Power QUICC™ II Pro Family of Communications Processors: A Broad Range of Advanced Functionality in IP Convergence

CRISTIAN DUMITRESCU, VALENTIN CIOCOI, MIRCEA POP
Freescale Semiconductor Romania SRL
45, Tudor Vladimirescu Street, Bucharest 050881
ROMANIA

Abstract: - This paper presents Freescale's MPC8360E Power QUICC™ II Pro communications processors family with QUICC Engine™ technology. It provides an overview of the internal architecture and the suitable networking applications for this processor, including some advanced functional features like interworking, switching, parsing and forwarding. It also discusses how the QUICC Engine™ enables the convergence of ATM, TDM, Ethernet and IP networks.

Key-Words: - Communications Processor, ATM, IMA, Ethernet, Interworking

1 Introduction

The MPC8360E communications processor is a programmable System on a Chip (SOC) targeted for processing the networking traffic at wire speed, according to the requirements of a wide range of communications applications. Its control plane and data plane programmability differentiates it from an Application Specific Integrated Circuit (ASIC) whose hard-wired logic is designed to implement a single application. The MPC8360E processor is the newest member of the Power QUICC™ II Pro family of communications processors.

2 MPC8360E Communications Processor

The main components of the MPC8360E are a control pane processor, a data plane coprocessor, an array of network interfaces and a security engine coprocessor. The on-chip memory controllers provide the connectivity with:

- DDR1/DDR2 external memory chips using either two 32-bit buses or a single 64-bit bus
- SRAM, SDRAM, Flash or other external chips using an 8/16/32-bit bus providing 8 chip selects and configurable timing capability

Other on-chip components are a Programmable Interrupt Controller (PIC), four DMA channels, a 32-bit PCI bus interface, a dual I^2C bus interface with a boot sequencer, a dual UART interface and system timers. The MPC8360E internal shared bus provides a common address space used for device configuration.

2.1 The e300 Power Architecture™ Technology Core

The control plane processor of the MPC8360E is a 32-bit e300 Power Architecture™ technology core. This high performance core is a superscalar processor which is able to issue three instructions (two plus a branch) and retire as many as five instructions in parallel per execution clock cycle. Although the core makes completion appear sequential, the instruction execution is out of order to achieve increased performance. The e300 core integrates five execution units:

- an integer unit (IU) with full multiply and divides
- a floating-point unit (FPU)
- a branch processing unit (BPU) with static branch prediction
- a load/store unit (LSU) for data transfers
- and a system register unit (SRU)

Most integer instructions execute in one clock cycle. On the e300 core, the FPU is pipelined, so a single-precision multiply-add instruction can be issued and completed every clock cycle.

The e300 core provides independent on-chip, 32-Kbyte, eight-way set-associative, physically addressed instruction and data caches with parity and integrated way lock capabilities. The processor also features independent on-chip instruction and data memory management units (MMUs).

As an added feature to the e300 core, the device can lock the contents of one up to all ways of the instruction and data cache, or even lock an entire

cache. This allows locking interrupt routines or other important time-sensitive instruction sequences inside the instruction cache or frequently accessed data structures into the data cache.

The e300 core has a high-performance 64-bit data bus and 32-bit address bus interfaces to the rest of the chip. The e300 core supports single-beat and burst data transfers for memory accesses and also memory-mapped I/O operations.

The e300 core usually serves to implement the control plane, along with some tasks of the higher network layers and the application layer. These tasks are different in nature from the data plane processing, thus requiring a different type of processing capability than wire speed packet processing. Due to the continuous evolution of the standards and of the application, the control functions are primarily implemented in software, on a processor with a standard instruction set which allows using a high level programming language such as C. Moreover, the control plane processor is required to run an operating system with real time capabilities, like Linux or VxWorks. This is why a high performance e300 core was selected as the control plane processor of the MPC8360E.

The control plane is also called the exception path since the bulk of the traffic does not traverse it. For example, one typical control plane task is to update the routing table according to the routing update traffic received from the network, which is just a small fraction of the overall traffic received from the network (typically less than 5%). Due to this reason, the control path processor is not required to have the ability to process packets at wire speed, as opposed to the data plane coprocessor.

2.2 The QUICC Engine™ Data Plane Coprocessor

The data plane coprocessor is responsible for processing at wire speed the packets received from the network. Since the received packets are sent out back to the network, the processing performed by this fast path is also known as packet forwarding. Common data plane functions include packet classification, packet header manipulation, protocol conversion, QoS implementation and payload encryption/ decryption (on MPC8360E, the latter is performed by the security engine coprocessor, thus off-loading the other processing entities of this task).

The QUICC Engine™ coprocessor of the MPC8360E provides the data plane capabilities of the MPC8360E. Its aim is to provide wire speed processing for the traffic received over the MPC8360E on-chip array of network interfaces:

- 8 TDM interfaces
- 2 Utopia Level 2 (UL-2)/Packet over Sonet (POS) interfaces, each configurable as a master with support for 128 ports or as a slave
- 2 Gigabit Ethernet (GE) interfaces configurable as GMII/ RGMII/ TBI/ RTBI or 8 Fast Ethernet (FE) interfaces configurable as MII/ RMII

The ability of the QUICC Engine™ coprocessor to provide termination, switching and interworking for a wide range of networking protocols is based on the programmability of its two 32-bit RISC cores which control a set of configurable hardware accelerators, I/O blocks, serial DMA channels, baud rate generators and timers.

The two RISC cores have a proprietary multi stage pipeline architecture optimized for packet processing. They are able to execute most of the instructions at a rate of one instruction per clock cycle, those requiring more than one clock cycle being off-loaded to the hardware accelerators. The program executed by the RISC cores is loaded at boot time from an internal ROM memory into an internal instruction RAM (I-RAM) memory, which provides the reason of naming this program as firmware or microcode. One beneficial feature is the possibility to override the original I-RAM contents, thus allowing for firmware updates after the chip manufacturing. Moreover, one way of adding support for new protocol features is to develop the new microcode in I-RAM and moving it to the on-chip ROM when reaching maturity.

The I/O blocks of the QUICC Engine™ coprocessor are called communication controllers. Together with the network interface MACs mentioned above, these configurable blocks provide hardware support for those network protocols that are not feasible for a software implementation. The QUICC Engine™ coprocessor possesses eight Universal Communication Controllers (UCCs) and one Multi-channel Communication Controller (MCC). The UCC is basically a bidirectional FIFO with some additional logic that links a MAC with the RISC core applying the packet processing. The UCC can be configured to provide support for a variety of fast protocols (like Ethernet or ATM over UL-2) or slow

protocols (like UART or ATM over TDM). The MCC provides support for up to 256 TDM channels over the eight existing TDM interfaces and is used to implement protocols like ATM over TDM, HDLC and SS7.

The QUICC Engine™ coprocessor (figure 1) has an internal dual port 48 kilobytes SRAM memory called the Multi User RAM (MURAM). Since its primary use is to store the configuration parameters for the various configurable internal blocks, it can be seen as a customizable configuration space supplementing the fixed layout register space. The MURAM also provides the means to implement the communication mechanism between the QUICC Engine™ coprocessor the e300 core when combined with the capability of the former to send interrupt requests to the latter.

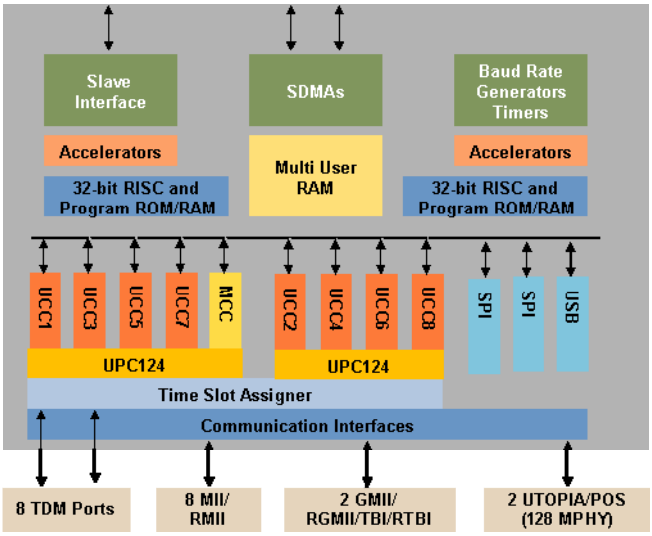

Fig. 1 QUICC Engine™ coprocessor block diagram

3 Implementation of the main Networking Protocols on MPC8360E

3.1 Ethernet Switch

The QUICC Engine™ coprocessor microcode implements a store-and-forward Fast Ethernet switch with up to 8 ports. Each port is implemented using an UCC and is connected to an external Fast Ethernet transceiver chip on an MII/RMII interface. The QUICC Engine™ coprocessor is responsible for the forwarding of the Ethernet frames according to the QoS policy while the e300 core is primarily responsible for the configuration of the Ethernet switch and for processing the incoming control frames which are identified by the data plane coprocessor and forwarded to the e300 core.

3.2 Asynchronous Transfer Mode (ATM)

The QUICC Engine™ coprocessor microcode implements the ATM and the AAL layers of the ATM protocol stack. It also implements the TC sublayer of the PHY layer for DS1/E1 physical links over the TDM interfaces using the Serial ATM microcode. The PMD sublayer of the PHY layer is implemented by external framing devices. There are several ways the ATM is supported by the QUICC Engine™ coprocessor:

- External ATM framers connected as slaves over one of the two UL-2 interfaces, either in single PHY (SPHY) or in multi-PHY (MPHY) mode
- External DS1/E1 framers without TC sublayer capability, each one connected to a TDM interface. The TC sublayer is provided by the Serial ATM microcode independently for each DS1/E1 link. The IMA microcode can further be employed to provide aggregation for several DS1/E1 links constituting an IMA group, which is presented to the ATM layer as a single virtual PHY device
- External DS1/E1 framers with TC sublayer capability connected as slaves over one of the two UL 2 interfaces, with the possibility of aggregating several DS1/E1 links as an IMA group

The QUICC Engine™ coprocessor can perform segmentation and reassembly of AAL5 frames at up to OC12 rate. For AAL2, the microcode implements the Common Part Sublayer (CPS) and the Service Specific Segmentation and Reassembly (SSSAR) sublayer.

On the ingress path, the lookup of the incoming ATM cells is performed using the VPI/VCI cell header fields. If the lookup operation is not successful, the cell is discarded, unless the cell is an Operation and Maintenance (OAM) cell, in which case it is enqueued to an ATM channel specially defined for this purpose. The ATM channel QoS policy is enforced by the traffic policer: even if the lookup operation is successful, the cell can still be discarded or marked by the policer according to a dual leaky bucket virtual scheduling algorithm.

On the egress path, the traffic scheduler allocates the available bandwidth between the registered ATM channels according to the bandwidth and service

class preconfigured for each channel. The real time traffic like constant bit rate (CBR) and real time variable bit rate (VBR RT) is prioritized against the non real time traffic such as non-real-time variable bit rate (VBR NRT), Unspecified Bit Rate (UBR) and UBR+.

The ATM microcode implements the VPI/VCI based automatic cell switching. For AAL2 traffic, the ATM microcode also implements the Channel ID (CID) based automated switching of CPS packets.

3.3 Inverse Multiplexing for ATM (IMA)

IMA provides a standardized way for the ATM networks to group multiple physical links of the same capacity into a single higher rate logical link referred to as the IMA group whose rate is approximately the sum of the rates of the physical links. The physical links which are employed are typically very well represented in the field, e.g. DS1/E1 links, as opposed to the aggregated bandwidth which is usually not readily available in the field, e.g. rates between the DS1/E1 and DS3/E3 levels. IMA describes a methodology to provide modular bandwidth for the ATM networks. The required amount of ATM traffic is split among the physical links of the IMA group.

Since the ATM traffic is made up of fixed length ATM cells, a simple round robin algorithm is used for dividing the traffic among the physical links of the group. The transmitter also periodically inserts special control cells into the data stream to allow the correct reconstruction of the ATM cell stream by the receiver. These cells, called IMA Control Protocol (ICP) cells, are specially defined OAM cells which allow the two ends of an IMA virtual link to synchronize with one another before the bidirectional exchange of data is allowed to proceed. The IMA protocol also defines the procedures for handling the run time events such as the user management commands for link addition, deletion, inhibiting and un-inhibiting, handling the occurrence of link defects and the link recovery after the defect goes off.

The IMA microcode utilizes the same UCC as the ATM microcode. The IMA microcode is responsible for the round-robin transmission and receive side reconstruction of the ATM cell stream, while the e300 core is responsible for maintaining an up to date state of the IMA group, as result of processing the ICP cells received from the other IMA end, handling the user group management commands and the events reported by the external DS1/E1 PHYs implementing the PMD sublayer, the Serial ATM microcode or the external DS1/E1 PHYs implementing the interface specific TC sublayer and the IMA microcode implementing the IMA specific TC sublayer. The current state of the IMA group is permanently communicated to the other IMA end using the transmitted ICP cells, whose update is triggered by the e300 core, although the actual moment an ICP cell is inserted into the outgoing ATM cell stream is dictated by the IMA microcode.

3.4 Interworking between ATM and Ethernet

In addition to the existing termination and switching support for Ethernet and ATM traffic, the QUICC Engine™ coprocessor microcode is also able to provide interworking between the two protocols. This means that traffic arriving on one interface can be sent out on another interface associated with a different protocol after performing the necessary protocol conversion while also maintaining the QoS. Without the interworking capability, the QUICC Engine™ block would not be a full data plane coprocessor, since the conversion between the two protocols would require the intervention of the e300 core, which would seriously undermine the capability of the entire chip to sustain the packet processing at wire speed.

Due to the interworking between Ethernet and ATM, conversion between complex protocol stacks becomes a reality. Other examples of protocol interworking supported by the microcode are:

- Interworking between Ethernet and IP over ML/MC-PPP over DS1/E1 links
- Interworking between ATM and IP over ML/MC-PPP over DS1/E1 links
- Interworking between Ethernet and IP over POS interface
- Interworking between ATM and IP over POS interface

4 Suitable Applications for the MPC8360E

4.1 Node B/BTS

The Node B network interface card (NIC) in UMTS Base Stations carries voice and data from the end user equipment to the Radio Network Controller

(RNC). The network backhaul is typically made of several T1/E1 links bundled through the Inverse Multiplexing over ATM (IMA) protocol. Higher bandwidth alternatives use an OC-3/STM-1 or OC-12/STM4 link, with ATM Adaptation Layers to carry voice (AAL2) and data (AAL5) traffic. Following the market trend of migrating to IP technology and packet networking, Release 5 of the 3GPP specification adds IP (UDP/IP) over ML-PPP as a secondary option to the Transport Network Layer. As far as the backplane is concerned, ATM over UTOPIA remains the predominantly used switching technology, but alternative technologies such as Ethernet are under consideration for future IP-ready solutions. When adding IP networking capability to an existing ATM based Node B, interworking functionality must be provided between the IP based network and the legacy ATM backplane. Service interworking at UDP level is also required for an ATM network and an Ethernet backplane.

In summary, specific NodeB NIC design challenges include:

- support for the increasing number of different network interfaces and speeds required, ranging from T1/E1 lines through OC-12 SONET and Gigabit Ethernet interfaces
- support for the convergence of ATM and IP packet networks by adapting to the different protocols used including: ATM (AAL2, AAL5), Ethernet, PPP
- QoS management to ensure prioritization of latency sensitive traffic such as voice
- flexibility to add new features and functions through in-field software upgrades

Figure 2 illustrates a typical NodeB NIC created with the MPC8360E, which provides all the processing, protocol and interworking functions required. The QUICC Engine™ coprocessor is able to transport voice, data and video using ATM or IP over eight T1/E1 links bundled with IMA or ML-PPP protocols between the NodeB and the RNC. In addition, one of the UCCs could be used to support either a Gigabit Ethernet (GMII) or ATM (Utopia 8/16bit) backplane interface. Another UCC could be used to implement an STM-1/OC-3 (AAL5 and AAL2) link to the RNC. The unused UCCs can be configured as serial (UART) or Ethernet (MII) for debug and control. The interworking function performed by the MPC8360E's QUICC Engine™ coprocessor is essential to offload the e300 CPU and accelerate data plane functions, including mapping of ATM ports to IP/UDP ports. The main system memory is provided by a dual 32-bit DDR SDRAM controller which offers the possibility of separating data plane from control plane processing. The local bus can be used for optional SDRAM or on-board FLASH EPROM.

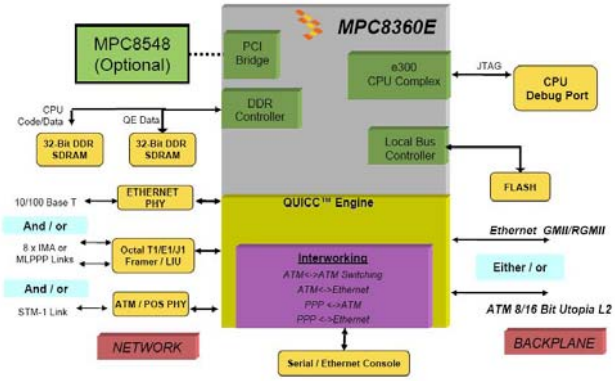

Fig. 2 Typical NodeB NIC implementation with MPC8360E

4.2 IP DSLAM

The Digital Subscriber Line Access Multiplexer (DSLAM) is a system that aggregates traffic from DSL end user lines towards the Central Offices (CO) of the Internet Service Providers. A typical DSLAM system consists of uplink cards, line cards and optionally control cards. As Access Infrastructure Networks are moving from the traditional centralized ATM-based architecture to a decentralized, packet switched/IP-based architecture capable of providing Triple Play services, the requirements for a DSLAM system in terms of connected users, data rates, bandwidths and port density are steadily increasing. Specific design challenges facing an IP-DSLAM solution include:

- technology migration from legacy ATM circuit-switched to Ethernet packet-switched solutions, including ATM to IP interworking at the data plane
- increasing port density on line cards – typically 24 to 96 ports
- steep bandwidth increase due to demand for high performance services/standards e.g. ADSL2/+ or VDSL for Triple Play
- QoS management to ensure prioritization of latency sensitive traffic to meet service level agreements
- hierarchical IP or ATM traffic scheduling and shaping capability
- intelligence moving towards the edge of the network
- ensuring the security/confidentiality of subscribers information

Figure 3 illustrates a typical connection between an IP-DSLAM and the customer premises equipment (CPE). The IP-DSLAM has to support interworking between ATM and Ethernet – ATM for the DSL connection to the CPE and Gigabit Ethernet for the uplink. The IP-DSLAM must be able to handle packet routing using either IP or PPP. Compliance to multi-protocol encapsulation over ATM is required for both IP-DSLAM and CPE systems. Finally, the IP-DSLAM must support Ethernet VLAN tagging and aggregation to ensure subscriber privacy.

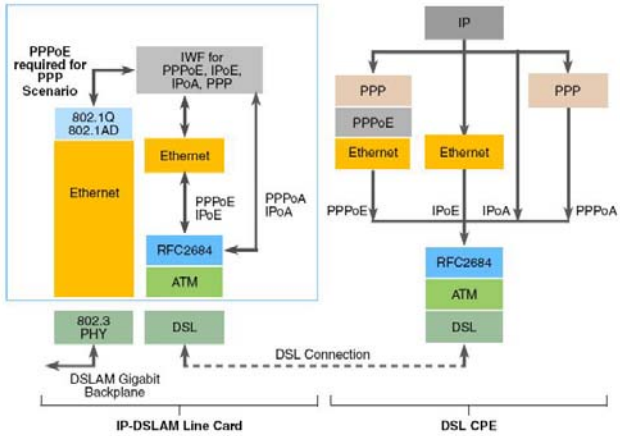

Fig. 3 Typical connection between IP-DSLAM and CPE

A typical CO IP-DSLAM application is presented in Figure 4. Subscribers connect to the IP-DSLAM via xDSL PHYs and the integrated UL-2 interface on the MPC8360E. The uplink connection to the Ethernet backplane or CO can be provided by one Gigabit Ethernet interface while the second interface could act as backup. The QUICC Engine™ performs ATM to Ethernet interworking and VLAN tagging to transfer data between the ATM-based DSL subscriber inputs and the Gigabit Ethernet uplink port. Additional Ethernet ports can be used for maintenance.

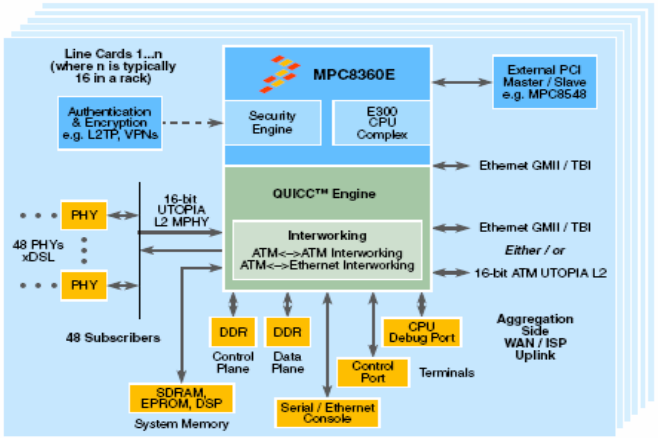

Fig. 4 Typical IP-DSLAM implementation with MPC8360E

5 Conclusions

The Freescale MPC8360E communications processor provides efficient termination, switching and interworking for the ATM and IP based networking protocol stacks. The control plane is implemented by the e300 Power Architecture™ technology core and the data plane is implemented by the QUICC Engine™ coprocessor which provides wire speed packet processing making it suitable for customer premises equipment and access applications.

References:

[1] www.freescale.com, *Reference Manuals and Application Notes for the Power QUICC™ II Pro Family of Communications Processors*

[2] David E. Culler, Jaswinder Pal Singh - *Parallel Computer Architecture. A Hardware / Software Approach*, Morgan Kaufmann Publishers, 1999

[3] Radia Perlman, *Interconnections. Bridges, Routers, Switches, and Internetworking Protocols*, Second Edition, Addison-Wesley, 2000

[4] David E. McDysan, Dave Paw, *ATM and MPLS Theory and Application. Foundations of Multi-Service Networking*, McGraw Hill/Osborne, 2002