VLSI Systems Design for 51.84 Mb/s ATM-LAN

Gi-Hong Im

Bell Laboratories/POSTECH
200 Laurel Avenue,
Middletown, NJ-07748,

Naresh R. Shanbhag

Coordinated Science Laboratory/ECE Dept.
University of Illinois at Urbana-Champaign,
Urbana, IL 61801. *shanbhag@uivlsi.csl.uiuc.edu*

Abstract

Presented in this paper are: 1.) system design issues for the implementation of 51.84 Mb/s ATM-LAN transceivers, 2.) an integrated VLSI design methodology underlying this design, and 3.) a pipelined fractionally-spaced linear equalizer (**FSLE**) architecture. The integrated design methodology incorporates algorithmic concerns such as signal-to-noise ratio (SNR) and bit-error rate (BER) along with VLSI constraints such as power dissipation, area, and speed, within a common framework. Characteristics of the channel and the modulation scheme are described. An adaptive **FSLE**, employed in the receiver, eliminates ISI, suppresses NEXT (in case of ATM-LAN) and provides robustness to timing jitter. A pipelined **FSLE** architecture is derived via the *relaxed look-ahead* technique for high-sample rate adaptation. Simulation and experimental results for high-speed digital CAP transceivers for LAN are also presented.

1 Introduction

Numerous high-bit rate digital communication systems are currently being proposed such as those for very high-speed digital subscriber loop (VDSL) [1], and asynchronous transfer mode (ATM) LAN [2]. Initially, ATM networks were envisioned to be a wide-area transport technology for delivering integrated services on public networks. However, the potential benefits of this technology has led to the acceptance of ATM technology in a new generation of LANs. In the ATM-LAN application, the bandlimited nature of the channel and the required performance levels necessitate the use of highly complex digital communications algorithms. Furthermore, the need to reduce costs is driving the industry towards increased levels of integration.

In this paper, we focus upon the design of 51.84 Mb/s transceivers for ATM-LAN. In particular, we present: 1.) system level issues in the design of data transceivers for ATM-LAN and 2.) an integrated VLSI design methodology and 3.) a hardware-efficient pipelined adaptive fractionally-spaced linear equalizer (**FSLE**).

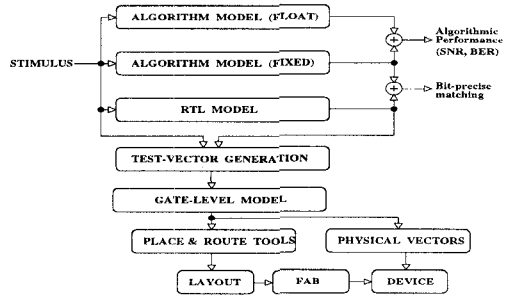

Figure 1: An integrated design methodology.

2 An Integrated VLSI Design Methodology

In this section, we will describe an integrated design methodology that was employed in the design of the 51.84 Mb/s ATM-LAN transceiver [3]. Such a methodology incorporates constraints from the VLSI domain into the algorithm design phase thus providing a design thread linking all levels of the VLSI design abstraction. Doing so enables the design, testing and system implementation of a functionally correct first silicon with a short design cycle. The proposed methodology (see Fig. 1) has four major steps:

- *STEP 1:* Floating-Point Algorithm Model Development.
- *STEP 2:* Fixed-Point Algorithm Model Development.
- *STEP 3:* Register-Transfer Level (RTL) Model Development.
- *STEP 4:* Logic, Circuit and Physical Design.

The proposed methodology differs from the traditional approach in that the latter consists only of *STEP 3* and *STEP 4*. The inclusion of *STEP 1* and *STEP 2* is what provides the integration between algorithm development and VLSI design.

In the first step (see Fig. 1), the establishment of the floating-point algorithm model (referred to as **FP**-model) requires:1.) regression between the measured

0-7803-3583-X/97 $10.00 ©1997 IEEE

and analytical channel models, 2.) inclusion of non-idealities such as component non-linearities and mismatch, and 3.) structural partitioning of algorithmic functions. This allows the subsequent phases in the VLSI design methodology to be executed smoothly. Furthermore, some accounting of the VLSI hardware speed and power in the **FP**-model also possible.

The second step involves determining the precisions of various signals in the algorithm model and simultaneous simulation of the **FL**-model and the corresponding fixed-point model (**FX**-model). The SNR at the slicer achieved via the fixed-point model (SNR_{fx}) was found to be within 0.1 dB of **FL**-model SNR. Similar results were obtained for different cable lengths and configurations justifying the choice of precision values.

Step 3 in (see Fig. 1) is to develop a register-transfer level model (**RTL**-model) of the algorithm, which is a clock-accurate and bit-accurate representation of the final chip. Our methodology ties in the **RTL**-model design to the algorithm design via the process of verification. This can be seen in Fig. 1, where we find that the input to the **RTL**-model and the **FX**-model is identical and outputs are compared in a *bit-precise* manner.

A key role of the **RTL**-model is to generate test vectors for both design verification and physical testing of the fabricated device. This can be seen in Fig. 1, where the inputs and the outputs of the **RTL**-model are formatted to generate the test vectors for the gate/logic level models. Finally, the chip-level test vectors were also employed as physical test vectors for the fabricated device. Fault-coverages in the high nineties were obtained in all cases. The first silicon for the ATM-LAN transceiver chip [3] exhibited correct functional behavior. Thus, we see that a tight link between the algorithm design and VLSI implementation is critical for a successful deployment of complex silicon systems especially those for high-bit rate communications.

3 The Channel

In this section, we will describe the physical environment and the CAP modulation scheme for ATM-LAN.

3.1 The LAN Environment

In Fig. 2, we show a vendor's view of an ATM-based LAN. The environment of interest for the unshielded twisted-pair category three (UTP-3) User Network Interface (UNI) consists of the "I1" and "I2" interfaces (see Fig. 2). The wiring distribution system runs either from the closet to the desktop or between hubs in the closets. The 16-CAP modulation scheme is the standard for ATM-LAN over UTP-3 at 51.84 Mb/s [2] and broadband access over copper wiring.

In the LAN environment, the two major causes of performance degradation for transceivers operating over UTP wiring are propagation loss and crosstalk generated between adjacent wire pairs. The propagation loss assumed is the worst-case loss given in the TIA/EIA-568 draft standard for category 3 cable [4] and is approximated by the following expression:

$$L_P(f) = 2.320\sqrt{(f)} + 0.238f, \qquad (3.1)$$

where the propagation loss $L_P(f)$ is expressed in dB per kilofoot and the frequency f is expressed in MHz. Near-end cross-talk (NEXT) occurs due to the coupling of the locally generated transmit signal to the received signal. The worst-case NEXT loss model for a single interferer (also given in the TIA/EIA draft standard [4]) has the following form:

$$L_N(f) = 43 - 15\log f, \qquad (3.2)$$

where the frequency f is in megahertz. The wavy curves in Fig. 3 give the measured pair-to-pair NEXT loss characteristics for three different combinations of twisted pairs in 100m category 3 cables. Notice also that the NEXT loss corresponding to the minima decreases with increasing frequency and tends to follow the smooth dotted curve of (3.2), which is also referred to as the 15 dB per decade model.

3.2 The CAP Transceiver Structure

The block diagram of a digital CAP transmitter is shown in Fig. 4(a). The 51.84 Mb/s bit stream to be transmitted is first passed through a scrambler (not shown in the figure) in order to randomize the data. The scrambled bits are then fed into an encoder, which maps blocks of m bits into one of $k = 2^m$ different complex symbols $a(n) = a_r(n) + ja_i(n)$. Given that 51.84 MHz and $m = 4$, we have the symbol rate $1/T =$ 12.96 Mbaud. The encoder can be implemented as a table look-up.

After the encoder, the symbols $a_r(n)$ and $a_i(n)$ are fed to digital shaping filters whose outputs are subtracted and the result is passed through a digital-to-analog (D/A) converter and an interpolating low-pass filter (LPF). The signal at the output of the CAP transmitter in Fig. 4(a) can be written as

$$s(t) = \sum_{n=-\infty}^{\infty} [a_r(n)p(t - nT) - a_i(n)\tilde{p}(t - nT)], \quad (3.3)$$

where T is the symbol period, $a_r(n)$ and $a_i(n)$ are discrete multilevel symbols, which are sent in symbol period nT, and $p(t)$ and $\tilde{p}(t)$ are the impulse responses of in-phase and quadrature passband shaping filters, respectively. In practice, the transmit bandwidth is made greater than $1/T$ by a fraction α, where α is also referred to as the *excess bandwidth*. For ATM-LAN, the excess bandwidth was 100% ($\alpha = 1.0$) in order to be able to supress NEXT.

The structure of a digital CAP receiver is shown in Fig. 4(b). It consists of an analog-to-digital (A/D) converter followed by a parallel arrangement of two adaptive digital filters. The A/D and the digital filters operate at a sampling rate $1/T' = i/T$, which

is typically the same as the sampling rate used at the transmitter. This parallel arrangement of adaptive filters in Fig. 4(b) is called a T/i fractionally spaced linear equalizer (**FSLE**). The function of the **FSLE** is to perform *NEXT suppression* (for ATM-LAN), *FEXT suppression* (for broadband access) and ISI removal. In addition, an **FSLE** based receiver also provides immunity to sampling jitter caused by the timing recovery circuit. However, from a VLSI perspective, implementing a high sample rate adaptive filter is a difficult task and pipelining techniques such as relaxed look-ahead [6] need to be employed.

The two outputs of the **FSLE** are sampled at the symbol rate $1/T$ and the results are fed to a decision device followed by a decoder, which maps the symbols into bits. The output of the decoder is then passed to a descrambler. Thus, we see that most of the signal processing in a CAP transceiver is done in the digital domain thus providing a robust VLSI implementation. The interested reader is referred to [5] for more details on the design of CAP receivers.

4 The Pipelined Adaptive FSLE Architecture

Designing adaptive **FSLE**'s with sufficient number of taps to meet a BER requirement of 10^{-10} at these sample rates is a challenging problem. This problem can be solved via the technique of pipelining, which will be described in the next subsection.

4.1 Pipelined FSLE

As mentioned in the previous subsection, adaptive **FSLE**s are needed in the applications of interest in this paper. As shown in Fig. 4(b), these two **FSLE**'s are adapted independently employing the in-phase error $e_R(n)$ (for the in-phase **FSLE** or **IFSLE**) and the quadrature-phase error $e_I(n)$ (for the quadrature phase **FSLE** or **QFSLE**) via the leaky LMS algorithm. The serial **IFSLE** (**QFSLE** can be similarly described) algorithm is described by,

$$e_R(nT) = Q[y_R(nT)] - \mathbf{W}_R^T((n-1)T)\mathbf{X}(nT) \quad (4.1)$$

$$\mathbf{W}_R(nT) = (1-\gamma)\mathbf{W}_R((n-1)T) + \mu e_R(nT)\mathbf{X}(nT) \quad (4.2)$$

where $\mathbf{W}_R(nT) = [w_{0,R}(nT), w_{1,R}(nT), \ldots, , w_{N-1,R}(nT)]^T$ is the vector of in-phase coefficients, $\mathbf{X} = [x(nT), x(nT-T_s), \ldots, x(nT-(N-1)T_s)]^T$ is the vector of input samples, μ is the step-size, γ is the tap-leakage factor and $Q[.]$ is the output of the decision device.

An architectural representation of (4.1-4.2) is shown in Fig. 5(a). Note that in Fig. 5(a), we have not yet exploited the fact that the output of the filter block is being sampled at baud-rate. This is done in order to facilitate the application of relaxed look-ahead [6], which applies at the algorithmic level. Doing so results in the following equations that describe the pipelined **IFSLE**:

$$e_R(nT) = Q[y_R(nT)] - \mathbf{W}_R^T((n-D_2)T)\mathbf{X}(nT) \quad (4.3)$$

$$\mathbf{W}_R(nT) = (1-\gamma)^{D_2}\mathbf{W}_R((n-D_2)T) + \mu \sum_{i=0}^{LA-1} e_R(nT-iT-D_1T)\mathbf{X}(nT-iT-D_1T) \quad (4.4)$$

Employing *product relaxation* [6], we can further simplify look-ahead technique allows us to further simplify (4.4) by approximating the term $(1-\gamma)^{D_2}$ by $(1-D_2\gamma)$. This is possible to do because the value of γ is small as compared to unity. The pipelined **FSLE** architecture is shown in Fig. 5(b), where D_1 and D_2 represent baud-rate delays. Note that MD_1' delays have been obtained by retiming D_1' baud-rate delays into the filter block, which operates at the sample rate. Thus, fine-grain pipelining of all the blocks has been achieved.

5 Simulation and Experimental Results

In this section, we describe some quantitative results for the performance of the 16-CAP transceiver. We employ noise margin as a measure of algorithm performance, where the noise margin is the difference between the $SNR_{o,ref}$ (the SNR required to achieve a given BER) and SNR_o (the actual SNR at the slicer). For 16-CAP, $SNR_{ref} = 23.25$dB corresponds to a $BER = 10^{-10}$. A positive margin implies that the transceiver operates with a BER that is better than the targeted BER.

In Table 1, we show the margin obtained with the ATM-LAN transceiver with a transmitter and receiver spans of $32T$ each. The first four rows of data in the Table 1 were obtained with the TIA/EIA NEXT loss model [4] for category 3 cables. The other entries in the table were obtained using measured NEXT loss characteristics. The terms ϕ_i $(i = 0, \ldots, 3)$ refer to the four relative phases that the clock of the NEXT transmitter can have with respect to the receiver clock.

The quantity of interest in Table 1 is the margin, which is given in the last column. Notice that all the margins are positive, so that the 16-CAP transceiver achieves a BER that is better than 10^{-10} for all cases considered in the table. Notice also in Table 1 that the values of the SNR at the input of the slicer, i.e. SNR_o, are much larger than the values of the SNR at the input of the receiver, i.e. SNR_i. Thus, the **FSLE** used in the receiver improves the SNR. This is possible because the **FSLE** performs NEXT suppression. It should be pointed out that the computer simulation results in Table 1 are consistent with similar results obtained in the laboratory.

6 Acknowledgment

The authors would like to acknowledge the encouragement and support of J.-J. Werner and Jit Kumar.

References

[1] P. S. Chow, J. C. Tu, and J. M. Cioffi, "Performance evaluation of a multichannel transceiver system for ADSL and VHDSL services," *IEEE J. Select. Areas in Commun.*, vol. 9, pp. 909-919, Aug. 1991.

[2] *IEEE Journal on Sel. Areas in Commun.*, vol. SAC-13, no. 4, May 1995, Special Issue on ATM LAN.

[3] N. R. Shanbhag, G. A. Wilson, R. Shariatdoust, J. Kumar, R. Townsend, "VLSI complexity of the 16-CAP transceiver", *ATM Forum PHY Subworking Group Contribution*, ATM-Forum/93-994, Nov. 16-19, 1993, Stockholm, Sweden.

[4] *Commercial Building Telecommunications Cabling Standard*, TIA/EIA-568-A Standard, 1994.

[5] W. Y. Chen, G.-H. Im and J.-J. Werner, "Design of digital carrierless AM/PM transceivers," AT&T/Bellcore Contribution T1E1.4/92-149, Aug. 19, 1992.

[6] N. R. Shanbhag and K. K. Parhi, *Pipelined Adaptive Digital Filters*. Kluwer Academic Publishers, 1994.

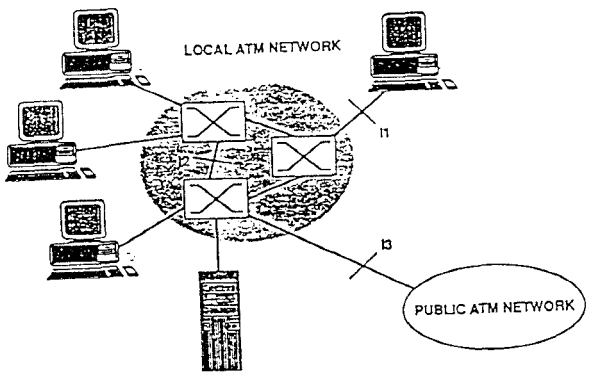

Figure 2: ATM-based LAN.

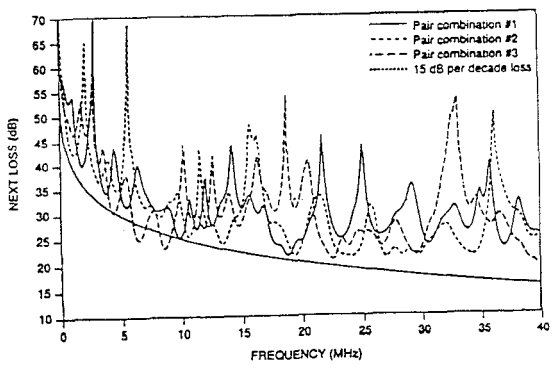

Figure 3: Measured NEXT loss.

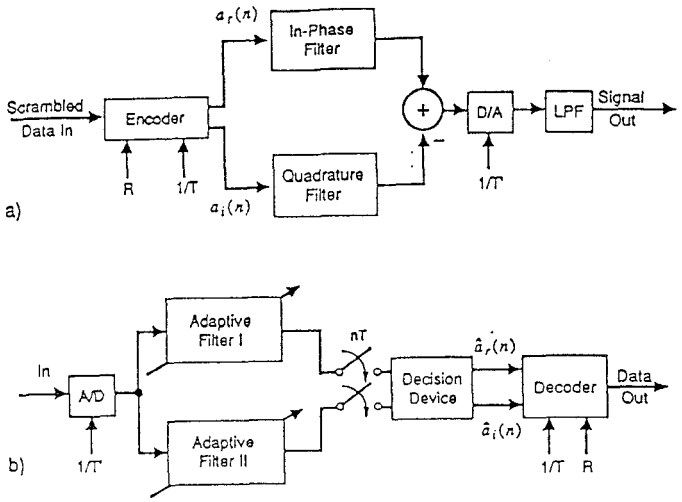

Figure 4: The CAP: (a) transmitter, and (b) receiver.

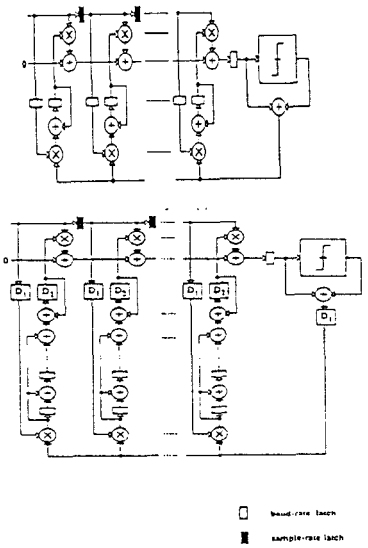

Figure 5: The FSLE:(a) serial and (b) pipelined.

Table 1 - Performance of 51.84 Mb/s 16-CAP Transceiver over 100m Category 3 Cable with One Cyclostationary NEXT Interferer

NEXT Model	ϕ_i (i T/4)	SNR_i (dB)	SNR_e (dB)	Margin† (dB)
TIA/EIA Model	ϕ_0	13.8	25.9	2.65
	ϕ_1	13.8	27.1	3.85
	ϕ_2	13.8	27.0	3.75
	ϕ_3	13.8	26.0	2.75
Worst-case Measured	ϕ_0	17.8	28.3	5.05
	ϕ_1	17.8	27.2	3.95
	ϕ_2	17.8	29.7	6.45
	ϕ_3	17.8	29.3	6.05

† Margins are with respect to $P_e = 10^{-10}$ for which $SNR_{e,ref} = 23.25$ dB