Modeling and Simulation of Discretized Data Transmission in Very High-Speed Digital Subscriber Line

Dogu Arifler, Ming Ding, and Zukang Shen

Abstract

We implement a Synchronous Dataflow (SDF) model for the discretized data transmission in Very High-Speed Digital Subscriber Line (VDSL) using Agilent Advanced Design System (ADS) electronic design automation software. Our ADS implementation will enable designers to simulate and optimize different VDSL transceiver designs. The filters used at the VDSL receiver and the bit allocations for sub-channels of a specific VDSL channel are designed using MATLAB and are integrated into our ADS implementation. The overall ADS implementation of the system is fully compliant with the performance requirements of the European Telecommunication Standards Institute VDSL specification.

1. INTRODUCTION

Very High-Speed Digital Subscriber Line (VDSL) technology is an emerging solution for the last kilometer of connection to a residence or small office [1]. VDSL enables very high bit rate applications over short-length (1 km or less) twisted pair copper wire.

The VDSL technical specification [2, 3] proposes the use of discrete multitone (DMT) as one of the two standardized line codes in VDSL systems. The use of DMT in VDSL has important advantages including the ability to maximize the transmitted bit rate and adapt to changing line conditions.

Designing VDSL systems is inherently complex. However, with recent advances in the digital signal processor (DSP) technology, programmable DSP based solutions are rapidly replacing application-specific integrated circuit based implementations. DSP based solutions decrease time-to-market of a product and allow rapid upgrades to the system after the product is shipped. Such trends necessitate a high-level abstraction of the system that can be reused and shared between product releases.

Synchronous dataflow (SDF) model of computation [4] is well suited for modeling data-driven, communication and signal processing applications such as VDSL. Efficient DSP code can be generated from statically scheduled SDF programs [5].

The major contribution of our project is a high-level SDF abstraction and a corresponding implementation in an electronic design automation (EDA) software of a discretized, DMT based VDSL transceiver that is compliant with the freely available European Telecommunication Standards Institute (ETSI) specification. Our implementation will enable designers to simulate, evaluate, optimize, and possibly synthesize different implementations of VDSL transceiver designs.

2. BACKGROUND

2.1. The VDSL Channel

We described the VDSL channel environment in detail in our literature survey [6]. In this project, we consider only crosstalk, which has a significant effect on the VDSL channel. Crosstalk is caused by electromagnetic radiation. Near-end crosstalk (NEXT) is caused by signals traveling in opposite directions, while far-end crosstalk (FEXT) is caused by signals traveling in the same direction [1]. Both NEXT and FEXT increase with frequency. Methods that mitigate crosstalk in VDSL systems with impulse noise are currently under investigation [7].

2.2. Discrete Multitone (DMT) Based VDSL Transceiver

DMT and the transceiver functions were explained in detail in our literature survey [6]. DMT divides the bandlimited communication channel into a large number of orthogonal, narrowband sub-channels to mitigate inter-symbol interference (ISI). A significant amount of research has shown that a DMT based system is capable of transmitting at the highest performance with proper allocation of bits and energies to the sub-channels. A proper allocation of bits is the one in which sub-channels with higher signal-to-noise ratios carry proportionately more bits [8]. In VDSL, the number of bits that can be carried by each sub-channel ranges from 0 to 15 [3]. The number of sub-carriers used in the system may be 512, 1024, 2048, or 4096 [3]. The assignment of the number of bits for each sub-channel, in the form of a *bit allocation table*, is exchanged between the transmitter and the receiver during *initialization and channel identification*.

A DMT based transceiver together with the channel is illustrated in Figure 1. Since we consider only the discretized data transmission, we omit the analog front ends from

the figure. In a DMT based transceiver, a bit stream is encoded/decoded into a set of Quadrature Amplitude Modulation (QAM) symbols, called *sub-symbols*, by the data encoders/decoders. Each sub-symbol is a complex number and carries the number of bits determined by the bit allocation on one sub-channel.

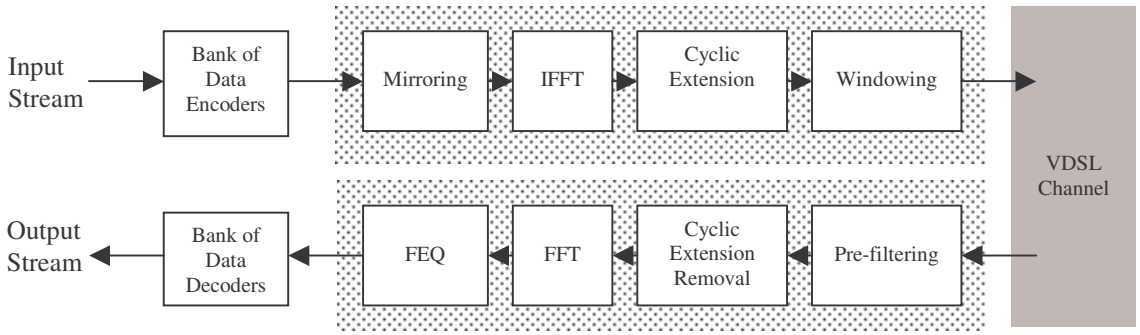


Figure 1. A discretized VDSL transmitter and receiver (transceiver).

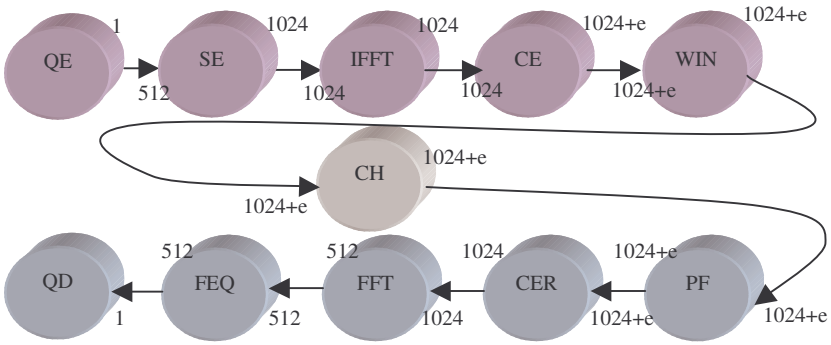
DMT modulation and demodulation are generally implemented using Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT). We impose the conjugate symmetry conditions (through a process called *mirroring*) onto the input sub-symbols to IFFT to make IFFT output a real number sequence for transmission on the channel. Reduced complexity FFT/IFFT implementations for DMT based VDSL transceivers on digital signal processors are always topics for further research [9].

The other functionalities of the VDSL transceiver include cyclic extension, windowing, pre-filtering, and frequency-domain equalization (FEQ). A T-tap pre-filter, or time-domain equalizer, is implemented if the channel duration is too long compared to the length of cyclic extension [1, 10]. Another use of the pre-filter is to suppress the radio frequency interference from amateur radio bands. FEQ is implemented to compensate for magnitude and phase distortion from the channel and the pre-filter [1].

3. MODELING

We use SDF [4] to model the discretized VDSL transceiver blocks and the discretized channel as illustrated in Figure 2. SDF is very suitable for modeling signal processing and communication systems in which a stream of data is processed by signal processing functional blocks. Inputs and outputs of the SDF blocks are the sub-symbols. The data consumption and production of each block in SDF are pre-determined.

The total number of bits encoded/decoded by the QAM encoder/decoder bank is fixed. The number of QAM encoders/decoders in the banks is equal to the number of sub-carriers used by the system. Each QAM encoder/decoder encodes/decodes a number of bits determined by the bit allocation table. Figure 3 shows the SDF blocks for an individual QAM encoder/decoder.



QE: Bank of QAM encoders	SE: Conjugate symmetric extension	CE: Cyclic extension
WIN: Windowing	CH: Channel	PF: Pre-filter
CER: Cyclic extension removal	FEQ: Frequency-domain equalization	QD: Bank of QAM decoders

Figure 2. Top-level SDF model for the discretized VDSL transceiver that uses 512 sub-carriers. Data type on arcs is sub-symbols. "e" is the length of cyclic extension in sub-symbols and is equal to 80.

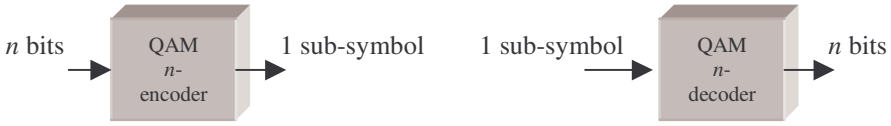


Figure 3. SDF blocks for a QAM encoder and a QAM decoder from the encoder and the decoder banks. Data consumptions and productions are also shown.

Finally, Figure 4 shows the SDF model for the VDSL channel. FIR_{ch} is the channel impulse response modeled as a finite impulse response (FIR) filter. In this project, we model only the NEXT and FEXT noises. The NEXT and FEXT noises are generated by passing White Gaussian Noise (WGN) through FIR filters (FIR_{next} and FIR_{fext}).

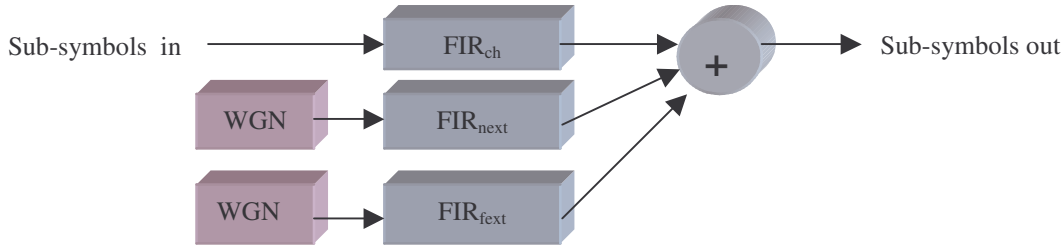


Figure 4. The SDF model for the discretized VDSL channel. Data consumption and production by the blocks are equal to one sub-symbol.

The VDSL transmitter obtains an estimate of the channel's impulse response and crosstalk noise spectrum before the data transmission begins through identification and channel initialization.

4. IMPLEMENTATION

We implemented the discretized VDSL transmission system using Agilent Advanced Design System (ADS) software [11]. We assumed that the transmitter and the receiver were perfectly synchronized. We further assumed that the bit allocation table did not change after initialization. In this project, we chose not to implement the windowing block at the transmitter, since the bit rate performance of the system did not improve significantly with windowing according to our MATLAB based design.

ADS enables hierarchical design of systems. We built most of the top-level transmitter, receiver, and channel SDF blocks using ADS's signal processing library

components. The SDF blocks for individual QAM encoders/decoders and the SDF block for cyclic extension were implemented using MATLAB.

We used the *linemod* program [12] from Stanford University to generate a model and implement a 1024-tap FIR filter for a particular VDSL transmission line (Loop #5). We used MATLAB to design two 500-tap FIR filters that were used to generate the NEXT and FEXT noises based on the ETSI VDSL specification [2].

Then, we designed the pre-filter and FEQ at the receiver using MATLAB for the specific VDSL channel. Our system specifications are listed in Table 1. The achievable bit rate based on these specifications was 17.9 Mbps for a bit error rate of 10^{-7}.

The filter coefficients designed using MATLAB were stored in files and were directly imported into ADS's filtering components.

Table 1. Specifications for the discretized VDSL transceiver. Each DMT *symbol* in a 512 sub-carrier system consists of 1024 sub-symbols plus 80 samples for the cyclic extension.

	Specification	Comments
Number of sub-channels	512	FFT size = 1024
Cyclic extension length	80	Mitigate ISI
Sampling frequency	4.416 MHz	4000 symbols/s
Length of pre-filter	5 taps	To shorten the channel
Length of FEQ	1 tap per sub-carrier	To compensate distortion
Transmitter power	-40 dBm/Hz	Uniform in 2 MHz band

5. RESULTS AND CONCLUSION

Table 2 shows the computational complexity of the receiver part of the discretized VDSL transceiver. Note that an N-point FFT requires $(N/2)\log_2 N$ complex multiplications and $N\log_2 N$ complex additions. The computational cost of QAM decoding is not very

significant since decoding is done by a table lookup that takes constant time assuming random access to the table.

Table 2. Computational complexity of the discretized VDSL receiver based on the specifications in Table 1.

	Real multiplications/symbol	Real additions/symbol
T-tap pre-filter	1104×T	1104×(T-1)
FFT	10240×2=20480	10240×(1+2)=30720
FEQ	512×4=2048	512×2=1024
Total	22528+(1104×T)	30640+(1104×T)
For T = 5	**28048**	**36160**

We simulate the ADS implementation of the discretized VDSL system to verify the functionality of the transceiver components. The simulation results show that the implemented VDSL transceiver is capable of high-rate data transmission over typical short-length twisted pair in bundled cable where significant crosstalk is present.

The performance of our system meets the payload bit rate specifications of ETSI [2]. More specifically, our achievable bit rate is 17.9 Mbps that can support up to Class I (A3) and Class II (S3) operations' payload bit rates [2, Table 5]. Note that our system only uses $1/8^{th}$ of the total available bandwidth. Higher payload bit rates for Class I (A4), Class II (S4), and Class II (S5) operations can easily be surpassed by allocating more sub-carriers (up to 4096) in our system.

The ADS implementation of the discretized VDSL transmission system is a framework for research on VDSL systems. The framework provides an easy-to-use simulation and test environment for evaluating VDSL transceiver designs.

References

[1] J. M. Cioffi, V. Oksman, J. Werner, T. Pollet, P. M. P. Spruyt, J. S. Chow, and K. S. Jacobsen, "Very-High-Speed Digital Subscriber Lines," *IEEE Communications Magazine*, vol. 37, no. 4, pp. 72-79, Apr. 1999.

[2] "Transmission and Multiplexing (TM); Access transmission systems on metallic access cables; Very high speed Digital Subscriber Line (VDSL); Part 1: Functional requirements," *ETSI TS 101 270-1 V1.2.1* (1999-10), Technical Specification.

[3] "Transmission and Multiplexing (TM); Access transmission systems on metallic access cables; Very high speed Digital Subscriber Line (VDSL); Part 2: Transceiver specification," *ETSI TS 101 270-2 V1.1.1 (2001-02)*, Technical Specification.

[4] E. A. Lee and D. G. Messerschmitt, "Synchronous Data Flow," *Proc. of the IEEE*, vol. 75, no. 9, pp. 1235-1245, Sep. 1987.

[5] E. A. Lee and D. G. Messerschmitt, "Static Scheduling of Synchronous Data Flow Programs for Digital Signal Processing," *IEEE Transactions on Computers*, vol. C-36, no. 1, pp. 24-35, Jan. 1987.

[6] D. Arifler, M. Ding, and Z. Shen, "Modeling and Simulation of Discretized Data Transmission in Very High-Speed Digital Subscriber Line," *Literature Survey* for EE 382C-9 Embedded Software Systems Course Project, The University of Texas at Austin, Spring 2002.

[7] H. Dai and H. V. Poor, "Crosstalk mitigation in DMT VDSL with impulse noise," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 10, pp. 1205-1213, Oct. 2001.

[8] P. S. Chow, J. M. Cioffi, and J. A. C. Bingham, "A practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels," *IEEE Transactions on Communications*, vol. 43, no. 2, part 3, pp. 773-775, Feb.-Mar.-Apr. 1995.

[9] A. Lakhzouri and M. Renfors, "Signal processor implementation of DMT based VDSL modems," *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2349-2352, 2001.

[10] N. Al-Dhahir and J. M. Cioffi, "Optimum Finite-Length Equalization for Multicarrier Transceivers," *IEEE Transactions on Communications*, vol. 44, no. 1, pp. 56-64, Jan. 1996.

[11] Advanced Design System 2002, Agilent Technologies, http://eesof.tm.agilent.com.

[12] *Linemod*, Stanford University, http://www.stanford.edu/class/ee379a.