A PROGRAMMABLE GENERAL PROTOCOL PROCESSOR - A PROPOSAL FOR AN EXPANDABLE ARCHITECTURE

Seppo Virtanen[1], Jouni Isoaho[2], Tomi Westerlund[2] and Johan Lilius[3]

[1]University of Turku (Laboratory of Electronics and Information Technology), Turku Centre for Computer Science
FIN-20014 Turku, Finland, E-mail seppo.virtanen@utu.fi
[2]University of Turku (Laboratory of Electronics and Information Technology)
[3]Åbo Akademi University (Department of Computer Science), Turku Centre for Computer Science

Abstract - We present an architecture proposal for a programmable microprocessor optimized for communications protocol processing.

Connections between networked devices require fast processing of different types of protocols and in turn the protocol processing requires fast hardware. Until recently, protocol processing has been done with ASICs or other custom pre-programmed hardware. In this paper, we present an architecture for a programmable microprocessor with communications protocol processing optimized design.

Protocol processing exhibits certain distinct characteristics from one protocol to another that can be taken advantage of when designing an architecture and functional subunits for a programmable general protocol processor. By examining the ATM, SDH, IP and IEEE 802.11 MAC protocols [2, 3] we have found that these protocols share a common need for certain protocol processing tasks. These tasks are listed in Table 1 [1, 2].

Table 1. Common Protocol Processing Tasks in ATM, SDH, IP and IEEE 802.11 MAC.

Task	Description
ERROR CHECKING	CRC and other (e.g. IP header checksums) error checking calculations. Must be a programmable function, e.g. CRC polynomials need to be changed. Also needed for calculating checksums for outgoing protocol data units.
RANDOM NUMBER GENERATION	Generation of statistically independent pseudo-random bitstrings. Random values of several different word sizes are needed.
TIMING AND COUNTERS	Updating of a system clock and several counters. The counters need to be programmable so that they can be reset, set to certain values, incremented and decremented either directly or at certain time periods.
SCALING	Bitwise shifting and scaling as well as the changing of word lengths. Also needed for synchronization with the incoming bitstream.
PACKET ASSEMBLY/ DISASSEMBLY	Assembling and disassembling of protocol data units. In the assembly process, the outgoing data may need to be split into multiple parts and each part needs to be given a header and possibly a trailer. In the disassembly process headers and possible trailers are removed from the protocol data units and the original data is reconstructed. The incoming data units may need to be reordered in the process.
STATE MACHINE CONTROL	Flow control for taking care of message passing etc. between protocol operation state machines
BITSTRING MATCHING	Matching bitstrings in the incoming data to bitstrings in the memory, and performing certain actions based on the result of the matching. Also needed for analyzing the headers in incoming protocol data units. When sending protocol data units, matching provides some of the information required to construct a protocol data unit header.

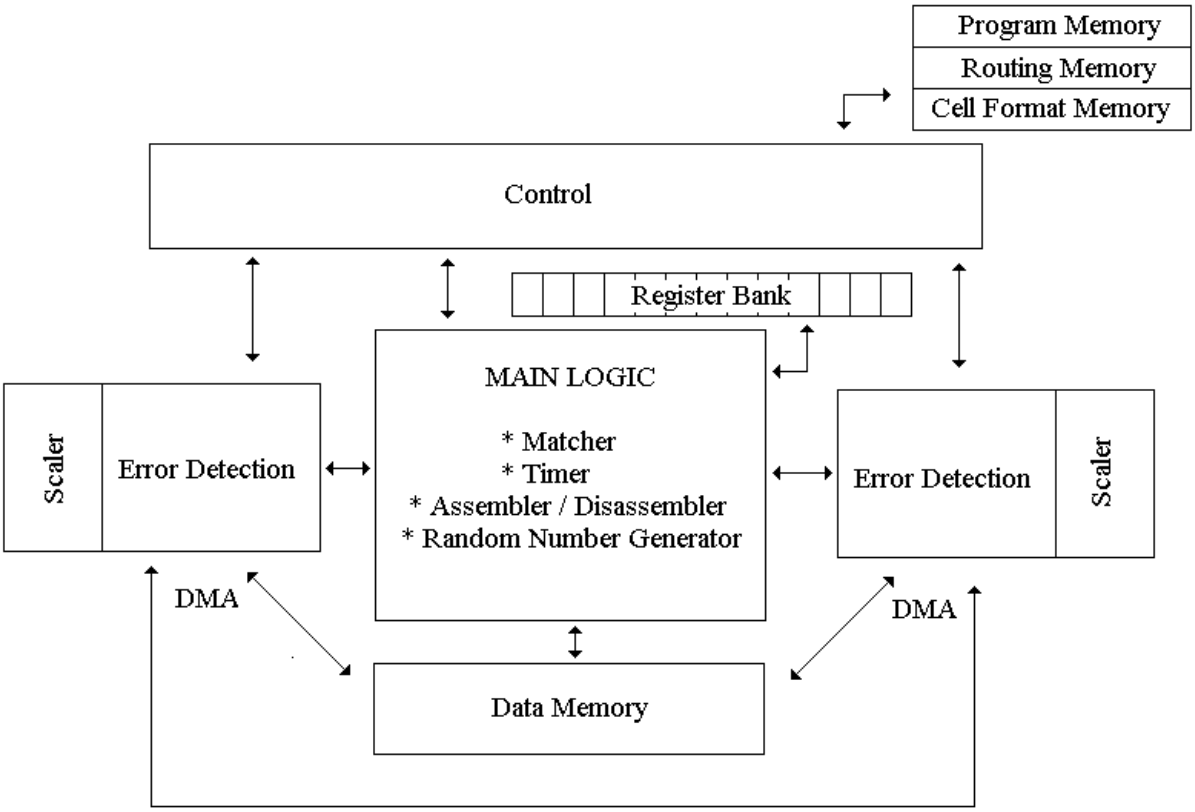

Figure 1. Architecture Block Diagram for a General Protocol Processor.

The architecture we suggest for a programmable general protocol processor is shown in figure 1. It takes into account the mentioned hardware requirements of protocol processing while maintaining the flexibility of a programmable microprocessor. The architecture features a control unit that handles the program flow according to information in its program, routing and cell format memory. The control unit directly controls the other functional blocks of the processor. There are also two Error Detection/Scaler blocks for parallel processing of input and output data streams.

REFERENCES

[1] A. Jantsch, J. Öberg and A. Hemani, "Is there a Niche for a General Protocol Processor Core?", Proceedings of the 16th IEEE NORCHIP Conference, pp. 93-100, Lund, Sweden, November 1998.

[2] S. Virtanen, "On Communications Protocols and their Characteristics Relevant to Designing Protocol Processing Hardware", to appear in TUCS Technical Report series, Turku Centre for Computer Science, Turku, Finland, September 1999.

[3] "IEEE Std 802.11-1997, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications." The Institute of Electrical and Electronics Engineers, Inc., New York, USA, June 1997.

Presented in URSI/IEEE XXIV Convention on Radio Science, Turku, Finland, 4-5 October 1999. Informo No. 181, Tuorla Observatory Reports, pp. 112-113, ISBN 951-29-1531-6