One of the biggest growth technologies in the industrial market is Ethernet based networking. Most industrial Ethernet standards use the IEEE 802.3 standard Ethernet protocol giving them the ability to carry standard network traffic as well as real time data. Each standard uses a different technique to deliver real-time performance—some use custom hardware, others use custom software and some use a completely standard Ethernet/TCP/IP implementation. The result is a set of operationally incompatible standards with different levels of performance and cost.

One increasingly popular way around the non-deterministic communication times of the Ethernet protocol is to implement a local clock in every device; as most devices contain a microprocessor and a (relatively) high speed clock, this is usually easy to implement. If accurate clock synchronization across the network can be implemented and maintained, while controlling the accurate timing of operations across the whole system, the only limitations are the latency of communication and accuracy of the system wide clock synchronization.

This type of system control is not suitable for applications like accurate motion control (for example precision speed control of a motor under varying load) as these require low latency communication between the controller and device, but it is very useful for accurate control of whole systems (for example a large printing press or long automated production line) where highly synchronous system wide control (e.g. a speed change) is required. If there is enough time to issue a command to every device, the only limitation to the accuracy of clock based control is the accuracy of the clock synchronization across the system.

Several industrial networking standards (and not just Ethernet based standards) are implementing the IEEE 1588 standard to provide this kind of control capability. IEEE 1588 provides a high accuracy master clock and proven mechanisms for clock synchronization that can be used to create and maintain a very precise system wide synchronization of all local clocks with the master clock.

Ethernet based networks are attractive due to the low cost and ease of implementation of Ethernet. Ethernet switches are a key component that helps deliver these benefits, and corporate systems rely heavily on them to implement high performance and easy to maintain infrastructures. The huge corporate market for switches means that they are readily available and low cost, however most of the switches presently available on the market have not been designed to deliver low latency performance or deterministic routing times, making them difficult to use in the industrial environment.

The IEEE 1588 system synchronizes the master and slave clocks by measuring the communication delay time between the master and slave. Having a switch placed between the master and slave clock introduces an additional latency as the switch has to analyze the packets and route them. This additional latency is not good, but it is not the major issue as it can be factored into the delay
correction. The problem is when traffic increases the time taken to route the packet gets significantly longer.

This is caused by the time taken to buffer, analyze, and route packets to many destinations. This variability greatly reduces the accuracy of the 1588 clock synchronization and hence significantly degrades the real-time control over the system. The measurement of the delay between the 1588 master and the slave clocks also relies on symmetrical communication times in both directions as it is calculated by halving the time taken for a time stamped message to travel from the slave clock to the master and back again. In most switches and Ethernet network implementations, this symmetry is unlikely to occur, again causing degradation in the accuracy of the clock synchronization.

![Figure 1. A Typical Block Diagram of a Boundary Clock Application.](image)

However, the IEEE 1588 standard offers a way to resolve this issue, if the switch also contains a clock (as shown in Figure 1) the time taken to route the packet can be measured and incorporated into the synchronization calculation. As corporate systems do not require this capability, switches that contain this type of 'boundary' clock are difficult to find and are usually expensive and tailored for a particular kind of network implementation. As the use of IEEE 1588-based networks is growing rapidly, manufacturers are encountering challenges of how to efficiently and cost-effectively implement an IEEE 1588 capability into their products and network infrastructures.

Developing a custom ASIC solution is possible, but with the rising cost of ASIC development and the rapid changes in industrial Ethernet standards, developing this kind of solution is slow, risky, and not cost-effective. It is possible to develop solutions for individual protocols using a microprocessor and third party ASICS or ASSPs that target individual networking protocols, but this means implementing a unique solution for each networking standard that again is expensive and inefficient. These types of solutions are also likely to run into issues with lack of flexibility and device obsolescence.

To date designers have worked around this limitation by carefully implementing the network to minimize the use of switches or to minimize the network traffic over the real time dependent sections of the network. This kind of network isolation can produce acceptable performance levels for some applications, but they are not easy to implement or maintain.

**Save Six To Nine Months**

Implementing an IEEE 1588 enabled switch in an FPGA device is an ideal solution to this problem. Altera, National Semiconductor, and MorethenIP GmbH have combined their expertise to provide Industrial Ethernet designers with an 8-port optimized switch design that will save six to nine months of engineering development time. Saving this kind of development time will provide
equipment manufacturers with a significant advantage in their time-to-market.

Figure 2 shows an Altera Stratix II FPGA-based development board from MoreThanIP that includes embedded 1588 capabilities. The board is delivered with a reference design that implements a 1588 enabled switch in a simple, cost-effective way that can be easily adapted to suit other systems and to meet changing market requirements. This is achieved by the flexibility of the FPGA and the incorporation of a 32-bit RISC processor core in the FPGA design.

![An 8-Port Switch Development Board with IEEE 1588 Timing Control.](image)

The intellectual property (IP) for the Ethernet MAC cores and switch fabric core with 1588 timing control and programmable uplink has been developed by MorethanIP GmbH. MorethanIP also provides the UDP and 1588 software protocol stacks that run on the soft core, 32-bit Altera Nios II RISC processor. In order to provide an optimal physical interface the 8-port switch design targets four dual-port PHY transceivers from National Semiconductor.

The reference design can be used for a wide range of applications and offers sub-100ns clock synchronization capability. This level of accuracy is critical for meeting strict communications latency and quality of service (QoS) requirements for industrial connectivity. Applications include switches using such industrial standards as Ethernet/IP, Profinet, Ethernet Powerlink, and other Ethernet protocols. **Reprogrammability Extends Product Lifecycles**

The programmability of an FPGA is the key to the benefits of this design. From a single hardware platform a designer can easily implement switches that support different Industrial Ethernet protocols (such as EtherCAT, Profinet, etc). The board can support different Industrial Ethernet protocols in the same system or even from the same Ethernet port.

This is done by implementing different media access controller (MAC) hardware blocks and embedded processor software to support different Ethernet standards as well as the IEEE 1588 capability. The ability to easily re-use previous designs and the availability of 'off the shelf' IP means that creating a configuration that supports new features can be done in a very short space of time compared to using ASIC or ASSP devices.

An FPGA loads the hardware configuration and embedded processor software from a serial flash memory device. The functionality of both the FPGA hardware and software is easily changed by
rewriting the contents of the flash device during production or even when distributed in the field.

The availability of programmable hardware and software processing capabilities within an FPGA means that designers can incorporate any additional features required via the application as hardware or software. The ability to implement new functionality simply by re-programming an FPGA future-proofs the product (e.g. to support IEEE 1588 v2.0) and enables a very fast turnaround time for introducing new features to customers.

As FPGAs have long lifetimes, equipment manufacturers should not worry about potential end-of-life announcements. As the design is IP-based it is also relatively easy to transfer the design to the next generation of FPGAs, thus offering the developer the ability to benefit from the possibility of lower costs or increased performance of the next generation of FPGA family. Combined with the ability to easily update the devices in the field these factors make an FPGA implementation the best way to develop a product that can be easily supported throughout the whole product life cycle.

This reference design uses a Stratix II FPGA that allows all of the Nios II processor code to be held in on-chip memory, but a lower cost version of the system can be implemented using the Altera Cyclone III family of FPGA devices.

![Figure 3. A Block Diagram of a MorethanIP 8-port switch with IEEE 1588](image)

Click here for full image.

**Embedded Switch Fabric IP**

Figure 3 shows the embedded 8-port switch fabric from MorethanIP GmbH that includes eight 10/100 megabits per second (Mbps) Ethernet 802.3-compliant MACs. Each MAC is 1588 enabled, that means it time stamps each incoming 1588 frame using a locally synchronized high precision clock from a programmable timer.

To implement boundary clock applications, the switch design implements both 1588 V1 Slave and Master applications. The port that communicates to the grand master is automatically configured as a slave port. From the slave port, the embedded 1588 application generates a precise clock and forwards the clock information to the other ports that are then, automatically configured as master
ports. A tightly coupled integration with the programmable timer ensures a synchronization precision with the grand master clock of less than 100ns.

In the switch up to two prioritized queues per port can be implemented to provide quality of service (QoS) to a critical class of services. The switch can also be programmed to identify and prioritize traffic that uses the 3-Bit VLAN priority field, the 6-bit DiffServ Layer 3 Code Point (IPv4) or the 8-bit class of service (IPv6).

The switch design supports IEEE 1588 version 1 boundary and version 2 transparent clock applications. It can be enhanced and modified to add custom logic, for example, bridge applications to legacy protocols or different system interfaces like PCI, to ease integration of the switch into an existing system.

![Figure 4. 1588 Block Diagram including Software and Hardware.](image)

In this design, the Nios II embedded processor supports the switch IP configuration and management as well as running the user datagram protocol (UDP) stack, the 1588 protocol stack, the precise timing synchronization and the PHY management and line diagnostic features of the dual 10/100 PHY transceivers as shown in Figure 4. The embedded processor can also be used for high level networking functions such as running Spanning Tree and Rapid Spanning Tree algorithms and termination of TCP/IP connections. The Spanning Tree Protocol (STP) and Rapid Spanning Tree Protocol (RSTP) are link management protocols that enable path redundancy that prevent undesirable loops in the network (for an Industrial Ethernet network to function properly only one active path can exist between two nodes).

**PHY Transceivers**

Each transceiver features two fully independent 10/100 ports for multi-port applications as shown in Figure 5. The device’s port switching also allows the two ports to be configured to provide fully-integrated range extension, media conversion, fast hardware-based fail-over in nanoseconds, and port monitoring.
This device integrates multi-port support for common Industrial Ethernet topologies. In particular, the designer gets redundancy support for different applications that demand the ability to deal with fail-over/under in a variety of conditions. Switching from one network stack to another can take upwards of hundreds of milliseconds, but some applications (for example, safety applications) require extremely fast fail-over that is best performed at the PHY layer. The transceivers on this reference design switch from one port to another at nanosecond speeds, even though the host still handles the control path. Architectural improvements in the signal path of the transceiver turbo-charges performance beyond the minimum PHY layer specifications to deal with such design issues as jitter and latency. Each Ethernet PHY layer is driven from a reference clock and to minimize jitter, the specification calls for an extremely accurate clock that is within 50 PPM of the transceiver's 25 MHz reference. Additionally, jitter must be very low to begin with to meet the specification. To resolve this issue, a higher jitter tolerance is built into the architecture. The device architecture also optimizes latency for real-time Ethernet operation ensuring that the switch latency is kept to a minimum.

In many real-time system implementations, Ethernet packet data transfer latencies are important parameters for proper system operation and fixed and variable components of the transmit and receive latencies within the Ethernet PHY can be critical components of system latency calculations.

This PHY transceiver is designed to limit the variability of the receive data latencies and as a result, provides for very deterministic system delay. In particular, the device does not suffer from a common non-determinism due to aligning receive data to the receive clock. Consequently, significantly more deterministic receive data latency is provided in both MII and RMII modes. Also, the transceiver reduces common non-determinism in the transmit RMII latency.

Another important design feature is built-in cable diagnostics featuring predictive diagnostics that is added to the conventional time domain reflectometry (TDR) method the transceivers use. This new dimension of fault isolation taps the signal processing strengths of the transceiver to track link quality while data is being transferred. This robust TDR implementation involves sending a pulse out on either transmit or receive conductor pair and observing the results on either pair. By observing the types and strength of reflections on each pair, software is deployed to determine cable shorts and opens, distance to a fault, identifying which pair has a fault and pair skew. Proactively
monitoring and correcting the changing or deteriorating link conditions can save lengthy system
downtime and costly repairs. This capability also detects faults during installation, saving significant
man-hours of debug work.

**Conclusion**

Industrial Ethernet technology has been evolving and growing in popularity, while designers are
facing a growing demand for cost-effective industrial switches. ASIC- and ASSP-based switches, with
their fixed architectures, leave virtually no latitude for customizing system features and usually
require a complete re-design to add the features required. This incurs extra design time and cost.
However, an FPGA-based design, such as the 1588-enabled switch discussed, can save from six to
nine months engineering time and provides designers with that much sought after flexibility for
implementing features like the precise timing protocol (PTP), support for different or multiple
Industrial Ethernet standards, additional standard interfaces or any of a host possible custom
features.

**About the authors**

Francois Balay is CEO at MorethanIP GmbH, Karlsfeld, Germany.

Andrew J. McLean is Director of the Atlanta Design Center, Interface Division at National
Semiconductor Corporation, Atlanta, GA.

James Adams is Industrial Business Group at Altera Corporation, San Jose, CA.