

Efficient Architectures for Improving VDSL

Author: Tim Styles

02-DP-1641

ClearSpeed Technology Ltd

130 Aztec West
Park Avenue
Bristol BS32 4UB
United Kingdom

Tel: +44 (0) 1454 629623

Fax: +44 (0) 1454 629624

info@clearspeed.com

www.clearspeed.com

Disclaimer

Information and data contained in this document, together with the information contained in any and all associated ClearSpeed documents including without limitation, data sheets, application notes and the like ('Information') is provided in connection with ClearSpeed products and is provided for information only. Quoted figures in the Information, which may be performance, size, cost, power and the like are estimates based upon analysis and simulations of current designs and are liable to change.

Such Information does not constitute an offer of, or an invitation by or on behalf of ClearSpeed, or any ClearSpeed affiliate to supply any product or provide any service to any party having access to this Information. Except as provided in ClearSpeed Terms and Conditions of Sale for ClearSpeed products, ClearSpeed assumes no liability whatsoever.

ClearSpeed products are not intended for use, whether directly or indirectly, in any medical, life saving and/ or life sustaining systems or applications. The worldwide intellectual property rights in the Information and data contained therein is owned by ClearSpeed. No license whether express or implied either by estoppel or otherwise to any intellectual property rights is granted by this document or otherwise.

The system vendor remains solely responsible for any and all design, functionality and terms of sale of any product which incorporates a ClearSpeed product including without limitation, product liability, intellectual property infringement, warranty including conformance to specification and or performance.

ClearSpeed reserves the right to make changes to the Information or the data contained therein at any time without notice.

© Copyright ClearSpeed Technology Ltd 2010. All rights reserved.

ClearSpeed, ClearConnect and the ClearSpeed logo are trade marks or registered trade marks of ClearSpeed Technology Ltd. All other brands and names are the property of their respective owners.

Contents

1	Introduction	5
2	Background	6
3	Processing Requirements	8
3.1	Low power	9
3.2	High performance	9
3.2.1	FEXT Pre-compensation	10
3.2.2	FEXT Cancellation	10
3.2.3	Other Receiver Enhancements	10
3.3	Required operations	11
3.3.1	Transmitter PMS block	11
3.3.2	Transmitter PMD block	12
3.3.3	Receiver PMD block	13
3.3.4	Receiver PMS block	14
4	Suitable Architectures	15
4.1	Existing systems	15
4.2	Implementation 1: Configurable pipeline	16
4.2.1	Implementation	18
4.2.2	Modelling Results	18
4.3	Implementation 2: SIMD array	19
4.3.1	Architectural Details	19
4.3.2	Implementation	20
4.3.3	Modelling Results	21
4.4	Architecture Comparison	21
4.5	SIMD Architecture Extension – Flexible DSLAM	22
5	Summary and Conclusions	24
6	About ClearSpeed	25
7	References	26

Tables

Table 1 - Dedicated hardware for VDSL processing	17
Table 2 - SIMD array for VDSL processing	20

Figures

Figure 1 - VDSL cables terminate at the Fiber fed DSLAMs	6
Figure 2 - PMS and PMD blocks in a VDSL transceiver	8
Figure 3 - Functions in the Transmitter PMS block	11
Figure 4 - Functions in the Transmitter PMD block	12
Figure 5 - Functions in the Receiver PMD block	13
Figure 6 - Functions in the Receiver PMS block	14
Figure 7 - System diagram of the iKanos Fusiv Vx180	15
Figure 8 - VDSL Processing diagram	16
Figure 9 - SIMD Array diagram	19
Figure 10 – DSLAM Architecture, Single Configurable DSLAM Block	22
Figure 11 - DSLAM Architecture, Two Configurable DSLAM Blocks	23

Abbreviations

ADC	Analogue to Digital Convertor
ADSL	Asymmetric Digital Subscriber Line
CM	Common Mode
DAC	Digital to Analogue Convertor
DFT	Discrete Fourier Transform
DM	Differential Mode
DMT	Discrete Multi-Tone
DSLAM	Digital Subscriber Line Access Multiplexer
FDD	Frequency Division Duplex
FEC	Forward Error Correction
FEXT	Far End cross Talk
IDFT	Inverse Discrete Fourier Transform
MAC	Multiply Accumulate
MIMO	Multiple Input Multiple Output
MISD	Multiple Instruction Single Data
NEXT	Near End cross Talk
ODMC	Off-Diagonal MIMO Canceller
PCI	Peripheral Component Interconnect
PMD	Physical Media Dependent
PMS	Physical Media Specific
PSD	Power Spectral Density
QAM	Quadrature Amplitude Modulation
RFI	Radio Frequency Interference
RISC	Reduced Instruction Set Computer
SIMD	Single Instruction Multiple Data
SINR	Signal to Interference and Noise Ratio
TEQ	Time domain Equalizer
USB	Universal Serial Bus
VDSL	Very high speed Digital Subscriber Line

1 Introduction

A modern digital infrastructure will provide fast, reliable connections to homes and other premises throughout the UK. A cost effective deployment involves laying fiber optic cables to street cabinets and continuing use of the existing copper wires from the street cabinets to the consumers. A high data rate can be achieved with the short loops of twisted pair copper wire and a recommendation for communication over this 'last mile' has evolved during the last decade.

The International Telecommunication Union has a recommendation for Very high speed Digital Subscriber Line (VDSL2) transceivers [1]. This recommendation details the parameters for a number of profiles that enable two transceivers to communicate with each other. Transceivers may include additional time-domain equalizers, echo cancellers and interference mitigation techniques to increase the data rate and supported line length. These techniques are still an active area of research and any processing technology targeting this recommendation should include provision for supporting future developments.

The architecture proposed in this document is software programmable and highly versatile. It supports the VDSL2 recommendation and is backwards compatible with the existing ADSL standards. This enables the same architecture to be used with existing transceivers during the transition between ADSL and VDSL2. Transceiver manufacturers are able to add value to a software programmable architecture, enabling them to differentiate themselves from other manufacturers. The architecture can be used in consumer equipment and also in the Digital Subscriber Line Multiplexer (DSLAM) at the operator's central office. Using the architecture in the DSLAM allows future advances in interference mitigation to be implemented without replacing the hardware. This reduces the cost and environmental impact of frequent DSLAM upgrades.

The architecture has been designed with the aim of balancing performance with low power consumption. This is important to the operator because power efficient hardware reduces total running costs, which over the lifetime of the hardware will be more significant than the cost of the components. While the running cost of a transceiver in the home is not significant to the consumer, the total environmental impact of many millions of transceivers is worth consideration. The carbon footprint of a modern digital infrastructure will be reduced by selection of a suitable low power, high performance architecture.

2 Background

The maximum data rate that can be achieved with twisted pair copper wire is limited by the Signal to Interference and Noise Ratio (SINR). Longer loops attenuate the signal, especially at high frequencies. The short loops considered for VDSL (< 1km) support frequencies up to 30MHz, which enables high data rates up to 200 Mbit/s [1].

High frequency transmissions are more susceptible to Radio Frequency Interference (RFI) from external sources and most significantly from electromagnetic coupling with other wires in the same wiring bundle, known as crosstalk. The RFI emitted by the wiring bundle can be reduced by managing the Power Spectral Density (PSD) of transmitted signals. The ITU recommendation includes PSD management and support for notching 16 arbitrary bands defined by the operator. The recommendation also specifies the use of Forward Error Correction (FEC) codes to correct bit errors generated by RFI received from external equipment and crosstalk.

VDSL uses Frequency Division Duplexing (FDD) to reduce Near End CrossTalk (NEXT). FDD uses separate bands for upstream and downstream traffic, which prevents a transmitter from interfering with other receivers at the DSLAM. The most significant source of interference is Far End CrossTalk (FEXT), where transmissions travelling in the same direction along a bundle of wires interfere with each other, resulting in a reduction in SINR at the receiver. This crosstalk is typically 10-15 dB larger than other noise sources and it affects upstream traffic to the receiver in the DSLAM as well as downstream traffic to the receiver at the consumer's premises.

All the wires generating crosstalk are likely to be terminated at the same DSLAM, either in a building, a street cabinet or an exchange (Figure 1). The knowledge of what is received on each wire at the DSLAM enables the crosstalk interference to be estimated and cancelled. Similarly, the knowledge of what will be transmitted on each wire enables the signals to be distorted such that the distortion interferes destructively with the crosstalk, resulting in a significant reduction in crosstalk at each receiver. This distortion is termed pre-compensation.

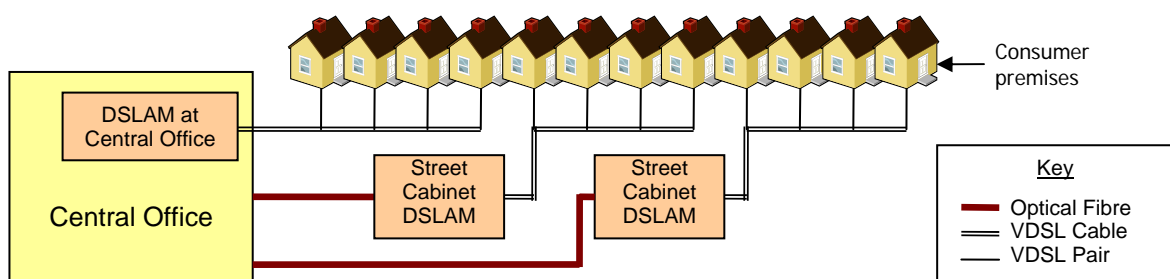


Figure 1 - VDSL cables terminate at the Fiber fed DSLAMs

The pre-compensation calculation at the DSLAM is a form of Multiple Input Multiple Output (MIMO) processing, where multiple input signals are fed into the cable bundle producing multiple output signals with

reduced crosstalk [2]. The computation required for pre-compensation and crosstalk cancellation can be reduced by concentrating on the most significant sources of crosstalk and the worst affected frequencies [3][4]. The use of frequency selection and line selection is termed partial crosstalk cancellation and it can provide 80% of the benefit for only 20% of the effort.

VDSL signals are transmitted as a Differential Mode (DM) signal on twisted pair copper wire. This reduces the transmitted RFI and allows received RFI to be rejected by considering only the difference between the signals received on each wire. The average of the signals relative to ground can also be measured to obtain the Common Mode (CM) signal, which is strongly correlated with crosstalk in the DM signal. The interference measured in the CM signal can be used to improve the SINR, especially on long lines where the DM signal is weak [5].

3 Processing Requirements

The ITU recommendation details two functional blocks in the VDSL transceiver: the Physical Media Specific (PMS) block and the Physical Media Dependent (PMD) block (Figure 2).

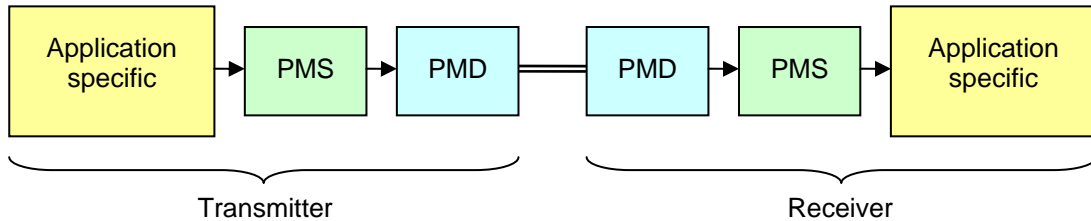


Figure 2 - PMS and PMD blocks in a VDSL transceiver

The processing architectures considered here implement the functions in the PMS and PMD blocks. The architecture also considers the analog circuitry between the transmitter and receiver so that the optimum interface can be provided. The functions in the application specific layer are likely to be implemented using an industry standard processor, which could be combined with one of the processing architectures considered here.

The PMS block combines multiple data streams and creates a bit stream with redundancy that is resilient to burst errors caused by impulse noise. The functions in the PMS block are as follows:

1. Multiplex channels and overhead channel.
2. Scramble the bit stream.
3. Add FEC octets (Reed Solomon codes).
4. Delay and interleave octets.
5. Multiplex data paths with different latencies.

The PMD block creates an analog Discrete Multi-Tone (DMT) signal from the bit stream with a spectrum tailored to meet the PSD requirements. Data bits are allocated across the spectrum such that the required SINR matches the expected SINR for each frequency tone. The functions in the PMD block are as follows:

1. Add timing information.
2. Allocate bits to each frequency tone.
3. Pass the bits for each tone through a trellis encoder.
4. Map the trellis encoded bits to an analog tone value in a Quadrature Amplitude Modulation (QAM) constellation.
5. Generate a time domain signal from the frequency tone values using an Inverse Fourier Transform (IDFT).

6. Extend the time domain signal with a cyclic prefix.

The details for each function are described in the ITU recommendation. Here we consider how each function could be implemented in silicon.

3.1 Low power

In general a low power, high performance processing architecture will follow these principles:

1. Minimize transmitter power:
Reduced transmitter power with dynamic spectrum management to match the required bit rate. This reduces crosstalk and enables a general reduction in power at the DSLAM while maintaining the required bit rate on each line [6][7].
2. Maximize processing efficiency:
The use of good noise and interference cancellation algorithms allows lower power transmissions to achieve the same bit rate [8][9].
3. Minimize clock rate and data movement:
A processor with a low clock rate can run at a lower voltage. This reduces both the dynamic power and the power dissipated through leakage.
4. Maximize performance:
High performance is achieved by performing a large number of operations on each cycle. This can involve:
 - a. Streaming the data through a pipeline such that different operations are performed in parallel.
 - b. Performing the same operation on different parts of the data stream in parallel (Single Instruction Multiple Data - SIMD).
 - c. A combination of both SIMD and a pipelined architecture.

The ITU recommendation includes details of a power back-off scheme and dynamic spectrum management to achieve a permitted PSD. Halving the transmitted power with dynamic spectrum management reduces impulse and spectral noise, and the bit rate is reduced by just 15%. Long lines can be unnecessarily affected by crosstalk from short lines, which do not need a high transmit power to achieve the same bit rate. The interference generated by the shorter lines can be reduced with power back-off, significantly improving the bit rate on longer lines [7].

3.2 High performance

The processing architecture should be reconfigurable or software programmable if it is to support multiple standards and future enhancements. Crosstalk mitigation is still a major area of development in VDSL. The following techniques are examples of the type of processing that will improve performance.

3.2.1 FEXT Pre-compensation

Transceivers in the DSLAM can treat the copper wires as paths in a MIMO network. The signal transmitted on each wire can include pre-compensation that destructively interferes with the expected crosstalk resulting in a channel to the receiver with reduced crosstalk (G.vector). In practice it is not necessary to consider the effect of every wire in a wire bundle, and the most significant lines causing crosstalk can be determined. It is also possible to determine the frequencies that are most affected by crosstalk.

In terms of the processing required, full pre-compensation requires one multiply operation for each tone and each user being compensated for. Experiments have shown that selecting half the lines and half the frequencies achieves over 80% of the compensation that could be achieved by selecting all the lines and frequencies. The compensation can double the data rate achieved on short lines, and increase the data rate on 1.2 km lines by 25%. The pre-compensation factors are established by sending training symbols. Forty symbols are sufficient and the results can be stored for all lines, even though only a quarter of the pre-compensation factors are used at any one time [10].

3.2.2 FEXT Cancellation

Transceivers in the DSLAM can also process the signal received on each wire to cancel crosstalk. This is similar and requires one multiply operation for each tone and each user. The coefficients used for crosstalk cancellation are related to those used for pre-compensation, and so the same training period can be used to establish both sets of coefficients. It is likely that a different set of lines and frequencies will be selected for cancellation and pre-compensation, since the sources of crosstalk will be different in upstream and downstream traffic.

The coefficients used in crosstalk cancellation and pre-compensation can be updated iteratively from noise measurements at the QAM decoder. It typically takes 200 iterations for an Off-Diagonal MIMO Canceller (ODMC) to converge on an optimum solution. Initialization requires $K(0.5N^2+2.5N+3)$ multiply operations and $K(N+1)$ log operations for K tones and N users, plus an integer sort of up to KN length. The processing required to update the coefficients can be shared across multiple channels at the DSLAM. The channel matrix can be computed using an approximation, or by pre-computation for each tone (requires more memory). This results in 60 MFLOPS when updating at 100 Hz [11].

3.2.3 Other Receiver Enhancements

A receiver in the home cannot make use of these crosstalk mitigation techniques, although other enhancements can be performed in the home and at the DSLAM. These include SINR improvement based on the CM signal and echo cancellation.

Inter-Symbol Interference (ISI) can be caused by reflections from bridged taps on the line and unfiltered extensions in the home. A Time domain Equalizer (TEQ) can be used at the receiver to cancel the reflections, but the improvement depends on the channel characteristics. For this reason it is desirable to include the TEQ only when it improves the performance. A typical TEQ performs 20 Multiply Accumulate

(MAC) operations on each sample received and the coefficients can be derived from the VDSL training symbols [12].

The use of alternative error correction schemes has been considered (G.inp). While the ITU recommendation gives details of a variable Reed Solomon code and a 4D trellis code, future standards may use turbo codes or other forms of FEC [13]. Error corrections could be passed from the receiver PMS block back to the PMD block so that the filters can be updated.

3.3 Required operations

3.3.1 Transmitter PMS block

The functions in the PMS block are shown in Figure 3 below. They combine multiple data streams and create a bit stream with redundancy that is resilient to burst errors. There may be two paths with different interleave settings for low latency and high resiliency paths. The results are multiplexed before being passed to the PMD block.

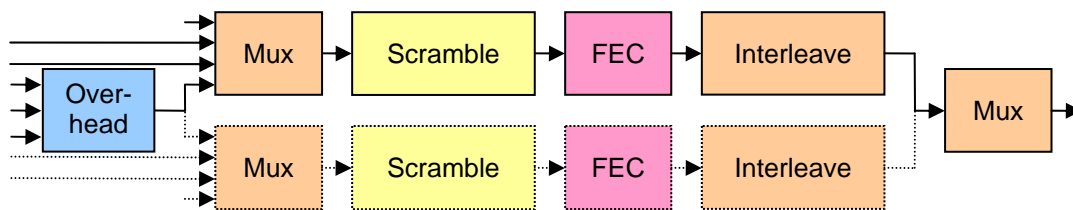


Figure 3 - Functions in the Transmitter PMS block

The first stage involves reading octets from a number of streams and multiplexing them together. These streams include data, management and synchronization. The bits are scrambled with the following formula before being passed to the FEC encoder:

$$x(n) = m(n) + x(n-18) + x(n-23)$$

where n is the index of a bit in the bit stream, $m(n)$ is an input bit and $x(n)$ is an output bit.

A Reed-Solomon code word contains $N=K+R$ bytes, comprised of R check bytes appended to K data bytes, where $32 \leq N \leq 255$ and $0 \leq R \leq 16$. The encoder and decoder can be implemented in hardware efficiently with 16 shift registers, multipliers and adders. A software implementation is also possible, but the performance is improved significantly if the processor core supports finite field multiplication.

The N encoded octets are split into blocks of size l octets, where N is an integer multiple of l and $N \leq 8l$. The blocks are read back out after a delay equal to $n(D-1)$ where n is the original index of the block and D is the interleaver depth. D and l are co-prime and $1 \leq D \leq 2048$ for most profiles. Some profiles allow values of D up to 4096. The interleaver and de-interleaver each require $(l-1)(D-1)/2$ octets of memory for each path.

3.3.2 Transmitter PMD block

The values read from the interleaver(s) are multiplexed together and passed to the bit allocator in the PMD block. The functions in the PMD block are shown in Figure 4 below. They create an analog DMT signal from the bit stream with a spectrum tailored to meet the PSD requirements. Data bits are allocated across the spectrum such that the required SINR matches the expected SINR for each frequency tone.



Figure 4 - Functions in the Transmitter PMD block

The bit allocator controls the Serial to Parallel (S2P) conversion of bits to tones using the bit table and tone table, which are set up during initialization. The bit table indicates the number of bits that are read from the multiplexer and the tone table indicates the tone that they will be transmitted on.

Up to 15 bits are read from the multiplexer for each tone. The bits are split into two groups ready for the QAM encoder and the two groups of bits form two integer values, x and y . If just one bit is extracted from the multiplexer then integer value x will equal y and will be 0 or 1. If trellis encoding is used then the tables will have been reordered so that single bit tones are grouped together, allowing pairs of single bit tones to be passed through the trellis encoder together.

The trellis encoder uses block processing of Wei's 16-state 4-dimensional trellis code [14]. The encoder and Viterbi decoder can be implemented in hardware efficiently with 1 bit shift registers and adders. A software implementation is also possible. The trellis encoder takes the three least significant bits read from the multiplexer and generates four bits using a convolutional encoder and some logic. The four bits are assigned two to each of the x and y values.

The two groups of bits are passed to the QAM encoder, which returns a complex value by mapping the bits onto a constellation. The constellation used depends on the number of bits. If the total number of bits is even then the output value is created by interpreting the two groups of bits as two integer components of a complex value. If the total number of bits is odd then the five most significant bits are used to generate the three most significant bits of the two components of a complex value. The remaining bits form the least significant bits of a complex value as with the even case.

The two components resulting from the QAM encoder are scaled to normalize the transmit power of the different constellations. They are also scaled by a PSD coefficient, which can include a gain adjustment. The two factors can be multiplied together in advance, so each complex output from the QAM is scaled by one real value.

The scaled complex values calculated for each tone are buffered at the input to the Inverse Discrete Fourier Transform (IDFT) block. The values are padded with zeros or optional additional values (where permitted) to increase the length to the next power of two.

The array of values can be mirrored with conjugate values, which doubles the length of the input to the IDFT and ensures the output values from the IDFT are real. However, it is more efficient to construct the input to the IDFT using two arrays of values. The first half of the input to the IDFT contains the sum of each pair from the two arrays, and the second half of the input contains the conjugate of the difference of each pair. The result of the IDFT can then be separated into the two real outputs for two symbols. The number of tones is between 1205 and 4095, so the length of the IDFT can be up to 8192. Most profiles require an IDFT of length 4096.

The largest QAM constellation requires 8 bits for each of the two components. The PSD coefficient has a resolution of 10 bits, which can also include a gain adjustment. The IDFT requires up to 12 bits to maintain precision, but the resulting total of 30 bits does not need to be transmitted, since the SINR will typically be 70 dB or worse. A 12 or 14 bit DAC is adequate, running at 70 MHz to cover all profiles.

Finally the time domain signal is extended by adding L samples from the end of the IDFT output to L samples from the end of the previous symbol and inserting the result at the beginning of the IDFT output. The samples are multiplied by FEXT pre-compensation coefficients and a further gain adjustment before the complex values are passed to the DAC. If common mode signals are to be transmitted then two DACs are necessary, however, this is not part of the VDSL recommendation due to the RFI that would be generated.

3.3.3 Receiver PMD block

The receiver will include an adjustable gain block and the ADC should have 14 or 16 bit resolution to avoid quantization distortion. If two ADCs are used then the common mode signal can be obtained. This can be used to compensate for crosstalk interference. The additional processing requires an addition and a multiply for each sample. Two ADCs would require two single ended amplifier inputs and a subtraction operation instead of one differential amplifier with two inputs.

The sampled signal is processed by the PMD block in the receiver (Figure 5). The samples may be multiplied by FEXT cancellation coefficients and passed through a Time domain Equalizer (TEQ) filter that would typically have 20 coefficients. These filters can be implemented efficiently in hardware with shift registers and multipliers. They may also be implemented using a Multiply Accumulate (MAC) operation on each coefficient for each received sample.



Figure 5 - Functions in the Receiver PMD block

The received samples are windowed to remove the L samples that were added as a cyclic prefix, and also to compensate for Fourier integral truncation. This can be included in the multiplications performed by the filters and crosstalk cancellation.

The windowed samples from two symbols are demodulated with a Discrete Fourier Transform (DFT). The sum and difference of the two halves of the result are taken (where one half is conjugated), giving the two sets of tones used to create the two symbols. The tone values are multiplied by a complex factor to adjust for frequency selective attenuation and delay in the channel, and then quantized to produce the number of bits indicated in the bit table. This is the reverse of the QAM encoding.

If the transmitted signal was trellis encoded, the received bits must be decoded with a suitable algorithm, such as the Viterbi decoder. The quantization and Viterbi decoder should be combined to minimize the bit error rate. This requires each sample to be processed by a state machine with 16 states.

3.3.4 Receiver PMS block

The Parallel to Serial (P2S) converter recombines the bits from each tone using the bit table and tone table to produce a serial bit stream, which is passed to the demultiplexer in the PMS block (Figure 6). The demultiplexer separates the two latency paths (if there is more than one) and stores the octets for each path in a buffer. The size of each buffer is the same as the size of the buffer used by the interleaver in the transmitter.

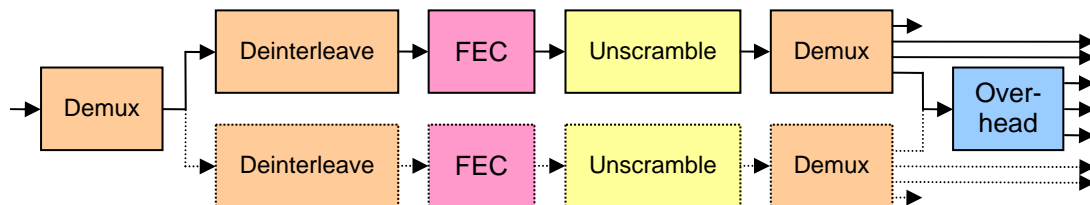


Figure 6 - Functions in the Receiver PMS block

The deinterleaver reads octets from the buffer and produces the original data stream generated by the Reed-Solomon FEC encoder (with potential bit errors). The Reed-Solomon decoder is then used to identify bit errors and correct them. It would be possible to use the bit error information to identify tones that were experiencing poor SINR and to help train interference cancellation and time domain equalizers.

The corrected bit stream is unscrambled and demultiplexed to obtain the original input data streams. The unscrambler requires the same 23 bit buffer as the scrambler. This could be implemented using shift registers in hardware, or with a logic unit and a small amount of register space.

4 Suitable Architectures

4.1 Existing systems

VDSL transceivers are already available integrated with application layer processors. Companies such as Broadcom and ikanos can supply xDSL devices that provide Ethernet, PCI, USB and voice connectivity. The system diagram for the ikanos Fusiv Vx180 is shown in Figure 7 below.

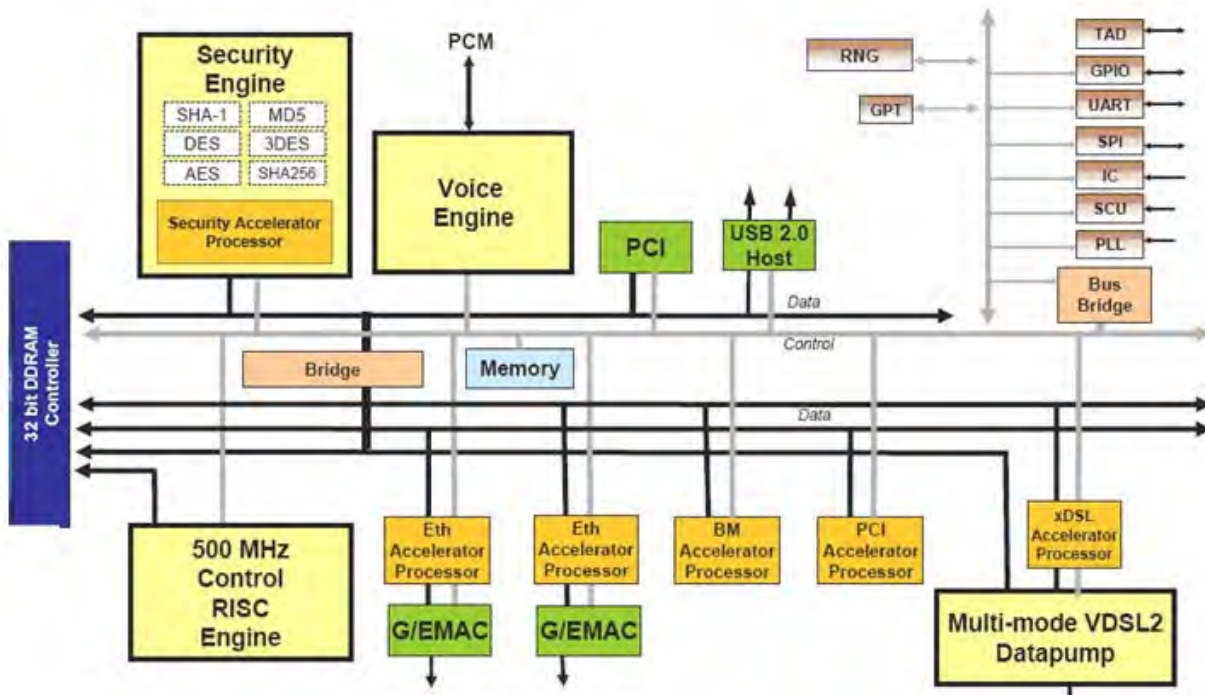


Figure 7 - System diagram of the ikanos Fusiv Vx180

The Multi-mode VDSL2 Datapump is controlled by a 400MHz processor, which allows it to run independently of the control RISC engine. This report compares alternative architectures for the VDSL Datapump.

The Broadcom BCM96519 DSLAM solution supports vectored VDSL, which mitigates transmitted and received crosstalk. The existing solution requires feedback from the receiver, which may be incorporated in the ITU G.vector recommendation.

4.2 Implementation 1: Configurable pipeline

The processing described in section 3.3 can be performed efficiently using Multiple Instruction Single Data (MISD) parallel architecture. This is effectively a dedicated hardware pipelined architecture as shown in Figure 8 and Table 1 below.

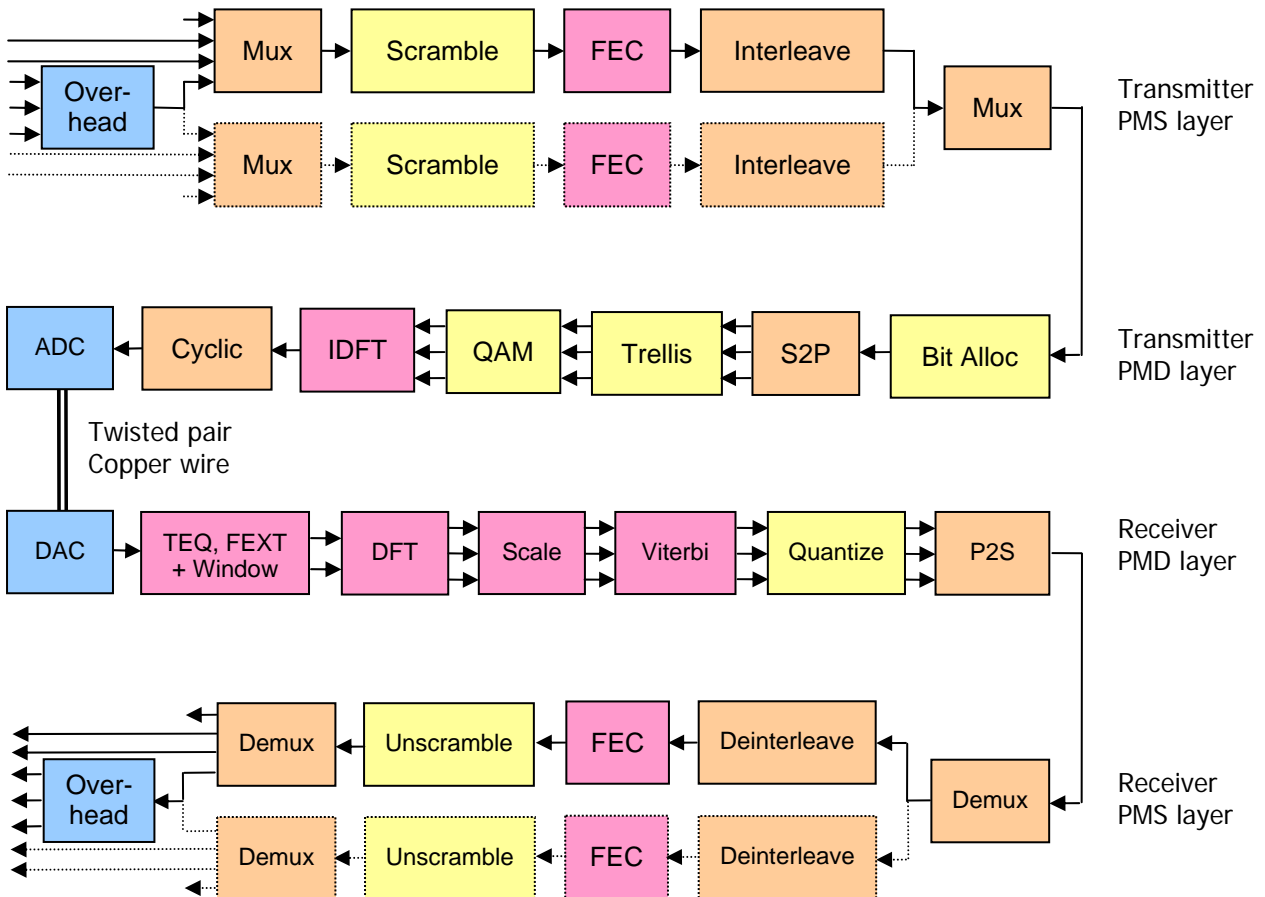


Figure 8 - VDSL Processing diagram

This architecture represents a simple approach, implementing each functional unit as it appears in the specification. It would be possible to add a degree of configurability to this architecture to give it some flexibility; however the discrete nature of the functional blocks will limit this.

Table 1 describes the operations required for each function, along with the type of data unit being operated on.

Function	Hardware	Data Unit
Multiplexer		byte
Scrambler	Bit wide shift register with taps at 18 and 23 Two XOR gates	bit
Reed-Solomon encoder	Byte wide shift register with 15 taps 15 multiply and add units	byte
Interleaver	64 KB buffer supporting 1 byte reads	byte
Multiplexer		byte
Convolutional encoder	3 x 1 byte adder, 4 x 1 byte delay, 4 x 1 byte adder	bit
Constellation mapping	Logic to generate 6 bits from 5	tone
Scaling	Two 8 bit multiplies	tone
IDFT	8192 point 16 bit FFT (6 stages 2048 Radix-4)	symbol
Cyclic prefix	Up to 4096 additions (typically 640)	symbol
FEXT / Gain / Window	16 bit multiply	sample
14bit DAC		sample
Line driver		
16bit ADC		sample
TEQ ¹ / Gain / Window	20 x 16 bit multiply and add	sample
DFT	8192 point 16 bit FFT (6 stages 2048 Radix-4)	symbol
Compensation	One complex multiply	tone
Viterbi decoder	16 state, 4 input, 2 output state machine	bit
Demultiplexer		byte
Deinterleaver	64 KB buffer supporting 1 byte reads	byte
Reed-Solomon decoder stage 1	32 Byte wide shift registers 32 multiply and add (to obtain syndrome)	byte
Reed-Solomon decoder stage 2 (on error)	Up to 16 iterations with 3 multiplies, 1 divide, 1 add and 2 tests (to obtain error polynomial) Up to 256 tests (to find roots of polynomial) Up to 16 convolutions (8 MACs each) and 16 divides	256 bytes
Unscrambler	Bit wide shift register with taps at 18 and 23 Two XOR gates	bit
Demultiplexer		byte

Table 1 - Dedicated hardware for VDSL processing

The rate at which the operations in Table 1 are performed depends on what the operation is performed on. This is indicated by the Data Unit for the operation, where the maximum rate of each unit is as follows:

- The bit rate is up to 200 Mbit/s
- The byte rate is up to 25 MB/s
- The tone rate is up to 30 MHz
- The symbol rate is 8.625 kHz
- The sample rate is up to 40 MHz

¹ The Time Domain Equalizer is only operational if there is Inter-Symbol Interference on the line

4.2.1 Implementation

The simple nature of this solution makes it relatively straightforward to architect and model, capture the design and verify, and therefore the functional section of the design would incur relatively low development costs. Capturing the physical design, whilst unlikely to pose any real problems, would not be as efficient as a design that possesses a more regular structure.

4.2.2 Modelling Results

Using industry average data for a 65nm silicon process and a clock speed of 200MHz, the required logic and memory resources to support the performance requirements of the VDSL specification are as follows:

Silicon Area (excluding analog circuitry) : 1.66 sq mm

Power Consumption: 85mW

4.3 Implementation 2: SIMD array

The processing described in section 3.3 can be performed using a Single Instruction Multiple Data (SIMD) processor as shown in Figure 9 below. This solution represents a more complex, but potentially much more flexible approach.

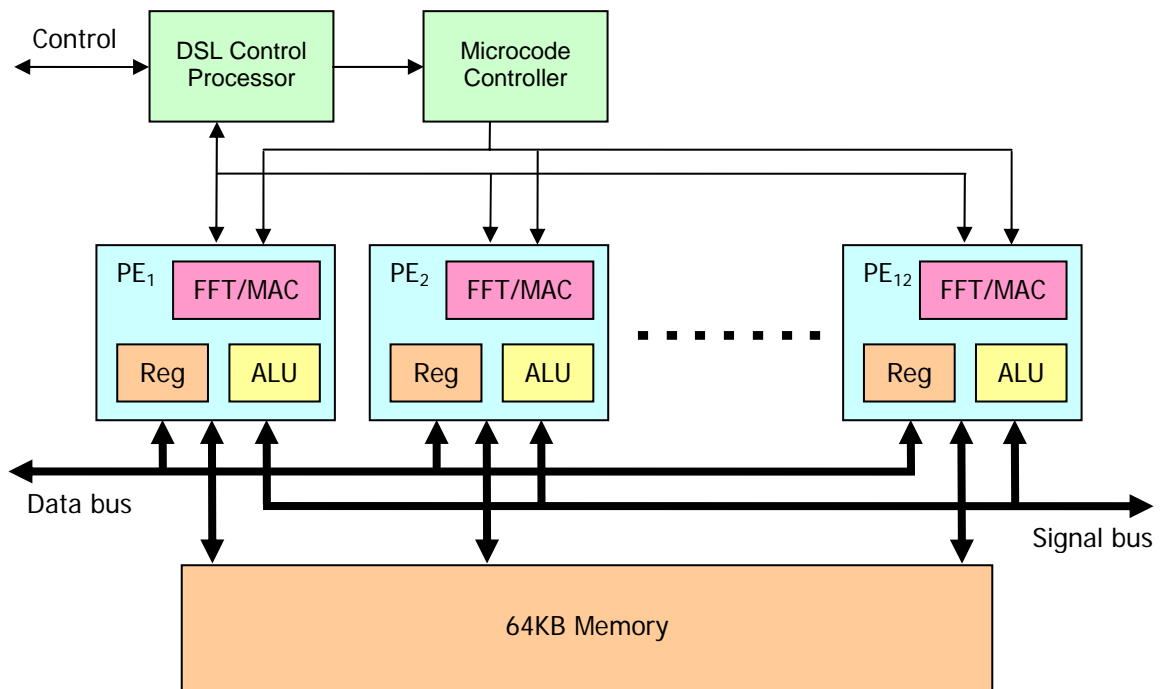


Figure 9 - SIMD Array diagram

4.3.1 Architectural Details

The maximum number of processing elements in the SIMD array can be derived from the data rate and symbol rate as follows:

- The maximum net data bit rate is 200 Mbit/s
- The maximum bit rate including Overhead bytes and FEC is 237.3 Mbit/s
- The symbol rate at this bit rate is 8625 Hz
- Each symbol carries 3438 bytes produced by the FEC encoder
- FEC decoding of 255 bytes is most easily performed on one processing element
- 3438 bytes divided by 255 bytes leads to a maximum of 13 processing elements

The Fourier Transform is performed using all of the processing elements in parallel, and this is most easily achieved if there are 8 or 12 processing elements. The FFT and interleaving require that each PE can access all of the available memory. The maximum interleaving and deinterleaving depth requires a total of

64 KB of memory. If the data rate is less than the maximum then some of the PEs can be disabled. The processing performed by the SIMD array is described in Table 2 below.

Function	Operations	Data Unit
Multiplexer		byte
Scrambler	1 shift and 3 adds	bit
Reed-Solomon encoder	15 MACs	byte
Interleaver	64 KB buffer supporting 1 byte reads	byte
Multiplexer		byte
Convolutional encoder	7 Adds	bit
Constellation mapping	Lookup to generate 6 bits from 5	tone
Scaling	Two 8 bit multiplies	tone
IDFT	8192 point 16 bit FFT (0.5 GOP)	symbol
Cyclic prefix	Up to 4096 Adds (typically 640)	symbol
FEXT / Gain / Window	16 bit multiply	sample
14bit DAC		sample
Line driver		
16bit ADC		sample
TEQ ¹ / Gain / Window	20 x 16 bit multiply and add	sample
DFT	8192 point 16 bit FFT (0.5 GOP)	symbol
Compensation	One complex multiply	tone
Viterbi decoder	2 test + jump	bit
Demultiplexer		byte
Deinterleaver	64 KB buffer supporting 1 byte reads	byte
Reed-Solomon decoder stage 1	32 MAC (to obtain syndrome)	byte
Reed-Solomon decoder stage 2 (on error)	Up to 48 multiplies, 16 divides, 16 adds and 32 tests (to obtain error polynomial). Up to 256 test + jump (roots of polynomial). Up to 16 convolutions (8 MACs each) and 16 divides	256 bytes
Unscrambler	1 shift and 3 adds	bit
Demultiplexer		byte

Table 2 - SIMD array for VDSL processing

4.3.2 Implementation

The more complex nature of this solution makes it more difficult to architect and model the design, however its regular structure will make it more straightforward to capture the design and verify, and therefore the functional section of the design would incur similar development costs to the pipeline architecture.

It is likely that the highly regular structure will simplify the physical design and therefore reduce the cost of physical implementation overall.

¹ The Time Domain Equalizer is only operational if there is Inter-Symbol Interference on the line

4.3.3 Modelling Results

Using industry average data for a 65nm silicon process and a clock speed of 200MHz, the required logic and memory resources to support the performance requirements of the VDSL specification are as follows:

Silicon Area (excluding analog circuitry) : 1.04 sq mm

Power Consumption: 64mW

4.4 Architecture Comparison

Both implementations require an analog front end (DAC, ADC), although the line driver could exist in a separate package. They could also include a high pass filter, implemented in CMOS [15], avoiding the need for an external filter. As these elements are required for all solutions, they have not been included in the comparison figures here. The table below compares the two architectures detailed above.

Architecture	Strengths	Weaknesses
Pipelined	<ul style="list-style-type: none"> • Simple architecture • Easy to ensure specification may be met 	<ul style="list-style-type: none"> • Lack of flexibility • Higher cost part • Higher power consumption • Redesigns necessary to keep pace with technology updates
SIMD	<ul style="list-style-type: none"> • Flexible architecture • Low cost • Low power consumption • Technology platform: upgrades by software as improvements to techniques are made. 	<ul style="list-style-type: none"> • Complex architecture • Harder to model and ensure specification may be met.

4.5 SIMD Architecture Extension – Flexible DSLAM

It is possible to extend the SIMD architecture of section 4.3, to allow for a highly flexible DSLAM part. The number of Processing Elements (PEs) per channel can be reduced where the bandwidth required for a channel is lower, for example where the length of the copper wire makes only a lower bandwidth connection possible. It is likely that the required number of PEs per channel will be between 4 and 12 depending on the possible connection bandwidth. Where fewer than 12 PEs are required it is possible to add further configurability to the architecture to reuse the spare PEs to create additional channels. This way a block of 12 PEs may be used to provide for a single channel of full bandwidth, or 3 channels of lower bandwidth requiring 4 PEs each, and this way most efficient use is made of the hardware resource. This is shown in Figure 10.

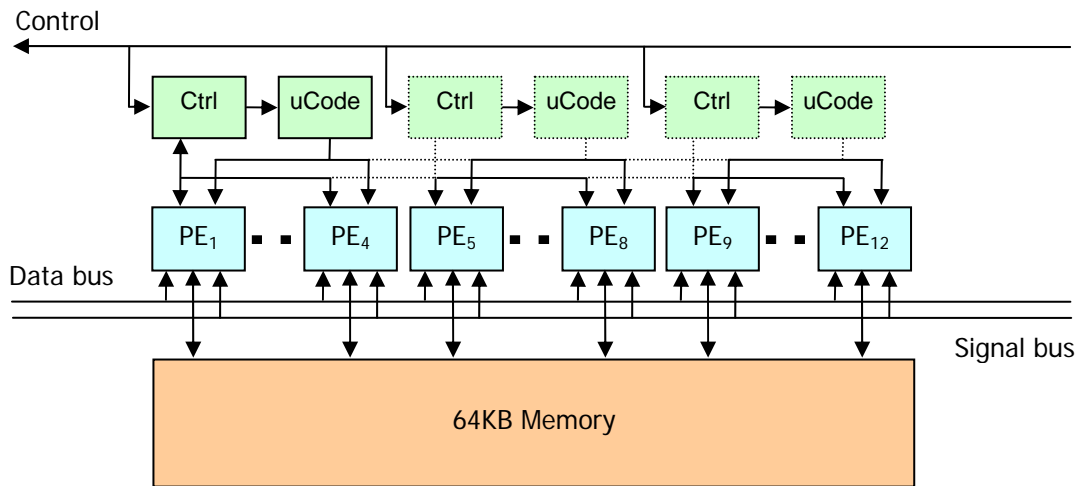


Figure 10 – DSLAM Architecture, Single Configurable DSLAM Block

For a DSLAM part, it may be appropriate to instance many blocks of 12 PEs to provide many more flexible bandwidth channels. Figure 11 shows a 24 PE architecture implemented as 2 blocks of 12, allowing 2 full bandwidth channels or 6 lower bandwidth channels. This approach can scale much larger than this, so for example a 288 PE device would allow for 24 full bandwidth channels or 72 lower bandwidth channels or a mix of both as required.

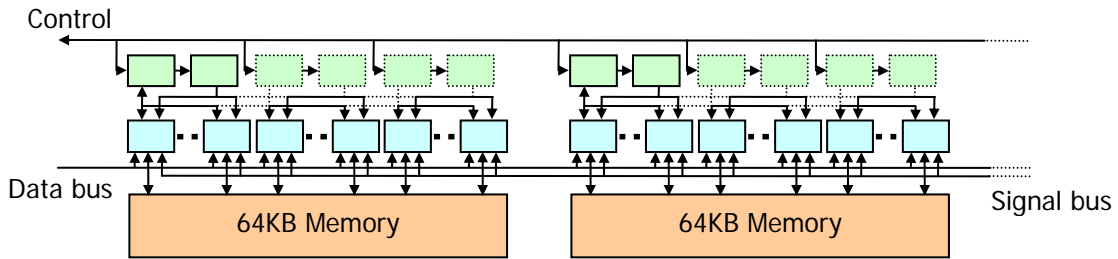


Figure 11 - DSLAM Architecture, Two Configurable DSLAM Blocks

5 Summary and Conclusions

It is possible to utilize different signal processing algorithms to improve the performance of copper wire connections, and enable higher bandwidth connections for an FTTC network upgrade. Improving the performance of existing copper infrastructure will provide significant cost advantages over new FTTH installations.

These improvement techniques are still evolving and to avoid the need for expensive upgrade or replacement programs it is highly desirable that the data processing sections of the VDSL PMD and PMS blocks are implemented as flexibly as possible, and support software upgrade to enable new techniques and deliver higher connection bandwidth without the need for new hardware.

The study began by reviewing data on the current techniques and algorithms to improve the performance of copper wire connections and has reviewed the processing requirements of these techniques together with the processing requirements of the VDSL specification. Two approaches, one configurable and one programmable were selected and analysed in more detail. Both approaches were modelled to estimate the required hardware resource to meet both the VDSL specification and the requirements for additional performance improvement, and the strengths and weaknesses of the two solutions compared.

The study concludes that there is considerable potential to develop a power efficient programmable architecture with the scope to provide increased performance and flexibility. The highlighted approach is modular and scalable and builds on proven approaches which have been demonstrated to ease physical implementation problems and benefit from architectural energy efficiencies.

The SIMD architecture presented in this report represents a flexible software defined solution, which has the additional benefits of being lower power and lower unit cost than alternatives. It would also appear that this solution will not incur any greater development cost than less flexible alternatives, and may even be lower. It will also provide a more flexible way of making a DSLAM part, allowing for trading the number of channels against the bandwidth of each channel.

6 About ClearSpeed

ClearSpeed Technology is a fabless semiconductor company based in Bristol, UK, and is a world leading provider of low energy, high reliability, high performance processors. Our integrated circuit, board level and intellectual-property product range enables software defined solutions for imaging, pattern matching, data encryption, synthetic aperture radar, digital signal processing, multidimensional FFT and many data processing applications especially where size, weight and power are constrained. Backed by a mature, easy to use software development kit based on the 'C' language and a hardware development kit, our processor products are easily integrated into many systems; they are designed for high reliability by incorporating error correction, and hardware redundancy.

ClearSpeed Technology is also able to provide silicon and software design consultancy services. The engineering team is highly experienced with expertise in designing complex, high performance, low power systems on chip. The team has a track record of right-first-time design and has good relationships with silicon foundries. Our software expertise includes development of FFT and DSP libraries and applications, compiler and tool chain development, and modelling of large SOCs.

7 References

- [1] ITU-T Recommendation G.993.2, "Very high speed digital subscriber line transceivers 2 (VDSL2)", Series G: Transmission systems and media, digital systems and networks, February 2006
- [2] George Ginis and John M. Cioffi, "Vectored Transmission for Digital Subscriber Line Systems", IEEE Journal on selected areas in communications, Vol. 20, No. 5, June 2002
- [3] Raphael Cendrillon, Marc Moonen, George Ginis, Katleen Van Acker, Tom Bostoen, Piet Vandaele, "Partial Crosstalk Cancellation for Upstream VDSL", EURASIP Journal on Applied Signal Processing 2004:10, 1520-1535
- [4] Raphael Cendrillon, George Ginis, Marc Moonen, Katleen Van Acker, "Partial crosstalk precompensation in downstream VDSL", Elsevier Signal Processing 84, April 2004
- [5] Thomas Magesacher, Per Odling, Per Ola Borjesson, Tomas Nordstrom, "Exploiting the Common-Mode Signal in xDSL"
- [6] Paschalis Tsiaflakis, Yung Yi, Mung Chiang, Marc Moonen, "Dynamic Spectrum Management for Green DSL"
- [7] Milos Jakovljevic, Drinton Statovci, Tomas Nordstrom, Rickard Nilsson, Santiago Zazo, "VDSL Power Back-Off Parameter Optimization for a Cable Bundle", 15th European Signal Processing Conference, 2007
- [8] Raphael Centrillon, George Ginis, Etienne Van den Bogaert, Marc Moonen, "A Near-Optimal Linear Crosstalk Canceler for VDSL", IEEE Transactions on Signal Processing, Vol. 54, No. 8, 2006
- [9] Raphael Centrillon, George Ginis, Etienne Van den Bogaert, Marc Moonen, "A Near-Optimal Linear Crosstalk Precoder for Downstream VDSL", IEEE Transactions on Communications, Vol. 55, No. 8, 2007
- [10] Amir R. Forouzan and Lee M. Garth, "Computationally Efficient Partial Crosstalk Cancellation in Fast Time-Varying DSL Crosstalk Environments", EURASIP Journal on Advances in Signal Processing, ID 72041, 2007
- [11] Patrick Duvaut, Pravesh Biyani, Amitkumar Mahadevan, Shailendra Singh, Satyavardhan Maheshwari, "Adaptive Off-Diagonal MIMO Cancellor (ODMC) for VDSL Upstream Self FEXT Mitigation", 16th European Signal Processing Conference (EUSIPCO 2008), August 2008
- [12] Yuan-Pei Lin, Lo-Han Liang, Pei-Ju Chung, See-May Phoong, "An Eigen-Based TEQ Design for VDSL Systems", IEEE Transactions on Signal Processing, Vol. 55, No. 1, January 2007

- [13] Sreekanth Marti, "A Bandwidth Efficient Turbo Coding Scheme for VDSL Systems", A Thesis in the Department of Electrical and Computer Engineering, Concordia University, April 2003
- [14] Lee-Fang Wei, "Trellis Coded Modulation with Multidimensional Constellations", IEEE Transactions on Information Theory, Vol. 33, No. 4, July 1987
- [15] Fang Lin, Xinyu Yu, S. Ranganathan, T. Kwan, "A 70dB MTPR integrated programmable gain/bandwidth 4th order Chebyshev highpass filter for ADSL/VDSL receivers in 65nm CMOS", IEEE Symposium on VLSI Circuits, August 2008