

VLSI Implementation of Forward Error Control Technique for ATM Networks

G. Padmavathi, R. Amutha, and S.K. Srivatsa

In asynchronous transfer mode (ATM) networks, fixed length cells of 53 bytes are transmitted. A cell may be discarded during transmission due to buffer overflow or a detection of errors. Cell discarding seriously degrades transmission quality. The quality degradation can be reduced by employing efficient forward error control (FEC) to recover discarded cells. In this paper, we present the design and implementation of decoding equipment for FEC in ATM networks based on a single parity check (SPC) product code using very-large-scale integration (VLSI) technology. FEC allows the destination to reconstruct missing data cells by using redundant parity cells that the source adds to each block of data cells. The functionality of the design has been tested using the Model Sim 5.7cXE Simulation Package. The design has been implemented for a 5×5 matrix of data cells in a Virtex-E XCV 3200E FG1156 device. The simulation and synthesis results show that the decoding function can be completed in 81 clock cycles with an optimum clock of 56.8 MHz. A test bench was written to study the performance of the decoder, and the results are presented.

Keywords: ATM network, forward error control, virtual circuit (VC), virtual path (VP), automatic repeat request (ARQ), BER, VLSI, VHDL.

Manuscript received Nov. 20, 2004; revised July 20, 2005.

G. Padmavathi (phone: +91 4111 262321, email: gpadma@svce.ac.in) is with ECE Department, Sri Venkateswara College of Engineering, India.

R. Amutha (email: ramutha@svce.ac.in) is with IT Department, Sri Venkateswara College of Engineering, India.

S.K. Srivatsa (email: profsks@annauniv.edu) is with the School of ECE, Anna University-MIT Campus, India.

I. Introduction

The data paths of asynchronous transfer mode (ATM) networks are composed of fiber optic links with very low bit error rates (BER). Resource sharing by a large number of high-speed, fluctuating-rate data streams is bound to cause congestion and buffer over flow. Cells are discarded at such times.

In ATM networks, the header (5 bytes) is checked for errors at transit nodes with the aid of a cyclic redundancy check (CRC) – 8 code contained in the header (last byte of the header) itself. The header format is shown in Fig. 1. If a header is found to be in error, the cell is discarded. Cell discarding seriously degrades transmission quality and is considered one of the major problems in ATM-based networks. Hence, an efficient implementation of forward error control (FEC) for ATM networks is required. Using very-large-scale integration (VLSI) technology, the delay

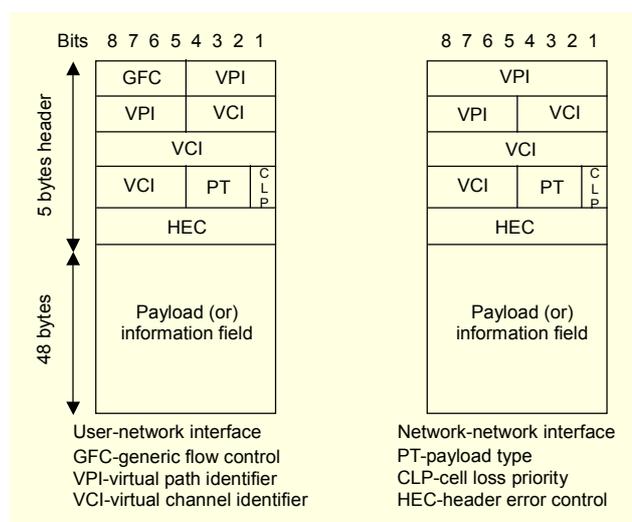


Fig. 1. ATM cell header.

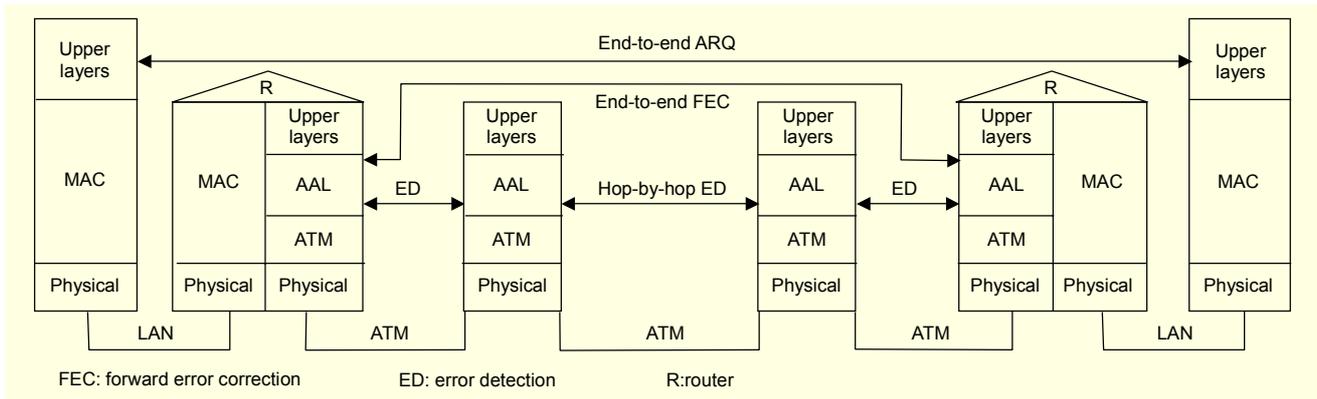


Fig. 2. Illustration of the hybrid ARQ/FEC scheme in LAN/ATM interconnection.

and size of the encoding and decoding equipment can be reduced and the power dissipation can be minimized.

In [1], the performance of a hybrid automatic repeat request (ARQ)/FEC coding scheme over ATM networks has been analyzed. The scheme proposes FEC to be applied to the virtual circuits (VCs) of an ATM network (between end routers) and ARQ to be applied end-to-end (between end systems), while cells are checked for errors at the virtual path (VP) terminals (hop-by-hop, that is, at each router) as shown in Fig. 2. For FEC, the coding matrix technique in [2] has been adopted.

In [2], a cell loss recovery method to be applied to the VPs of ATM networks has been proposed. The method consists of cell-loss detection (CLD) and lost cell regeneration. The data cells are arranged in a matrix of size $(M-1) \times (N-1)$. Each row of data cells is terminated by a specially designed cell, called a CLD cell, while each column is terminated by a parity cell formed based on a single parity check code. The CLD cell is generated from the virtual circuit identifier (VCI) and the sequence number (SN) fields of all cells in that row. The SN is assigned in the ATM adaptation layer (AAL) as the first four bits of the header of this layer, which forms the first four bits of the payload field of the ATM cell in the ATM layer, as shown in Fig. 3. The header of the CLD cell is the same as that of the data cells in that row, and a CRC-24 is also attached at the end. Thus, the design of a CLD cell is a time-consuming operation during the encoding process at the transmitter side. The scheme in [2] was slightly modified in [3] to extend the row size of the coding matrix and thus make

the scheme more robust to burst cell loss.

In our design, we have proposed using the sequence number attached to each cell in the ATM adaptation layer (AAL) instead of using a CLD cell for the same purpose as suggested in [1]. This is possible because the hybrid ARQ/FEC scheme was proposed to be applied to VCs and not to VPs of the ATM network. Hence, the complex design of CLD cells can be eliminated, thereby making the encoding process faster. Since the sequence number assigned in AAL is of modulo-16, the size of the row is limited to 16.

This paper proposes an FEC based on a single parity check (SPC) product code technique. SPC codes are some of the most popular error detecting codes because of their ease of implementation. Product codes have the proper structure for burst error correction without the need for extra interleaving. Interleaving is used in communication networks to transform a burst error into a random error, which then can be corrected by FEC codes. The proposed method consists of cell loss detection and lost cell regeneration. It is intended to be applied to VCs of the ATM network. When there is no cell loss, the decoder checks for bit errors and corrects them to the possible extent, as described in section IV.

The rest of the paper is organized as follows.

An encoding scheme is described in section II. The design of decoding equipment is described in section III. Section IV describes the performance of the designed equipment. Implementation and results are discussed in section V. Section VI presents our conclusions.

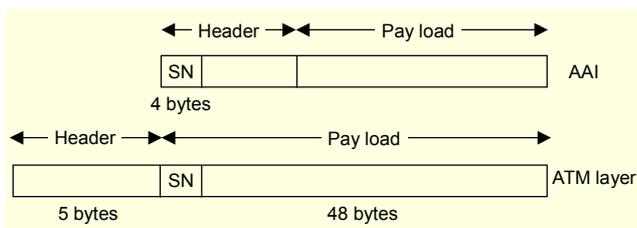


Fig. 3. Illustration of an SN field attached in AAL.

II. Encoding

The coding matrix, that is, the structure of the SPC product code, is shown in Fig. 4.

The data cells are arranged in a matrix of size $(M-1) \times (N-1)$. To every row of $(N-1)$ data cells of 53 bytes each, the encoding equipment adds a vertical parity cell, which is also of 53 bytes

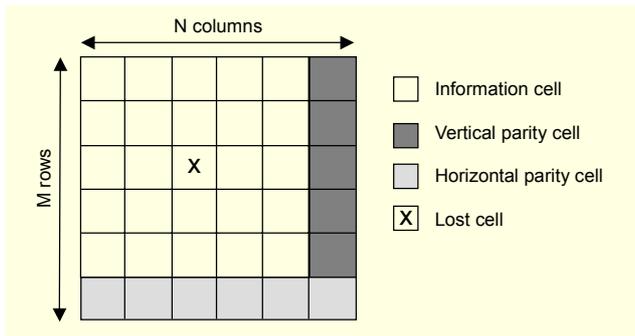


Fig. 4. Structure of SPC product code.

(424 bits) in length. The i -th bit of the vertical parity cell of a particular row is given by

$$C_{N,i} = \left(\sum_{j=1}^{N-1} C_{j,i} \right) \text{ mod } (2) \quad i = 1, 2, 3, \dots, 424, \quad (1)$$

where $C_{N,i}$ is the i -th bit of the N -th column of a particular row.

Similarly, a horizontal parity cell is added to each column of $(M-1)$ data cells that is generated as

$$C_{M,i} = \left(\sum_{j=1}^{M-1} C_{j,i} \right) \text{ mod } (2) \quad i = 1, 2, 3, \dots, 424, \quad (2)$$

where $C_{j,i}$ is the i -th bit of the j -th cell.

$C_{M,i}$ is the i -th bit of the M -th row of a particular column.

The parity cells are also assigned sequence numbers separately by the encoding equipment.

III. Decoding

The received cells are arranged in $M \times N$ matrix form in the decoder at the receiving side.

Decoding is performed in two steps. Lost cells are detected by row-wise operations with the aid of the sequence number attached in AAL. A gap in the sequence number indicates a loss of cell. Next, the lost cells are recovered by either row-wise or column-wise operations with the aid of vertical and horizontal parity cells, depending on whether a single cell is lost or a burst of cells is lost.

A block diagram of the decoder at the receiver side is shown in Fig. 5. The decoder has been designed for a 5×5 matrix of data cells as shown in Fig. 5.

The serial-in parallel-out (SIPO) block converts the bits received serially into a byte-wide parallel form. The bytes are stored in the storage buffer (store_buf) until the entire 53 byte ATM cell is received. Once a full ATM cell is received, the cell is transferred to the matrix (Mat) module, where the cells (data and parity) are arranged in a 6×6 matrix. The 6th column contains the vertical parity cells and the 6th row contains the

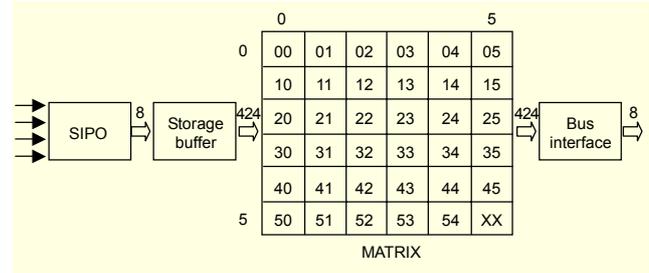


Fig. 5. Block diagram of the decoder.

horizontal parity cells. After the matrix module is processed for cell loss recovery or bit error correction, which is described in detail in the following section, the cells are given to the bus interface unit which strips off the vertical and horizontal parity cells and delivers only the data cells to the receiver. The INT signal is used in the matrix block to synchronize the data write operation with the operation of the store buffer block. An ARQ signal indicates that the error correction is incomplete and an ARQ is needed in the upper layer.

IV. Performance Analysis

1. Single Cell Loss Recovery

The received cells are arranged in a matrix of size $M \times N$ in the decoder at the receiving side. The cells are arranged row-wise. When a cell is received in the matrix block of the decoder, first its sequence number is checked. Modulo-16 numbering is followed for the assignment of a sequence number in the AAL of an ATM network. A gap in the sequence number indicates a cell loss, and hence a dummy cell with the expected sequence number in the 4-bit SN field and all other bits as zero is generated in place of the gap. After all the $(N-1)$ data cells and the VPC of that row are received, the dummy cell is replaced by the result of a bit-wise modulo-2 addition of all the cells in that row including the parity cell.

2. Recovering a Burst of Lost Cells

The coding scheme presented here can recover a burst of missing cells of any length less than or equal to $N-1$. Such recovery is done with the aid of column parity, that is, horizontal parity cells only. In row parity, the vertical parity cells are used to recover additional missing cells scattered over the array.

If a row suffers only a single missing cell, that cell can be recovered after the last cell of the row is received. On the other hand, if two or more cells are missing in a row, they must wait until their respective columns are fully received. If two or more cells are missing in a row and in the corresponding columns, then the row and column processing can be repeated two or more times until the cells are recovered to the greatest possible extent. In each repetition, at least one cell can be recovered. In our implementation,

the row and column processing is done only once.

When the recovery process terminates completely or partially, any remaining lost cells can be informed to the upper layer (by the ARQ signal) where ARQ takes place.

3. Handling Bit Errors

Although bit errors are rare in ATM networks, they nevertheless do occur and must be anticipated. If a single bit error occurs in a row, in which no cell is missing, the bit in the vertical parity that is in the same position in the cell as the erroneous bit shows a mismatch. The intersection of the column and row identifies the bit in error, thereby allowing the decoding equipment to correct it. It is possible to correct multiple single-bit errors occurring at different rows and different columns. Depending on the position of the errors, some or all errors can be corrected.

A test bench was written to analyze the performance of the designed decoder. The cell loss recovery performance and the bit error correction performance of the decoder are shown in Tables 1 and 2, respectively.

The following cases have been analyzed to study the cell loss recovery performance:

1. A single cell loss located anywhere in the matrix is 100% recoverable.
2. A single cell loss per row is 100% recoverable even if all the cell loss is in the same column, as shown in Fig. 6. Recovery of the lost cell in a particular row is done after the last cell of that row is processed.
3. A single cell loss per column is 100% recoverable even if there are two or more cells missing in the same row, as shown in Fig. 7. If two or more cells are missing in a row, they must wait until their respective columns are fully processed.
4. A single burst of cell loss of any length $\leq N-1$ is 100% recoverable, as shown in Fig. 8.
5. In the case of multiple random cell loss, only certain cells are recoverable, others are unrecoverable. Figure 9 shows an unrecoverable pattern of cell loss.

Table 1. Cell loss recovery.

Type of cell loss	Percentage of recovery
Single cell loss (located anywhere in the matrix)	100
Single cell loss per row	100
Single cell loss per column	100
Single burst of cell loss (length $\leq N-1$)	100
Multiple random cell loss	Depending upon the positions of the lost cells, some or all can be recovered.

Table 2. Bit error correction.

Type of error	Percentage of correction
Single bit error per column with odd number of bit errors in a row	100
Multiple single bit errors	Depending on the position of bit errors, some or all can be corrected.

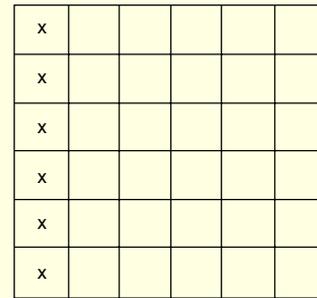


Fig. 6. Single cell loss per row.

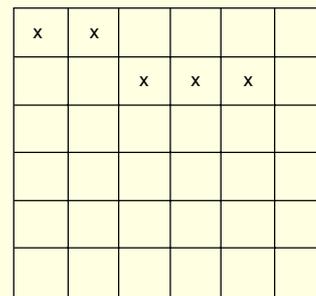
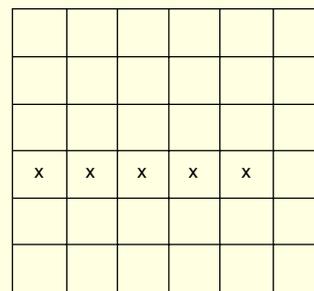


Fig. 7. Recoverable pattern of single cell loss per column.



x- Lost cell

Fig. 8. Single burst of cell loss.

V. Implementation and Results

The decoder has been designed using the very high speed integrated circuit (VHSIC) hardware description language (VHDL), which is a powerful tool aiming especially at high

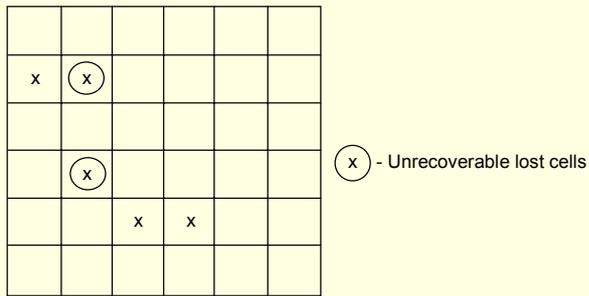


Fig. 9. Multiple random cell loss.

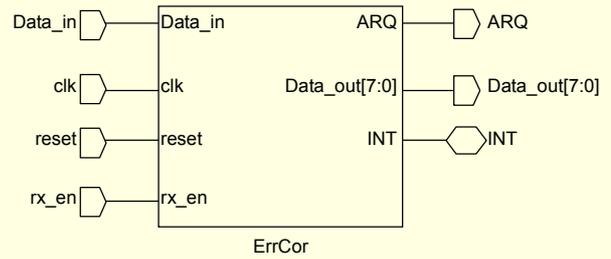


Fig. 11. Pin diagram of the decoder.

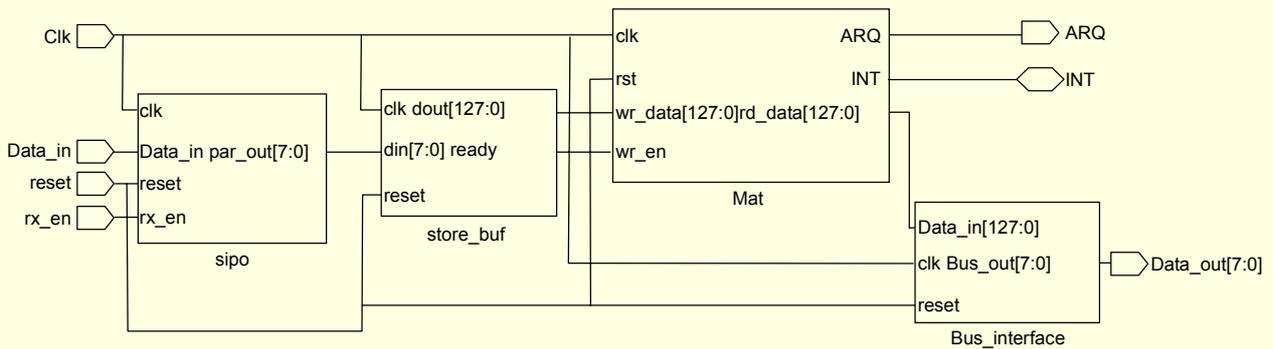


Fig. 10. RTL schematic of the decoder.

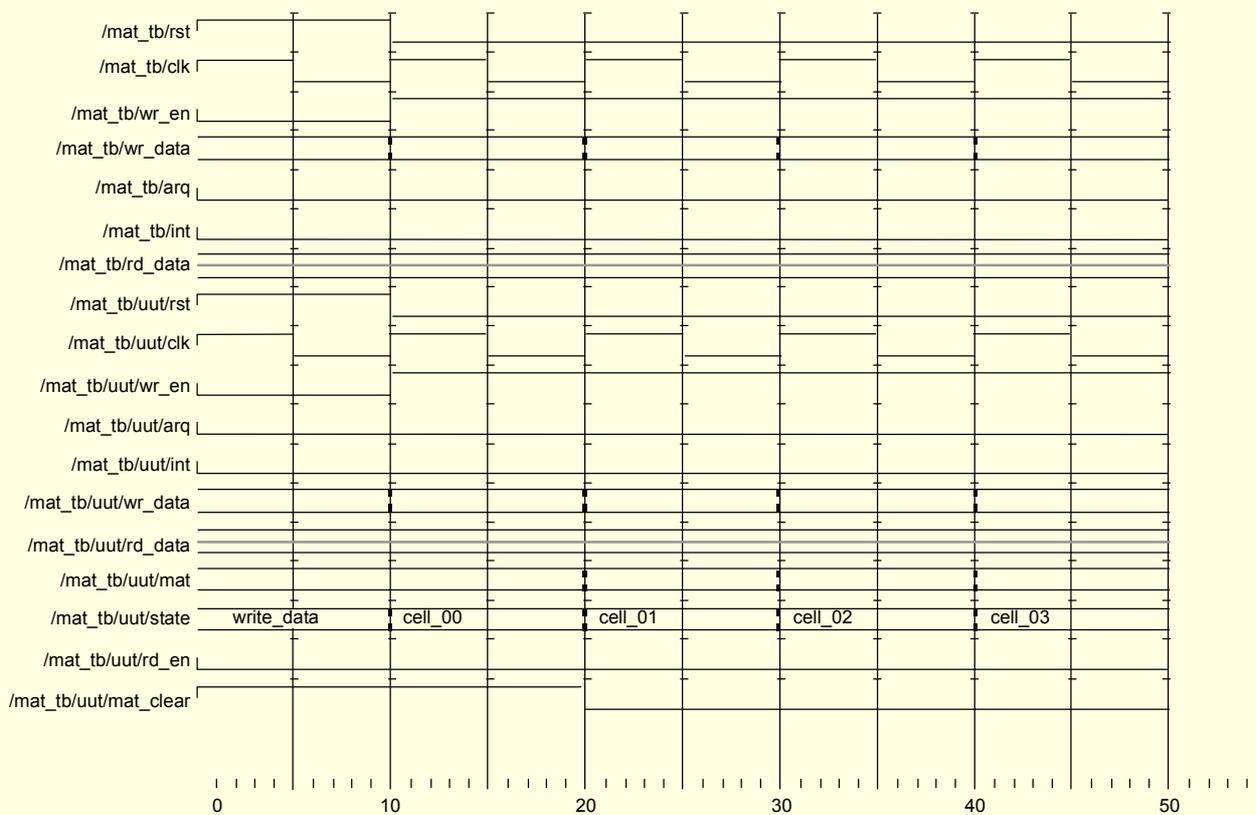


Fig. 12. A portion of the simulation results.

Table 3. Device utilization for V 3200E FG1156.

Resource	Used	Available	Utilization
IOs	14	804	1.74%
Function generators	22220	64896	34.24%
CLB slices	11110	32448	34.24%
Dffs or latches	5511	67308	8.19%

level abstractions, portability, and design automation. Since VHDL allows a design to be independent of design tools and technology, or of the vendor of the end product, a circuit can be designed and archived in VHDL and later fabricated with the most advanced technology. The functionality of the decoder is tested using a Model Sim 5.7c XE simulator, and synthesis has been done targeting the Virtex-E: XCV 3200E FG1156 device. Virtex-E devices feature a flexible, regular architecture that comprises an array of configurable logic blocks (CLBS) surrounded by programmable input/output blocks (IOBs), all inter connected by a rich hierarchy of fast, versatile routing resources. Virtex-E FPGAs are SRAM-based and are customized by loading configuration data into internal memory cells. The register transfer level (RTL) schematic and pin diagram of the decoder, generated by the synthesis tool, are shown in Figs. 10 and 11.

A portion of the simulation results is presented in Fig. 12. The simulation results show that the decoding function would require 81 clock cycles with an optimum clock frequency of 56.8 MHz (from the synthesis report).

VI. Conclusions

ATM technology is gaining popularity in high-speed networks, and hence an efficient implementation of FEC is required. In this paper, the design of decoding equipment based on an SPC product code has been presented for a 5×5 matrix of data cells. Since the size of the matrix taken for data cells is 5×5 , the overhead (the redundant bits attached to the data cells for FEC) is 30 percent. The percentage of overhead can be reduced by increasing the matrix size. The design has been implemented in a Virtex-E XCV 3200E FG1156 device. The simulation and synthesis results show that the decoding function would require 81 clock cycles with an optimum clock frequency of 56.8 MHz, that is, 1.426 micro seconds. This timing is suitable if the data rate of the ATM network is 155.52 Mbps or less. For ATM networks with a data rate of 622.08 Mbps, the decoding function should still be speeded up by modifying the design to have an additional matrix block (the two matrix blocks should be used alternately) and by specifying the speed constraint in the VLSI implementation process. The cell loss recovery and bit error correction performances of the decoder are also presented.

References

- [1] M.A. Kousa, A.K. Elhakeem, and Hui Yang, "Performance of ATM Networks under Hybrid ARQ/FEC Error Control Scheme," *IEEE/ACM Trans. on Networking*, vol. 7, no. 6, Dec. 1999, pp. 917-925.
- [2] H. Ohta and T. Kitemi, "A Cell Loss Recovery Method Using FEC in ATM Networks," *IEEE J. on Selected Areas of Comm.*, vol. 9, Dec. 1991, pp. 1471-1483.
- [3] H.T. Lim and J.S. Song, "Cell Loss Recovery Method in B-ISDN/ATM Networks," *Electronic Lett.*, vol. 31, no. 11, May 1995, pp. 848-851.
- [4] N. Schacham, "Packet Recovery in High Speed Networks Using Coding and Buffer Management," *Proc. of IEEE INFOCOM' 90*, 1990, pp. 124-131.



G. Padmavathi received the BE in electronics and communication engineering from Madurai Kamaraj University, Madurai, India in 1985 and the MS in electronics and control from Brila Institute of Technology and Science, Pilani, India in 1993. Since 1993 she has been on the Faculty in the Electronics and Communication Engineering Department of Sri Venkateswara College of Engineering, Sriperumbudur, India. Her research interests are channel coding, system design using microprocessors and microcontrollers and FPGA design technology. She has published papers in reputed journals/conferences.



R. Amutha received the BE in electronics and communication engineering from Madurai Kamaraj University, Madurai, India and the ME in applied electronics from Bharathiar University, Coimbatore, India. She has been on the Faculty in Information Technology Department of Sri Venkateswara College of Engineering, Sriperumbudur, India from 1993. Her areas of interest are in coding theory and high speed networks. She has published papers in international conferences and journals. She has also submitted her research thesis for a PhD degree in Anna University, Chennai, India.



S.K. Srivatsa received the BE in electronics and telecommunication engineering from Jadavpur University, the ME in electrical communication engineering and the PhD from Indian Institute of Science, Bangalore, India. He had been a Professor of electronics engineering in Anna University, Chennai, India for nearly 20 years. He is the author of 191 publications in reputed journals/conference proceedings. His research interests include computer networks, digital logic design, design of algorithms, and robotics.