support EEE as a standard feature.

- The Intel 82579LM Gigabit Ethernet Connection delivers significant power savings to desktop and mobile platforms by reducing idle power by 90% when in EEE mode. This product is a key component of all second-generation Intel® Core™ vPro™ processor family systems, and power efficiency features in the processor, chipset, and network connection combine to deliver better performance than previous generation systems while significantly reducing power requirements.
- The Intel Ethernet Controller I350 is the industry's first fully integrated quad-port Ethernet controller to support EEE. This controller, which combines both MAC and PHY functionality, is optimized for Intel® Xeon® processor-based platforms and supports EEE across all four ports. Power consumption is reduced by approximately 50% when in EEE idle mode.

For more information visit <http://www.intel.com/go/ethernet>

© 2011 Cisco and/or its affiliates. Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company (1005R)

© 2011 Intel Corporation. All rights reserved. Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.