

# Reprint

## VIRUS: A Pseudo-Framing Method for Cell-Based Interfaces

*T. Antonakopoulos, A. Maniatopoulos, A. Pantazi and V. Makios*

Computer Communications

---

VOL. 26 NO. 2, FEBRUARY 2003 pp 129-142

---

**Copyright Notice:** This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted or mass reproduced without the explicit permission of the copyright holder.



ELSEVIER

Computer Communications 26 (2003) 165–176

computer  
communications

[www.elsevier.com/locate/comcom](http://www.elsevier.com/locate/comcom)

# VIRUS: a pseudo-framing method for cell-based interfaces

Theodore Antonakopoulos\*, Alex Maniatopoulos, Aggeliki Pantazi, Vassilios Makios

*Department of Electrical Engineering and Computers Technology, University of Patras, 26500 Rio Patras, Greece*

Received 11 July 2001; revised 2 May 2002; accepted 2 May 2002

## Abstract

This paper presents the coding scheme and the timing recovery method of an interface for high-speed cell-based networks. The synchronization technique used in this interface is based on the functionality resulting from combining functions of the fiber channel with the basic features of pure asynchronous transfer mode. The main advantage of this interface is its implementability at high transmission rates, since most of its functions, like cell delineation and cell header error detection and correction, are performed at byte-level by using a pseudo-framing structure. The paper presents the basic principles of this interface, the analytic formulas for calculating its mean synchronization time under various traffic conditions and describes how the various parameters affect the method's performance.

© 2002 Elsevier Science B.V. All rights reserved.

*Keywords:* Asynchronous transfer mode; VIRUS interface; Idle order set

## 1. Introduction

The introduction of the asynchronous transfer mode (ATM) technology in the previous decade changed the way broadband and local area networks are being developed and how various user requirements and services are supported. All ATM-like cell-based networks transmit data by relaying fixed length user cells (UCs) using various transmission media, formats and speeds. For synchronization purposes many cell delineation mechanisms have been proposed [1], but the most interesting one, which is used in standard ATM networks, is based on the header error check (HEC) code. This mechanism is independent to the used line code and of any special data pattern, but malicious simulation of HEC in the cells' payload results to incorrect synchronization. Another method is based on the use of start-of-cell commands (with or without scrambler reset) before the transmission of each cell [2].

ITU and the ATM forum developed a number of physical layer interfaces [3], having transmission rates in the range of 1.544–622.08 Mbps, using either PDH, Sonet/SDH, Block Coded or Clear Channel interfaces (CCI) and using either copper (STP, UTP or Cu) or fiber (MM or SM). The 25.6 Mbps over twisted pair cable interface [2] uses 4B5B block coding, based on 16 valid data nibbles and one escape code (X). Cell delineation is achieved by preceding each ATM cell with 1

byte that consists of two valid commands (X\_X or X\_4). SDH STM-1/Sonet STS-3c at 155.52 Mbps [4] and SDH STM-4c/Sonet STS-12c at 622.08 Mbps [5] use SDH frames for transporting ATM cells, while Sonet framing is used at the 51.84 Mbps UTP-3 physical layer interface [3]. Finally, CCIs are non-framed interfaces and are mostly used for LAN connections. The 25.6 Mbps CCI, sometimes called mid-range user network interface (MUNI), uses the same physical signaling as token-ring and cells are sent asynchronously, while CCI at 622.08 Mbps uses STM-4c framing without using the management channels [6]. Although the work presented in this paper is related to user network interfaces (UNIs), it is worth to mention that ATM is also used in network-to-network interfaces (NNIs) at speeds of 2.5 Gbps (OC-48) and 10 Gbps (OC-192). Recently, the ATM forum has approved its '2.4 Gbps physical layer' for SONET/SDH support over six different single mode fiber interfaces and the '622 and 2488 Mbps cell-based physical layer' for supporting ATM cells directly on the physical media without using any frame structure [7,8].

In this paper, a new interface for high-speed cell-based communication systems is presented, emphasizing on its coding and synchronization functions. This interface is called VIRUS and its name comes from a special type of idle cells that it uses, which are called idle order sets (IOS).<sup>1</sup> The interface is able to support point-to-point communications for exchanging data cells, to provide the services

\* Corresponding author. Tel.: +30-61-997-286; fax: +30-61-997-342.  
E-mail address: [antonako@ee.upatras.gr](mailto:antonako@ee.upatras.gr) (T. Antonakopoulos).

<sup>1</sup> The acronym IOS means 'virus' in Greek.

defined by ITU for the physical layer and to be easily implemented in high-speed networks. A drawback of pure ATM is that the cell delineation mechanism for determining the cell boundaries is performed at the bit data rate. The VIRUS interface uses a different approach in order to get synchronization by exploiting the line code characteristics. The VIRUS interface was developed to support high-speed point-to-point interfaces and to be connected to other ATM systems with the minimum possible complexity.

The VIRUS physical layer has to provide to the ATM layer the services determined by the respective ITU recommendations, to operate at much higher cell rates and to be implementable with commercially available components and programmable devices. The maximum transmission rate in each interface was estimated to be 800 Mbps and the nodes to be interconnected using multi-mode fiber links [9].

## 2. The VIRUS interface

When high transmission rates are used in communication channels, the implementation of the physical layer functions becomes difficult and costly, especially when these functions are performed at the line data rate. Recovering byte timing using a framing structure decreases the processing requirements, but adds more complexity to the interface structure. In order to minimize its complexity, we use the notion of pseudo-frames in the VIRUS interface. As a pseudo-frame is defined, the information transmitted between two control groups of bytes plus the first control group. Before proceeding with the pseudo-frames of VIRUS, its coding scheme is presented.

Since the main requirement was to support the basic ATM functions at high speeds, various coding schemes and transmission methods were considered. The fiber channel (FC) technology was the most promising in developing the physical layer. The disadvantage of using the 8B/10B code of FC in a cell-based environment, where single bit error correction is required, is that due to its coding structure, two unacceptable conditions may be met. First, an error bit in the transmitted serial stream may be spread into two bits in the decoded data, making the single error correction capability of TC sub-layer non-applicable. Second, a code violation may be obtained erroneously, due to an error occurred in a previous bit position, which altered the running disparity of the bit stream, but did not result in a detectable error at the respective transmission character [10,11]. Therefore it was finally decided that the VIRUS interface has to use a different code, based on the combination of two block codes. The user data are encoded using the 4B1C code [12, 13], while the 8B/10B code is used only for control and synchronization purposes. The 4B1C code is applied twice to each data byte, while the idle cells, which are used at the physical layer for cell rate decoupling and cell delineation, are constructed as a sequence of K28.5 and D21.5 (FC

terminology) bytes [11]. Following this coding scheme, the FC requirement that the maximum run-length of the encoded data must not exceed five bits is satisfied and the correct phase relationship of the extracted clock with the received data is maintained.

ATM uses idle cells for rate decoupling between adjacent nodes. The ATM idle cells are 53 bytes long and their length is equal to the length of the UCs. For rate-decoupling purposes, the VIRUS interface uses a smaller group of bytes, which is called IOS. Each IOS is composed of two identical words, each word consisting of four characters, two K28.5 special characters and two D21.5 data characters, as shown in Fig. 1(a). K28.5 is the FC special character used as the first character in each control word. The VIRUS receiver uses the K28.5 characters for detecting byte boundaries before acquiring cell boundaries. Since the 4B1C coding is used for the user data, the K28.5 character contains code violation and, before achieving synchronization, it may be detected erroneously in the incoming serial bit stream, as the result of a combination of two consecutive data bytes, as it is shown in Fig. 1(b). In order to overcome this false sync case, two consecutive K28.5 characters have been used in the IOS structure. The D21.5 characters have been selected to conform to the FC idle words structure and are used mainly for maintaining the clock synchronization. The two D21.5 characters are not used in the cell delineation procedure, while other patterns can be used to support operation, administration and maintenance (OAM) functions.

As it will be shown in the synchronization method description, the IOS length must be equal or greater to the cell header length, which is 5 bytes long, and therefore the IOS was determined to be eight characters long. The IOSs are also used for cell rate decoupling. They are removed at the receiver side in order to absorb the differences of the actual clocks.

The VIRUS physical layer is subdivided into two sub-layers. The lower sub-layer, which is called VIRUS channel adapter (VCA), is related to the FC functionality and deals with physical medium aspects. The upper sub-layer, which is called transmission convergence (TC), deals with the cell stream multiplexing/demultiplexing, cell header error detection and correction, IOS insertion and extraction and coding/decoding of data. VCA is based on the FC architecture and uses the FC-0 functions of the FC (transmission media adaptation, transmitters, receivers and their interfaces) and a combination of 8B/10B and 4B1C transmission codes. The physical layer uses bit and word boundary synchronization functions and contains the required functionality for adapting ATM cells into VIRUS pseudo-frames (VPF). As was mentioned previously, a VPF is considered as the number of UCs contained between two IOSs including the first IOS. Fig. 2 shows an example of VIRUS transmissions and the respective pseudo-frames. The minimum length of a pseudo-frame is the length of an IOS, since the next IOS starts a new pseudo-frame, while the maximum length depends on the rate decoupling

IOS Structure : <K28.5><K28.5><D21.5><D21.5><K28.5><K28.5><D21.5><D21.5>

Special Character	<K28.5>	<D21.5>
Binary Equivalent	0 0 1 1 1 1 1 0 1 0	1 0 1 0 1 0 1 0 1 0

(a)

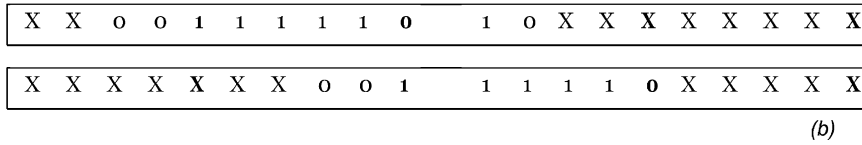


Fig. 1. The IOS structure (a) and two examples of data words resulting to erroneous K28.5 character detection (b).

mechanism, which determines the maximum number of UCs between two IOSs.

2.1. The VIRUS transmitter

In the transmit direction, the physical layer accepts complete cells from the ATM layer and generates the fifth byte (HEC) of the cell header. Then the cell passes through the VIRUS interface, which generates the appropriate VPF format and multiplexes the ATM cells in its structure. The IOSs are generated in the TC sub-layer when there are no cells available for transmission or when the number of UCs transmitted after the last IOS has reached a predetermined value.

Fig. 3 shows the VIRUS interface state diagram for the transmit direction. It includes a Tx-state counter,  $c_H$ , which measures the number of UCs transmitted since the last transmitted IOS. Initially  $c_H$  is set to zero. When a UC is transmitted, the counter increases. When the counter reaches its maximum value  $k$ , the UC buffer service is ceased, an IOS is transmitted and the Tx-state counter is cleared. The value of  $k$  is related to the clock accuracy used in the VIRUS interface, since a higher  $k$  value requires a more accurate clock. As it will be shown in the performance analysis section the value of  $k$  affects the interface performance and is a compromise between system performance and system cost.

2.2. The VIRUS receiver

At the receiving side, the IOSs are detected and removed at the TC sub-layer, where the cell delineation

mechanism is implemented based on the HEC method and on the ‘indication of IOS’. A byte boundary detector is used which operates under the control of the cell delineation mechanism. The boundaries detector accepts encoded serial data for performing clock recovery and is capable of detecting K28.5 characters for performing timing resynchronization. Whenever the resynchronization section of the boundary detector is active and a K28.5 character has been detected, an indication is provided to the TC sub-layer synchronizer. When character synchronization is achieved, the boundary detector is deactivated for avoiding interpretation of specific patterns in the user data. When resynchronization is required, the character boundary detector is reactivated. As it is shown in Fig. 4, initially the synchronizer of the cell delineation mechanism is in the HUNT state and the resynchronization part of the boundaries detector is activated for detecting K28.5 characters. Whenever a valid IOS is detected, the synchronizer changes to the PRESYNC state, the characters’ boundaries are considered detected and the synchronizer validates both the cell headers and the IOS structure. The synchronizer uses the Rx-state counter,  $c_k$ , for monitoring the cell delineation procedure. When the receiver is in the HUNT state, the value of the Rx-state counter is not used, while the Rx-state counter becomes zero when the receiver enters into the PRESYNC state. At this state, the Rx-state counter increases by one or two, whenever a correct UC header or a valid IOS is detected.

This procedure continues until the receiver enters in the SYNC state and the Rx-state counter becomes equal to  $k$ . Since the UCs and the IOSs have different lengths, two

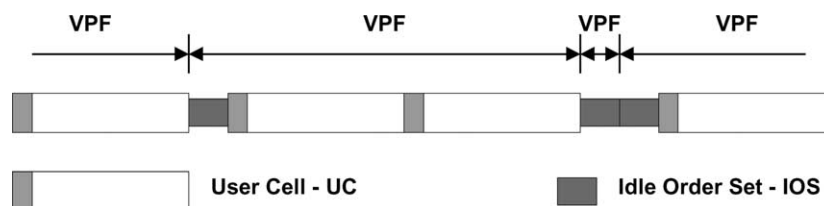


Fig. 2. VIRUS pseudo-frame examples.

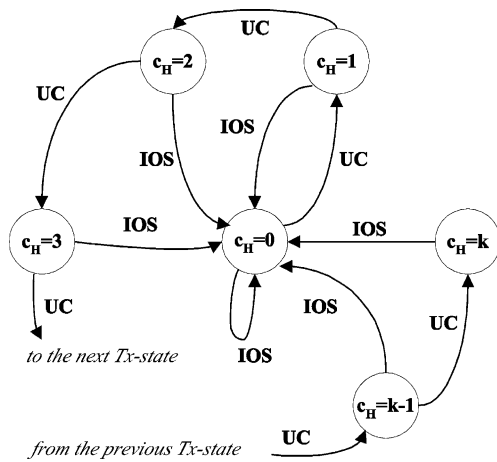


Fig. 3. The VIRUS Tx-state diagram.

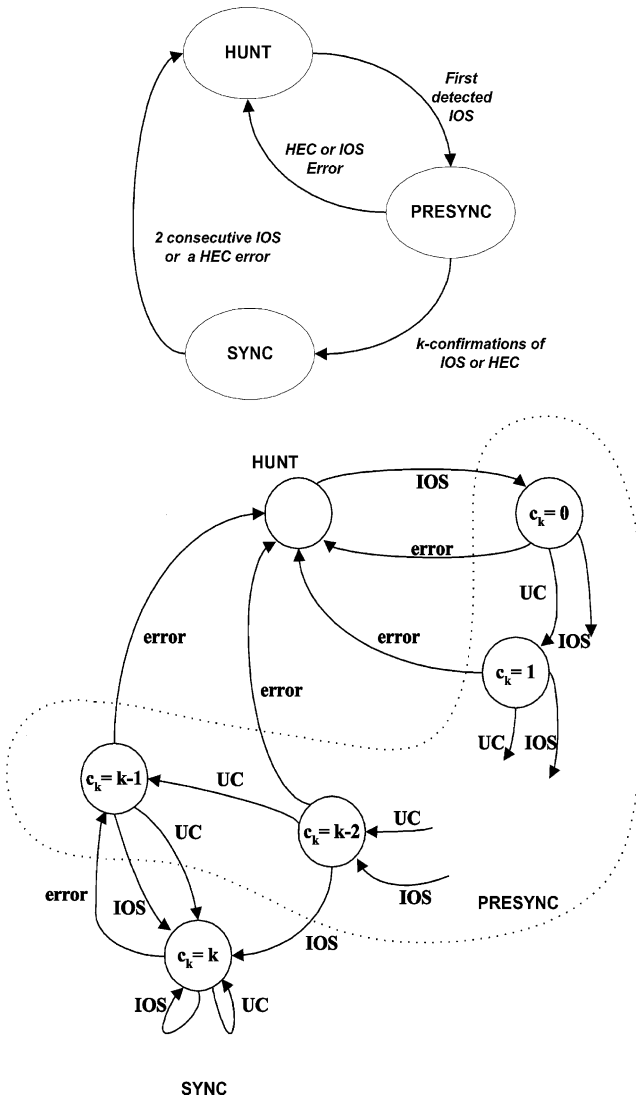


Fig. 4. The VIRUS Rx-state diagram.

validation modules are triggered, the HEC module and the IOS module. The HEC module detects the existence of a cell by validating its fifth byte using the CRC method, while the IOS module detects an IOS by matching the last eight characters with the IOS structure. If a valid indication is available at the end of the eighth character, a new cell or IOS boundary is determined and then, the synchronizer waits for a new boundary to restart the boundary validation procedure. If the synchronizer is in the PRESYNC state, the internal counter is increased once or twice, depending on the valid indication. A UC results to one time increase, while an IOS to two times increase, since it contains two FC transmission words. If no valid indication is available at the end of the eighth character, the validation procedure is restarted. If no valid indication is given for a second time, the validation procedure is cancelled, the internal counter is cleared, the character boundary detector is re-activated and the cell delineation mechanism goes to the HUNT state. When cell or IOS boundaries have been confirmed and the internal Rx-state counter gets its maximum value, the synchronizer switches to the SYNC state. These validation procedures are performed in parallel by using the character stream, so high processing rate can be achieved, especially compared to the pure ATM cell delineation procedure, which is performed at the serial bit stream.

In the SYNC state, the synchronization state machine continues to validate the cell boundaries, following a two-steps procedure. In each valid boundary, the previously described procedure is performed for estimating the new boundary. If no valid indication is received at the end of the eighth character of this validation, the validation procedure is restarted immediately, the internal counter is decreased but the system remains in the SYNC state. At the same time, a new cell or IOS boundary is estimated, assuming that errors occur in an IOS with higher probability than in a cell header. This is due to the single bit error correction mechanism used for protecting the cell header. If the new cell header or IOS validation procedure does not give a positive result, it is assumed that an error occurred into a UC and the next cell boundary is estimated to be 37 ( $53 - 2 \times 8$ ) bytes after the second erroneous validation procedure. If the new cell header or IOS validation procedure does not give a positive result, the Rx-state counter is cleared, the cell delineation mechanism goes to the HUNT state and the character boundaries detector is activated. The rest of this paper presents the performance analysis of the proposed method and describes how the method's performance is affected by the traffic conditions.

### 3. Analysis of the synchronization procedure

For the evaluation of the VIRUS synchronization method, the mean value of the synchronization time has been studied. For the rest of this paper, the following terminology is used:

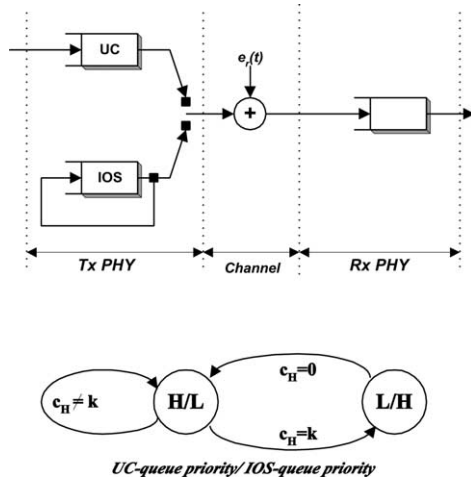


Fig. 5. The VIRUS model and the Tx PHY state transition diagram.

- $T_{UC}$  the transmission time of an UC,
- $T_{IOS}$  the transmission time of an IOS,
- $\lambda_{UC}$  the rate of UC arrivals,
- $\lambda_{IOS}$  the rate of IOS generations,
- $R$  the channel data rate in bytes/s,
- $\bar{t}_{HUNT}$  the mean value of the time the system spends in the HUNT state,
- $\bar{t}_{PRESYNC}$  the mean value of the time the system remains in the PRESYNC state,
- $\bar{t}_{SAT}$  the mean value of the time needed to acquire synchronization after desynchronization has been detected,
- $\bar{t}_{i,IOS}$  the mean value of the time needed for the transmission of the next IOS when the transmitter is in the  $i$ -state,
- $P_i$  the probability the transmitter is in the  $i$ -state at a given instant,
- $p_e$  the probability the UC-buffer is empty,
- $m_{UC}$  the length of a user cell (in characters),
- $m_{IOS}$  the length of an IOS cell (in characters),
- $k$  the maximum number of user cells transmitted between two IOSs,
- $c_H$  the value of the Tx-state counter,
- $c_k$  the value of the Rx-state counter,
- $n_i$  the number of cells in the UC-queue at the  $i$ -departure epoch.

The evaluation of the VIRUS synchronization method is based on the model shown in Fig. 5. Although the actual interface has only one buffer for the UCs, in this model we consider that there are two buffers in the transmit direction, one for UC and the other for IOS cells. The second buffer is used to model the IOS generation procedure. According to this model, an IOS is always available for transmission. Whenever an IOS cell is transmitted in the channel, it is also copied back to the IOS-buffer. The length of the IOS-buffer is equal to  $m_{IOS}$  characters.

The receiver's behavior is analyzed by considering the transmitter state and the channel behavior. The UC-queue has priority over the IOS queue as long as the value of the Tx-state counter,  $c_H$ , is lower than its maximum value. Whenever the Tx-state counter gets its maximum value, the IOS-queue has the highest priority for transmitting the next cell, regardless of the number of cells in the UC-queue. On the other hand, no message interruption is allowed during transmission. So, the model uses non-preemptive priorities, which are not constant but toggle, depending on the Tx-state counter value. Fig. 5 also shows the system priorities according to the value of the Tx-state counter. The Tx-state counter takes values from 0 to  $k$ , following the state diagram shown in Fig. 3. Although the  $c_H$  counter has  $k + 1$  states, two of them are its basic states. The 0-state ( $c_H = 0$ ) indicates that the last transmitted cell was an IOS cell, while the  $k$ -state ( $c_H = k$ ) implies that the next transmitted cell must be an IOS cell. Assuming that the system was in the  $i$ -state at the last departure, its state at the end of the next departure depends on the value of  $c_H$  and on the number of cells in the UC-queue before the last departure. If the UC-queue was empty, the system goes back to the 0-state, otherwise changes to the  $(i + 1)$ -state, assuming that  $i < k$ .

The system analysis relies on the properties of the embedded Markov chain, where the Markov chain is embedded at cell transmission epochs. It is assumed that UCs arrive in the UC-queue independent to the system state, they follow a Poisson process with  $\lambda_{UC}$  mean value and are  $m_{UC}$  characters long. In this case, the system can be described by two state equations. The first state equation is valid whenever there are cells in the UC-queue and the Tx-state counter has not reached its maximum value:

$$n_{i+1} = n_i - 1 + \alpha_1, \quad c_{i+1} = c_i + 1 \quad \text{for } c_i < k \text{ and } n_i > 0 \quad (1)$$

where  $c_i$  is the value of the Tx-state counter and  $a_1$  is the number of UCs that arrive during an UC transmission.

If the Tx-state counter has its maximum value and there are cells in the UC-queue, or if the UC-queue is empty, then the second state equation is valid:

$$n_{i+1} = n_i + \alpha_2, \quad c_{i+1} = 0 \quad \text{for } c_i = k \text{ or } n_i = 0 \quad (2)$$

where  $\alpha_2$  is the number of UCs that arrive during an IOS cell transmission.

Before using Eqs. (1) and (2) for obtaining the analytic expressions of system's behavior, we will use the state diagram of Fig. 3 for calculating the various state probabilities. Since the transmission of an IOS cell forces the system to the 0-state, the system goes to the next state only when there is at least one UC in the UC-buffer, otherwise the system remains in the 0-state and a new IOS is transmitted. Thus, assuming that the system is in steady

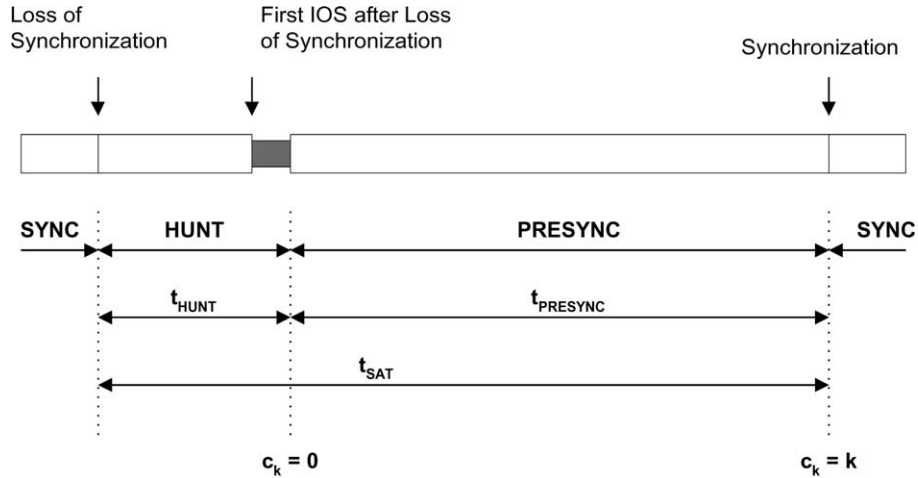


Fig. 6. The synchronization procedure timing diagram.

state conditions, the following equalities are valid:

$$\begin{aligned}
 &P[i - \text{state at the current departure}] \\
 &= P[(i - 1) - \text{state at the previous departure}] \\
 &\quad \times P[\text{a UC was transmitted}] \quad 1 \leq i \leq k
 \end{aligned}$$

and

$$\begin{aligned}
 &P[0 - \text{state at the current departure}] \\
 &= P[i - \text{state at the previous departure} / 0 \leq i \\
 &\leq k] P[\text{UC queue is empty}] + P[k \\
 &\quad - \text{state at the previous departure}]
 \end{aligned}$$

Therefore, the system equilibrium equations are:

$$P_i = P_{i-1}(1 - p_e) \quad P_0 = \sum_{i=0}^{k-1} P_i p_e + P_k \quad (3)$$

and by solving the system equilibrium equations, it is obtained that:

$$P_i = \frac{p_e(1 - p_e)^i}{1 - (1 - p_e)^{k+1}} \quad \forall i \in [0, k] \quad (4)$$

As described previously, there are two distinct events in the system dynamics:

UC transmission:  $c_i < k$  and  $n_i > 0$  and IOS transmission:  $c_i = k$  or  $n_i = 0$ .

By defining that  $\Pi_I$  is the probability that an IOS is served and  $\Pi_U$  is the probability that a UC is served, then

$$\Pi_I \triangleq p_e \sum_{i=0}^k P_i + P_k(1 - p_e) = \frac{p_e}{1 - (1 - p_e)^{k+1}} \quad (5)$$

and

$$\Pi_U = 1 - \Pi_I = \frac{(1 - p_e)[1 - (1 - p_e)^k]}{1 - (1 - p_e)^{k+1}} \quad (6)$$

The probability that the UC-buffer is empty,  $p_e$ , depends on the arrival process of UCs and on the parameter  $k$ . The IOS arrival process, which is equal to the departure process from the IOS queue, depends on the UCs arrival process, since the system transmits data continuously, and therefore:

$$\lambda_{UC} m_{UC} + \lambda_{IOS} m_{IOS} = R \quad (7)$$

The probability that an IOS is served is equal to the ratio of the bandwidth occupied by the transmitted IOS cells to the total bandwidth, thus

$$\Pi_I = \frac{\lambda_{IOS} m_{IOS}}{R} = 1 - \frac{\lambda_{UC} m_{UC}}{R} \quad (8)$$

By using Eqs. (5) and (8), the following equation can be used for calculating the probability  $p_e$  that the UC-buffer is empty:

$$1 - (1 - p_e)^{k+1} - p_e \frac{R}{R - \lambda_{UC} m_{UC}} = 0 \quad (9)$$

### 3.1. Synchronization time

As it is shown in Fig. 6, the synchronization acquisition time,  $t_{SAT}$  is given by  $t_{SAT} = t_{HUNT} + t_{PRESYNC}$  and thus

$$\bar{t}_{SAT} = \bar{t}_{HUNT} + \bar{t}_{PRESYNC} \quad (10)$$

since  $t_{HUNT}$  and  $t_{PRESYNC}$  are independent variables. We will calculate these two mean values as functions of the probability the UC-queue being empty ( $p_e$ ) and the system state parameter  $k$ . By definition, the mean value of the HUNT-time is equal to the mean value of inter-IOS times,



that is:

$$\bar{t}_{\text{HUNT}} = \sum_{i=0}^k P_i \bar{t}_{i,\text{IOS}} + (1 - P_0)(T_{\text{UC}} - T_{\text{IOS}}) \quad (11)$$

where

$$\bar{t}_{i,\text{IOS}} = T_{\text{IOS}} + \left[ 1 - (1 - p_e)^{k-i} \right] \frac{(1 - p_e)}{p_e} T_{\text{UC}} \quad (12)$$

$$\forall i \in [0, k]$$

as it can be derived from the Tx-state diagram of Fig. 3. The second part of Eq. (11) is related to the time required for starting the resynchronization process, if synchronization is lost during a UC. By using Eqs. (4) and (12) in Eq. (11),  $\bar{t}_{\text{HUNT}}$  is calculated as:

$$\bar{t}_{\text{HUNT}} = \frac{[p_e^2 T_{\text{IOS}} + (1 - (1 - p_e)^k (1 + k p_e (1 - p_e))) T_{\text{UC}}]}{p_e (1 - (1 - p_e)^{k+1})} \quad (13)$$

Since the UCs transmitted between two IOS cells belong to the same pseudo-frame, and up to  $k$  UC belong to a frame, the number of positive indications that have to be received for entering the SYNC state is  $k + 1$ ,  $k$  during the PRESYNC state and 1 for the IOS detection in the HUNT state. Therefore, the maximum length pseudo-frame is sufficient for achieving system synchronization.

Fig. 4 shows the state diagram the receiver follows when it is in the PRESYNC state. When the system enters the PRESYNC state, the Rx-state counter becomes zero. Whenever an error is detected, either in a UC or an IOS, the system returns to the HUNT state. In order to calculate the mean value of the time the system stays in the PRESYNC state,  $\bar{t}_{\text{PRESYNC}}$ , the methodology used in the HUNT state has to be followed.

As it was described previously, the receiver enters into the SYNC state whenever its Rx-state counter becomes equal to  $k$ . Therefore, the Rx-state counter value during the

PRESYNC state is given by:

$$c_{k,i+1} = c_{k,i} + 1 \quad (14)$$

if a correct HEC indication was received

or

$$c_{k,i+1} = c_{k,i} + 2 \quad \text{if a correct IOS indication was received} \quad (15)$$

When the receiver enters into the SYNC state, the following equation is valid:

$$c_{k,i+1} \geq k \begin{cases} \text{if } c_{k,i} = k - 1 \text{ following a correct HEC indication} \\ \text{either } c_{k,i} = k - 2 \text{ following a correct IOS indication} \\ \text{or } c_{k,i} = k - 1 \text{ following a correct IOS indication} \end{cases} \quad (16)$$

Under heavy traffic conditions, the IOS that forces the receiver to the PRESYNC state, is followed by up to  $k$ -UC, while in light load, it is followed by other IOSs. That gives the minimum and maximum values of  $t_{\text{PRESYNC}}$ :

$$\left[ \frac{k}{2} \right] T_{\text{IOS}} \leq t_{\text{PRESYNC}} \leq k T_{\text{UC}} \quad (17)$$

Due to the cell delineation mechanism, when the system is in the PRESYNC state, the reception of an IOS is the result of an empty UC-buffer condition. Assuming that  $k_1$  and  $k_2$  are the UCs and IOSs, respectively, which arrived when the receiver was in the PRESYNC state, then  $k_1 + 2k_2 \geq k$ . We define as  $P_j[m]$  the probability that  $m$  IOSs have been used in the PRESYNC state so that  $c_k = j$ . As it is shown in Fig. 4, the probability that  $k_2$  IOSs have been used for achieving synchronization is equal to:

$$P_k[k_2] = (1 - p_e) P_{k-1}[k_2] + p_e (P_{k-2}[k_2 - 1] + P_{k-1}[k_2 - 1]) \quad k_2 > 0 \quad (18)$$

But for  $j < k$  and  $j \geq 2k_2$ :

$$P_j[k_2] = \frac{(j - k_2)!}{(j - 2k_2)! k_2!} (1 - p_e)^{j-2k_2} p_e^{k_2} \quad (19)$$

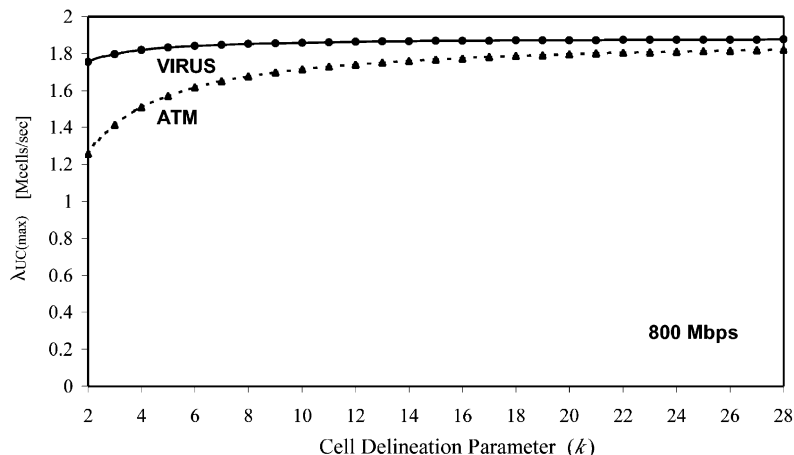


Fig. 7. The maximum supported cell rate as a function of  $k$ .



thus

$$P_k[k_2] = \begin{cases} \frac{(k - k_2)!}{(k - 2k_2)!(k_2 - 1)!} (1 - p_e)^{k-2k_2} p_e^{k_2} \left( \frac{1}{k_2} + \frac{(1 - p_e)}{(k - 2k_2 + 1)} \right) & k_2 > 0 \\ (1 - p_e)^k & k_2 = 0 \end{cases} \quad (20)$$

while the respective duration of the PRESYNC state is given by:

$$T_{k_2} = \begin{cases} [(k - 2k_2)T_{UC} + k_2 T_{IOS}] (1 - P_{k-1}[k_2 - 1]) + [(k - 2k_2 + 1)T_{UC} + k_2 T_{IOS}] P_{k-1}[k_2 - 1] = (k - 2k_2 + P_{k-1}[k_2 - 1])T_{UC} + k_2 T_{IOS} & k_2 > 0 \\ kT_{UC} & k_2 = 0 \end{cases} \quad (21)$$

The  $\bar{i}_{PRESYNC}$  time is given by:

$$\bar{i}_{PRESYNC} = \begin{cases} \frac{k+1}{2} p_e^{(k+1)/2} T_{IOS} + \sum_{k_2=0}^{(k-1)/2} P_k[k_2] T_{k_2} & \text{if } k \text{ is odd} \\ \sum_{k_2=0}^{k/2} P_k[k_2] T_{k_2} & \text{if } k \text{ is even} \end{cases} \quad (22)$$

Due to the cell delineation mechanism and to the assumption that the normalized offered load is always less or equal to 1, the maximum value of  $\lambda_{UC}$  is obtained by:

$$(\max)\lambda_{UC} \leq \frac{kR}{km_{UC} + m_{IOS}} \quad (23)$$

By using Eqs. (10), (13), and (22), the mean synchronization time can be calculated for various system parameters. At the end of this section various numerical results are presented based on actual system parameters.

### 3.2. VIRUS pseudo-frame length

As it was mentioned previously, the VPF is defined as the number of UCs contained between two IOSs, plus the first IOS. The minimum length of VPF is the length of an IOS,

while the maximum length depends on the rate decoupling mechanism, which determines the maximum number of UCs between two IOSs. Thus, if  $l_{VPF}$  is the VPF length, then:

$$m_{IOS} \leq l_{VPF} \leq km_{UC} + m_{IOS} \quad (24)$$

The mean length of the VPF is calculated by:

$$\begin{aligned} \bar{l}_{VPF} &= m_{IOS} \\ &+ \sum_{i=0}^k P(\text{the pseudo-frame consists of } i \text{ UC-cells}) i m_{UC} \\ &= m_{IOS} + m_{UC} \frac{(1 - p_e)[1 - (1 - p_e)^k]}{p_e} \end{aligned} \quad (25)$$

Thus, the portion of a pseudo-frame that is used for useful data transmission, called pseudo-frame utilization,  $U_{frame}$ , is defined as the ratio of the useful data transmitted versus the total transmitted data and is given by:

$$U_{frame} = \frac{(1 - p_e)[1 - (1 - p_e)^k] m_{UC}}{p_e m_{IOS} + (1 - p_e)[1 - (1 - p_e)^k] m_{UC}} \quad (26)$$

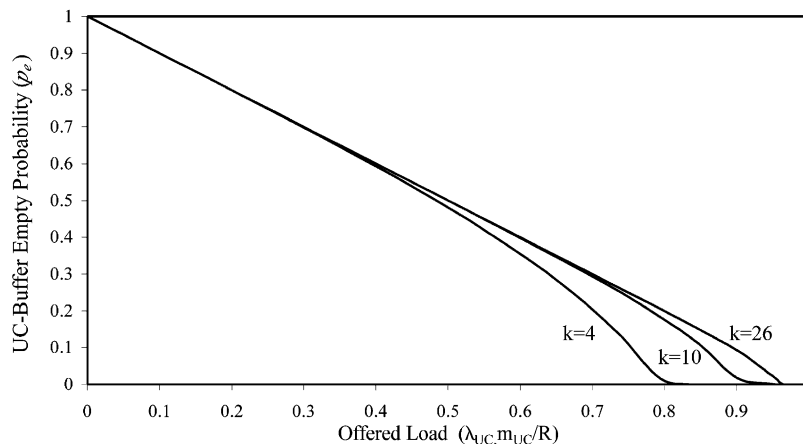


Fig. 8. The probability that the UC-buffer is empty as a function of the offered cell rate.

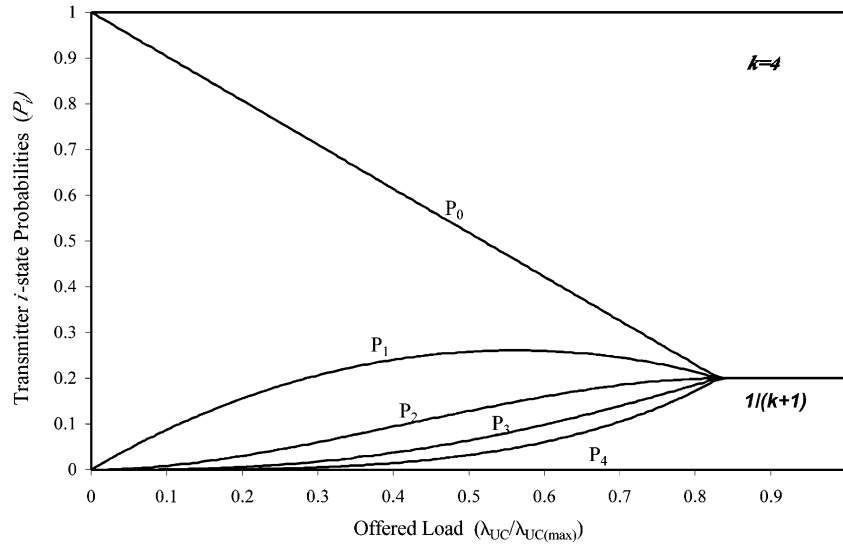


Fig. 9. The VIRUS Tx-state probabilities.

#### 4. Numerical results

The rest of this paper analyses the various VIRUS parameters and how they affect the system performance. ATM PHYs use a cell rate decoupling function that inserts one idle cell every 26 cells received from the ATM layer and that results to 96.3% efficiency of the available data rate (the Sonet framing overhead is not included). There is an option that ATM layer may insert unassigned cells to perform cell rate decoupling, which is not required in the VIRUS interface, as this function is done at the physical layer by inserting the IOS cells. Fig. 7 shows how the maximum rate of UC cells is affected by the cell delineation mechanism (the value of  $k$ ), for VIRUS and ATM interfaces at 800 Mbps. Based on the figure, VIRUS will support the same maximum

cell rate with existing ATM systems ( $k = 26$ ), when  $k = 4$ , while using  $k = 26$  for VIRUS, the maximum supported cell rate increases by 3.4%. The cell delineation mechanism used in all ATM interfaces is independent to the actual transmission rate and the relative performance for the maximum supported cell rate shown in Fig. 7, also applies at higher transmission rates.

Fig. 8 shows how the probability of finding the UC-buffer empty is affected by the offered cell rate and the cell delineation mechanism. This probability is used to relate the offered cell rate to all parameters used for measuring the performance of VIRUS, according to Eqs. (11)–(26).

Fig. 9 shows the Tx-state probabilities for various traffic conditions. When the traffic is low, the probability of generating an IOS is high, since the UC-buffer becomes

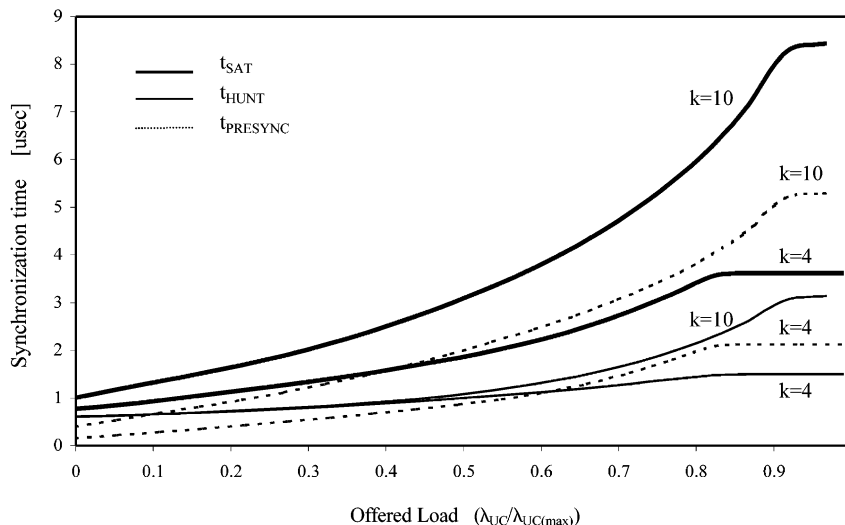


Fig. 10. The synchronization time and its components versus the traffic load and the cell delineation parameter ( $k$ ).

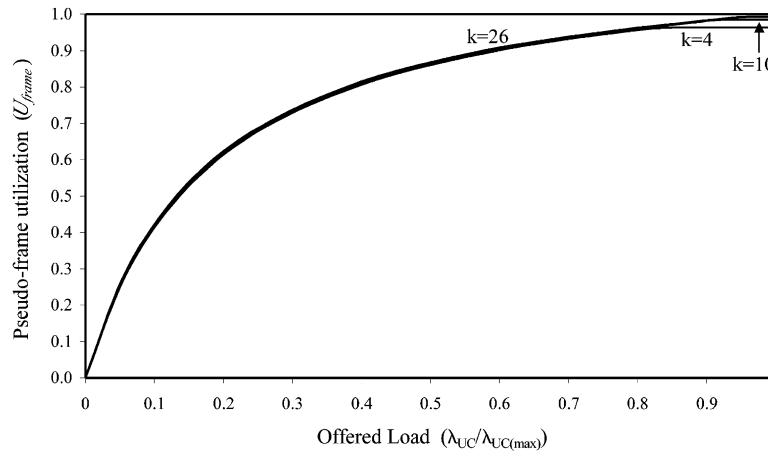


Fig. 11. The VIRUS pseudo-frame utilization,  $U_{\text{frame}}$ .

empty frequently and then the system passes from state-0. As the traffic increases, the other state probabilities increase but still IOSs are generated since the UC-buffer becomes empty. Only when the offered load becomes greater than 80% of the maximum supported cell rate, IOSs are generated because the transmitter is forced to transmit an IOS irrespective if there are any UCs available. When the offered load approaches 1, all state probabilities approach  $1/(k+1)$  since the transmitter uses all states for generating its output traffic.

For measuring the performance of the proposed synchronization method, the resynchronization acquisition time is used. Resynchronization acquisition time is the time elapsed from the detection of loss of synchronization (entering the HUNT state) up to the acquisition of full synchronization (entering the SYNC state). In ATM systems, the synchronization time has a fixed value and is equal to seven times the cell duration, thus it is  $3.71 \mu\text{s}$  at 800 Mbps (assuming that no transmission errors occurred during the synchronization procedure). In the proposed method the synchronization

time is not fixed and depends on the user offered load. Fig. 10 shows how the offered load influences the total synchronization time and its components for  $k=4$  and 10. The synchronization time depends strongly on the traffic conditions and its mean value approaches the value of  $1.5kT_{\text{UC}}$  when the offered load approaches its maximum value. The PRESYNC time contributes stronger than the HUNT time, since during the PRESYNC state the receiver has to wait for one or more pseudo-frames for getting the proper number of valid indications, while the HUNT state starts at the desynchronization point, which is uniformly distributed during a pseudo-frame.

Using various values of  $k$  and Eqs. (9) and (26), the pseudo-frame utilization  $U_{\text{frame}}$ , is shown in Fig. 11. As the offered load remains low, IOS cells are generated since the UC-buffer is empty; no IOS cell is generated due to the cell delineation mechanism, thus less than  $k$  UC cells are used in each pseudo-frame. As the offered load increases, the probability of finding the UC-buffer empty decreases, the IOS insertion mechanism is

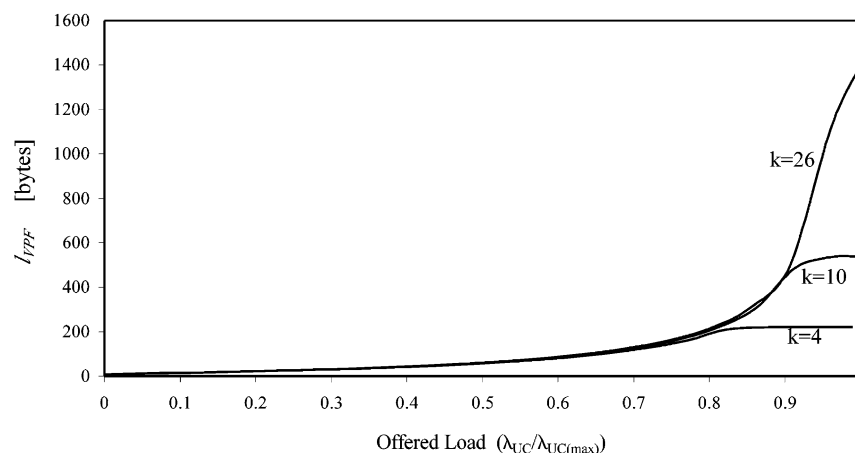


Fig. 12. The mean pseudo-frame length versus the traffic load.

activated and the number of UC cells used in each pseudo-frame increases.

Finally, the mean pseudo-frame length as a function of the offered load is shown in Fig. 12. As the offered load increases, more UC cells are used in each pseudo-frame. When the offered load approaches its maximum value, the pseudo-frame length approaches  $T_{IOS} + kT_{UC}$ , its maximum value. The above results show how the VIRUS interface performance is affected by the cell delineation mechanism.

## 5. Conclusions

In this paper, a new interface called VIRUS for use in cell-based systems was presented. The VIRUS interface supports the implementation of various functions (cell delineation, cell header error detection and correction, synchronization etc.) for transmitting a cell stream using the basic functions of the FC. The use of two block codes decreases the physical layer complexity by minimizing the number of functions performed at the channel bit rate and allows the implementation of all other functions at much lower speed. The presented performance analysis showed how the VIRUS transmitter and receiver behave and how their performance is affected by the cell delineation mechanism.

## References

- [1] M. de Prycker, Asynchronous Transfer Mode: Solution for Broadband ISDN, Ellis Horwood Ltd, Chichester, UK, 1991.
- [2] The ATM Forum. Physical Interface Specifications for 25.6 Mbps over Twisted Pair Cable, af-phy-0040.000, 1995.
- [3] H.J.R. Dutton, P. Lenhard, Asynchronous Transfer Mode—ATM, Prentice-Hall RTP, Upper Saddle River, USA, 1995.
- [4] The ATM Forum. ATM Physical Medium Dependent Interface Specification for 155 Mb/s over Twisted Pair Cable, af-phy-0015.000, 1994.
- [5] ITU-T Recommendation I.432. B-ISDN User Network Interface—Physical Layer Interface Specification, November, 1994.
- [6] The ATM Forum, Cell-Based Transmission Convergence Sublayer for Clear Channel Interfaces, af-phy-0043.000, November, 1995.
- [7] The ATM Forum, 2.4 Gbps Physical Layer Specification, af-phy-0133.000, October, 1999.
- [8] The ATM Forum, 622 and 2488 Mbps Cell-Based Physical Layer, af-phy-0128.000, July, 1999.
- [9] A. Maniatopoulos, T. Antonakopoulos, V. Makios, Design and Implementation of a New Synchronization Method for High-Speed Cell-based Network Interfaces, The IEEE Conference on Computer Design—ICCD'96, Austin, TX, 1996, October.
- [10] P.A. Franaszek, A.X. Widmer, Byte Oriented DC Balanced 8B/10B Partitioned Block Transmission Code, USA Patent No. 4,486,739, December, 4, 1984.
- [11] ANSI, Fiber Channel—Physical and Signaling Interface (FC-PH) Rev. 3.0. ANSI X3T9 I/O Interface, Working Draft, June, 1992.
- [12] Y. Takasaki, M. Tanaka, N. Maeda, K. Yamashita, K. Nagano, Optical pulse formats for fiber optical communications, IEEE Transactions on Communications COM-24 (4) (1976) 404–413.
- [13] N. Yoshikai, K. Katagiri, T. Ito, mBIC code and its performance in an optical communication system, IEEE Transactions on Communication COM-32 (2) (1984) 163–168.

**Theodore Antonakopoulos** was born in Patras, Greece in 1962. He received the Electrical Engineering Diploma degree in 1985, and his PhD degree in 1989 from the Department of Electrical Engineering at the University of Patras, Patras, Greece. In September 1985, he joined the Laboratory of Electromagnetics at the University of Patras participating in various R&D projects for the Greek Government and the European Union, initially as a research staff member and subsequently as the senior researcher of the Communications Group. Since 1991 he has been on the faculty of the Electrical Engineering Department at the University of Patras, where he is currently an Associate Professor. His research interests are in the areas of data networks with emphasis on performance analysis, efficient hardware implementation and rapid prototyping. He has more than 70 publications in the above areas and is actively participating in several R&D projects of the European industry. Dr Antonakopoulos is a Senior member of the IEEE, serves in the Program Committee of various International Conferences and Workshops and is a member of the Technical Chamber of Greece.

**Alex Maniatopoulos** was born in Athens, Greece in 1970. He received his Engineering Diploma degree in 1992 from the School of Electrical Engineering at the University of Patras, Patras, Greece. In September 1992, he joined the Laboratory of Electromagnetics at the University of Patras working towards his PhD and participating in various R&D projects. His research covers all aspects of the Physical layer, the Baseband and Medium Access Control mechanisms for wired and wireless local area networks, including architecture, modeling, analysis, design and implementation. He received his PhD in 1997. He is currently working as a Product Development Director in the Multimedia and Communication Department of Atmel Corporation, Patras, Greece. His research and development interests include architecture, prototyping and system-on-chip design for networking and communication applications. Dr Maniatopoulos has 12 publications in the above areas and he is a member of the IEEE and the Technical Chamber of Greece.

**Aggeliki Pantazi** was born in Athens, Greece in 1973. She received the Electrical Engineering Diploma degree in 1996 from the School of Electrical Engineering at the University of Patras, Patras, Greece. Since 1997 she is a research engineer in the Laboratory of Electromagnetics at the University of Patras, participating in various R&D projects for the Greek Government and the European Union. Her research interests are in the areas of communication networks with emphasis on performance analysis and protocols development. Ms Pantazi is a member of the Technical Chamber of Greece.

**Vassilios Makios** was born in Kavala, Greece. He received his Electrical Engineering degree (Dipl. Ing.) from the Technical University in Munich, Germany in 1962 and his PhD degree (Dr Ing.) from the Max Planck Institute for Plasmaphysics and the Technical University in Munich in 1966. From 1962–67, he was a Research Associate in the Max Planck Institute for Plasmaphysics in Munich, where he was associated with microwave interaction studies of plasmas. He served as Assistant Professor in 1967–70, Associate Professor in 1970–73 and Full Professor in 1973–77 in the Department of Electronics, Carleton University of Ottawa, Canada, where he was involved with teaching and research in microwave and optical communications, radar technology, remote sensing and CO<sub>2</sub> laser development. From 1977, he is an honorary Research Professor of Carleton University. Since 1976 he has been Professor of Engineering and Director of the Laboratory of Electromagnetics in the Electrical Engineering Department of the University of Patras, Greece, where he is involved in teaching and research in microwave and optical communications, data communications networks, LANs, MANs, and B-ISDN with emphasis on efficient hardware implementations and rapid prototyping. He was also involved in research in photovoltaic systems. He has published over 140 papers and holds numerous patents in the above fields. He has participated in the organizing committees of numerous IEEE and European Conferences and was the Technical Program Chairman of the 5th Photovoltaic European Community Conference in Athens 1983 and Co-Chairman of the EURINFO 1988 Conference of the European Community. He is the recipient of the silver and golden medal of the German Electrical Engineering Society (VDE). He is a senior member of the IEEE, member of the Canadian Association of Physicist, the German Physical Society and the VDE, Professional Engineer of the Province of Ontario and member of the Greek Technical Chamber.