Inverse Multiplexing for ATM (IMA) Solutions with Spartan-II FPGAs

WP107 (v1.0) January 11, 2000

Authors: James D. Beatty, Krishna Rangasayee

Introduction

Aggressive Process technology adoption has allowed PLDs to obtain more die per wafer, provide more logic, offer increased performance, and accommodate the various ASIC-like features required to allow system integration. This has been fundamentally instrumental in narrowing the wide gap between FPGAs and ASSPs. PLD vendors, by virtue of the benefits reaped through process technology now have the capability to service the needs of today's ASSP designers.

Spartan™-II FPGAs offer more than 100,000 system gates at under $10 and are the most cost-effective PLD solution ever offered. The Spartan-II family is extremely well positioned to offer a low-cost programmable ASSP alternative and expands the time-to-market advantage that PLDs traditionally offer. It also increases the value of the ASSP by allowing end users to customize their solutions.

The Spartan-II family, combined with a vast soft IP portfolio is the first programmable logic solution to effectively penetrate the ASSP marketplace. The ATM IMA-8 solution from Applied Telecom ported on the XC2S150 is a good example, highlighting the concept of a programmable ASSP. Applied Telecom is a member of the Xilinx AllianceCORE program and brings a wealth of expertise in ATM, SONET, telecommunications, and networking applications.

Applied Telecom, Inc. is a leading supplier of embedded communications products and technologies that meet the needs of communications equipment and semiconductor device manufacturers. The technical focus includes but is not limited to communications standards such as SONET/SDH, PDH, and ATM. Software and hardware products from Applied Telecom provide vital standards-compliant system functions and are designed with well defined interfaces and features to work in many different system architectures and applications. For more information, visit http://www.xilinx.com/products/logicore/alliance/app_telecom/applied.htm.

The IMA-8 core, developed, sold, and supported by Applied Telecom, targets network access systems such as adapters, multiplexers, and switches. Several leading manufacturers, including Alcatel, Ericsson, Nokia and Nortel, are already using Applied Telecom's Xilinx-based IMA technology in production systems.

The IMA-8 core is available immediately for use in Spartan-II FPGAs. An evaluation board and the DRV-IMA software are also available now. All IMA products can be purchased directly from Applied Telecom.

What is IMA?

IMA stands for Inverse Multiplexing for Asynchronous Transfer Mode (ATM) and makes possible the transmission of a high-bandwidth stream of ATM cells over multiple T1 (1.544 Mbps) or E1 (2.048 Mbps) facilities (or circuits). First standardized (v1.0) by the ATM Forum in 1997, and recently updated (v1.1) in 1999, IMA is applicable to both public and private networks and allows end users to enjoy the many benefits of ATM (e.g., Quality of Service (QoS) provisioning, scalability, and the ability to easily mix data, voice, and video) but circumvent the high cost and unavailability of broadband transmission facilities such as T3, E3, and SONET/SDH by using only as many lower cost, lower bandwidth facilities as necessary. With the advent of Digital Subscriber Line (DSL) technology, the case for IMA is even greater.

IMA Applications

IMA is applicable to many different types of ATM Wide Area Network (WAN) access equipment including ATM switches and routers with WAN ports, ATM access concentrators and multiplexers, and communications servers with WAN NICs. Typically, IMA is used as the WAN interface for general purpose access multiplexers, traffic aggregators, and access switches. Another common application is in Digital Subscriber Line Access Multiplexers (DSLAMs) where IMA can be used either to interconnect the DSLAM with a Remote Access Multiplexer (RAM) or as the high speed network side interface. Emerging applications are extending the IMA protocol beyond T1 or E1 to include other facilities including DSL circuits.

IMA Operation

Figure 1 illustrates the basic IMA mechanism for sending a single ATM cell stream over a number of lower speed transmission facilities or links.

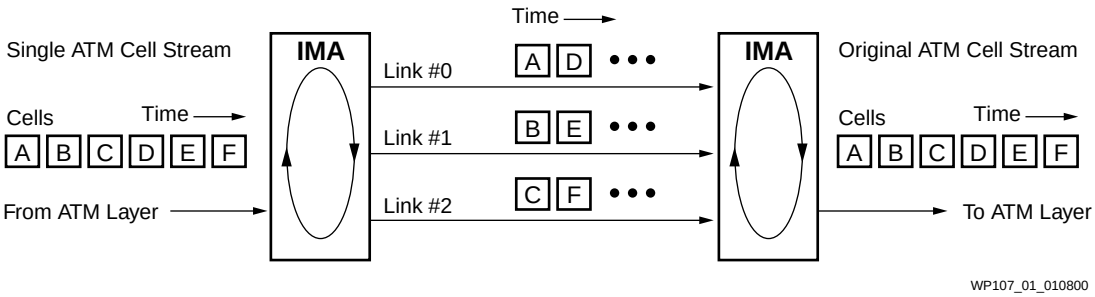

Figure 1: Simplified IMA Process

The IMA links contain three types of cells: ICP cells, filler cells, and ATM layer cells. The first two are IMA-specific cells. IMA Control Protocol (ICP) cells are the IMA overhead cells that carry the IMA control and status information between the two ends of the link, assuring synchronization and common configuration. The purpose of filler cells is rate decoupling: they are inserted into the IMA stream if no ATM layer cells are available. ATM layer cells are the data "payload" carried by the IMA group.

When splitting an ATM virtual circuit among N T1/E1 links, the IMA Subsystem must insert the IMA- specific cells into ATM cell streams of the transmit T1/E1 links. In the receive direction, a cell-based IMA framing process is used to locate and remove the IMA-specific cells so that only the ATM cells are passed to the ATM Layer. A variable number of physical links and ATM bandwidth rates can be supported and mechanisms are specified for accommodating differential delay variations present in the transmission links and for handling link failures and changes to the available transmission bandwidth.

Layer Reference Model

In terms of the protocol layer reference model, the IMA sublayer is considered to be an extension of the Physical Layer (PHY), sitting below the ATM Layer (ATM) and, as much as possible, transparent to the ATM Layer device. Table 1 illustrates the functions performed by the IMA sublayer.

Table 1: IMA Sublayer Reference Model

Layer	Sub-layer	Functions
ATM Layer		
Physical (PHY) Layer	IMA Specific Transmission Convergence Sublayer	ATM cell stream splitting and reconstruction
		Differential delay accommodation
		IMA Control Protocol (ICP) cell insertion / removal
		Cell rate decoupling
		IMA frame synchronization
		Cell stuffing, asynchronous facility compensation
		Discard cells with HEC errors
	Interface Specific Transmission Convergence (TC) Sublayer	Header error correction
		HEC generation / verification
		Cell scrambling / descrambling
		Cell delineation
		Scrambling / descrambling
		Transmission frame generation / recovery
	Physical Media Dependent (PMD) Sublayer	Bit timing, line coding
		Physical medium

From an ATM layer perspective, the behaviors exhibited by IMA groups are different than those of real PHY facilities. Two examples include the effects of IMA group start-up and variations in bandwidth caused by the activation/deactivation and addition/deletion of links. Typical PHY facilities are either active or inactive; IMA links include these conditions but also exhibit a number of intermediate states. Bandwidth changes caused by the varying number of active (i.e., traffic carrying) links within IMA groups must be reflected in the QoS of the virtual circuits carried by the IMA group.

In addition to the PHY layer, IMA also affects the management layer. The management layer handles alarm detection and processing, making it possible to configure IMA groups, add and delete links, and maintain IMA sublayer statistics.

PHY Layer Considerations

A typical PHY layer implementation with IMA is shown in Figure 2.

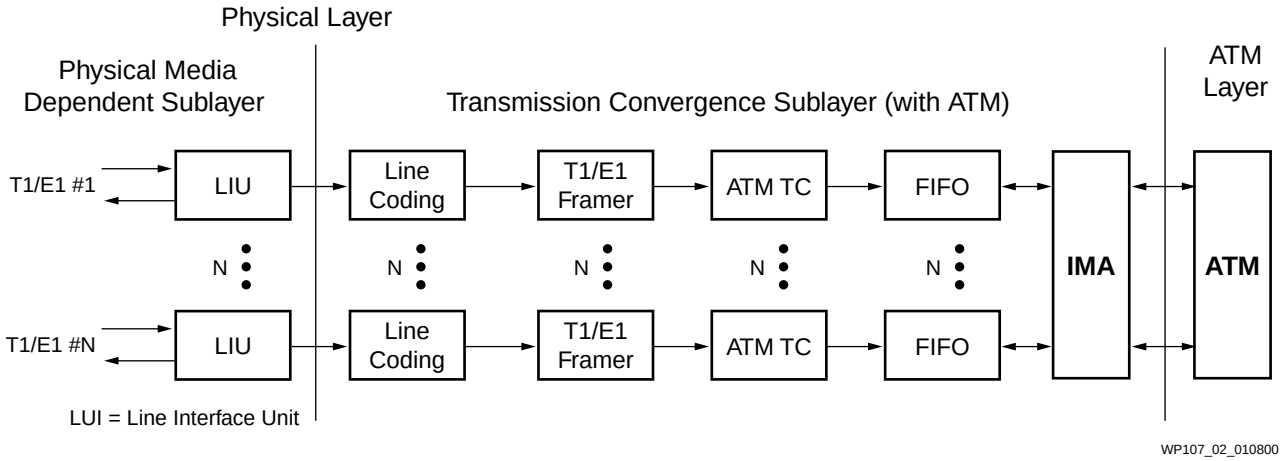

Figure 2: Typical PHY Implementation

In many such implementations, the Physical Layer function is resident on a "line card"" which is physically separate from the ATM layer device to allow the ATM device to serve many facilities. This co-location of IMA on the line card restricts the facilities which can be allocated to an IMA group because only the specific T1 or E1 facilities attached to that line card can be grouped, limiting the configurability of the system. One solution, for maximum configurability, is to place the IMA function with the ATM layer device. But this is usually difficult to implement and causes some system level functional partitioning problems (e.g., splitting up the PHY layer across modules). A more common solution that sacrifices some of the flexibility in assigning facilities to IMA groups is to develop the IMA functionality and the line card so that each T1/E1 facility can be independently configured to be part of an IMA group or be bypassed around the IMA function to be accessed uniquely by the ATM layer device. With this solution, the line card is no longer a "dedicated" IMA card.

IMA Solution

Given the availability of many off-the-shelf devices providing the T1/E1 line interface, framing, and ATM Transmission Convergence (TC) sublayer functionality plus the wide acceptance and usage of the ATM Forum's UTOPIA bus interface for ATM cell transfer, it is natural to define an IMA solution that can be inserted between the TC function and ATM layer. With the availability of the Spartan-II family, a complete IMA solution can be implemented using a single XC2S150 device, an external SRAM device, and a software driver.

IMA-8

The **IMA-8** product is an XC2S150 device solution that supports up to eight links and four IMA groups. The external interfaces for this FPGA device are shown in Figure 3. The IMA implementation has been partitioned in such a way that the real-time processes are performed in the FPGA and all non-real time processes are performed by a software driver. For example, all IMA link state machines are implemented in the FPGA but the IMA group state machines are implemented in software. This partitioning eliminates interrupts from the FPGA and allows the software to operate as a periodic background task on the processor.

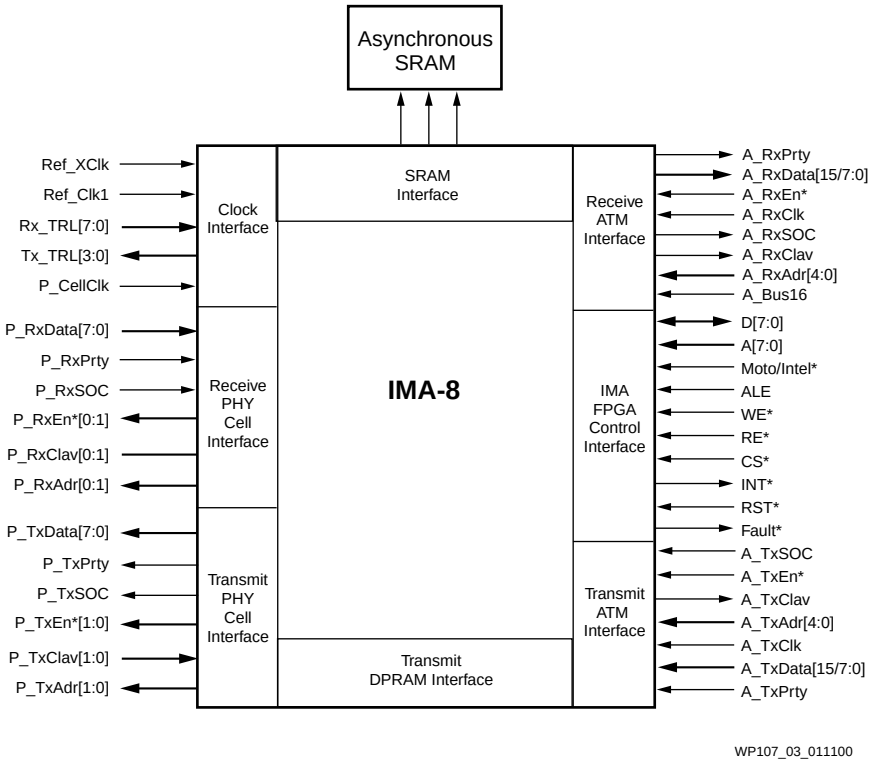

Figure 3: IMA-8 Interfaces

Functional Description

A simplified block diagram of the IMA FPGA solution is shown in Figure 4. The solution is composed of four main functional areas: the IMA clock generators, the Transmit IMA processing, the Receive IMA processing, and the Microprocessor Bus Interface. An IMA software driver completes the IMA implementation.

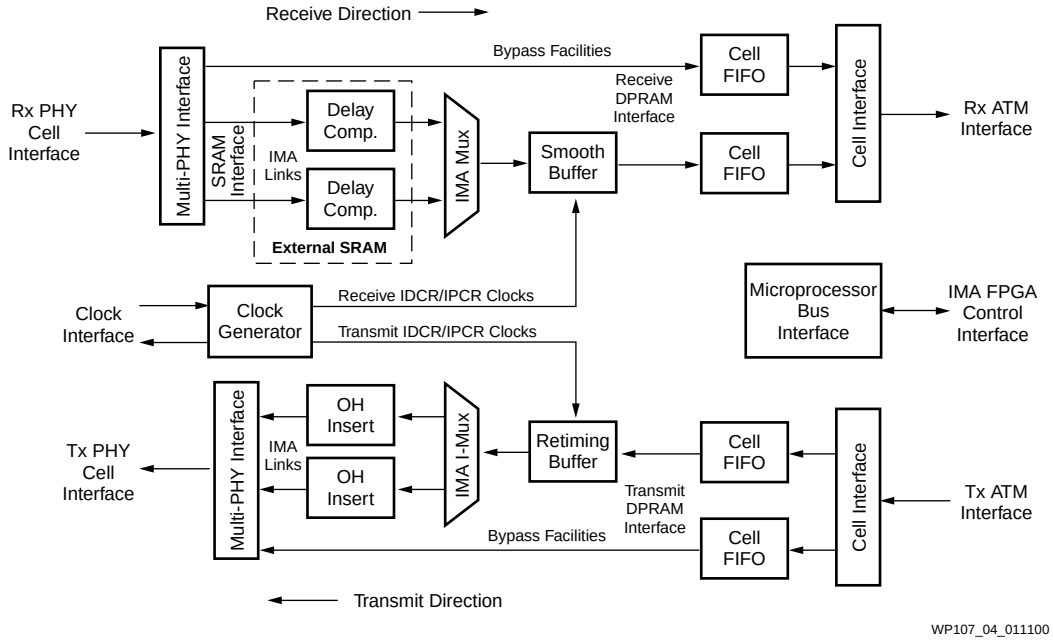

Figure 4: Functional Block Diagram

IMA Clock Generators

The IMA clock generator is responsible for producing the IMA Data Cell Rate (IDCR) and IMA PHY Cell Rate (IPCR) clocks that are used by both the Receive and Transmit IMA Processors. This block allows for significant flexibility in clock selection for both the Transmit and Receive IMA Processors.

Transmit IMA Processing

The Transmit IMA Processing section receives ATM cells from the ATM layer via the Utopia Transmit ATM Interface (Tx ATM I/F). The cell data is input to a bank of cell FIFOs implemented by a Dual-Port RAM (DPRAM). In the IMA-8 Spartan-II solution, the DPRAMs are implemented internal to the FPGA using BlockRAM. The Retiming Buffer extracts the ATM cells from the cell FIFOs to synchronize the data to the Transmit IMA Data Cell Rate (Transmit IDCR) clock. The IMA I-Mux inverse multiplexes the Transmit ATM Interface data into the individual IMA links. The OH Insert block generates the IMA frame and inserts the IMA Overhead (ICP, SICP, and Filler) cells for each IMA link. The IMA encoded cell stream is then passed to the Transmit PHY Cell Interface (Tx PHY Cell I/F). The Transmit PHY Cell Interface serves as a Utopia Level 2 master to one or multiple PHY devices. These PHY devices encapsulate the ATM cell data for transmission over the T1/E1 facility link. The Transmit IMA Processor also supports non-IMA facilities by allowing individual T1/E1 ports to operate in a pass-through mode whereby ATM cell data is passed from the Transmit ATM Interface to the Transmit PHY Cell Interface without IMA processing.

Receive IMA Processing

The Receive IMA Processing section provides the inverse operation of the Transmit IMA Processor. The Receive IMA Processor accepts cell streams from multiple T1/E1 facilities via the Utopia Level 2 Receive PHY Cell Interface (Rx PHY Cell I/F). The Receive PHY Cell interface data is monitored for IMA framing and then written to the external SRAM memory via the SRAM Interface. The SRAM memory provides storage for Delay Compensation (a minimum of 25 ms is required), which is necessary because of potential differences in transmission delay experienced by the individual T1/E1 links in an IMA group. The IMA Mux extracts the realigned cell data from the SRAM Interface and removes the IMA control overhead. The cell data is then multiplexed into higher speed ATM cell streams according to the assigned IMA groups. The Smooth Buffer is used to retime the cell data to the Receive IMA Data Cell Rate (Receive IDCR) clock and then passed to the Spartan-II DPRAM which provides a cell FIFO for the Receive ATM Interface (Rx ATM I/F). The Receive ATM Interface distributes the cells to the ATM layer. The Receive IMA Processor also supports non-IMA facilities by allowing individual T1/E1 ports to operate in a pass-through mode whereby ATM cell data is passed from the Receive PHY Cell Interface to the Receive ATM Cell Interface without IMA processing.

Microprocessor Bus Interface

The IMA FPGA Control Interface provides a standard Motorola or Intel compatible microprocessor bus interface for configuration, control, and status. The DRV-IMA software, available from Applied Telecom, uses this interface to communicate with the IMA FPGA.

DRV-IMA

The DRV-IMA product is an ANSI C software program that complements the IMA FPGA device and completes the IMA solution. The DRV-IMA device driver provides a common, high level application programming interface (API) for the IMA-8 device. The DRV-IMA software interfaces with higher level application software and lower level utility functions through a series of ANSI C function calls and data structures. The interface and parameters of the DRV-IMA are compatible with the IMA MIB specified by the ATM Forum. It is designed to be independent of the microprocessor and the Operating System (OS) or real-time executive environment and can be used in object-oriented systems. The DRV-IMA software is composed of many subsystem components: Configuration, Diagnostics, IMA Group, Failure Monitoring, Performance Monitoring, and the embedded monitor.

Conclusion

Early deployment of IMA technology meeting the IMA v1.0 standard began in late 1997 but due to different interpretations of this specification, true multi-vendor interoperability was not really possible until the completion and acceptance of the IMA v1.1 specification in 1999. Throughout this period, the changes to the applicable technical standard and the lessons learned through inter-vendor testing required the flexibility of FPGA and software implementations of IMA. At present, stability in standardization plus large scale IMA deployment have set the stage for the introduction of standard silicon IMA products. But IMA is still an emerging technology with limited test equipment support and compliance test suites. That combined with different line card architectures (partitions and sizes) and non-T1/E1 IMA applications, suggests that IMA will remain somewhat of a moving target for some time to come. An FPGA based IMA solution with efficient partitioning of hardware and software functions provides the necessary scalability and flexibility to handle all of these applications and allow for tracking of new standards. With the introduction of the Spartan-II family of devices and the IMA-8 and other Xilinx based IMA core solutions, an FPGA based IMA implementation is simple and economical.

References

James D. Beatty, President, Applied Telecom

Krishna Rangasayee, Manager, Strategic Applications, Xilinx

Revision History

Date	Version	Revision
01/11/00	1.0	Initial Xilinx release