(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
7 January 2010 (07.01.2010)

PCT

(10) International Publication Number
WO 2010/002908 A2

(51) International Patent Classification:
H04L 29/02 (2006.01)

(21) International Application Number:
PCT/US2009/049283

(22) International Filing Date:
30 June 2009 (30.06.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/077,169 1 July 2008 (01.07.2008) US

(71) Applicant *(for all designated States except US)*: **VECTOR SILICON, INC.** [US/US]; 728 Alester Avenue, Palo Alto, CA 94303 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SANDS, Nicholas, P.** [US/US]; 111 Pope Street, Menlo Park, CA 94025 (US). **FISHER, Kevin, D.** [US/US]; 728 Alester Avenue, Palo Alto, CA 94303 (US).

(74) Agent: **SYLKE, Thomas, C.**; Sylke Law Office, LLC, 756 N. Milwaukee Street, Suite 210, Milwaukee, WI 53202-3719 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*

(54) **Title**: REDUCED MEMORY VECTORED DSL

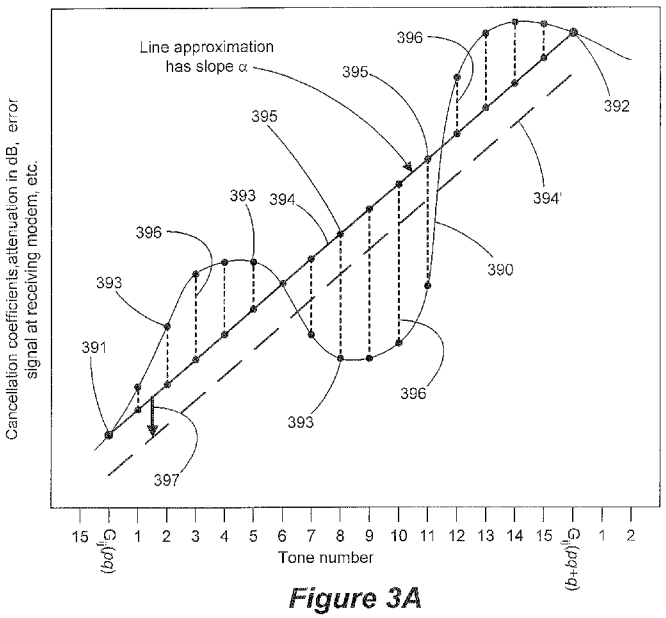


Figure 3A

(57) **Abstract**: A reduced-memory vectored DSL system includes methods and apparatus for reducing the bandwidth and memory storage demands on a vectored DSL system in which FEXT data is transmitted and stored. An upstream-end device such as a DSLAM communicates with a plurality of downstream-end devices such as CPE modems. When test signal data, such as training and/or tracking data, is sent to determine FEXT characteristics of the DSL system, error signals are available for all or substantially all of the upstream and/or downstream frequency band DSL tones used in the system. Dividing a frequency band into sub-bands, only a subset of tones in each sub-band is used for deriving FEXT data, such as a FEXT channel response, FEXT channel coefficients and/or FEXT cancellation coefficients. For tones in the sub-band subsets, full-precision FEXT data values can be derived. For other tones, approximations of the FEXT data can be derived. Alternatively, FEXT data can be defined as a base point and associated differential/incremental values that define full-precision and/or approximation FEXT data with fewer bits by utilizing a Δ value relative to either a sub-band base point or relative to another tone's value. Memory is reduced in both the transmission of such FEXT data (between upstream and downstream ends of the DSL system) as well as within an upstream-end device such as a DSLAM that performs vectoring using a separate or internal vectoring processing apparatus. Memory also is reduced in the storage of such FEXT data in or off of the DSLAM or other upstream-end device.

WO 2010/002908 A2

REDUCED MEMORY VECTORED DSL

PRIORITY CLAIM AND CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of priority under 35 U.S.C. § 119(e) and any other United States or other law of the following:

U.S. Serial No. 61/077,169 (Atty. Docket No. 2202-p92p) filed July 1, 2008, entitled ADAPTIVE INTER-TONE DIFFERENTIAL ENCODING OF FAR-END CROSSTALK CHANNEL ESTIMATES IN A DISCRETE MULTI-TONE SYSTEM, the entire disclosure of which is incorporated herein by reference in its entirety for all purposes.

TECHNICAL FIELD

The present invention relates generally to the field of vectored DSL communication systems.

BACKGROUND

Digital Subscriber Line (DSL) communications suffer from various forms of interference, including crosstalk. Various techniques have been developed to combat this interference and its deleterious effects on data transmission. Vectored DSL has been one such measure in development for some time. The amount of data that must be transmitted between downstream and upstream modems is substantial for vectored DSL systems and this vectoring data pertaining to FEXT interference also must be stored and manipulated/processed by upstream device such as vectoring modules and the like. Systems, apparatus and/or other hardware implementations that provide reduced memory transmission and storage of vectored DSL data would represent a significant advancement in the art.

SUMMARY

Embodiments of the present invention include apparatus and methods pertaining to reduced-memory generation, transmission, use and storage of FEXT data in DSL systems. In approximation-based embodiments, the relevant frequency bands of the DSL system are further divided into sub-bands, wherein each sub-band typically (though not always) contains an equal number of tones. For example, where a given DSL frequency band is divided into sub-bands, the number of DSL tones in the divided frequency band might not be an even multiple of the number of uniform sub-bands, so a minor number of sub-bands might not have the standard tone count. Therefore, in dividing a frequency band, "substantially all sub-bands" will have the same number of DSL tones per sub-band.

Test signals received as a receiving modem contain information relating to FEXT in substantially all tones of the DSL system in which the receiving modem operates. Only the FEXT data pertaining to a subset of the tones in each sub-band is used in generating an approximation of the FEXT channel or other FEXT data. In some embodiments, the FEXT data pertaining to the subset of tones is used to generate a linear or polynomial approximation of the FEXT cancellation coefficient plot. The tones for which actual FEXT error signals are available can be used to generate full-precision FEXT cancellation coefficients, while the tones not included in the sub-band subsets have their FEXT cancellation coefficients approximated. Where the FEXT cancellation coefficients are used in pre-cancelling downstream FEXT, less bandwidth is needed for CPE modems or the like to return FEXT data for vectoring – rather than sending FEXT data for all or substantially all tones in the DSL system (or all or substantially all downstream frequency band tones for the DSL system), only FEXT data for the subset of tones in each sub-band need to be transmitted upstream to the DSLAM or other upstream-end device performing vectoring. In the upstream-end device, less memory storage is required to store vectoring data because of the use of the approximation of the FEXT cancellation coefficients. This also can reduce upstream-end bandwidth usage where a separate vectoring processing system is used that is coupled to the DSLAM or upstream modems and where FEXT data (for example, error signals representing FEXT effects, FEXT channel response data, FEXT cancellation coefficients, etc.) are transmitted between the vectoring processing apparatus and the upstream modems or DSLAM. Where upstream FEXT is involved, the downstream-to-upstream bandwidth transmission bandwidth is not an issue, but

FEXT data storage and upstream-end bandwidth usage still benefits substantially from the approximation techniques of some embodiments of the present invention.

In other embodiments of the present invention, methods and apparatus utilize a differential/incremental value system that characterizes or defines FEXT data (for example, FEXT cancellation coefficients) in terms of a sub-band base point and incremental or differential values that describe each tone's FEXT cancellation coefficient for a given tone in the sub-band as either (1) a base point value plus differential value for the particular tone, or (2) as the previous tone's coefficient value plus a differential/incremental value for the next tone's coefficient.

In still other embodiments of the present invention, both the approximation and differential/incremental value methods, apparatus and/or techniques can be combined to provide other vectored DSL systems and operation.

BRIEF DESCRIPTION OF DRAWINGS

The present invention will be readily understood by the following detailed description in conjunction with the accompanying drawings, wherein like reference numerals designate like structural elements, and in which:

Figure 1 is a (MIMO) system in which embodiments of the present invention can be implemented.

Figure 2 is a typical FEXT channel response from 0 to 30MHz from a telco binder cable.

Figure 3A illustrates one embodiment of the present invention employing a linear approximation of FEXT-related data.

Figure 3B illustrates one embodiment of the present invention employing a polynomial approximation of FEXT-related data.

Figure 3C illustrates one embodiment of the present invention employing a differential/incremental value plot and/or approximation of FEXT-related data.

Figures 4 and 5 are flow diagrams of operation of DSL systems in which an approximation and/or differential/incremental value definitions of FEXT-related data are used to reduce bandwidth and/or memory storage demands.

Figures 6A-6F are block diagrams of vectored DSL systems according to one or more embodiments of the present invention.

Figure 7 is a block diagram of a DSL data processing stream in which embodiments of the present invention can be implemented..

Figure 8 is a block diagram of a DSLAM modem in which embodiments of the present invention can be implemented.

Figures 9A-9D are block diagrams of DSLAM modem chip interfaces with a vectored DSL system according to one or more embodiments of the present invention.

Figures 10A-10E are various block diagrams of vectoring data network configurations according to one or more embodiments of the present invention.

Figure 11 is a block diagram showing an exemplary data transmission configuration for use in connection with one or more embodiments of the present invention.

Figure 12 is an exemplary logic implementation of a vector processor unit according to one or more embodiments of the present invention.

Figure 13 is an exemplary vector processor implementing two VPUs and interfacing with at least one modem chip according to one or more embodiments of the present invention.

Figure 14 is a DSLAM architecture embodiment showing a dedicated vectoring module with four line cards providing user data for vectoring.

Figure 15 illustrates clock synchronization of components in a vectored DSL system according to one or more embodiments of the present invention.

Figure 16 is an example of a packet structure according to one or more embodiments of the present invention.

DETAILED DESCRIPTION

The following detailed description will refer to one or more embodiments, but the present invention is not limited to such embodiments. Rather, the detailed description and any embodiment(s) presented are intended only to be illustrative. Those skilled in the art will readily appreciate that the detailed description given herein with respect to the Figures is provided for explanatory purposes as the invention extends beyond these limited embodiments.

Certain terms are used throughout the description and claims to refer to particular system components. As one skilled in the art will appreciate, computer and other companies may refer to components by different names. This disclosure does not intend to distinguish between components that differ insubstantially. Phrases such as "coupled to" and "connected to" and the like are used herein to describe a connection between two devices, elements and/or components and are intended to mean physically and/or electrically either coupled directly together, or coupled indirectly together, for example via one or more intervening elements or components or via a wireless connection, where appropriate. The term "chip" refers broadly to a hardware device that operates in a prescribed manner, for example to process data, and can include various types of such devices (for example, a field-programmable gate array (FPGA), a digital signal processing (DSP) chip, an application-specific integrated circuit (ASIC), an integrated circuit (IC), etc.). The term "system" refers broadly to a collection of two or more components and may be used to refer to an overall system (e.g., a computer and/or communication system or a network comprising one or more computers, communication components, etc.), a subsystem provided as part of a larger system (e.g., a subsystem within an individual computer), and/or a process or method pertaining to operation of such a system or subsystem. In this specification and the appended claims, the singular forms "a," "an," and "the" include plurals unless the context clearly dictates otherwise. Unless defined otherwise, technical and scientific terms used herein have the same meanings that are not inconsistent to one of ordinary skill in the art relevant subject matter disclosed and discussed herein.

Reference in the specification to "some embodiments," "one embodiment," "an embodiment," etc. of the present invention means that a particular feature, structure or characteristic described in connection with such embodiment(s) is included in at least one embodiment of the present invention. Thus, the appearances of the noted phrases in various

places throughout the specification are not necessarily all referring to the same embodiment.

In the absence of crosstalk, the existing copper telephone infrastructure can in theory be utilized to carry from tens to hundreds of megabits per second over distances up to approximately 5000 feet using discrete multitone (DMT) digital subscriber line (DSL) modem technology. DMT modems divide the available bandwidth into many sub-carriers (also referred to as "tones" or "frequency bins") that are synchronized and independently modulated with digital QAM data to form an aggregate communication channel between the network and subscriber. DMT-based DSL systems typically use Frequency Division Multiplexing (FDM) and assign particular sub-carriers to either downstream (that is, from DSLAM/ CO to CPE/user) or upstream (from CPE/user to DSLAM/CO) directions. This FDM strategy limits near end crosstalk (NEXT). DMT systems are typically very robust in the presence of radio frequency interference (RFI) and other types of frequency-selective noise (or interference) or channel dispersion, because each sub-carrier can be independently modulated with an appropriate amount of data and power in order to meet the system requirements and the desired bit error rate.

Typical telephone binder cables typically contain between 10 and 50 unshielded copper twisted pairs (these binders are then bundled into larger cables). Due to variations in manufacturing and imperfect cable installation (for example, bending, etc.), significant levels of far-end crosstalk (FEXT) occur between each twisted pair within a binder (and between adjacent binders). More specific discussions concerning FEXT can be found in the various patent applications filed by Adaptive Spectrum and Signal Alignment of Redwood City, California and Stanford University. In particular, the following patents and published applications are incorporated herein by reference for all purposes:

Document No.	Pub. date	Title
US 6,512,797	01/28/03	Peak to average power ratio reduction
US 6,424,681	07/23/02	Peak to average power ratio reduction
US 6,314,146	11/06/01	Peak to average power ratio reduction
US 7,394,752	07/01/08	Joint reduction of NEXT and FEXT in xDSL systems
US 7,158,563	01/02/07	Dynamic digital communication system control
US 7,027,537	04/11/06	Iterative multi-user detection
US 6,990,196	01/24/06	Crosstalk identification in xDSL systems
US 6,829,307	12/07/04	Express bit swapping in a multicarrier transmission system
US 5,479,447	12/26/95	Method and apparatus for adaptive, variable bandwidth,

		high-speed data transmission of a multicarrier signal over digital subscriber lines
US 20020136397	09/26/02	Crosstalk identification in xDSL systems
US 20030086362	05/08/03	Joint reduction of NEXT and FEXT in xDSL systems
US 20030086514	05/08/07	Dynamic digital communication system control
US 20070081582	04/12/07	Dynamic digital communication system control
WO/2005/057857	06/23/05	DSL system estimation and parameter recommendation
WO/2005/057315	06/23/05	Adaptive margin and band control
WO/2005/057837	06/23/05	Adaptive FEC codeword management
WO/2005/094052	10/06/05	High speed multiple loop DSL system
WO/2005/114861	12/01/05	FEXT determination system
WO/2005/114924	12/01/05	Incentive-based DSL system
WO/2006/051359	05/18/06	Interference cancellation system
WO/2006/018716	02/23/06	DSL data collection system
WO/2006/048770	05/11/06	Communication device identification
WO/2006/092730	09/08/06	DSL system estimation including known DSL line scanning and bad splice detection capability
WO/2006/103557	10/05/06	DSL state and line profile control
WO/2006/120511	11/16/06	QR decomposition in DSL equalizers using real givens rotations
WO/2006/120521	11/16/06	DSL system estimation and control
WO/2006/120510	11/16/06	Phantom use in DSL systems
WO/2006/129141	12/07/06	Tonal precoding in multi-user vectored DSL transmission
WO/2006/129140	12/07/06	DSL system training
WO/2006/129143	12/07/06	Adaptive GDFE
WO/2006/131792	12/14/06	User-preference-based DSL system
WO/2006/131794	12/14/06	User-preference-based DSL system
WO/2006/131793	12/14/06	DSL system loading and ordering
WO/2006/131791	12/14/06	Vectored DSL nesting
WO/2006/136883	12/18/06	Dynamic minimum-memory interleaving
WO/2006/129145	12/07/06	Self-learning and self-adjusting DSL system
WO/2006/120513	11/16/06	Binder identification
WO/2007/008835	01/18/07	DSL system estimation
WO/2007/008836	01/18/07	Adaptive margin and band control
WO/2007/044326	04/19/07	DSL system
WO/2007/130877	11/15/07	Methods and apparatus to combine data from multiple sources to characterize communication systems
WO/2007/130878	11/15/07	Methods and apparatus to perform line testing at customer premises
WO/2007/130879	11/15/07	Video streaming diagnostics
WO/2007/146048	12/21/07	Vectored DSL system
WO/2008/016585	02/07/08	Method and apparatus for analyzing and mitigating noise in a digital subscriber line
WO/2008/045332	04/17/08	Interference cancellation system
WO/2008/045525	04/17/08	High speed multiple user multiple loop DSL system
WO/2008/073327	06/19/08	DSL ring signal compatibility

As a result of typical cable construction, installation and deployment techniques, each twisted pair is physically adjacent to many other twisted pairs in (or outside) its binder for a significant distance in the cable. While there is variation in the FEXT levels as a result of manufacturing variation, the net result is that every copper pair imparts some amount of FEXT onto virtually every other copper pair in the cable. In many instances, the FEXT can be as much as or more than 40 dB higher than the background noise level on the line and the crosstalk typically increases with frequency. Cable measurements have shown that each particular line has at least, and often more than, 5 to 10 disturbers that impact the line's performance. As a result, FEXT is the dominant impairment in reaching high data rates in DMT DSL systems on relatively short loops (for example, below 4,000 feet).

An advanced signal processing technique called "vectoring" can cancel FEXT between twisted pairs within copper telephone cables and permit dramatically higher data rates. Vectoring is well known to those skilled in the art, and is explained in more detail in "Vectored Transmission for Digital Subscriber Line Systems," by G. Ginis and J. Cioffi (IEEE JSAC special issue on twisted-pair transmission, Vol. 20, Issue 5, pp. 1085-1104, June 2002); and "Vectored-DMT: A FEXT Canceling Modulation Scheme for Coordinating Users," by G. Ginis and J. Cioffi (Proceedings of IEEE ICC 2001, Vol. 1, Helsinki, Finland, pp. 305-09, June 2001), both of which are incorporated by reference herein for all purposes.

The basic principle behind vectoring is coordinating communications between every twisted pair in a telephone cable. Because all twisted pairs in the cable terminate in one central location known as a DSLAM (DSL access multiplexer), the DSLAM is the natural location to coordinate this communication. To vector the modems effectively in the vectoring systems, the modems must be synchronized to the same DMT symbol clock and must have synchronized training/tracking periods with pre-arranged training data patterns (unique to each modem) to allow victim modems to identify the crosstalk channel characteristics between particular disturber-victim modem pairs. As will be appreciated by those skilled in the art, there are a variety of techniques for identifying crosstalk in such systems, including one or more that may be described in various applicable standards pertaining to DSL systems and their implementation.

Using "crosstalk topography" or mapping, which typically is viewed or considered as a

matrix representing the various interactive/crosstalking relationships between DSL lines determined during training, tracking and the like, transmit pre-compensation (for example, precoding) can be used in the downstream direction (from DSLAM to customer site). In effect, the transmit waveform for a given modem is pre-distorted based on the transmit waveforms being used by many other modems (that is, the other modems whose FEXT materially affect the given modem's signals), such that the interfering FEXT is effectively balanced (counteracted) and the FEXT's effects thus eliminated by the time the pre-distorted modem signal reaches its customer side terminus. In the upstream direction (from customer site to DSLAM), MIMO (multiple-input-multiple-output) spatial filtering is used at the DSLAM to cancel upstream FEXT at the upstream receive side. Identification and tracking of the FEXT filter cancellation coefficients may be performed using the LMS (Least Mean Squares) adaptive algorithm or other Least-Squares type methods, as is well known to those skilled in the art. Training data (for example, training signals) to support identification of the FEXT filter coefficients may be integrated into a modified version of a known DSL standard, such as VDSL2.

Embodiments of the present invention utilize techniques and apparatus to reduce the memory needed for both transmission/communication from downstream CPE apparatus to upstream DSLAM apparatus or the like of what will be referred to as "FEXT data" (for example, received error signals or other signals that are proxies for the FEXT interference on a given DSL line, FEXT compensation filter coefficients, FEXT channel estimates or coefficients, a FEXT compensation matrix G, back-channel communication signals conveying any of this information, etc.) as well as storage of FEXT data used for vectoring in DSLAM side modems. Embodiments of the present invention can be used to realize advantageous reductions in the amount of data needed/used for FEXT channel estimates and/or FEXT crosstalk cancellation coefficients. Moreover, these embodiments reduce memory storage requirements, reduce memory bandwidth usage and reduce sync symbol error communication from the customer premises equipment (CPE) side during training and/or tracking.

Embodiments of the present invention use one or a combination of two approaches - (1) the use of approximations of crosstalk attenuation data, crosstalk cancellation data and/or crosstalk cancellation coefficients, and/or (2) the use of incremental/differential value based representations of such crosstalk attenuation data, crosstalk cancellation data and/or crosstalk cancellation coefficients. Again, implementation of one or a combination of these embodiments

yields (1) a vectored DSL system that imposes less of a bandwidth burden for transmissions of data from CPE side collectors of crosstalk data to DSLAMs, and/or (2) a vectored DSL system that has lower data storage requirements at the DSLAM side for crosstalk cancellation coefficients, as well as lower memory bandwidth demands associated with accessing the coefficient data for processing within the DSLAM or the like.

As illustrated in Figure 1, a multiple-input multiple-output (MIMO) system can be characterized by its system (or channel) responses from each input to each output of the system. For linear systems, the system responses are linear functions. For example, inputs 1 through U and outputs 1 through U have system responses H_{ij} (FEXT channel responses/coefficients for $i \neq j$, and direct channel responses/coefficients for $i=j$), namely H_{11}, H_{12}, …, H_{1U}, H_{21}, …, H_{U1}, H_{U2}, …, H_{UU}. Any given $Output_u$ is thus the aggregate response from every input (1 to U) to that output. To communicate data reliably and at high rates over such a MIMO system, the system responses must be known. To identify these system responses, H_{ij} with $(i,j=1:U)$, the inputs can be stimulated with test signal input data (for example, pilot and/or orthogonal training or tracking sequences, as is well known to those skilled in the art) and the test signal output data of the system received, observed, measured and/or collected by CPE modems and/or other downstream-end devices. The data received or computed by a receiving modem is typically an error signal. That is, each modem knows what the training or tracking data inputs are and what the output signal is that the modem should receive if no crosstalk is present. The receiving modem calculates an error signal that is representative of the difference between the expected transmission output and the actual (FEXT perturbed) transmission output. These error signals can be sent to a DSLAM or the like as an indication of a given DSL line's FEXT interference. The DSLAM collects these error signals from all of the relevant CPE modems and correlates the data to determine the $H_{ij}(k)$ coefficients that represent crosstalk effects in the relevant DSL lines.

Thus, each sub-channel becomes an independent MIMO system whose channel response can be independently (and simultaneously) identified. In such a DSL MIMO system, the direct system responses, H_{ii}, correspond to the desired modem response signal for a single sub-channel from each modem *u*'s own twisted pair and the remaining terms, H_{ij}, where $i \neq j$, correspond to FEXT from other twisted pairs in the binder cable (and sometimes from a different cable). Such a MIMO system exists in both the upstream and downstream directions where groups of tones

are assigned to either upstream or downstream communication. As noted above, the upstream and downstream directions use frequency division multiplexing to avoid excessive impact from NEXT. In this type of system, all telco-side modems terminate in the DSLAM, and the CPE modems are widely dispersed at many subscriber facilities. Such a crosstalk canceling DMT DSL modem system is called vectored DSL. Various examples of DSL systems using one or more embodiments of the present invention will be described herein in connection with a DSLAM coupled to a number of CPE modems. However, as will be appreciated by those skilled in the art, embodiments of the present invention can be used in other DSL transmission and vectoring settings and structures. For example, a DSL line may couple a DSLAM to an intermediate DSL line device such as an optical networking termination point (ONT), a signal repeater, etc. Likewise, the upstream end of a DSL line might terminate with a device or apparatus other than a DSLAM alone. That is, embodiments of the present invention can be used in a variety of DSL settings in which the DSL line terminates with a downstream transceiver and an upstream transceiver.

Receive-side upstream vectoring can be performed at the DSLAM side, as all DSLAM-side modems terminate in one location and the necessary FEXT data is available (or can be more easily obtained) to perform crosstalk cancellation. Transmit-side downstream vectoring is performed using transmit pre-compensation and FEXT cancellation (receive-side vectoring usually is not feasible at the CPE side because of the CPE modems' typically wide dispersion in many subscriber locations, except in the case of bonded DSL lines where CPE modems might be co-located).

For previous DSLAM-side vectoring systems to perform transmit pre-compensation effectively, each coupled CPE modem measured/collected its FEXT response for each tone in the DMT system (based, for example, on its training data transmitted from the telco side modems) and sent each tone's measurement back to the telco/DSLAM side modem (over some type of "back-channel") for further processing and FEXT channel estimation. As is well known to those skilled in the art, each value/datum sent to the DSLAM by the CPE after crosstalk training may be a proxy for H_{ij}, rather than the value of H_{ij} itself. When discussed herein, such minor variations in commonly used and/or known notation and/or quantities is not discussed in detail. Alternatively, the CPE modem could perform any needed FEXT channel estimation processing and then send these channel estimates back to the telco/DSLAM side over the back-

channel. Because FEXT channel characteristics can vary over time, these channel estimates typically are tracked and updated regularly over time.

Upstream back-channel communication of FEXT data can be a significant bottleneck and subsequent storage of FEXT data can be a substantial memory burden in implementing DMT-based vectored DSL systems. Back-channel communication of FEXT channel measurements or the like is an especially big problem during system startup, as only a low bit-rate control channel is available during this preliminary startup period (prior to full modem training and initialization). This communication bottleneck can slow system-training time, thus postponing useful operation of the system from the user's perspective. Back-channel communication also is a problem during normal system operation, as most DSL frequency band schemes are biased in favor of higher downstream communication rates at the expense of lower upstream communication rates (for example, by a ratio of up to 10 to 1). Thus back-channel communication may constitute a significant percentage of a total available upstream data rate. If sufficient bandwidth is not allocated for updating FEXT channel measurements, such updates may be slowed to a level that prevents adequate tracking of channel variation over time.

Storage of the FEXT channel data is a related problem. Generally, a MIMO system with U users has U^2 total channel responses (including U desired direct channel responses and U*(U-1) FEXT channel responses) that are calculated and stored for each DMT system tone. While some optimization techniques can reduce this storage burden (for example, using optimized FEXT channel training sequences and techniques that ignore FEXT channels with smaller channel response values), the storage requirements can still be very large and burdensome.

DMT systems use the discrete Fourier transform to break down the channel into a large number of parallel sub-channels (for example, ADSL1 uses a full tone set of 256 tones, VDSL2 uses a full tone set of 4,096 tones) that are closely spaced in the frequency domain. Vectored FEXT cancellation requires deriving FEXT channel responses between all users at many or all of the tones (also referred to as "frequency bins") in use. Measurements on DSL cables show that the incremental change between adjacent tones' FEXT coefficients is small. This does not imply that FEXT is approximately constant across all tones, but rather that FEXT changes relatively smoothly from one tone to the next. For example, Figure 2 shows a typical FEXT channel response curve from 0 to 30MHz from a telco binder cable. Sub-channel spacing in

DMT-based DSL systems is typically on the order of 4 to 8kHz. As can be seen, incremental changes in FEXT between nearby sub-cannels is relatively small even though there is large variation over the complete 0 to 30MHz band.

Using previous FEXT cancellation techniques, a pre-canceller or post-canceller crosstalk cancellation value had to be derived for each tone of each user, where the derived crosstalk cancellation value is related to the FEXT interference coefficient in one or more ways. In most cases, the goal is a combined system in which the relationship between system inputs x and system outputs y is defined for downstream vectoring as $y=\mathbf{HG}x+n$, where $\mathbf{G}$ is pre-applied to the input data, where the matrix product $\mathbf{HG}$ approximates a diagonal matrix, where $\mathbf{H}$ is a matrix containing the MIMO system transmission data values H_{ij} discussed above, where $\mathbf{G}$ is a matrix of FEXT pre-canceller (pre-coder) coefficients G_{ij} derived by the system, and where n is noise (or other uncorrelated distortion sources in the channel or electronics). For upstream vectoring, FEXT compensation is implemented as a post-canceller and $y=\mathbf{G}(\mathbf{H}x+n)$. For either downstream or upstream vectoring, one method for deriving $\mathbf{G}$ with full precision (that is, calculating each coefficient value of G_{ij} with full precision) utilizes the matrix inverse, so that if the FEXT coefficient between user i and user j at tone k is $H_{ij}(k)$, the pre-canceller or post-canceller coefficient is $G_{ij}(k) = H_{ij}^{-1}(k)$. As will be appreciated by those skilled in the art, there are other methods to compute these crosstalk cancellation coefficients. However, because incremental change from $H_{ij}(k)$ to $H_{ij}(k+1)$ is small, embodiments of the present invention exploit the corresponding minor incremental variations of $G_{ij}(k)$ in consecutive tones.

In the disclosure herein, as will be appreciated by those skilled in the art, any examples of embodiments of the present invention given in connection with downstream vectoring and the provision of downstream FEXT data by a CPE (that is, FEXT data such as error signals collected by the CPE and sent back upstream to the DSLAM for further processing) can be applied equally to upstream vectoring, wherein the CPEs of a given group send test signal input data upstream to the DSLAM or other upstream-end device to generate FEXT data relating to upstream transmissions. The only material difference is that the benefits of reduced-memory transmissions to reduce upstream bandwidth usage are irrelevant because the FEXT error signals are received, observed, measured and/or collected by the DSLAM (or other upstream-end modems or devices) rather than the CPE modems.

A practical exemplary system can be defined as having 128 users using 2,500 tones, leading to 32 crosstalk cancellation coefficients per user (to cancel the 32 most significant FEXT sources for each user). Such a configuration operating with full precision for calculating and implementing inverse matrix **G** requires the storage of 10 million coefficients, each requiring approximately 20-32 bits. As will be appreciated by those skilled in the art, in such an exemplary system, materially reducing the bandwidth and memory required to transmit and store these coefficients represents a substantial benefit to such a system. Not only is total memory storage capacity/size in the DSLAM side equipment a concern, but also the coefficient memory bandwidth is a bigger concern in many cases. If this exemplary system uses a 4kHz DMT symbol rate, 40 Giga-words of memory bandwidth are required (because each coefficient must be loaded for each DMT symbol). The two options for storage – on-chip and off-chip memory – both have drawbacks that negatively affect system performance. On-chip FPGA, ASIC, etc. embedded memory cannot cost-effectively handle 40 MB without compression. Off-chip memory with commodity DRAM has adequate storage space, but memory bus bandwidth between the DRAM and any processor(s) performing vectoring becomes an issue.

Various embodiments of the present invention, relying on the small incremental changes in consecutive values of $G_{ij}(k)$, provide reduced-memory vectoring data using either or a combination of two techniques. One technique uses differential-based coefficient definition based on incremental value differences between successive tones, thus greatly reducing the bandwidth and memory needed for use of a full-precision G matrix. A second technique involves using one or more approximations for $G_{ij}(k)$ between full-precision values of $G_{ij}(k)$, again greatly alleviating bandwidth and storage burdens while nevertheless implementing a reliable vectored DSL system. In other embodiments, the differential-based coefficient definition technique is combined with the use of the approximation technique to reduce bandwidth and memory storage needs. Embodiments of the present invention are illustrated in Figures 4 and 5, which illustrate one or more methods for reducing data transmission (both upstream/downstream and upstream-end-internal communication) bandwidth usage and memory storage.

In some embodiments of the present invention, a number of sub-bands are defined within a full set of DSL tones. The full set of DSL tones might be all tones used in the DSL system, or might be divided into a full set of downstream tones and a full set of upstream tones,

or alternatively into a number of downstream or upstream frequency bands (each of which has a full set of tones). It should be kept in mind that when "all tones" are referred to, this phrase is to be interpreted as "substantially all tones" in light of the fact that a minimal number of tones in a given range, especially a large frequency range, might not be involved in data transmission, etc.

In embodiments of the present invention, each sub-band can have an equal number of tones, if desired, but this is not required in some embodiments of the present invention. For example, sub-band definition may change (for example, the number of DSL tones in each sub-band may change) on the basis of frequency band position. Higher frequencies suffer from different FEXT problems than lower frequencies, so having smaller sub-band sizes in higher frequency ranges might be appropriate. Other such variations will be apparent to those skilled in the art, and are deemed to be equivalents of the specific embodiments disclosed and claimed herein. In relatively simple implementations of an approximation system, some embodiments of the present invention break down the full set of tones used in a given DSL system (for example, 256 or 4,094 tones in an entire system, or a full set or frequency band used in just the downstream or upstream direction) into a number of sub-bands that each contain the same number of tones (or frequency bins). Using one or a small number of full-precision values for $G_{ij}(k)$ in each sub-band, approximations of the remaining tones' $G_{ij}(k)$ values are approximated and used. Through careful selection of an approximation model and definition, back-channel communication and memory storage requirements of such an approximation-based vectored DSL system can be reduced substantially. As will be appreciated by those skilled in the art, while $G_{ij}(k)$ is being used as the quantity for which full-precision and approximation values are being determined, other FEXT-related quantities/values (for example, $H_{ij}(k)$ or some proxy for one of these values, such as an error signal generated as a DSL line output at a modem) could equivalently be used instead.

In some simple embodiments a base point for a given sub-band is calculated to determine a $G_{ij}(k)$ full-precision value (for example, a 16-bit complex number representing the $G_{ij}(k)$ crosstalk cancellation value). A line approximating the remaining $G_{ij}(k)$ points in the sub-band is generated using consecutive sub-bands' base points. This linear approximation can thus be defined using a base point and slope to define the line on which $G_{ij}(k)$ approximation values (as opposed to full-precision values) are found. In an exemplary system, a full tone set of 4,096

tones can be divided into 256 sub-bands of 16 tones per sub-band. If a full-precision value is calculated for the first tone in each sub-band, then the remaining 15 tones' approximation values can be defined by a line connecting the first tone of the given sub-band and the first tone of the succeeding sub-band. In this exemplary system the CPE only needs to return the 256 full-precision values of H_{ij} for the first tone in each sub-band, and can disregard the remaining 3,840 tones' H_{ij} full-precision measurements during each FEXT channel determination and/or update. In a more generalized sense then, if the full DSL tone set has K tones (i.e., the 4,096 tones in the above example) that are divided using a decimation factor of p (i.e., the 16 in the above example, defining the number of tones in each sub-band) to create K/p sub-bands (i.e., 256 sub-bands in the above example), then p full-precision values are all that needs to be transmitted by the CPE to the DSLAM to generate the simple linear approximation, while the remaining p-1 (i.e., 15 above) tones per sub-band are approximation values. The DSL back-channel communication channel thus avoids having to transmit 3,840 other fully retracted position data points corresponding to the remaining $(p\text{-}1)*K/p$ FEXT measurements per user that would be sent using an earlier system. Memory storage space requirements at the DSLAM-side are likewise reduced substantially.

Sub-band endpoints in the full DSL tone set can be defined as corresponding to tones pq for the number of sub-bands, where q is defined as $q=1:K/p$. The actual coefficients for endpoints $G_{ij}(pq)$ can be calculated and stored with full precision. Coefficient values of G_{ij} between sub-band endpoints $G_{ij}(pq)$ then can be linearly interpolated to gain a linear approximation of the FEXT channel response coefficients between those endpoints. Thus the endpoints $G_{ij}(pq)$ and $G_{ij}(pq+p)$ can be used to approximate coefficients for $G_{ij}(pq+1)$, $G_{ij}(pq+2)$, … , $G_{ij}(pq+q\text{-}1)$ on a straight line connecting the endpoints and having a slope that can be determined from the sub-band's endpoints. Each sub-band's entire set of crosstalk cancellation coefficients can then be defined in memory as a "start point" (for example, the sub-band's lower endpoint, a midpoint, or the upper endpoint) and a slope that is defined by the linear approximation values for that sub-band's tones. As will be appreciated by those skilled in the art, this technique does not require that a sub-band endpoint, *per se*, be used; the full precision FEXT channel response coefficient for a tone in the middle of a sub-band also can be used. As also will be appreciated by those skilled in the art, as the number of tones in each sub-band is reduced, the accuracy of the linear approximation is likely to improve. This allows the

DSL system to balance improved FEXT approximation with memory usage.

Figure 3A illustrates one or more embodiments using a simple linear approximation. The actual FEXT channel response 390 includes sub-band endpoints 391 and 392 corresponding to $G_{ij}(pq)$ and $G_{ij}(pq+q)$, respectively, in the DSL system's complete tone set. $G_{ij}(pq)$ to $G_{ij}(pq+q-1)$ define a sub-band of 16 tones and can be designated as sub-band q. In the example of Figure 3A, because each sub-band is 16 tones, there are 15 intermediate tones that will use approximation values between each sub-band endpoint. Each of these 15 intermediate tones has its own full-precision G_{ij} value 393 (for example, a dB or linear attenuation value, a crosstalk cancellation coefficient, etc.) on the actual FEXT channel response curve 390. In a simple linear approximation embodiment of the present invention, straight line 394 connects points 391 and 392 and is used as the linear approximation for the FEXT channel response in the defined sub-band q. A reduced-memory estimation of the FEXT channel response thus can be stored by using the starting point 391 for sub-band q, along with a slope α calculated for line 394 using its endpoints 391, 392. Each point 395 along line 394 can then be used for the FEXT channel response without having to transmit (from CPE to DSLAM), store and update all full-precision values for such intermediate tones for the entire DSL tone set. All that must be stored in memory in this basic linear approximation embodiment is the sub-band base point 391 (for example, a sub-band endpoint or midpoint) and the slope α for that particular sub-band's linear approximation. In some embodiments, the base point can be stored and the slope determined "on the fly" whenever it is needed. Of course, storing just the base point and slope simplifies memory access since each sub-band can be processed in parallel without having to cross-reference or cross-access memory contents. The base point and slope typically can be represented/stored as 16-bit complex numbers.

This basic linear approximation embodiment can be enhanced by calculating and storing/implementing a "line offset" to the linear approximation, wherein the line offset value moves the entire linear approximation (and thus all $G_{ij}(k)$ approximation values) a given constant value to achieve better error performance when compared to the full-precision FEXT crosstalk cancellation coefficient set the linear approximation is seeking to approximate. In this case, it again would be necessary to provide back-channel communication from the CPE side only with respect to only one tone per sub-band (that is, sending only a tone number and G_{ij} coefficient value for each sub-band).

Using this line offset variation of the basic linear interpolation embodiment, the linear approximation of the FEXT channel response (or crosstalk cancellation coefficients) for a given sub-band can be evaluated for overall and/or aggregate error as compared to the actual FEXT channel response (or full-precision crosstalk cancellation coefficients) for that sub-band. If shifting the linear approximation line improves the aggregate sub-band error performance, then such a "shifting adjustment" can be made for the linear approximation of that sub-band and a "shift value" or "line offset" can be calculated and stored along with the endpoint and slope data. As illustrated in Figure 3A, a line offset 397 is used to move line 394 to line 394' as an adjusted linear approximation for the FEXT channel response. Line 394' might be used, for example, when the FEXT channel response 390 (or set of crosstalk cancellation coefficients) contains substantially "higher" or substantially "lower" values over the relevant sub-band, so that the adjusted line 394' more accurately approximates the FEXT channel response with the line offset in use.

Another embodiment of the present invention uses a polynomial approximation of the FEXT channel response (or set of crosstalk cancellation coefficients) rather than a linear-only approximation. As illustrated in Figure 3B, the full-precision FEXT channel response 370 (or set of crosstalk cancellation coefficients) has points 373 for each intermediate tone in a sub-band defined by endpoints 371, 372. Linear approximation 374 provides the simple, first order approximation. However, a second order polynomial approximation 377 provides a close fit to the full-precision FEXT channel response 370 (or set of crosstalk cancellation coefficients). As with the linear approximation techniques above, this polynomial approximation 377 can be used by itself, with a curve offset constant, or with one or more error vectors to implement an appropriate estimation of the FEXT channel response 370 (or set of crosstalk cancellation coefficients).

The polynomial approximation can be obtained, for example, by using 3 points for a given sub-band. The two endpoints 371, 372 can be used, along with another full-precision point 378 on plot 370. As will be appreciated by those skilled in the art, given these three points in the sub-band's range of tones, any known fitting techniques can be used to generate a polynomial expression for use as an approximation. Therefore, in the polynomial expression:

$$G_{ij}(q+r) = G_{ij}(q) + \alpha_q * r + \beta_q * r^2$$

where r is the sub-band tone index (for example, r=0:15 in a situation where each sub-band is 16 tones), where $G_{ij}(q)$ is endpoint 371 in sub-band q, α_q represents sub-band q's linear component slope and β_q represents the adjustment to the squared term for sub-band q. Here, each sub-band's index q would be stored with the sub-band's base point $G_{ij}(q)$ and polynomial expression coefficients α_q and β_q.

In another embodiment, an error vector (also referred to a "differential value vector" or "incremental value vector") is calculated and stored for each sub-band to represent a number of incremental adjustments, wherein each incremental adjustment is needed to a given value of $G_{ij}(k)$ to get to the next or another $G_{ij}(k)$ value on the FEXT channel response (or set of crosstalk cancellation coefficients) for each tone in the sub-band. Because the $G_{ij}(k)$ coefficients do not change dramatically within in a given sub-band, the error vector values can be represented in a relatively small number of bits when compared to the full-precision $G_{ij}(k)$ value for a given tone. That is, this error vector can be stored as a number of differential or incremental values representing each change in G_{ij} from the previous tone in the sub-band, once again taking account of the smooth FEXT channel response (or set of crosstalk cancellation coefficients). Such an incremental value format can be even more memory efficient.

Thus as seen in Figure 3C, another reduced-memory embodiment uses the sub-band base point 381 and an incremental value vector D_q for sub-band q having various incremental values Δ_{q1}, Δ_{q2}, etc. That is, base point 381 for sub-band q can be calculated using a full-precision value sent by the CPE after training. Each intermediate tone in sub-band q can then be determined by adding the Δ_q value to the preceding tone's G_{ij}. Each incremental Δ value represents the difference between the desired point and either (1) the base point 381, or (2) the preceding point in the sub-band. To minimize the bit-wise data representation of the sub-band's vector D_q, the point-to-point differential technique means that only the difference between adjacent intermediate tones in a sub-band need be provided, not the difference between each such intermediate tone and the base point. Therefore, each point 383 can be reconstructed using the whole base point differential 386, or incremental changes to each point 386.

Full reconstruction permits reduced-memory storage burdens for DSLAM-side equipment, but does not help much with regard to FEXT data transmission from CPE to DSLAM. To utilize the differential/incremental value technique and realize more reduced-

memory advantages, the differential/incremental value technique can be combined with one of the approximation techniques.

Using a subset of FEXT data points sent by the CPE, such as the sub-band endpoints or possibly a small number of points per sub-band (for example, 4 points per 16 tone sub-band), the DSLAM-side processor can construct smaller linear approximation models between fully retracted position points and use the differential/incremental value technique for determining intermediate tones within that structure. Memory might thus store D_q (base point, slope, e_{q1}, e_{q2}, …, e_{q15}) wherein each e_{qd} is the incremental change value for an intermediate tone in the sub-band and wherein the "base point" might or might not be a sub-band "endpoint," *per se*. The e_{qd} values are typically small due to the small dynamic range of sub-bands and thus could be represented using a relatively small number of bits. To further reduce memory demands, the e_{qd} values can be stored in differential form not just relative to the slope line but also with respect to the previous e_{qd-1} value in the sub-band, such that for example $e_2=e_1+\Delta_2$ and what is stored becomes (base point, slope, Δ_1, Δ_2, Δ_3, … , Δ_{15}) where $e_1= \Delta_1$ and $e_i=e_{i-1}+ \Delta_i$ for $i>1$.

This provides yet another approximation technique to reduce the memory needed for both communication of FEXT data as well as storage of FEXT data used for vectoring in the DSLAM side modems. As noted above, embodiments of the present invention can be used to realize advantageous compression of FEXT channel estimates and/or FEXT crosstalk cancellation coefficients. Moreover, these embodiments reduce memory requirements, reduce memory bandwidth usage and reduce sync symbol error communication from the CPE side during training and/or tracking.

To implement vectoring, each modem in a chassis must exchange real-time FEXT information with the other modems in the chassis (or even multiple chassis units). Typical DSL modems use a DMT symbol rate in the range of 4kHz to 8kHz, and FEXT cancellation must be done over the vectored bandwidth once per DMT symbol with a processing delay on the order of a few DMT symbols (or less) to avoid degrading the overall latency of the end-to-end modem system. This creates a complex and challenging data networking environment in which each line card shares real-time FEXT information from each of its modems with each of the other modems in the system (which might include modems on another line card or chassis in the system). In a typical DSLAM system, the communication flow of vectoring information can

total tens of gigabits per second. Current generation DSLAM copper backplanes generally cannot handle this additional communication load, and future DSLAM copper backplanes need a practical method for managing this communication flow with reasonable complexity and operational performance.

A typical telephone cable or grouping of multiple cables terminating in one telco "central office," "CO," "node," "cabinet," etc. may contain anywhere from 50 to 2,000 (or more) twisted pairs, potentially spread among a dozen or more binder cables. As a result of telco deployment practices, modems assigned to twisted pairs in the same binder cable may not be connected to the same line card or DSLAM chassis. This occurs because telcos add line cards to a chassis (or add one or more additional chassis) as customers subscribe to DSL service. Modem ports are assigned on a "first-come-first-served" basis, rather than being grouped together based on user identity, downstream terminus location, etc. Even if the telco practices were different, it would be an operational (and error-prone) challenge to segregate phone lines according to binder of origin (a practice known as binder group management). Empirical cable measurements have shown that significant FEXT coupling can come from adjacent binders or even different cables that terminate in the same telco node due to the use of a "cross box" or patch panel in the telco node. These patch panels are used to map individual copper pairs (possibly from different binders or telco cables) to specific modem ports on the DSLAM line cards.

Another related issue is the practice of bonding multiple modem ports together. Bonding can increase the customer data rate in situations where a single modem port cannot provide the desired data rate. A customer who has started with a single modem port can have a second port added as an upgrade to provide additional services to the customer (the multiple-port DSL service thus has the potential to carry more data, operate at faster speeds, etc.). Many current systems require bonded modem ports to be connected to the same line card. This is impractical for most telcos for the same reasons that binder group management is impractical, as discussed above.

Finally, the computational demands of DSL vectoring – especially the real-time processing of crosstalk information in connection with user data signals – create data transmission and processing bottlenecks and other problems. The large volume of data that

needs to be processed quickly (to reduce latency/delays) must be transmitted and processed efficiently. Embodiments of the present invention provide systems, apparatus and methods that provide such efficient handling and processing of these large amounts of data while having little negative effect on the latency and/or performance of the DSL lines being vectored.

Embodiments of the present invention include apparatus, systems, methods, etc. that eliminate or substantially reduce the communication bottlenecks that otherwise would confront vectored DSL systems. These embodiments are scalable, from small port-count systems (for example, a single line card system) up to much larger systems, with thousands of modem ports spread across multiple line cards and/or multiple chassis. Embodiments of the present invention work with both optical interconnect technologies (in the case of multiple chassis systems or line card upgrades to legacy chassis equipment) and also work with future copper interconnect technologies in which all communications flow within a single DSLAM on a copper backplane or using high bandwidth copper interconnect. Embodiments of the present invention also permit "virtual bonding" that allows telcos to virtually bond modem ports across multiple line cards and/or multiple chassis.

Figure 6A illustrates a vectoring data communication system 100 in which line cards 110 contain DSL modems 112 (for example, multi-port devices) that control communications on twisted-pair lines 114. Multiple line cards 110 are connected via a high speed communication apparatus, such as XAUI lines 108 or the like in Figure 6A, to a vectoring control entity (VCE) that can be or include a centralized vectoring module 120 (which can be, for example, a vectoring card). XAUI is the standard for extending the XGMII (10 Gigabit Media Independent Interface) between the MAC and PHY layer of 10 Gigabit Ethernet (10GbE) and such high-speed data communication lines 116 can be used to connect modems 112 to line card vector router components (VRC-Ls) 118. The VRC-Ls 118 form an abstraction layer for the modem 112, as the modem 112 needs to connect to only one VRC-L 118 and the complexity of the specific vectoring deployment (for example, number of ports, line cards, etc.) is thus hidden from each modem 112.

The vectoring data flow from each modem 112 to its respective VRC-L 118 includes frequency domain samples for downstream and upstream communications – that is, IFFT-input transmit (TX) data for downstream vectoring and/or FFT-output receive (RX) data for upstream

vectoring. The data returned to each modem 112 from the vectoring module 120 via a VRC-L 118 is the modem's crosstalk-adjusted (that is, vectored) IFFT-input and/or FFT-output data that is conditioned and/or processed to prevent and/or remove crosstalk interference from other vectoring system modems. The VRC-L 118 in each line card 110 acts as an interface between that line card's modems 112 and the vectoring module 120. High-speed communication lines 108 (for example, 10-40Gbps or higher optical or copper interconnect) network a VRC-L 118 on each line card 110 to a companion VRC-V 122 on the vectoring module 120. 10-40Gbps is a common data communication requirement that can be implemented between the vectoring module 120 and each line card 110. Today this would mostly likely be an aggregation of 5Gbps or 10Gbps XAUI lines or similar, whether over the electrical backplane or the optical cable.

The VRC-Vs 122 on vectoring module 120 subdivide the modem vectoring data stream into sub-bands for subsequent crosstalk cancellation in one or more vector processors (VPCs) 124, as defined by system requirements. The vector processors may also be referred to as "vector processor components," "computational devices" and/or the like. That is, data is removed from a normal (that is, non-vectored) data stream in each modem and is reorganized into data bundles defined by frequency characteristics so that the data can be crosstalk-processed on a frequency basis (for example, tone-by-tone, groups of tones, etc.). Once processed, the data is then again reorganized from the frequency-based bundles used for crosstalk-removal processing and is reassembled for transmission/use by the modems.

For example, upstream and downstream bands can be vector routed by one or more VRCs (for example, a VRC-L/VRC-V pair) to individual VPCs. A vector router is a specialized data networking device or subsystem that implements a specialized "private" data network, which can be similar to an Ethernet network, for the purpose of efficiently moving vectoring data between modems and vector processors to avoid processing or data transmission bottlenecks. Packets of vectoring data can contain headers and/or other state information enabling efficient routing of the vectoring data over the data network without the need for dedicated links between each modem and vector processor device. To this end, a vector router also converts vector data packets from a format readily supplied by the modems into a format that is naturally utilized by the vector processors, then converting back again after vectoring has been performed (for example, interleaving and de-interleaving of the modem vectoring data stream). This task may be split between VRC-Ls and VRC-Vs, or performed in only one or the

other, depending on the configuration. Alternately, VPC assignment can be based on evenly spaced sub-bands (independent of upstream and downstream band allocation). Data transmission between VRC-Vs 122 and VPCs 124 on the vectoring module 120 can be performed using high speed interconnect lines 126 (for example, XAUI or the like).

The exemplary packet structure illustrated in Figure 16 allows for real-time vectoring of data between a number of DSLAM-side modems (for example, VTU-Os) and a vectoring control entity (VCE). Request packets containing payload data in the form of unprocessed user data (that is, user data for which crosstalk effects have not been compensate – also referred to as "uncorrected" data or signals herein) for vectoring are sent to a vector processor or the like, where vectoring processing compensates for crosstalk effects in the unprocessed user data DSL signals. The processed DSL data is then repacked as payload data and sent in reply packets back to the respective modems that supplied the unprocessed user data (request and reply packets typically use similar configurations). A packet is constructed (for example, by the VRC-L) with non-vectored data sent from every modem, and this packet is sent for vectoring processing once per symbol for each tone and modem/port range. Essentially, non-vectored data sent to a VPC gets unpacked, vectoring processed, re-packed in reply packets that are similar to the non-vectored data packets, and subsequently transmitted back to modems in reply packets, which return vectored data in the same format as the non-vectored data that was transmitted by the modems. The packing and unpacking of data by a VRC-L or the like follows the line card and/or modem port map so that vectored data is returned to the modem/port from which its corresponding non-vectored data was extracted (that is, request packets and reply packets can be identified by their modem or modem port of origin).

Other types of packets can be used in connection with embodiments of the present invention. For example, along with the vector data packets described, vector error packets, vector configuration packets and vector control packets can be sent throughout the vectoring data network to configure, adjust, etc. the vectored DSL system as appropriate (for example, to configure, control or synchronize modems chips or to configure, control or synchronize vector processors). Any of the packets described herein can include one or more modem port identifiers, a modem chip identifier, a line card identifier, a chassis identifier, or information identifying at least one DMT tone range so that tracking and addressing of the packets is achieved.

For vector request packets, the payload data can include inverse fast Fourier transform samples for a plurality of modem ports and tone sets identified in the request packet header, fast Fourier transform samples for a plurality of modem ports and tone sets identified in the request packet header, downstream frequency domain data for a plurality of modem ports and tone sets identified in the request packet header, and/or upstream frequency domain data for a plurality of modem ports and tone sets identified in the request packet header. Additional fields and components of the packet can be utilized (for example, a cyclic redundancy check or CRC). A vector error packet can include upstream training error samples for the modem ports and tone sets identified in the request packet header, upstream tracking error samples for the modem ports and tone sets identified in the request packet header, upstream DMT sync symbols for the modem ports and tone sets identified in the request packet header, downstream training error samples for the modem ports and tone sets identified in the request packet header, downstream tracking error samples for the modem ports and tone sets identified in the request packet header, and/or downstream DMT sync symbols for the modem ports and tone sets identified in the request packet header.

The embodiment of Figure 16 shows the vectoring data packet format defined for modems serving N ports and vectoring data transmitted to/from the modems over the DSL vectoring data network (also referred to as the "private network," "routing subsystem," "vector routing network," "special network" or the like) has this format. The VRC-L adds a header on such a packet to identify the modem chip and line card, so the source of the data is known as the packet flows to the VRC-V (or, if no VRC-V is on the vectoring card, then to whatever destination is used in the vectoring card) and then on to the VPC/VPU. As vector-processed data comes back from the vectoring card to the VRC-L (or, if no VRC-L is used, then to the modems from which the vectoring data was extracted and transmitted), this header is removed before distributing the packets back to the appropriate modem chips. The VRC-L and/or VRC-V may optionally re-assemble packets from multiple modem chips into a format for use by the VPCs, depending on how the modems provide the data. Use of such packets and headers (coupled with vector router use in some embodiments) allows the vectored DSL system to scale to larger (or smaller) numbers of ports, and no hardwired/dedicated data bus is required between each modem and vector processor. Appropriate error signal packets and configuration and control messages also can flow over the network accordingly.

The VRC pairing configuration 118, 122 effectively converts data consideration from a per-port modem basis to a per-band (or per-frequency-set) VPC basis, at least for crosstalk cancellation pre-processing. Because crosstalk cancellation is most effectively and easily performed in the frequency domain on a per-tone, per-band or per-tone-set basis, this conversion greatly reduces the complexity of the vectoring system's implementation. Matching sub-bands (that is, groupings of the same one or more frequencies, referred to as "tone groupings," "tone sets," "sub-bands" and the like) from every modem 112 in the vectoring system 100 are collected and aggregated to enable crosstalk cancellation inside a VPC 124 between each victim modem and any set or subset of disturber modems.

As will be appreciated by those skilled in the art, the sub-bands assigned to each VPC 124 do not have to consist of contiguous frequencies in the system. For example, if there are 6 VPCs 124 in vectoring module 120, each consecutive VPC 124 may take the next frequency or frequency bin, meaning that the "first" VPC 124 will get frequencies 1, 7, 13, 19, etc. and the "fourth" VPC 124 will get frequencies 4, 10, 16, etc. Alternately, each VPC 124 can be sent a tone set, for example groups of 8 tones each, so that the first VPC 124 gets tones 0-7, the second VPC 124 gets tones 8-15, etc. Such distributions of data can provide a more "even" usage of available resources, for example preventing "clogging" of one or more given communication lines 126 due to a data dump from the relevant VRCs 122 to a single VPC 124 while the remaining VPCs 124 and lines 126 are idle.

The data on the private vectoring data network of system 100 consists of special purpose packets identifying the source modem 112 and destination VPC 124. Alternatively, the source and destination address can be implied by the packet timing. Each packet consists of IFFT input (and/or FFT output) data for one sub-band to/from one modem 112. Because DMT-based DSL systems operate using a DMT symbol period (for example, 4kHz or 8kHz), data communications from the modems 112 to the VPCs 124 and back again are repeated for every DMT symbol, possibly using pipelining of the communication and computation to use available networking bandwidth and computational resources more efficiently. Per-tone training/tracking error signals, diagnostic data to/from the modems 112 and other configuration and/or management data can use the same networking communication path and packet structure.

Various hardware devices and/or components can be used to implement embodiments of

the present invention. Various current FPGAs and the like are suitable for constructing configurations according to embodiments illustrated herein. For example, Altera Stratix IV GX FPGA devices (e.g., EP4SGX70, EP4SGX110, EP4SGX230, etc.) are appropriate for such implementations. In some embodiments of the present invention, each VPC 124 is a separate chip (FPGA, ASIC, etc.) on which are programmed multiple VPUs as defined below. Other hardware configurations will be apparent to those skilled in the art.

As illustrated in Figure 6B, each VPC 124 contains or implements one or more dedicated vector processing units 242 (VPUs), each of which performs vectoring for a subset of the modems and/or disturbers. Allocation and/or sharing of the VPU resources may be advantageous in some cases and can be accomplished in various ways. For example, each VPC 124 can contain a centralized set of VPUs 242 that are time-shared among all of the modems 112 for vectoring within a given sub-band. Alternately, each VPC 124 can contain a distributed set of VPUs 242 in which each VPU 242 is dedicated to a particular modem 112 within the sub-band assigned to the VPC 124. Multiple VPUs 242 can coordinate communication between their host VPC 124 and one or more VRC-Vs 122 using a VPU dispatcher 244 or the like.

A VPU 242 processes all of the IFFT/FFT data for a tone set (for example, a single tone, a group of tones, a band or sub-band of tones, etc.) for all of the modems making up the vectored DSL system. The VPU 242 processes this data once per DMT symbol, in a somewhat periodic manner. As noted above, in some embodiments, each VPC 124 is a collection of one or more VPUs 242, where each VPU 242 independently processes its batches of tones without any explicit communication with other VPUs 242. In a simple exemplary setting, a system might use 4096 tones, 2 VPUs, and N modems. Each VPU 242 handles tones in batches of 16 tones. Tones can be assigned to VPUs as follows:

VPU0: tones 0-15, 32-47, ...
VPU1: tones 16-31, 48-63, ...

In this example, the modems provide their FFT/IFFT data to the VRC-based private network in tone order from 0 to 4095. When the FFT/IFFT data gets routed to the VPCs, the VRC distributes data like "dealing cards" to the two VPUs in batches of 16 tones. This keeps the data networking pathway and VPUs uniformly busy without bottlenecks. For each tone, the VPU performs the equivalent of an NxN matrix multiply to perform vectoring (or maybe less than a

full NxN multiply if the vectoring matrix is "sparse").

Each VPU has a pre-determined processing capability level (for example, number of complex multiplications per second), based on the logic design and chip technology utilized. Additionally, the required level of processing power "per tone" for the vectoring system will increase as the number of modems, N, increases. Thus, the required number of VPUs is a function of the number of modems in the system and the number of vectored tones over the DMT bandwidth. Therefore, systems with a large number of vectored tones and/or a large number of modems will consequently need a larger number of VPU resources (and consequently VPC chips) to perform vectoring. That is, as the number of modems increases in a vectored system, the number of tones that each VPU can process will decrease due to the additional modems' data that need to be processed. As will be appreciated by those skilled in the art, the size of each sub-band can be determined by cost and complexity limits for a given implementation, as well as the DSL band profiles that are supported by the design. For example, in an implementation for a system having a 17MHz VDSL profile, 384 modems, and vectoring up to 64 disturbers per modem, a practical system would have on the order of 128 VPUs with each VPU processing 32 tones per DMT symbol. As the VPU logic is reconfigured to reflect any changing dimensions of the system (number of modems, number of vectored tones, etc.), vector routing is adjusted accordingly.

In embodiments of the present invention such as the one shown in Figure 6C, the vectoring system 300 may exceed the practical complexity for a single card used as the vectoring module 120 (for example, power consumption, card area, etc.). The system illustrated in Figure 6C permits splitting or allocating vectoring processing of system 300 onto multiple vectoring cards 320 (in a chassis, in a module or in multiple modules) where each card 320 performs crosstalk cancellation over some fraction of the total DSL frequency band. With this configuration, vectoring can be performed in systems with hundreds or even thousands of simultaneously operating modems. VPCs typically do not need to communicate with each other explicitly, although they will be roughly synchronized implicitly by the flow of the FFT/IFFT data and training signals coming from the modems.

In the embodiment of the present invention illustrated in Figure 6D, the vectoring data transmission path from each modem 412 to a vectoring card 420 can be further utilized to serve

a dual function and perform "virtual bonding." In addition to passing crosstalk cancellation data, each modem 412 in the bonded modem group (that is, two or more modems serving a user in a bonded fashion) can pass customer data streams (for example, IP packets) along this same communication path. One or more bonding routers 438 (acting as a supplemental VRC (sVRC), bonding agent and/or controller) on vectoring card 420 connect via VRC-Vs 418 to each VRC on its respective line-card. Each bonding router 438 performs routing between VRC-Vs and, thus, their respective VRC-Ls. This effectively creates a data path between and/or among any two or more modems in the vectoring system and multiple modem ports can then be bonded in the vectoring system, provided that one of the two modems performs the aggregation function and presents the total bonded data stream to the uplink path of the DSLAM, as will be appreciated by those skilled in the art.

In yet another embodiment of the present invention illustrated in Figure 6E, a simplified system 500 fits onto a single line-card 510 (see also Figure 10A). Once again, VRC resources 518 (a single VRC, for example, or alternately two VRCs back-to-back) provide routing between modem ports and the VPCs 524 that provide processing of designated sub-bands.

In low port-count systems with only modest networking communication bandwidth, the VRC functionality 618 can be situated directly inside each VPC 624, as illustrated in the embodiment shown as system 600 in Figure 6F. For example, where an FPGA is used as the VPC 624, a portion of the FPGA configuration is dedicated to the routing functionality. In this case, the cost savings realized by removing an independent vector routing device from the system outweigh the additional communication requirements. Figure 6F shows both daisy chain and mesh network connection options 650 between VPCs 624. In practice, one or the other would be utilized (not both) based on the communication requirements and tradeoffs for the system.

Figure 7 illustrates general operation of the vectoring operation 700, wherein efficient implementation of the downstream FEXT pre-compensator 701 and upstream FEXT post-compensator 702 can be accomplished using one or more embodiments of the present invention. In the downstream direction, a plurality of QAM data symbols from a plurality of modems is FEXT pre-compensated by coordinating or "vectoring" communication among all modems on a per-tone basis, followed by subsequent IFFT and digital-to-analog conversion operations on a

per-modem basis to transmit the FEXT-compensated signal onto the Telco copper pair. In the upstream direction, a plurality of received FFT samples is FEXT post-compensated by coordinating or "vectoring" communication among all modems on a per-tone basis, followed by subsequent decoding of the QAM data symbols on a per-modem basis.

Figure 8 shows a generic modem port for a commonly available DSLAM side VDSL modem device 800 that has been modified to support vectored communication using one or more embodiments of the present invention. The upstream FFT 802 outputs and the downstream IFFT 804 inputs are intercepted by a vector packet encoder/decoder 806 (or "vectoring data extractor") and sent to a vector processor 808 via interface 807 to allow vectoring of the data stream among all modems in the vectoring group. By removing, processing and "reinserting" data from the data stream, such a vector processor system may introduce a small number of symbols of latency in the modem's receive and transmit data paths.

Figures 9A-9D provide several specific embodiments of a DSLAM side modem configuration for communicating with a vector processor chip, specifically showing possible interfaces between a modem chip and a VRC. Figure 9A shows the configuration of a 12-port, ITU 993.2 Profile 30a profile vectored VDSL DSLAM side modem chip that supports vectoring (for example, a 5G serializer/deserializaer or "Serdes" type). In order to efficiently make use of I/O pins on the multi-port modem chip and reduce power consumption, a high-speed serial interface such as XAUI or the like can be used to aggregate FFT/IFFT data from each modem device over a single data networking link either directly to a vector processor chip or to a vector router chip for subsequent forwarding to a vector processor. Figure 9B illustrates a 12-port, ITU 993.2 Profile 30a profile vectored VDSL DSLAM side modem chip that supports vectoring. In this embodiment, low voltage differential signaling (LVDS) interface circuits (1 LVDS per modem port) are used. Figure 9C illustrates a 12-port, ITU 993.2 Profile 17a vectored VDSL DSLAM side modem chip that supports vectoring (for example, a 5G Serdes type). Figure 9D illustrates a 12-port, ITU 993.2 Profile 17a vectored VDSL DSLAM side modem chip that supports vectoring (for example, a LVDS type, with 1 LVDS interface per 2 modem ports).

Figure 10A illustrates one embodiment of a 48-port, single line card vectoring data communication system in which a single line card holds all modem ports, a vector router chip

and any needed vector processing chips (in this example, 3 VPCs). The 48 ports are served in a relatively simple configuration and the system makes use of either copper backplane or optical cable communication to efficiently aggregate the FFT/IFFT data and interface between the line card and the vectoring card. Figure 10B illustrates a multiple line card vectoring data communication system that serves 192 modem ports in an ITU 993.2 Profile 17a example. In the embodiment shown in Figure 10C, no VRC-V is required, so the VPUs communicate directly with each line card's VRC-L. Similarly, Figure 10D shows an embodiment in which no VRC-L is required, and each modem chip communications directly with a VRC-V. In this embodiment, modem chips on the "front" two line cards communicate with the "front" VRC-V, while modem chips on the "back" two line cards communicate with the "back" VRC-V. Figure 10E shows yet another embodiment.

Exemplary system configurations are noted in Table 1:

ITU 993.2 Profile	Modem ports/chip	Modem chips/line card	Modem ports/line card	System line cards	System modem ports	System VRC-Vs	System VPCs
8a	12	4	48	4	192	2	6
12a	12	4	48	4	192	2	8
17a	12	4	48	4	192	4	8
30a	8	3	24	4	96	4	6

The ITU 993.2 Profile 8a with 192 ports generates an estimated power dissipation of ~200mW/port and 40W for the total system (250mW/port and 48W for the 17a profile with 192 ports and 400mW/port and 40W for the 30a profile).

Figure 11 illustrates an embodiment of the present invention in which the vector router chips 1110, 1115 communicate data from six modems 1105 (the number of modems can vary as appropriate) to four VPCs 1125 (again, the number of VPCs can vary as appropriate) using a sweep-select 1112 to collect data from the modems 1105 and a sweep-distribution 1117 to distribute the collected data to the vector processor chips 1125. As can be seen in Figure 11, the VRC-L 1110 is coupled to the VRC-V 1115 by a suitable interconnect 1120. The bus bandwidth used/needed/required can be determined in instance as follows:

$$\text{Bandwidth} = \text{nPorts} * (\text{Data_Precision}) * 2 * 4\text{kHz}$$

in which "2" indicates complex arithmetic. Finally, bandwidth bottlenecks can be avoided using the "striping" shown in Figure 11.

Figure 12 illustrates one exemplary logic implementation of a vector processor unit. In this example, the sub-unit can process a single tone and Figure 12 traces the data path for a vector processor core according to one or more embodiments of the present invention. This exemplary VPU is capable of processing one DMT tone at a time. As IFFT/FFT samples from each modem arrive from the vector router, the VPU determines the list of "victim" modems that must be FEXT canceled for this disturber. Using pre-determined FEXT cancellation coefficients from the coefficient table and the victim address list, the impact of this disturber can be accumulated into the partial results table for each victim modem (partial result of the FEXT cancellation operation). After the full set of disturber data has been received from the vector router for this tone and FEXT cancellation has been completed, the partial results table will then contain the resultant vectored IFFT/FFT samples for the current DMT tone, and this vectored data in then sent back to the vector router for transmission back to the modem ports.

Figure 13 shows an exemplary vector processor 1324 implementing two VPUs 1342 and interfacing with at least one modem chip 1312. Vector processor 1324 could be implemented in some embodiments of the present invention on a single FPGA, ASIC, etc. connected to modem chip 1312 or the like.

The vectoring data communication system of Figure 14 illustrates a DSLAM architecture embodiment showing a dedicated vectoring module 1430 with four line cards providing user data for vectoring. Line cards 1410 are mounted to a chassis or the like 1405 and each line card includes several modem chips 1415 and a vectoring interface device 1420. Each interface device 1420 communicates via optical interconnect 1425 with the vectoring module 1430, which can be a pizza box or a line card. Each module 1430 contains a line card interface chip 1435 that corresponds to one of the interface devices 1420 on line cards 1410. A number of processing chips 1440 are available for vector processing of user data.

As illustrated in the system 1500 having a master line card 1510 and a slave line card 1511 in Figure 15, in order for vectoring to work properly, all modems 1512 in the vector group must be synchronized to both an ADC/DAC sample clock 1534 and a DMT symbol clock 1535, shown as 35.328MHz and 4kHz, respectively. The DSLAM side modems 1512 are connected

via DSL line 1514 to a number of customer-side DSL modems 1513. The high-speed sample clock is assumed to be distributed in some manner over a backplane or dedicated cable. The VRC-Ls 1518 can then phase lock to the high-speed sample clock and redistribute a low-noise sample clock to the modem chips 1517. In order to avoid sending a dedicated 4kHz DMT symbol clock over the backplane or dedicated cable, it would be advantageous to use the high-speed communication interface intended to transmit vectoring data (XAUI or similar) to send timing information in data packets. The high-speed interface operates at a sufficiently high speed, that the approximate DMT symbol timing can be determined from such timing packets. As there is some tolerance to the exact DMT symbol timing (made possible, for example, by cyclic prefix and cyclic extension in the VDSL protocol), the VRC-L 1518 needs to determine only the approximate symbol timing provided that it will remain consistent thereafter (that is, synchronized to some fixed number of the high-speed sample clock periods).

Many features and advantages of the invention are apparent from the written description, and thus, the appended claims are intended to cover all such features and advantages. Further, numerous modifications and changes will readily occur to those skilled in the art, so the present invention is not limited to the exact operation and construction illustrated and described. Therefore, described embodiments are illustrative and not restrictive, and the invention should not be limited to the details given herein but should be defined by the following claims and their full scope of equivalents, whether foreseeable or unforeseeable now or in the future.

CLAIMS

What is claimed is:

1. A method for operating a vectored digital subscriber line (DSL) system, wherein the DSL system comprises a DSL access multiplexer (DSLAM) coupled to a plurality of customer premises equipment (CPE) modems using a plurality of DSL lines, wherein the plurality of CPE modems comprises a first CPE modem, the method comprising:

identifying a plurality of DSL tone sub-bands in a downstream frequency band used for transmitting data from the DSLAM to the plurality of CPE modems, wherein each sub-band contains a plurality of DSL tones;

identifying a subset of the DSL tones in each sub-band, wherein the number of DSL tones in the subset of DSL tones is not more than half of the number of DSL tones in each sub-band;

transmitting test signal input data from the DSLAM on the plurality of DSL lines;

the first CPE modem receiving test signal output data based on the test signal input data;

the first CPE modem computing an error signal representative of a far-end crosstalk (FEXT) interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line i and test signal output data received by DSL line j for DSL tone k, further wherein each DSL tone k is in one of the subsets of DSL tones in one of the sub-bands;

the first CPE modem transmitting the computed error signal to the DSLAM;

the DSLAM generating a full-precision FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to the computed error signal transmitted by the first CPE modem;

the DSLAM generating an approximation of a FEXT cancellation coefficient function G for each sub-band based on a plurality of the full-precision FEXT cancellation coefficient values $G_{ij}(k)$;

the DSLAM using the approximation to generate approximate FEXT cancellation

coefficient values $G_{ij}(m)$, wherein each DSL tone m is not in one of the subsets of DSL tones in one of the sub-bands; and

performing vectored DSL processing using the full-precision FEXT cancellation coefficient values and the approximate FEXT cancellation coefficient values.

2. A method for operating a vectored DSL system, wherein the DSL system comprises a DSLAM coupled to a plurality of CPE modems using a plurality of DSL lines, wherein the plurality of CPE modems comprises a first CPE modem, the method comprising:

identifying a plurality of DSL tone sub-bands in an upstream frequency band used for transmitting data from the first CPE modem to the DSLAM, wherein each sub-band contains a plurality of DSL tones;

identifying a subset of the DSL tones in each sub-band, wherein the number of DSL tones in the subset of DSL tones is not more than half of the number of DSL tones in each sub-band;

transmitting test signal input data from the first CPE modem to the DSLAM;

the DSLAM receiving test signal output data based on the test signal input data;

the DSLAM computing an error signal representative of a FEXT interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line i and test signal output data received by DSL line j for DSL tone k, further wherein each DSL tone k is in one of the subsets of DSL tones in one of the sub-bands;

the DSLAM generating a full-precision FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to the computed error signal;

the DSLAM generating an approximation of a FEXT cancellation coefficient function G for each sub-band based on a plurality of the full-precision FEXT cancellation coefficient values $G_{ij}(k)$;

the DSLAM using the approximation to generate approximate FEXT cancellation

coefficient values $G_{ij}(m)$, wherein each DSL tone m is not in one of the subsets of DSL tones in one of the sub-bands; and

performing vectored DSL processing using the full-precision FEXT cancellation coefficient values and the approximate FEXT cancellation coefficient values.

3. The method of Claim 1 or 2 wherein the approximation of the FEXT cancellation coefficient function G comprises a linear approximation or a polynomial approximation.

4. The method of Claim 1 or 2 wherein substantially all sub-bands in the frequency band contain the same number of DSL tones per sub-band; and further wherein each subset of DSL tones contains the same number of DSL tones.

5. The method of Claim 4 wherein each sub-band contains one of the following:

64 DSL tones; 32 DSL tones; 16 DSL tones; 8 DSL tones; 4 DSL tones.

6. The method of Claim 1 or 2 wherein the test signal input data comprises DSL training data or DSL tracking data.

7. The method of Claim 1 or 2 wherein the subset of DSL tones in each sub-band is an endpoint tone in the sub-band;

further wherein a full-precision FEXT cancellation coefficient value corresponds to the endpoint tone in each sub-band; and

further wherein the approximation of the FEXT cancellation coefficient function G for a first sub-band comprises a linear approximation of G generated by connecting the

endpoints of the first sub-band and a second sub-band, further wherein the first sub-band is adjacent to the second sub-band.

8. The method of Claim 7 further comprising storing approximation data for use in performing vectored DSL processing, the approximation data comprising:

a sub-band base point corresponding to the full-precision FEXT cancellation coefficient value for the endpoint tone in the first sub-band; and

a slope computed using the linear approximation of G for the first sub-band.

9. The method of Claim 8 with the approximation data further comprises a first sub-band line offset value.

10. The method of Claim 1 or 2 wherein the subset of DSL tones consists of 3 or less tones in each sub-band, including at least one endpoint tone in each sub-band;

further wherein a full-precision FEXT cancellation coefficient value corresponds to each tone in the subset of DSL tones in each sub-band; and

further wherein the approximation of the FEXT cancellation coefficient function G for each sub-band comprises a polynomial approximation of G generated using a plurality of the full-precision FEXT cancellation coefficient values.

11. A method for operating a vectored DSL system, wherein the DSL system comprises a plurality of DSL lines, each DSL line comprising a transmitter coupled to a receiver, including a first DSL line comprising a first transmitter and a first receiver, the method comprising:

identifying a plurality of DSL tone sub-bands in a frequency band of DSL tones used for transmitting data on the plurality of DSL lines, wherein each sub-band contains a

plurality of DSL tones;

identifying a subset of the DSL tones in each sub-band, wherein the number of DSL tones in the subset of DSL tones is not more than half of the number of DSL tones in each sub-band;

transmitting test signal input data from the first transmitter to the first receiver;

the first receiver receiving test signal output data based on the test signal input data;

the first receiver transmitting a computed error signal representative of FEXT interference to the first transmitter for each DSL tone in the subset of DSL tones in each sub-band;

the first transmitter generating a full-precision FEXT cancellation coefficient value corresponding to the computed error signal transmitted by the first receiver;

the first transmitter generating an approximation of a FEXT cancellation coefficient function for each sub-band based on a plurality of the full-precision FEXT cancellation coefficient values;

the first transmitter using the approximation to generate approximate FEXT cancellation values for DSL tones that are not part of any subset of DSL tones in any sub-band; and

the first transmitter performing vectored DSL processing using the full-precision FEXT cancellation coefficient values and the approximate FEXT cancellation coefficient values.

12. The method of Claim 11 wherein the approximation is a linear approximation or a polynomial approximation.

13. A method for operating a vectored DSL system, wherein the DSL system comprises a DSLAM coupled to a plurality of downstream-end modems, the method comprising:

transmitting test signal input data from the DSLAM to the plurality of downstream-end modems;

a first downstream-end modem receiving test signal output data based on the transmitted test signal input data;

the first downstream-end modem transmitting to the DSLAM an error signal representative of FEXT interference for each DSL tone in a subset of DSL tones in each of a plurality of sub-bands in a first frequency band;

an upstream-end apparatus generating a plurality of full-precision FEXT cancellation coefficients corresponding to a plurality of error signals from the downstream-end modem;

the upstream-end apparatus generating a FEXT cancellation coefficient approximation for each sub-band based on a plurality of the full-precision FEXT cancellation coefficient;

the upstream-end apparatus using the FEXT cancellation coefficient approximation to generate approximate FEXT cancellation coefficients for DSL tones in the frequency band that are not part of the subset of DSL tones in each sub-band; and

the upstream-end apparatus performing vectored DSL processing using the plurality of full-precision FEXT cancellation coefficients and the approximate FEXT cancellation coefficients.

14. The method of Claim 13 wherein the upstream-end apparatus comprises the DSLAM.

15. The method of Claim 14 wherein the upstream-end apparatus further comprises a plurality of vectoring modules configured to perform vector processing on DSL signals.

16. A method for performing vectored DSL in a DSL system using a downstream frequency band, the method comprising:

a CPE modem transmitting to a DSLAM FEXT-representative error signals for a subset of DSL tones in the downstream frequency band, wherein the number of tones in the subset of DSL tones is not more than 90% of the number of DSL tones in the downstream frequency band;

the DSLAM computing full-precision FEXT cancellation coefficients corresponding to the error signals transmitted by the CPE modem;

the DSLAM deriving an approximation for FEXT cancellation coefficients for all DSL tones that are not part of the subset of DSL tones in the downstream frequency band; and

a DSL vectoring module performing vectored DSL processing using the full-precision FEXT cancellation coefficients and the FEXT cancellation coefficient approximation.

17. A method for performing vectored DSL in a DSL system using an upstream frequency band, the method comprising:

providing a DSLAM with FEXT-representative error signals for a subset of DSL tones in the upstream frequency band, wherein the number of tones in the subset of DSL tones is not more than 90% of the number of DSL tones in the upstream frequency band;

the DSLAM computing full-precision FEXT cancellation coefficients corresponding to the provided error signals;

the DSLAM deriving an approximation for FEXT cancellation coefficients for all DSL tones that are not part of the subset of DSL tones in the upstream frequency band; and

a DSL vectoring module performing vectored DSL processing using the full-precision FEXT cancellation coefficients and the FEXT cancellation coefficient approximation.

18. The method of Claim 16 or 17 wherein the approximation is a linear approximation and further wherein one of the following is stored for use in connection with vectored DSL processing:

a line definition set comprising a full-precision FEXT cancellation coefficient and a slope defining a line from the full-precision FEXT cancellation coefficient;

a line definition set comprising a full-precision FEXT cancellation coefficient and a slope defining a line from the full-precision FEXT cancellation coefficient, the line definition set further comprising a line offset value;

19. The method of Claim 16 or 17 wherein the approximation is a polynomial approximation.

20. The method of Claim 16 wherein the number of tones in the subset of DSL tones is not more than half of the number of DSL tones in the downstream frequency band.

21. The method of Claim 16 wherein the number of tones in the subset of DSL tones is not more than 25% of the number of DSL tones in the downstream frequency band.

22. The method of Claim 17 wherein the number of tones in the subset of DSL tones is not more than half of the number of DSL tones in the upstream frequency band.

23. The method of Claim 17 wherein the number of tones in the subset of DSL tones is not more than 25% of the number of DSL tones in the upstream frequency band.

24. A method for operating a vectored digital subscriber line (DSL) system, wherein the DSL system comprises a DSL access multiplexer (DSLAM) coupled to a plurality of customer premises equipment (CPE) modems using a plurality of DSL lines, wherein the plurality of CPE modems comprises a first CPE modem, the method comprising:

identifying a plurality of DSL tone sub-bands in a downstream frequency band used for transmitting data from the DSLAM to the plurality of CPE modems, wherein each sub-band contains a plurality of DSL tones;

identifying a base point DSL tone in each sub-band;

transmitting test signal input data from the DSLAM on the plurality of DSL lines;

the first CPE modem receiving test signal output data based on the test signal input data;

the first CPE modem computing a base point error signal representative of a far-end crosstalk (FEXT) interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line i and test signal output data received by DSL line j for DSL tone k, further wherein each DSL tone k is a base point DSL tone;

the first CPE modem transmitting the computed base point error signal to the DSLAM;

the DSLAM generating a full-precision base point FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to the computed base point error signal transmitted by the first CPE modem;

for a first sub-band, the DSLAM generating differential values $\Delta_{ij}(m)$ based on the base point of the first sub-band, wherein each DSL tone m is a tone in the first sub-band other than the base point; and

performing vectored DSL processing using the full-precision base point FEXT cancellation coefficient values and the differential values.

25. A method for operating a vectored digital subscriber line (DSL) system, wherein the DSL system comprises a DSL access multiplexer (DSLAM) coupled to a plurality of customer premises equipment (CPE) modems using a plurality of DSL lines, wherein the plurality of CPE modems comprises a first CPE modem, the method comprising:

identifying a plurality of DSL tone sub-bands in an upstream frequency band used for transmitting data from the first CPE modem to the DSLAM, wherein each sub-band contains a plurality of DSL tones;

identifying a base point DSL tone in each sub-band;

transmitting test signal input data from the plurality of CPE modems to the DSLAM;

the DSLAM receiving test signal output data based on the test signal input data;

the DSLAM computing a base point error signal representative of a far-end crosstalk (FEXT) interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line i and test signal output data received by DSL line j for DSL tone k, further wherein each DSL tone k is a base point DSL tone;

the DSLAM generating a full-precision base point FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to the computed base point error signal;

for a first sub-band, the DSLAM generating differential values $\Delta_{ij}(m)$ based on the base point of the first sub-band, wherein each DSL tone m is a tone in the first sub-band other than the base point; and

performing vectored DSL processing using the full-precision base point FEXT cancellation coefficient values and the differential values.

26. The method of Claim 24 or 25 wherein each $\Delta_{ij}(m)$ represents one of the following:

the difference between the full-precision base point FEXT cancellation coefficient value and a full-precision intermediate tone FEXT cancellation coefficient value for tone m;

the difference between a linear-approximated FEXT cancellation coefficient value for tone *m* and a full-precision intermediate tone FEXT cancellation coefficient value for tone *m*;

the difference between a polynomial-approximated FEXT cancellation coefficient value for tone *m* and a full-precision intermediate tone FEXT cancellation coefficient value for tone *m*;

the incremental difference between a full-precision FEXT cancellation coefficient value for tone *m* and a full-precision FEXT cancellation coefficient for tone *m*-1;

the incremental difference between a linear-approximated FEXT cancellation coefficient error value for tone *m* and a linear-approximated FEXT cancellation coefficient error value for tone *m*-1, wherein a coefficient error value is the difference between a linear approximation of a coefficient value and a full-precision coefficient value for a given DSL tone; or

the incremental difference between a polynomial-approximated FEXT cancellation coefficient error value for tone *m* and a polynomial-approximated FEXT cancellation coefficient error value for tone *m*-1, wherein a coefficient error value is the difference between a polynomial approximation of a coefficient value and a full-precision coefficient value for a given DSL tone.

27. The method of Claim 24 or 25 wherein each $\Delta_{ij}(m)$ represents the difference between the FEXT cancellation coefficient value for tone *m* and the FEXT cancellation coefficient value for the tone preceding tone *m*.

28. The method of Claim 24 further comprising the CPE modem transmitting a differential error signal for a plurality of intermediate DSL tones, wherein each intermediate DSL tone is a tone that is not a base point DSL tone, wherein each differential error signal represents the difference between each intermediate DSL tone and the base point DSL tone that is in the same

sub-band as the intermediate DSL tone.

29. The method of Claim 25 further comprising the DSLAM receiving a differential error signal for a plurality of intermediate DSL tones, wherein each intermediate DSL tone is a tone that is not a base point DSL tone, wherein each differential error signal represents the difference between each intermediate DSL tone and the base point DSL tone that is in the same sub-band as the intermediate DSL tone.

30. The method of Claim 24 further comprising:

the CPE modem computing a plurality of incremental values, wherein each incremental value is representative of the difference between a first base point error signal and a differential error signal, wherein the differential error signal corresponds to a DSL tone that is not a base point DSL tone;

the CPE modem transmitting the incremental values to the DSLAM; and

for the first sub-band, the DSLAM generating the differential values $\Delta_{ij}(m)$ based on the any incremental values received from the CPE modem corresponding to the first sub-band.

31. The method of Claim 24 wherein the DSLAM generating differential values $\Delta_{ij}(m)$ comprises at least one of the following:

the DSLAM using full-precision error signals from the first CPE modem representative of FEXT crosstalk interference coefficients for DSL tones m;

the DSLAM using full-precision differential error signals from the first CPE modem representative of FEXT crosstalk interference coefficients for DSL tones m;

the DSLAM computing and using a linear approximation based on full-precision base

point FEXT cancellation coefficient values;

the DSLAM computing and using a polynomial approximation based on full-precision base point FEXT cancellation coefficient values.

32. A method for operating a vectored digital subscriber line (DSL) system, wherein the DSL system comprises a DSL access multiplexer (DSLAM) coupled to a plurality of customer premises equipment (CPE) modems using a plurality of DSL lines, wherein the plurality of CPE modems comprises a first CPE modem, the method comprising:

identifying a plurality of DSL tone sub-bands in a downstream frequency band used for transmitting data from the DSLAM to the plurality of CPE modems, wherein each sub-band contains a plurality of DSL tones;

identifying one or more base point DSL tones in a first sub-band;

transmitting test signal input data from the DSLAM on the plurality of DSL lines;

the first CPE modem receiving test signal output data based on the test signal input data;

the first CPE modem computing a base point error signal for each base point in the first sub-band, wherein each base point error signal is representative of a far-end crosstalk (FEXT) interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line i and test signal output data received by DSL line j for DSL tone k, further wherein each DSL tone k is a base point DSL tone;

the first CPE modem transmitting any computed base point error signals for the first sub-band to the DSLAM;

the DSLAM generating a full-precision FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to each computed base point error signal transmitted by the first CPE modem;

the DSLAM generating an approximation of a FEXT cancellation coefficient function G based on any generated full-precision FEXT cancellation coefficient values $G_{ij}(k)$;

the DSLAM generating differential values $\Delta_{ij}(m)$ based on the approximation, wherein each DSL tone m is a tone in the first sub-band other than the base point; and

performing vectored DSL processing using the full-precision FEXT cancellation coefficient values and the differential values.

33. A method of performing vectored DSL processing, the method comprising:

dividing a DSL frequency band into a plurality of sub-bands;

selecting a subset of DSL tones within each sub-band;

computing full-precision FEXT cancellation coefficients for the subset of DSL tones;

deriving an approximation of FEXT cancellation coefficients for substantially all DSL tones that are not part of a subset of DSL tones; and

performing vectored DSL processing using the full-precision FEXT cancellation coefficients and the approximation.

34. The method of Claim 33 wherein the DSL frequency band is an upstream frequency band and further wherein the vectored DSL processing is receive-side vectoring.

35. The method of Claim 33 wherein the DSL frequency band is a downstream frequency band and further wherein the vectored DSL processing is transmit-side vectoring.

36. A method of performing vectored DSL processing, the method comprising:

dividing a DSL frequency band into a plurality of sub-bands;

selecting a base point DSL tone within a first sub-band;

computing a full-precision FEXT cancellation coefficient for the base point DSL tone of the first sub-band;

deriving a differential value representations of FEXT cancellation coefficients for intermediate DSL tones in the first sub-band that are not the base point DSL tone, wherein an intermediate DSL tone is a DSL tone that is not the base point DSL tone in the first sub-band; and

performing vectored DSL processing using the full-precision FEXT cancellation coefficient and the differential value representations.

37. The method of Claim 36 wherein each differential value representation of an intermediate DSL tone FEXT cancellation coefficient for the first sub-band is an incremental value to be added to the full-precision FEXT cancellation coefficient of the first sub-band.

38. The method of Claim 36 wherein each differential value representation of a first intermediate DSL tone FEXT cancellation coefficient for the first sub-band is an incremental value to be added to the FEXT cancellation coefficient of the DSL tone preceding the first intermediate DSL tone.

39. The method of Claim 36 further comprising generating an approximation of a FEXT cancellation coefficient function based on one or more full-precision FEXT cancellation coefficients and using the approximation to derive approximate FEXT cancellation coefficients for one or more of the intermediate DSL tones.

40. A vectored DSL system comprising a DSLAM coupled to a plurality of CPE modems, the vectored DSL system configured to perform a method according to one of Claims 1-38.

41. The vectored DSL system of Claim 40 wherein vectored DSL processing is performed by a vectoring module coupled to a plurality of line cards coupled to the DSLAM.

42. The vectored DSL system of Claim 41 wherein:

each line card comprises:

a plurality of DSL modems, wherein each DSL modem comprises a plurality of DSL ports, further wherein each DSL port is coupled to a DSL line; and

a line card vector router coupled to each of the plurality of DSL modems;

the vectoring module comprises:

a plurality of vectoring module vector routers, wherein each vectoring module vector router is coupled to a different line card vector router; and

a plurality of vector processors, wherein each vector processor is coupled to each vectoring module vector router;

wherein each vector processor comprises a plurality of vector processing units, further wherein each vector processing unit is configured to vector process data from one of the following:

all of the modems in the plurality of modems for a set of DMT tones; or

a subset of the plurality of modems, wherein the subset comprises the primary crosstalk disturbers in the vectored DSL system;

further wherein vector processing by a vector processing unit compensates for crosstalk interference.

43. The method of any one of Claims 1-39 wherein any FEXT-related data transmitted by a CPE modem to a DSLAM comprises data in a reduced-memory format.

44. The method of any one of Claims 1-39 wherein any FEXT-related data used in performing vectored DSL processing comprises data stored in a reduced-memory format, wherein the reduced-memory format comprises at least one of the following:

approximation values;

differential values;

incremental values.

45. A reduced-memory vectored DSL system comprising:

a DSLAM coupled to a plurality of CPE modems including a first CPE modem, the DSLAM configured to transmit test signal input data to the plurality of CPE modems;

the first CPE modem configured to compute an error signal representative of a far-end crosstalk (FEXT) interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line i and test signal output data received by the first CPE modem on DSL line j for DSL tone k, further wherein each DSL tone k is in a subset of DSL tones in a sub-band of a downstream DSL frequency band;

the first CPE modem further configured to transmit the computed error signal to the DSLAM;

the DSLAM configured to generate a full-precision FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to the computed error signal transmitted by the first CPE modem;

the DSLAM further configured to generate an approximation of a FEXT cancellation coefficient function G for each sub-band based on a plurality of the full-precision FEXT cancellation coefficient values $G_{ij}(k)$;

wherein the DSLAM uses the approximation to generate approximate FEXT cancellation coefficient values $G_{ij}(m)$, wherein each DSL tone m is not in one of the

subsets of DSL tones in one of the sub-bands; and

wherein the DSL system is configured to perform vectored DSL processing using the full-precision FEXT cancellation coefficient values and the approximate FEXT cancellation coefficient values.

46. A reduced-memory vectored DSL system comprising:

a DSLAM coupled to a plurality of CPE modems including a first CPE modem, the first CPE modem configured to transmit test signal input data to the DSLAM;

the DSLAM configured to:

compute an error signal representative of a far-end crosstalk (FEXT) interference coefficient $H_{ij}(k)$, wherein $H_{ij}(k)$ is based on test signal input data sent on DSL line *i* and test signal output data received by the DSLAM on DSL line *j* for DSL tone *k*, further wherein each DSL tone *k* is in a subset of DSL tones in a sub-band of an upstream DSL frequency band;

generate a full-precision FEXT cancellation coefficient value $G_{ij}(k)$ corresponding to the computed error signal; and

generate an approximation of a FEXT cancellation coefficient function G for each sub-band based on a plurality of the full-precision FEXT cancellation coefficient values $G_{ij}(k)$;

wherein the DSLAM uses the approximation to generate approximate FEXT cancellation coefficient values $G_{ij}(m)$, wherein each DSL tone *m* is not in one of the subsets of DSL tones in one of the sub-bands; and

wherein the DSL system is configured to perform vectored DSL processing using the full-precision FEXT cancellation coefficient values and the approximate FEXT cancellation coefficient values.

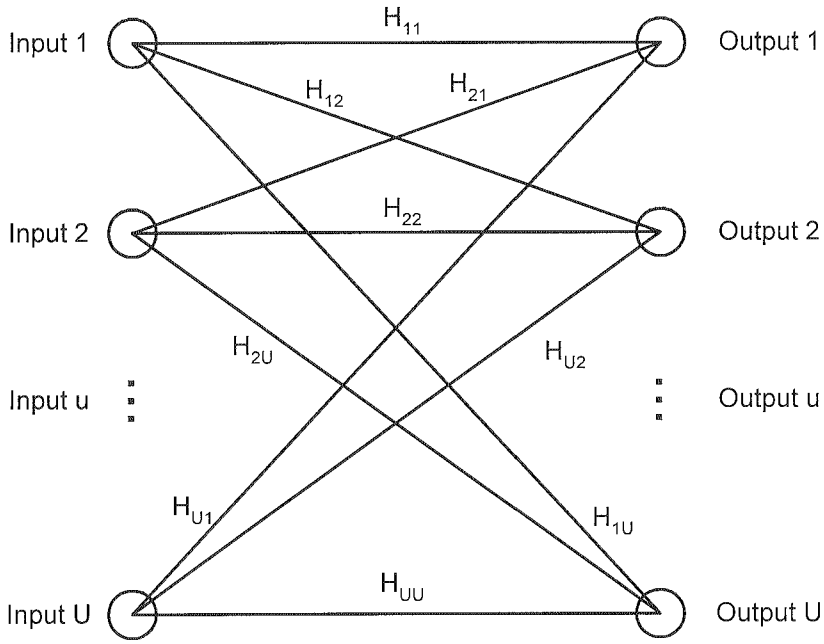

Input 1
Input 2
Input u
Input U
Output 1
Output 2
Output u
Output U
H_{11}
H_{12}
H_{21}
H_{22}
H_{2U}
H_{U2}
H_{U1}
H_{1U}
H_{UU}

Figure 1

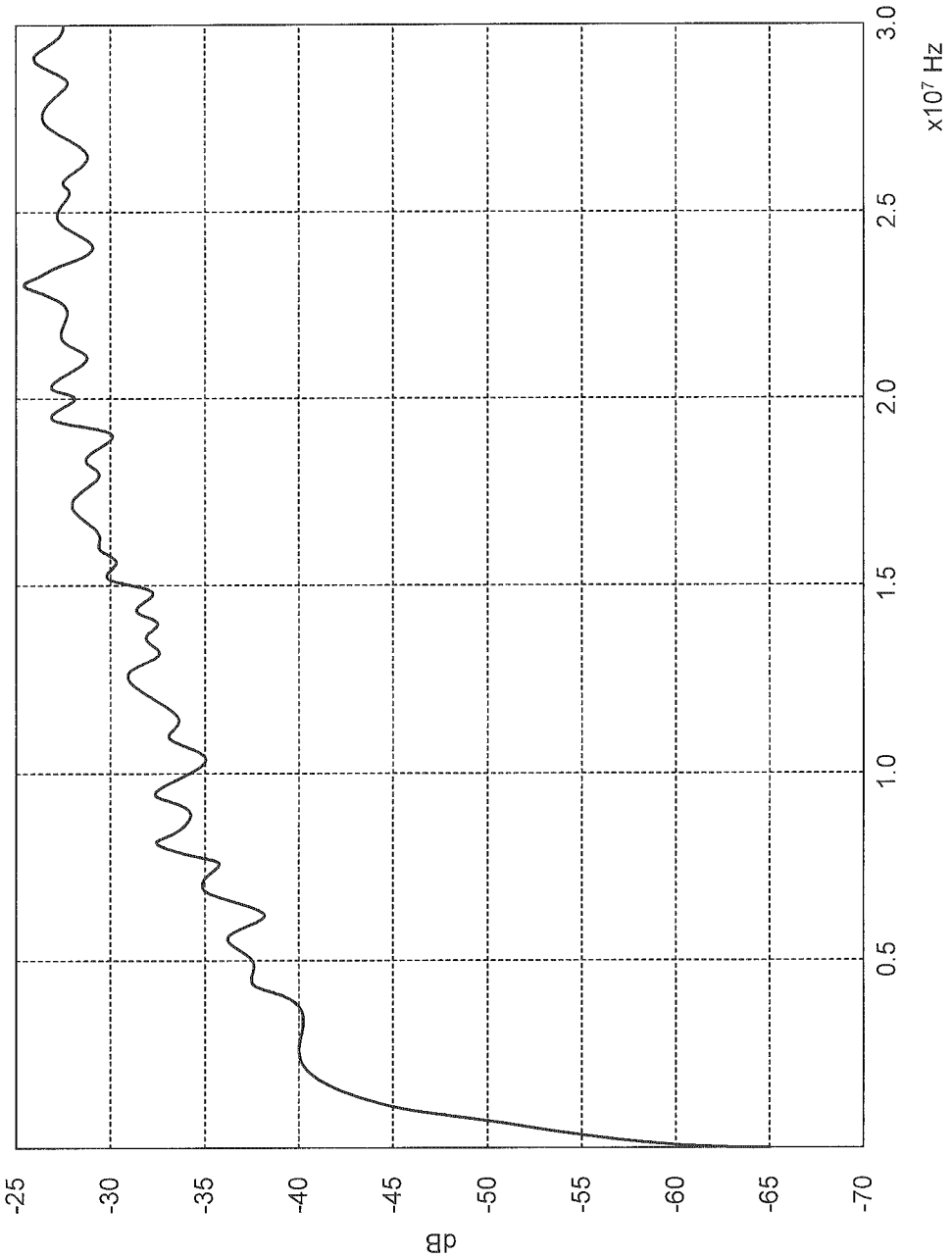


Figure 2

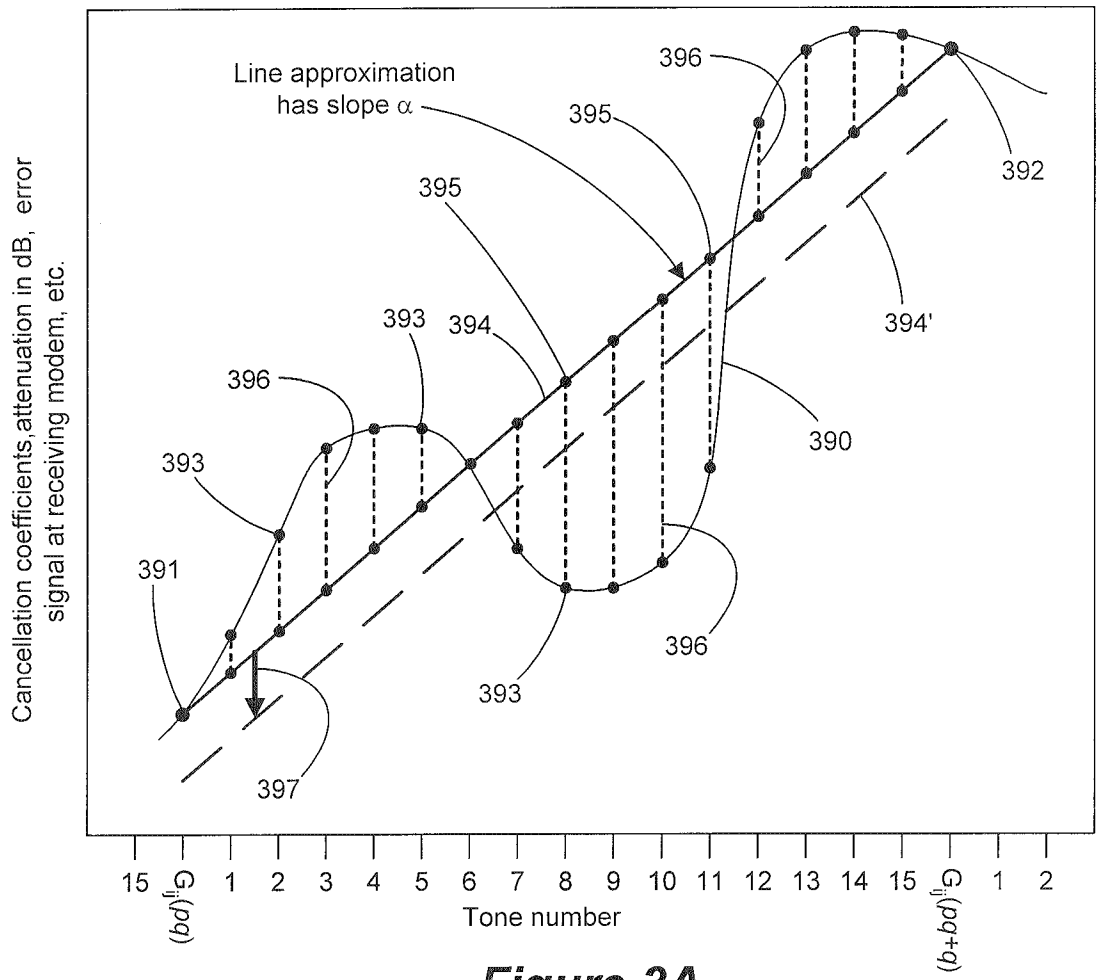

Line approximation
has slope α
396
395
395
392
394'
393
394
396
390
393
391
396
393
397
Cancellation coefficients,attenuation in dB, error
signal at receiving modem, etc.
15
$G_{ii}(bd)$
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
$G_{ii}(pq+q)$
1
2
Tone number

Figure 3A

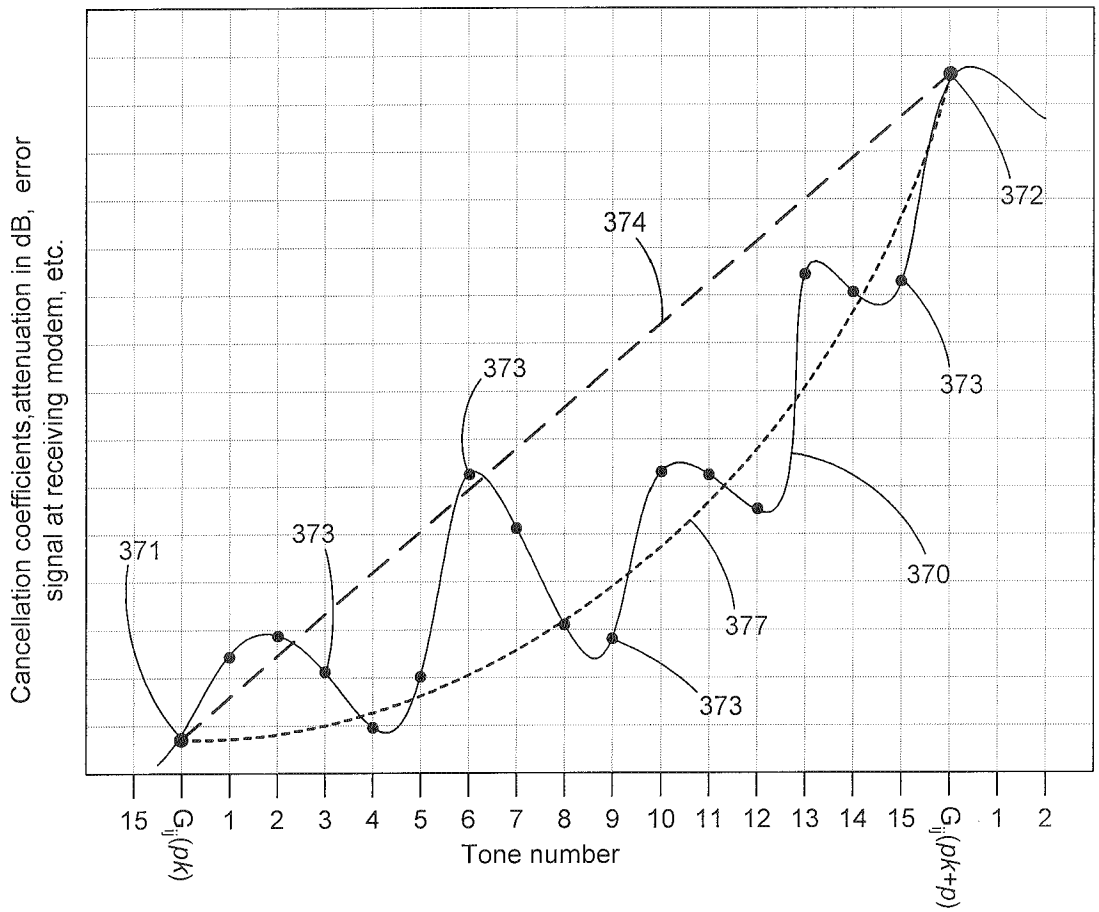

Cancellation coefficients,attenuation in dB, error signal at receiving modem, etc.
371
373
373
374
373
372
373
377
370
373
15
$G_{ii}(pk)$
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
$G_{ii}(pk+p)$
1
2
Tone number

Figure 3B

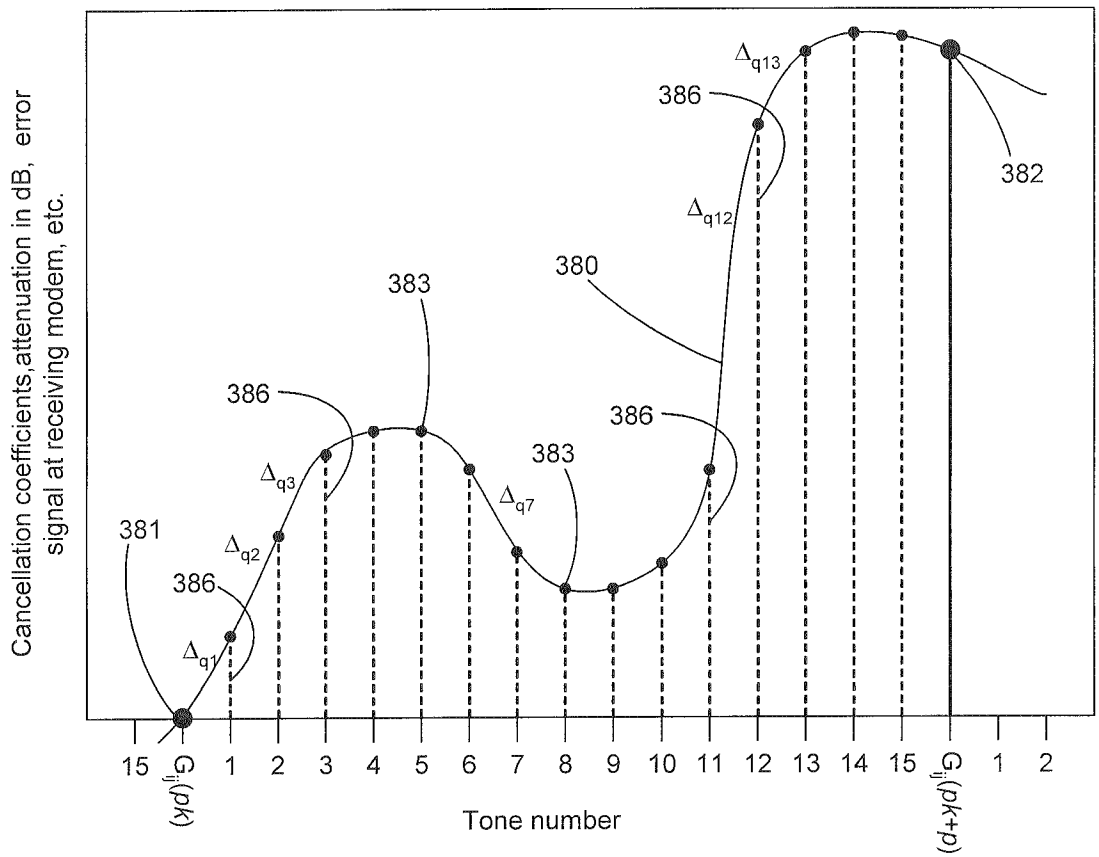

Cancellation coefficients,attenuation in dB, error signal at receiving modem, etc.
Tone number
15
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
1
2
$G_{ij}(pk)$
$G_{ij}(pk+p)$
381
382
383
383
386
386
386
386
380
Δ_{q1}
Δ_{q2}
Δ_{q3}
Δ_{q7}
Δ_{q12}
Δ_{q13}

Figure 3C

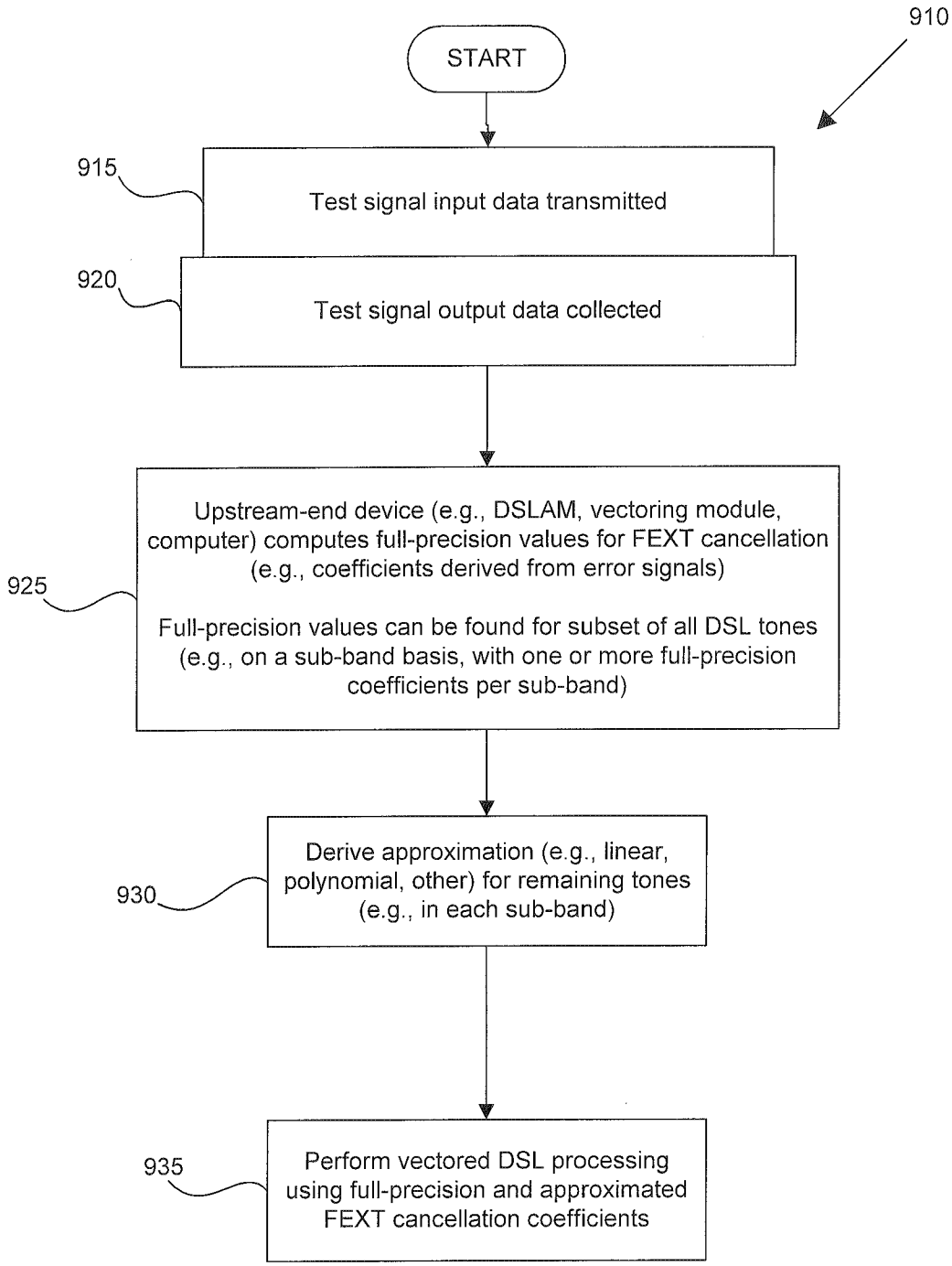

910
START
915
Test signal input data transmitted
920
Test signal output data collected
925
Upstream-end device (e.g., DSLAM, vectoring module, computer) computes full-precision values for FEXT cancellation (e.g., coefficients derived from error signals)
Full-precision values can be found for subset of all DSL tones (e.g., on a sub-band basis, with one or more full-precision coefficients per sub-band)
930
Derive approximation (e.g., linear, polynomial, other) for remaining tones (e.g., in each sub-band)
935
Perform vectored DSL processing using full-precision and approximated FEXT cancellation coefficients

Figure 4

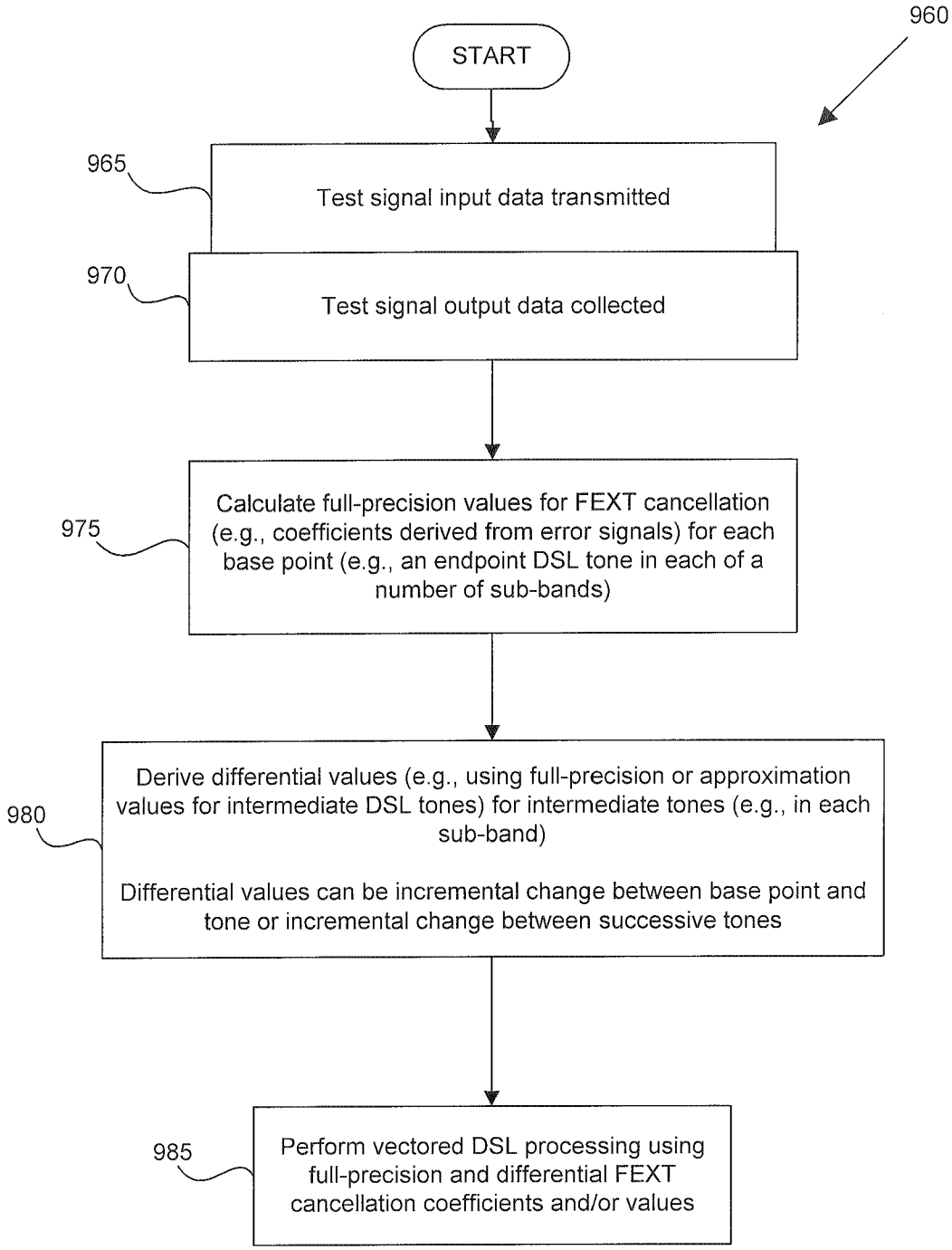

960
START
965
Test signal input data transmitted
970
Test signal output data collected
975
Calculate full-precision values for FEXT cancellation (e.g., coefficients derived from error signals) for each base point (e.g., an endpoint DSL tone in each of a number of sub-bands)
980
Derive differential values (e.g., using full-precision or approximation values for intermediate DSL tones) for intermediate tones (e.g., in each sub-band)
Differential values can be incremental change between base point and tone or incremental change between successive tones
985
Perform vectored DSL processing using full-precision and differential FEXT cancellation coefficients and/or values

Figure 5

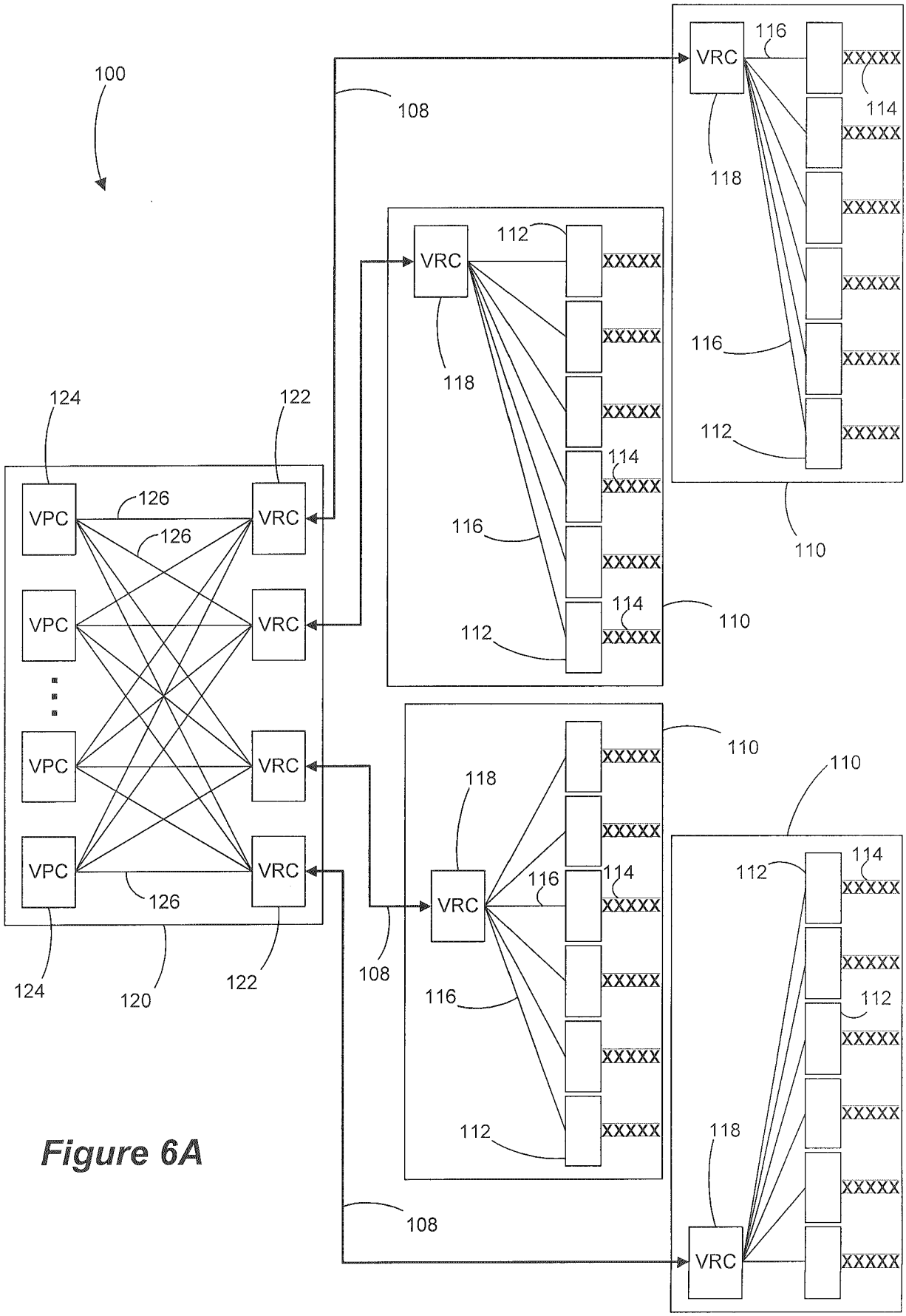
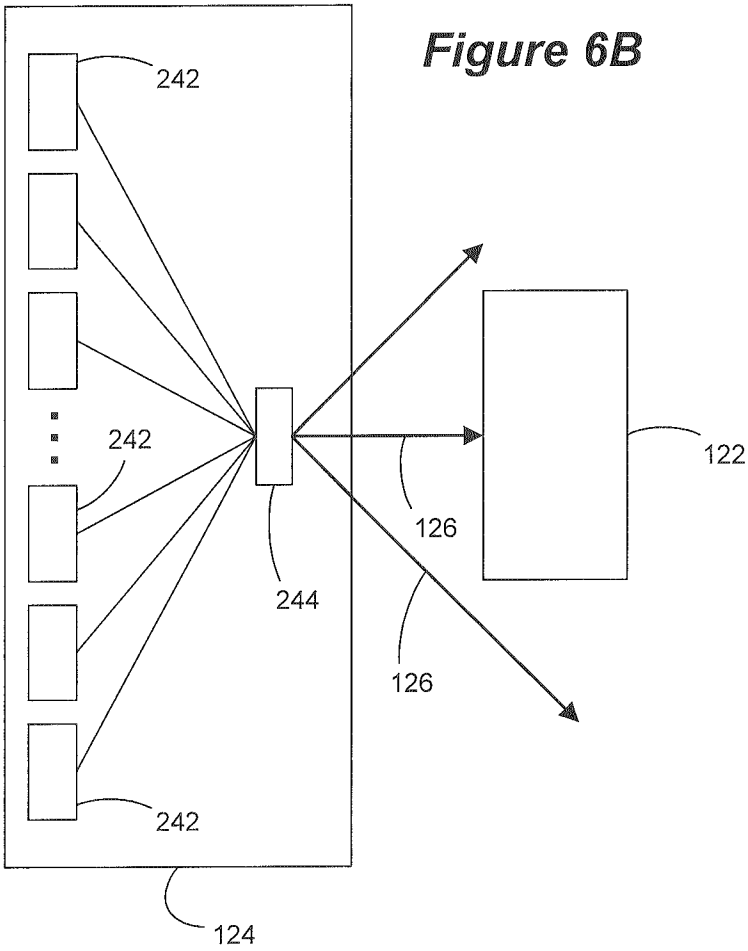


Figure 6A

Figure 6B



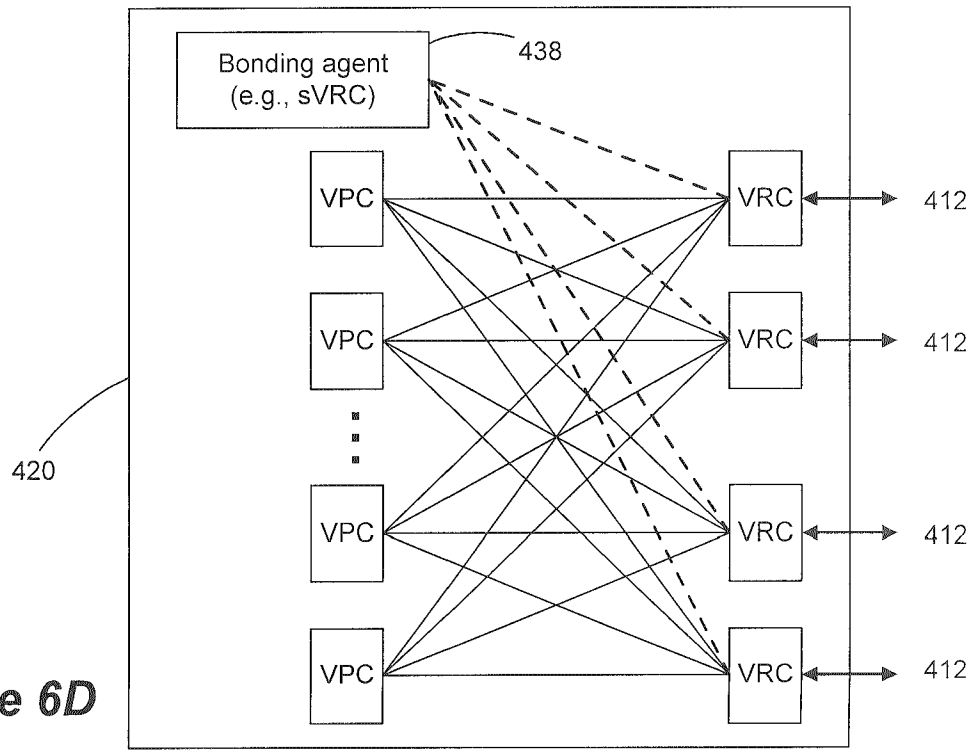


Figure 6D

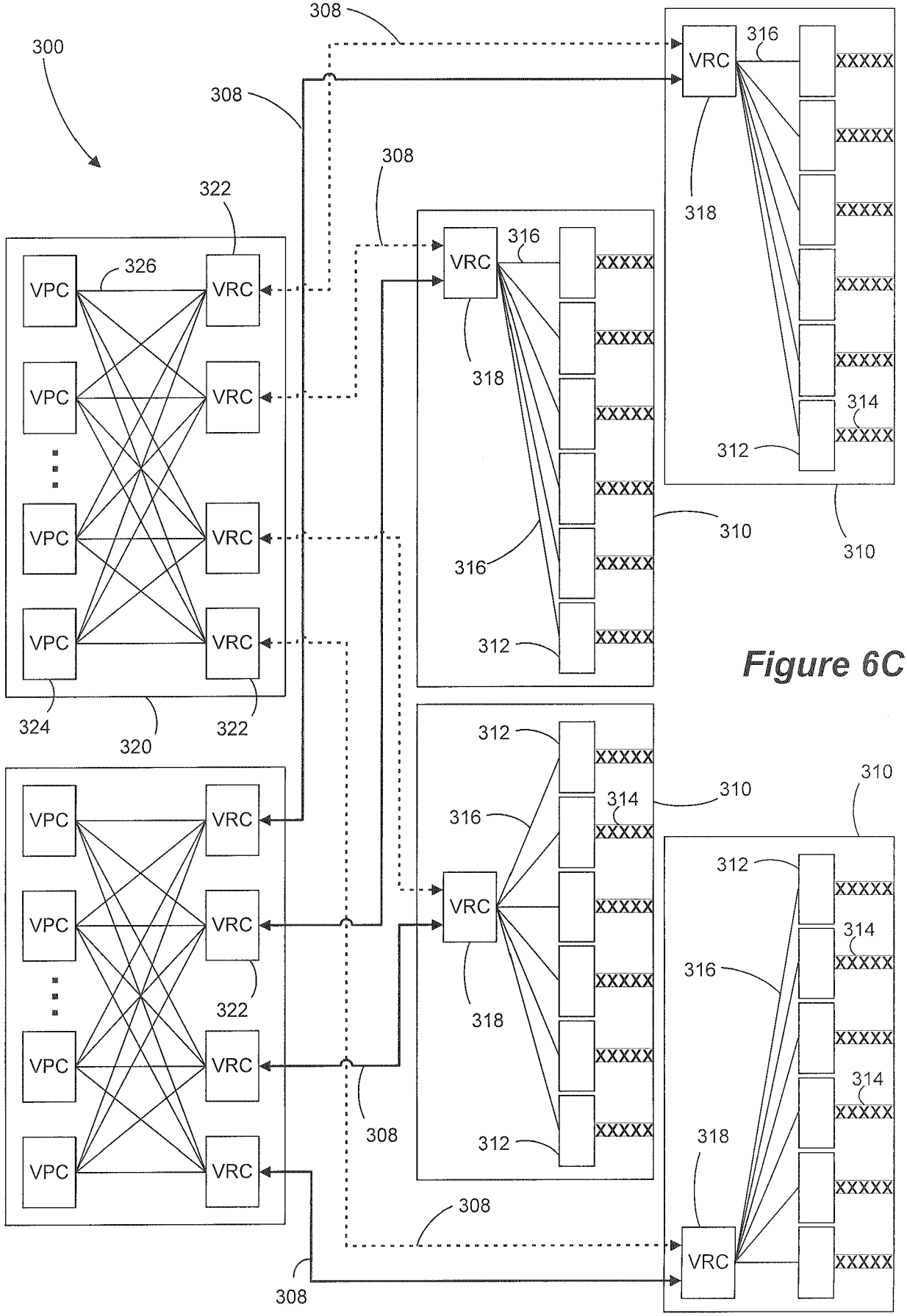

300
308
308
308
322
326
VPC
VRC
316
XXXXX
318
314
312
310
324
320
322

Figure 6C

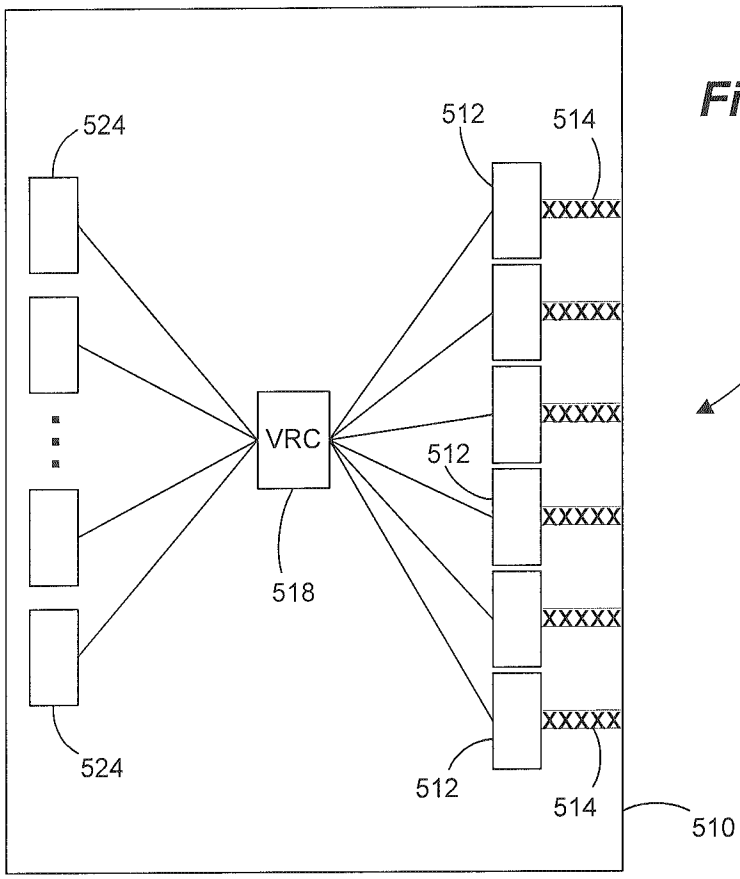


Figure 6E

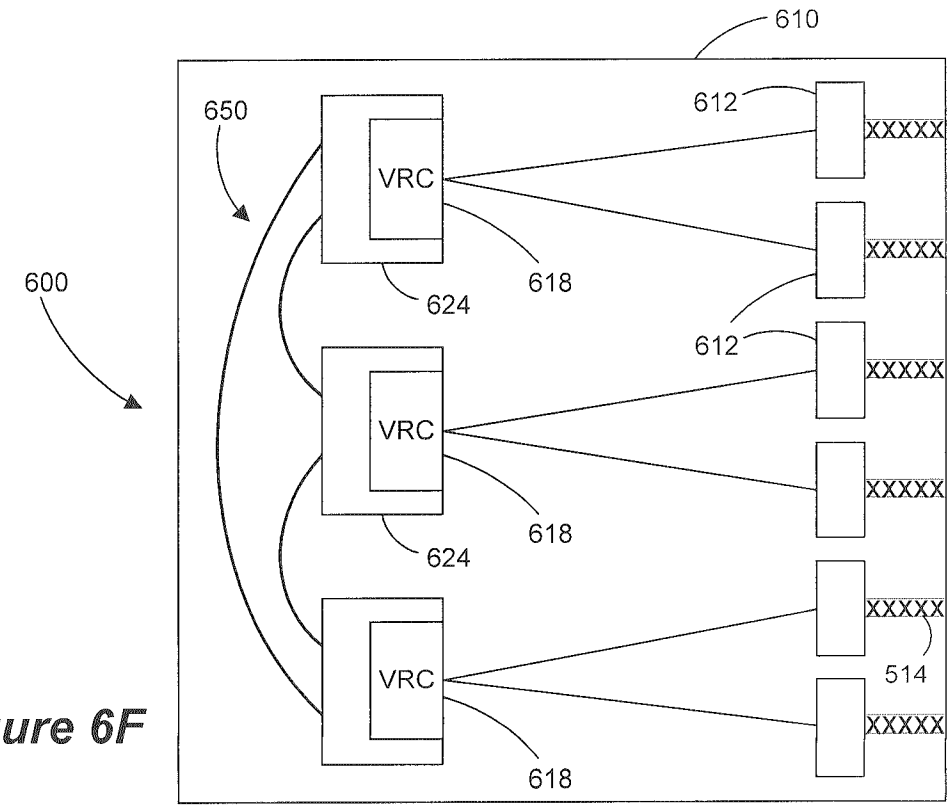


Figure 6F

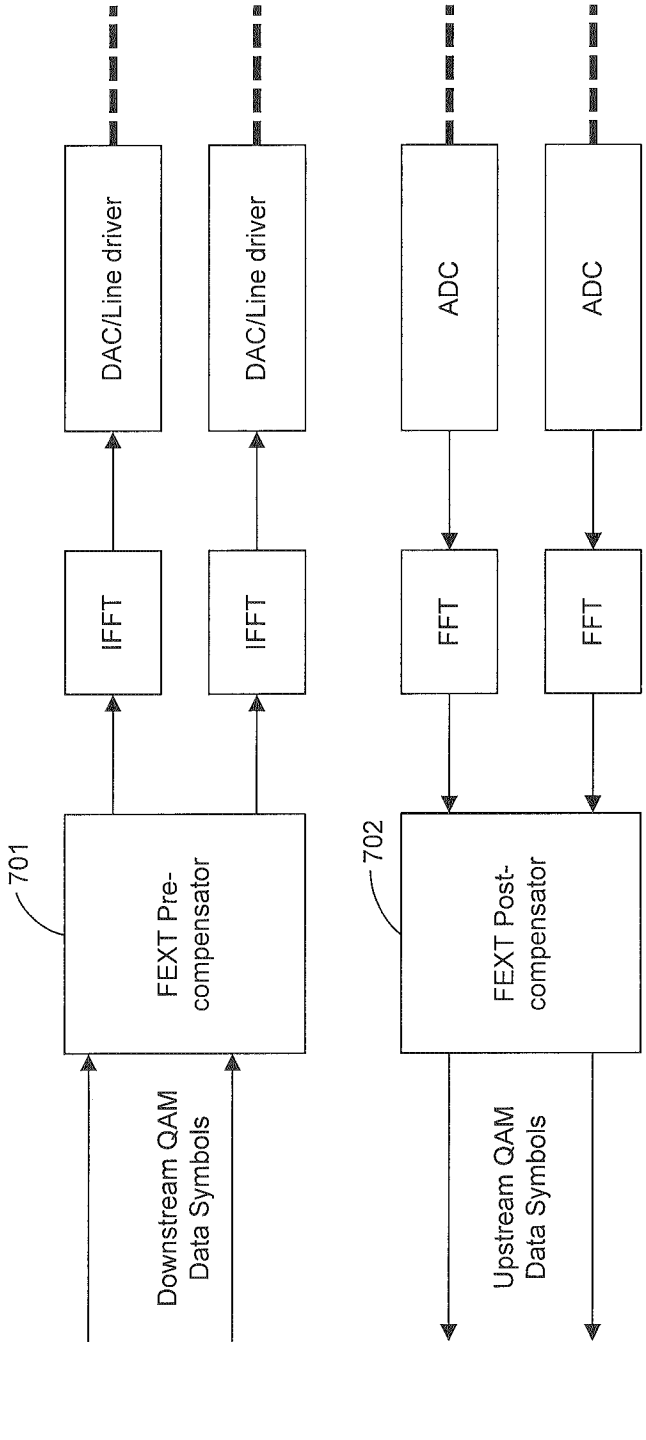

DAC/Line driver
DAC/Line driver
ADC
ADC
IFFT
IFFT
FFT
FFT
701
FEXT Pre-compensator
702
FEXT Post-compensator
Downstream QAM Data Symbols
Upstream QAM Data Symbols
700

Figure 7

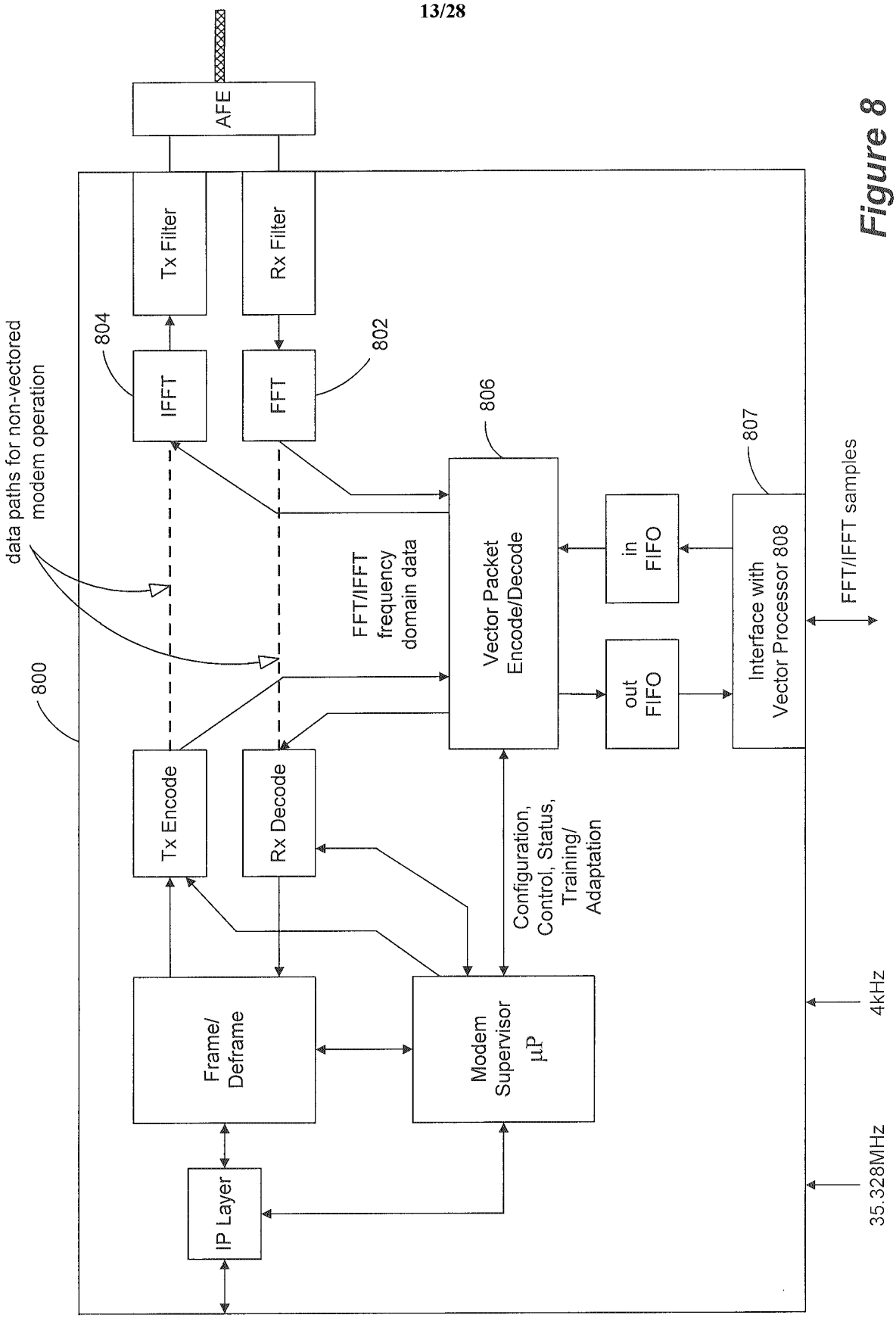
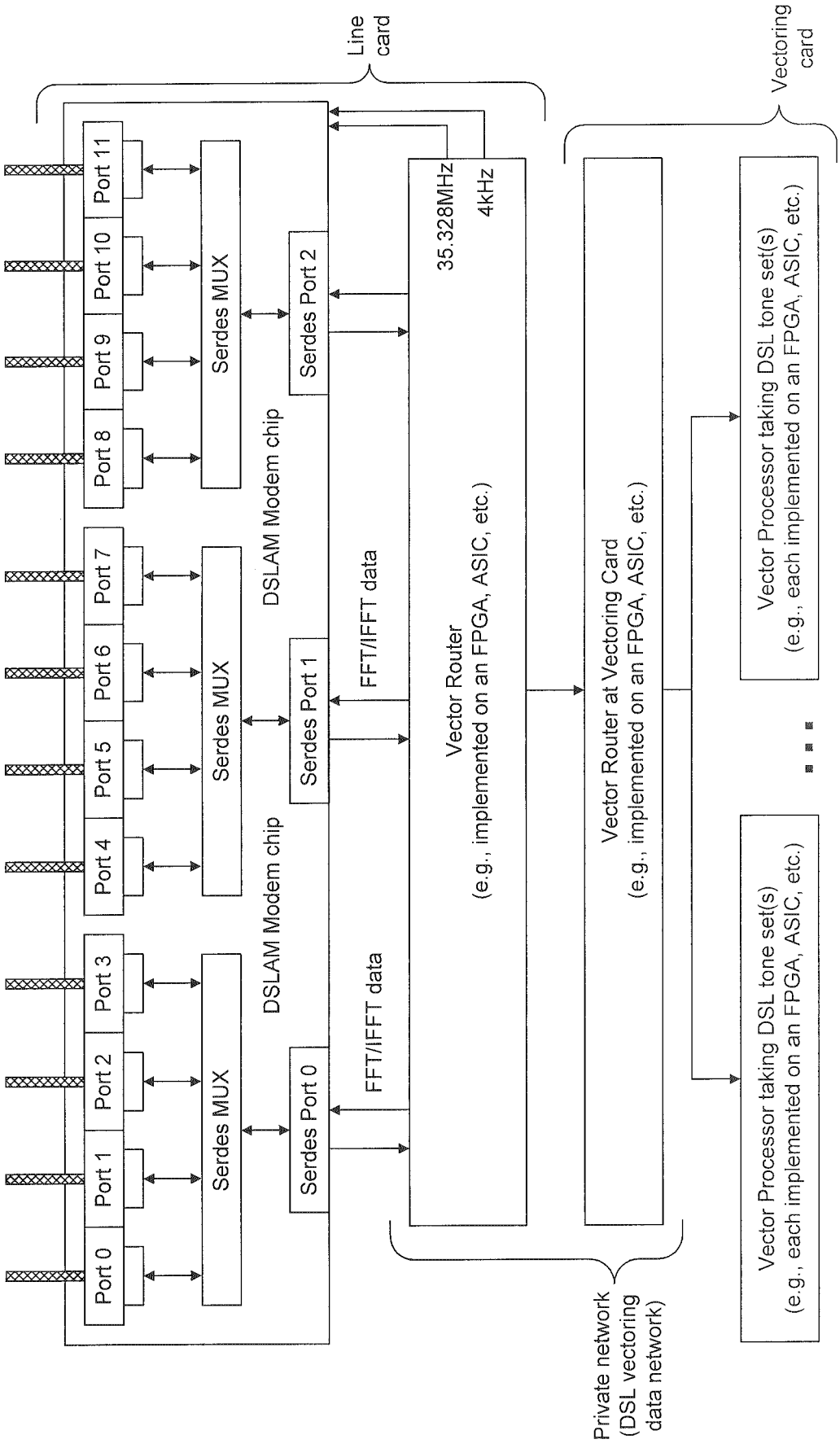

AFE
Tx Filter
Rx Filter
804
802
IFFT
FFT
806
807
data paths for non-vectored modem operation
FFT/IFFT frequency domain data
Vector Packet Encode/Decode
in FIFO
out FIFO
Interface with Vector Processor 808
FFT/IFFT samples
800
Tx Encode
Rx Decode
Configuration, Control, Status, Training/ Adaptation
Frame/ Deframe
Modem Supervisor µP
IP Layer
4kHz
35.328MHz

Figure 8

Figure 9A


Line card
Vectoring card
Port 0
Port 1
Port 2
Port 3
Port 4
Port 5
Port 6
Port 7
Port 8
Port 9
Port 10
Port 11
Serdes MUX
Serdes MUX
Serdes MUX
DSLAM Modem chip
DSLAM Modem chip
Serdes Port 0
Serdes Port 1
Serdes Port 2
FFT/IFFT data
FFT/IFFT data
35.328MHz
4kHz
Vector Router
(e.g., implemented on an FPGA, ASIC, etc.)
Vector Router at Vectoring Card
(e.g., implemented on an FPGA, ASIC, etc.)
Vector Processor taking DSL tone set(s)
(e.g., each implemented on an FPGA, ASIC, etc.)
Vector Processor taking DSL tone set(s)
(e.g., each implemented on an FPGA, ASIC, etc.)
Private network
(DSL vectoring
data network)

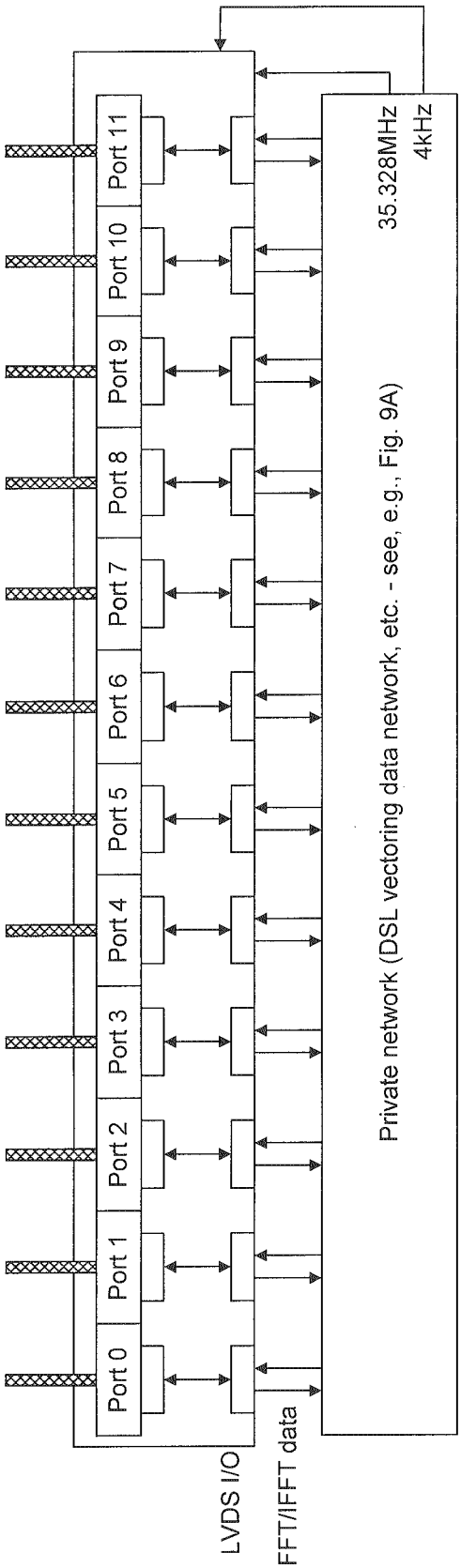

Port 0
Port 1
Port 2
Port 3
Port 4
Port 5
Port 6
Port 7
Port 8
Port 9
Port 10
Port 11
LVDS I/O
FFT/IFFT data
Private network (DSL vectoring data network, etc. - see, e.g., Fig. 9A)
35.328MHz
4kHz

Figure 9B

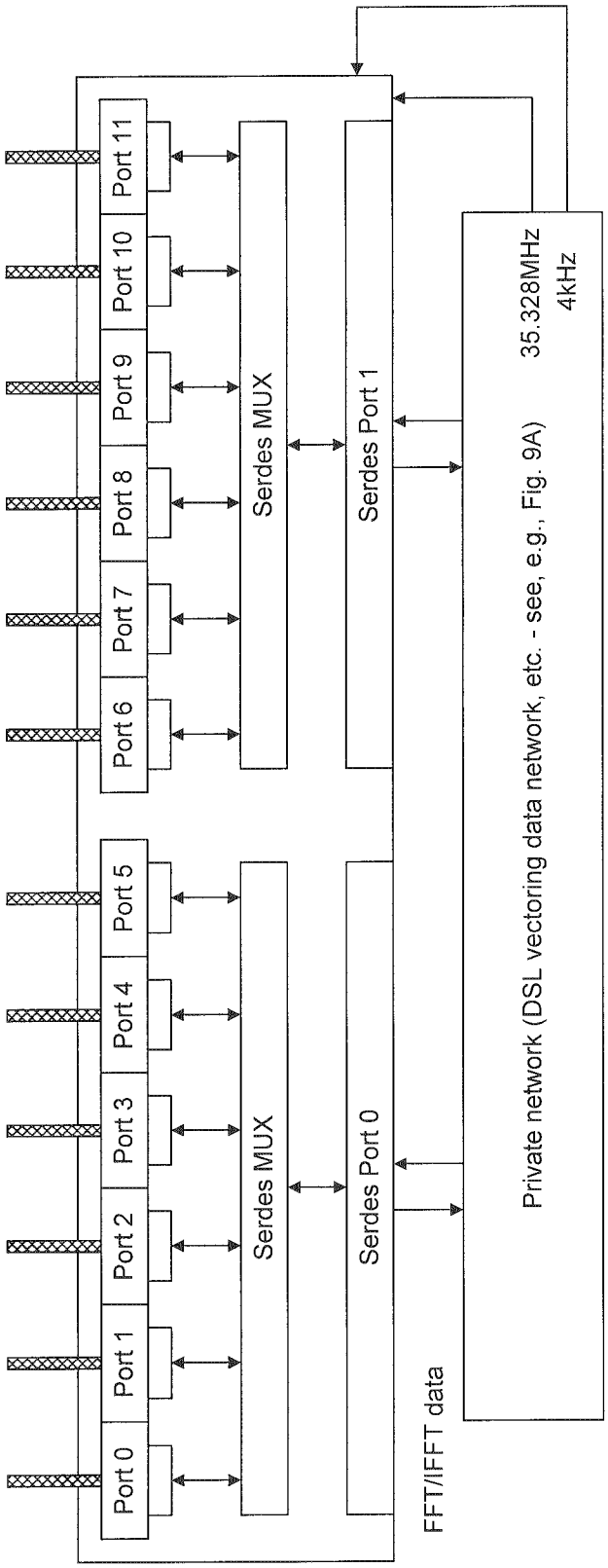

Port 0
Port 1
Port 2
Port 3
Port 4
Port 5
Port 6
Port 7
Port 8
Port 9
Port 10
Port 11
Serdes MUX
Serdes MUX
Serdes Port 0
Serdes Port 1
FFT/IFFT data
Private network (DSL vectoring data network, etc. - see, e.g., Fig. 9A)
35.328MHz
4kHz

Figure 9C

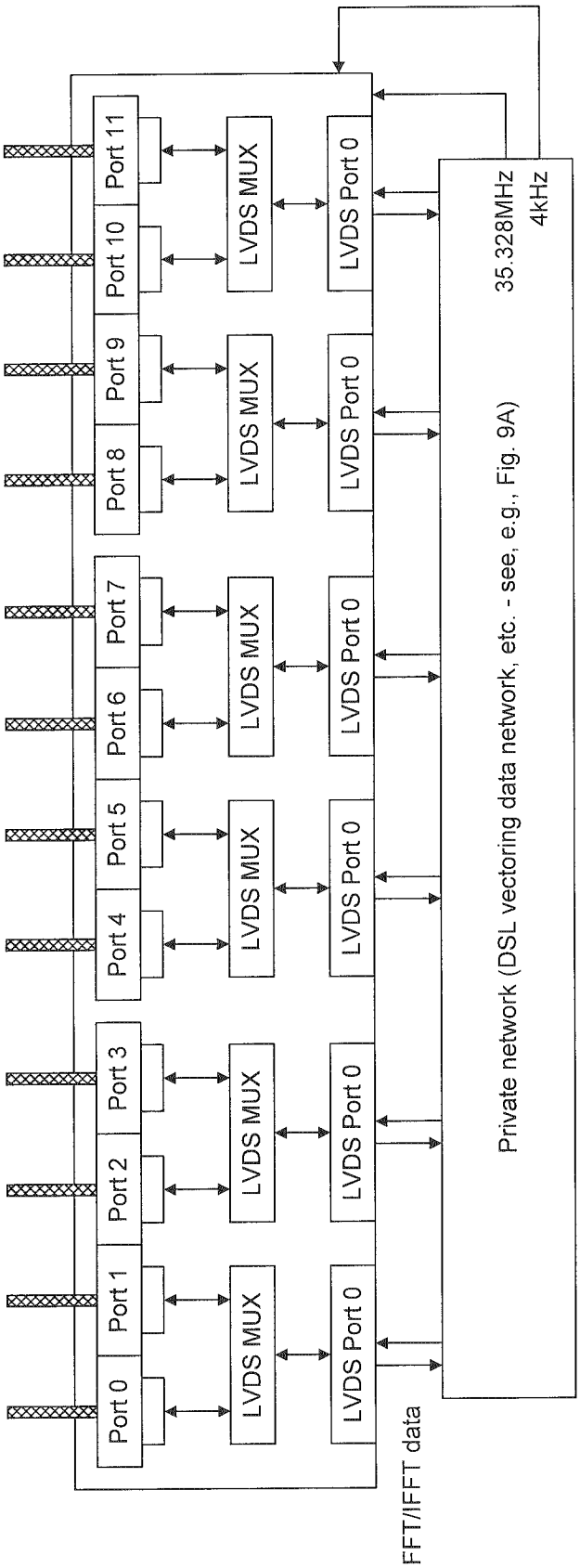

Port 0
Port 1
Port 2
Port 3
Port 4
Port 5
Port 6
Port 7
Port 8
Port 9
Port 10
Port 11
LVDS MUX
LVDS Port 0
FFT/IFFT data
Private network (DSL vectoring data network, etc. - see, e.g., Fig. 9A)
35.328MHz
4kHz

Figure 9D

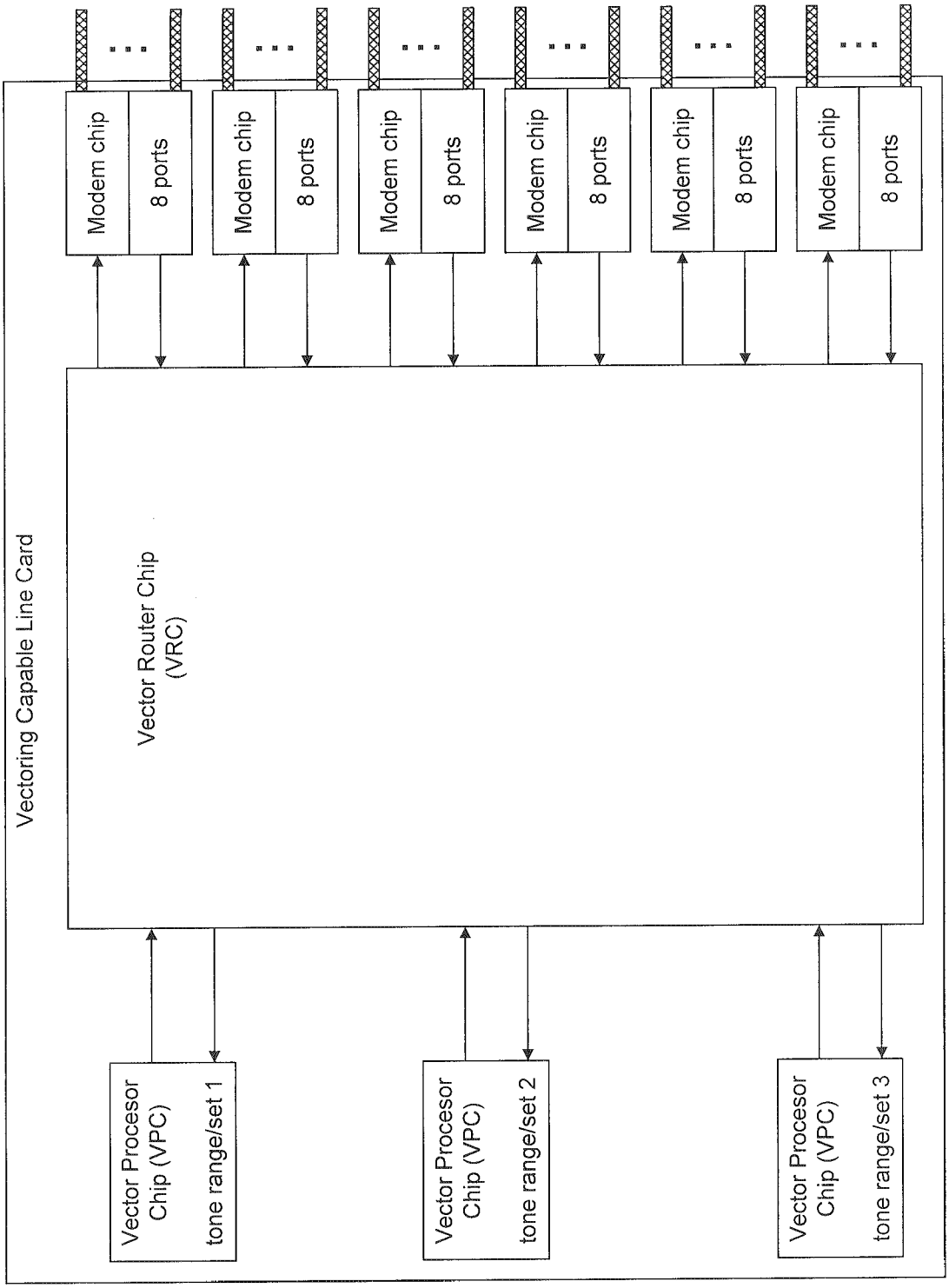

Vectoring Capable Line Card
Vector Router Chip (VRC)
Modem chip
8 ports
Vector Procesor Chip (VPC)
tone range/set 1
Vector Procesor Chip (VPC)
tone range/set 2
Vector Procesor Chip (VPC)
tone range/set 3

Figure 10A

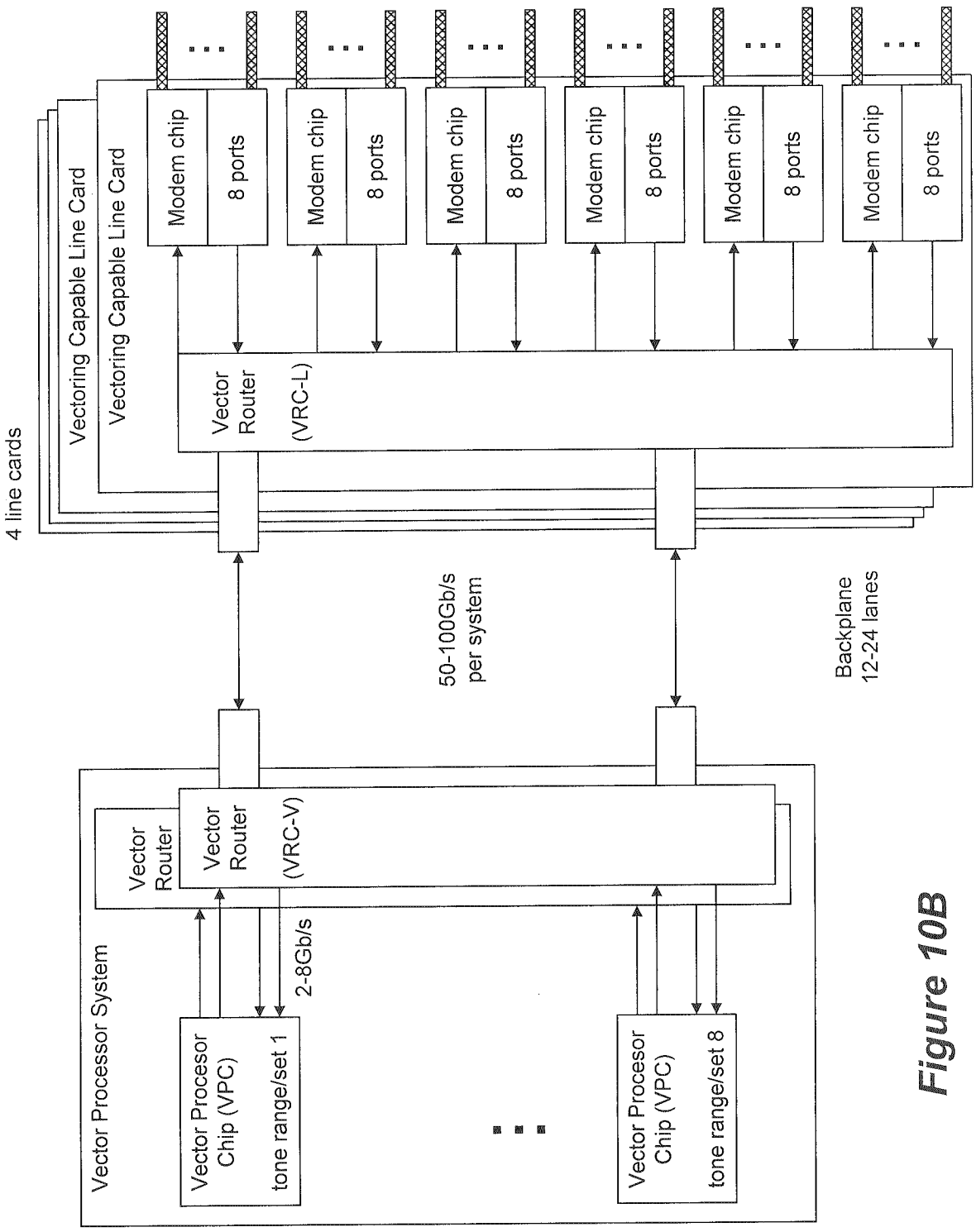

4 line cards
Vectoring Capable Line Card
Vectoring Capable Line Card
Modem chip
8 ports
Vector Router (VRC-L)
50-100Gb/s per system
Backplane 12-24 lanes
Vector Router
Vector Router (VRC-V)
2-8Gb/s
Vector Processor System
Vector Procesor Chip (VPC)
tone range/set 1
Vector Procesor Chip (VPC)
tone range/set 8

Figure 10B

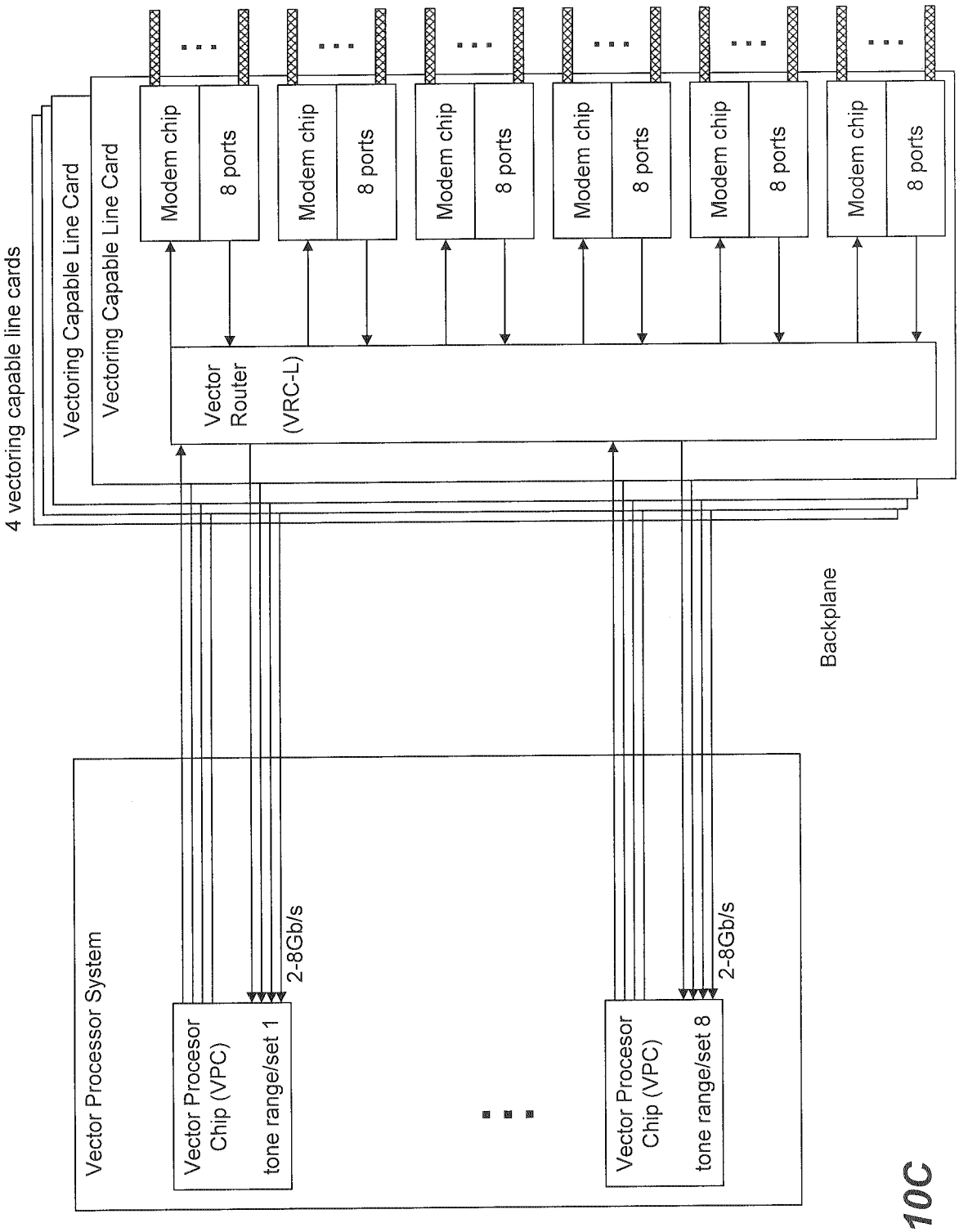

4 vectoring capable line cards
Vectoring Capable Line Card
Vectoring Capable Line Card
Vector Router (VRC-L)
Modem chip
8 ports
Backplane
Vector Processor System
Vector Procesor Chip (VPC)
tone range/set 1
Vector Procesor Chip (VPC)
tone range/set 8
2-8Gb/s
2-8Gb/s

Figure 10C

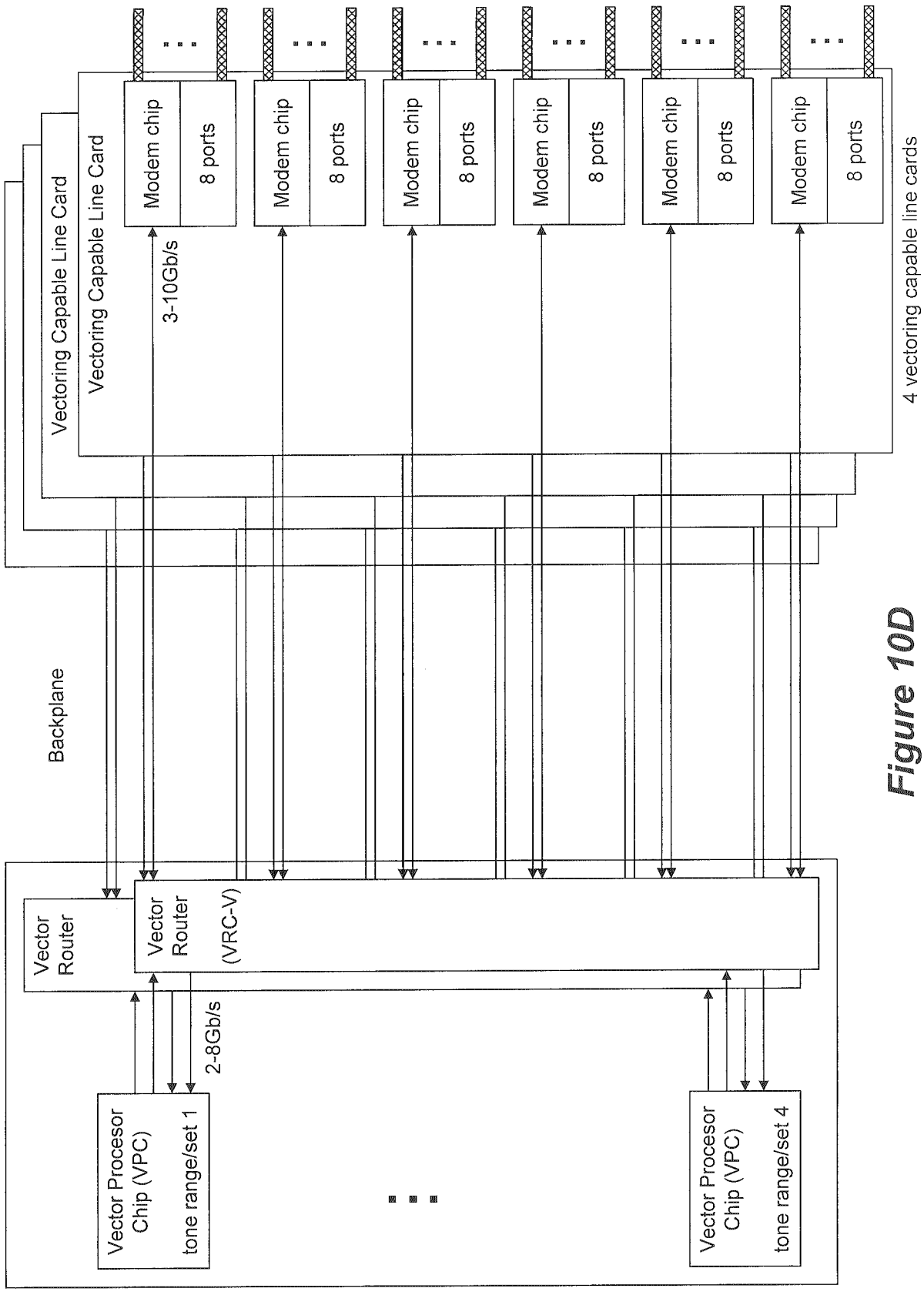

Vector Procesor Chip (VPC)
tone range/set 1
Vector Procesor Chip (VPC)
tone range/set 4
2-8Gb/s
Vector Router
Vector Router (VRC-V)
Backplane
Vectoring Capable Line Card
Vectoring Capable Line Card
3-10Gb/s
Modem chip
8 ports
4 vectoring capable line cards

Figure 10D

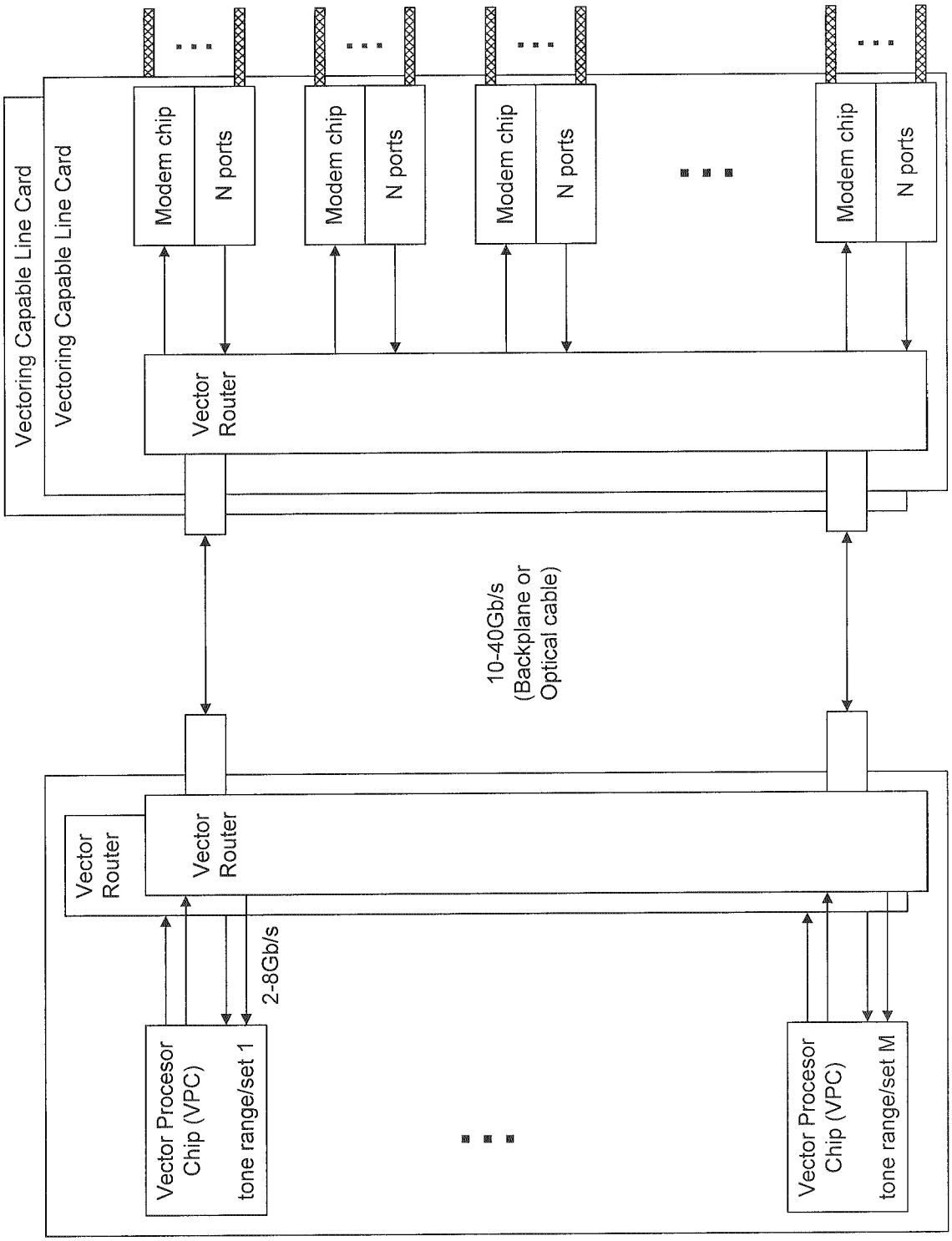

Vectoring Capable Line Card
Vectoring Capable Line Card
Modem chip
N ports
Modem chip
N ports
Modem chip
N ports
Modem chip
N ports
Vector Router
10-40Gb/s
(Backplane or
Optical cable)
Vector Router
Vector Router
2-8Gb/s
Vector Procesor
Chip (VPC)
tone range/set 1
Vector Procesor
Chip (VPC)
tone range/set M

Figure 10E

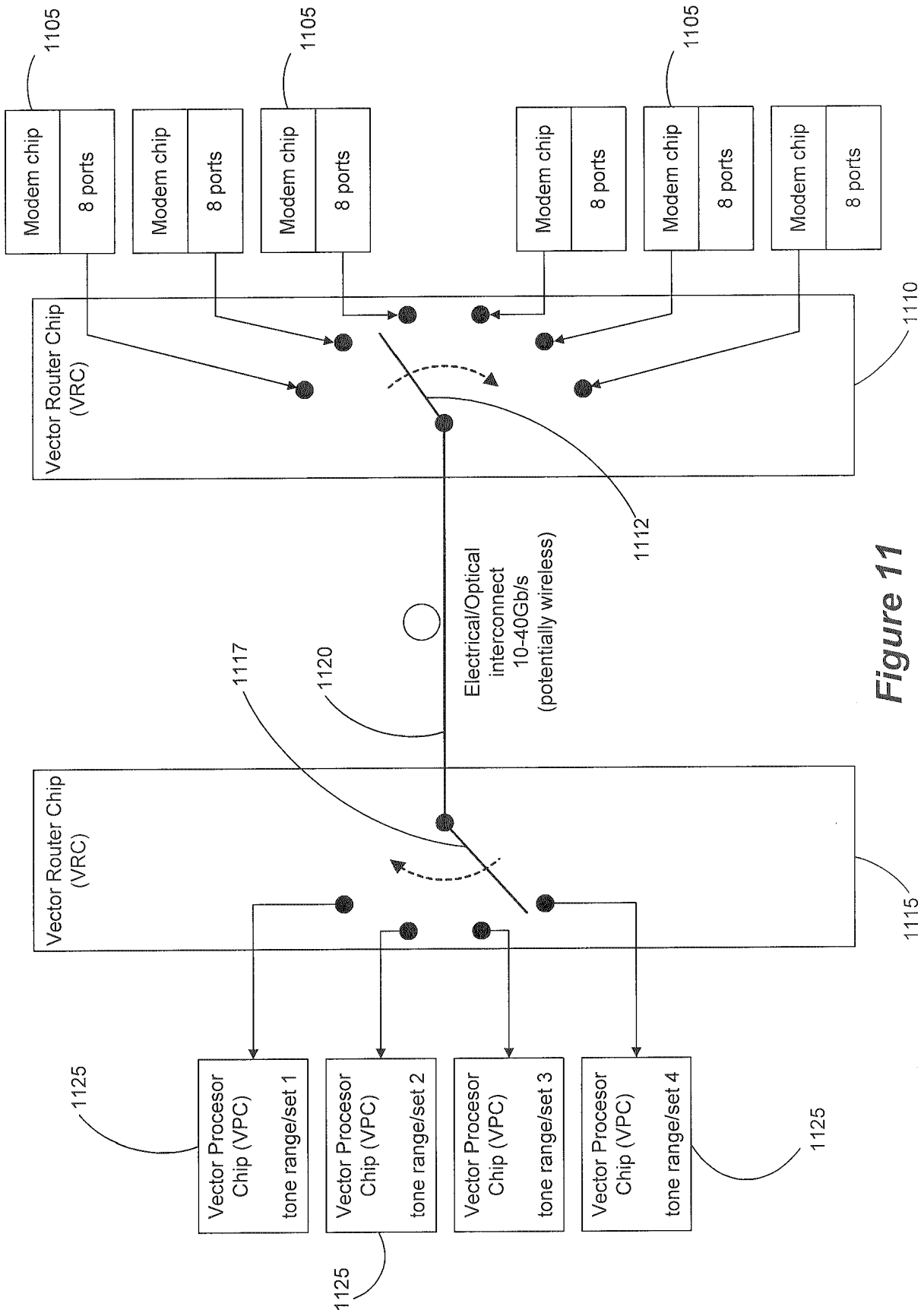

1105
Modem chip
8 ports
Vector Router Chip (VRC)
1110
1112
Electrical/Optical interconnect 10-40Gb/s (potentially wireless)
1117
1120
1115
Vector Procesor Chip (VPC)
tone range/set 1
tone range/set 2
tone range/set 3
tone range/set 4
1125

Figure 11

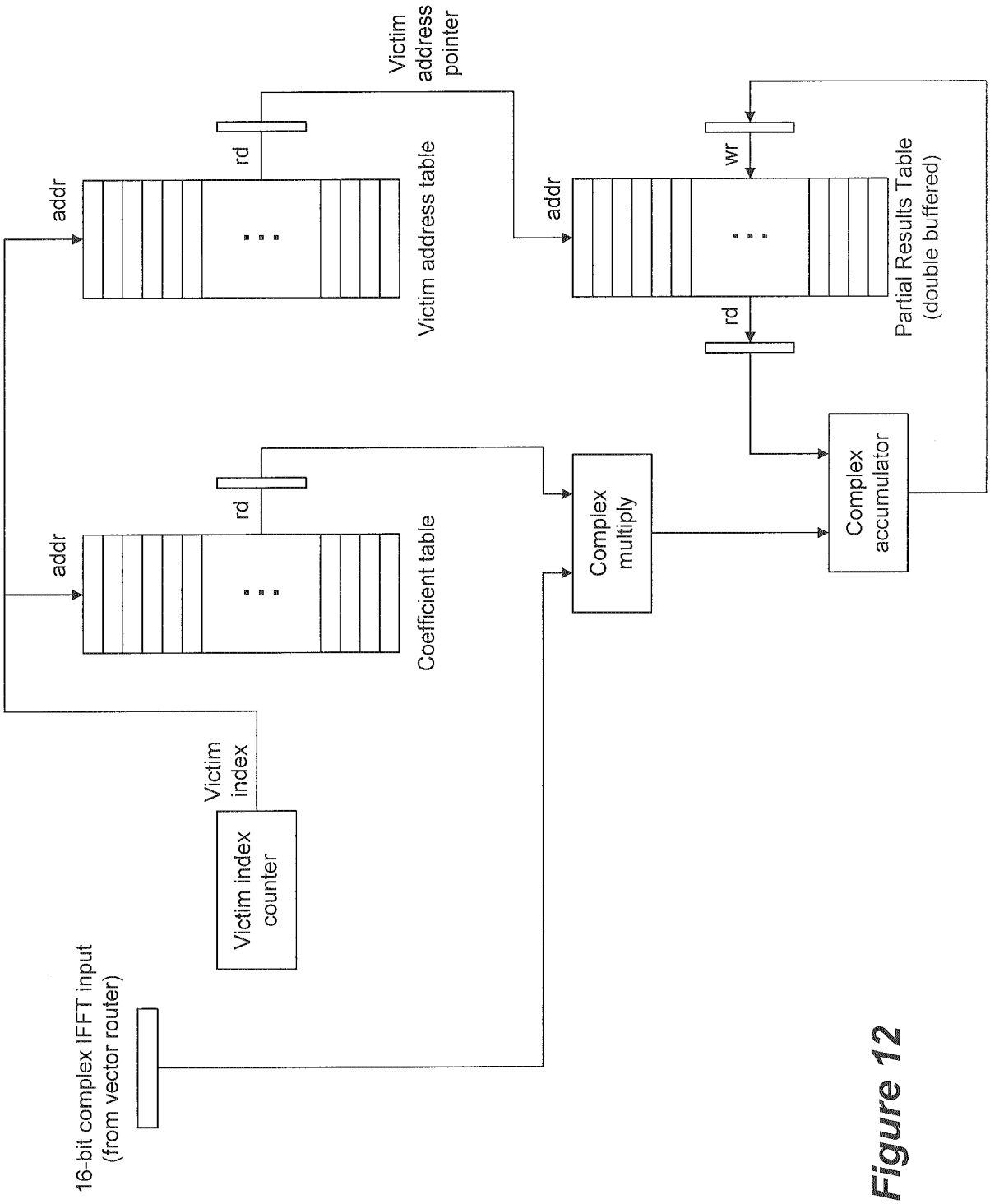

16-bit complex IFFT input
(from vector router)
Victim index counter
Victim index
addr
rd
Coefficient table
Complex multiply
addr
rd
Victim address table
Victim address pointer
addr
wr
rd
Partial Results Table
(double buffered)
Complex accumulator

Figure 12

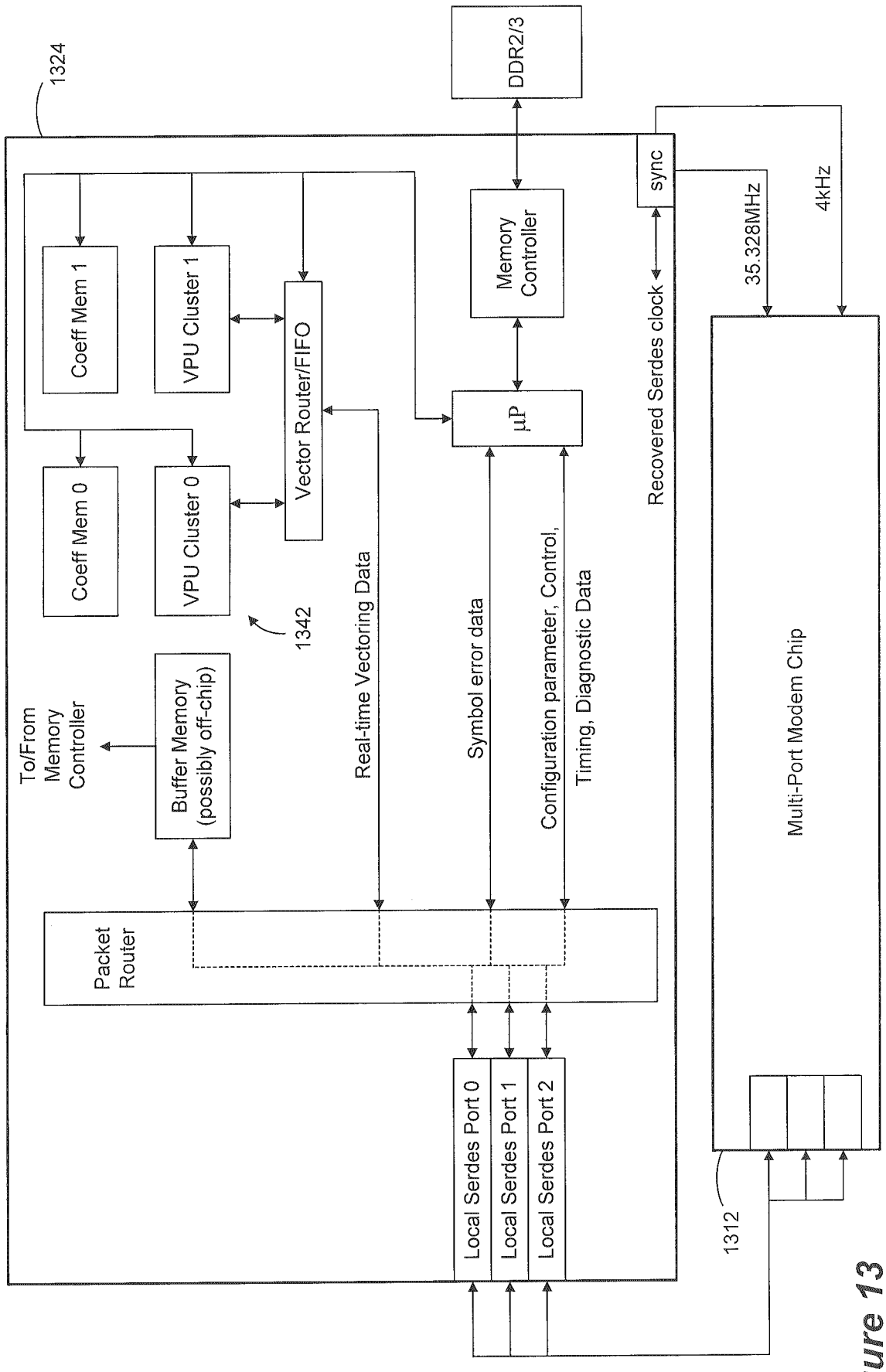
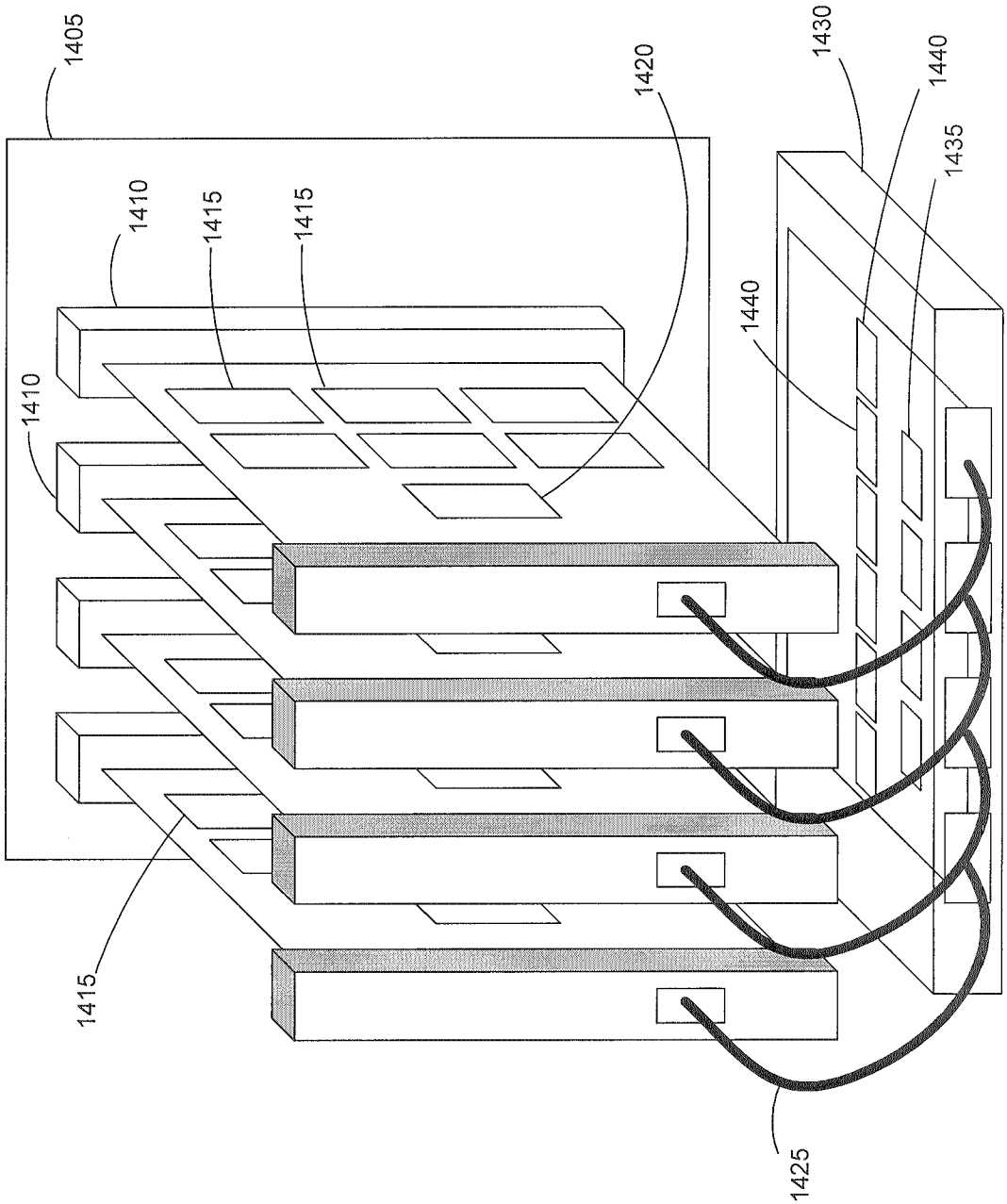

1324
DDR2/3
Memory Controller
sync
35.328MHz
4kHz
Recovered Serdes clock
Coeff Mem 1
VPU Cluster 1
Vector Router/FIFO
µP
Coeff Mem 0
VPU Cluster 0
1342
Real-time Vectoring Data
Symbol error data
Configuration parameter, Control,
Timing, Diagnostic Data
To/From Memory Controller
Buffer Memory (possibly off-chip)
Packet Router
Local Serdes Port 0
Local Serdes Port 1
Local Serdes Port 2
Multi-Port Modem Chip
1312

Figure 13

Figure 14



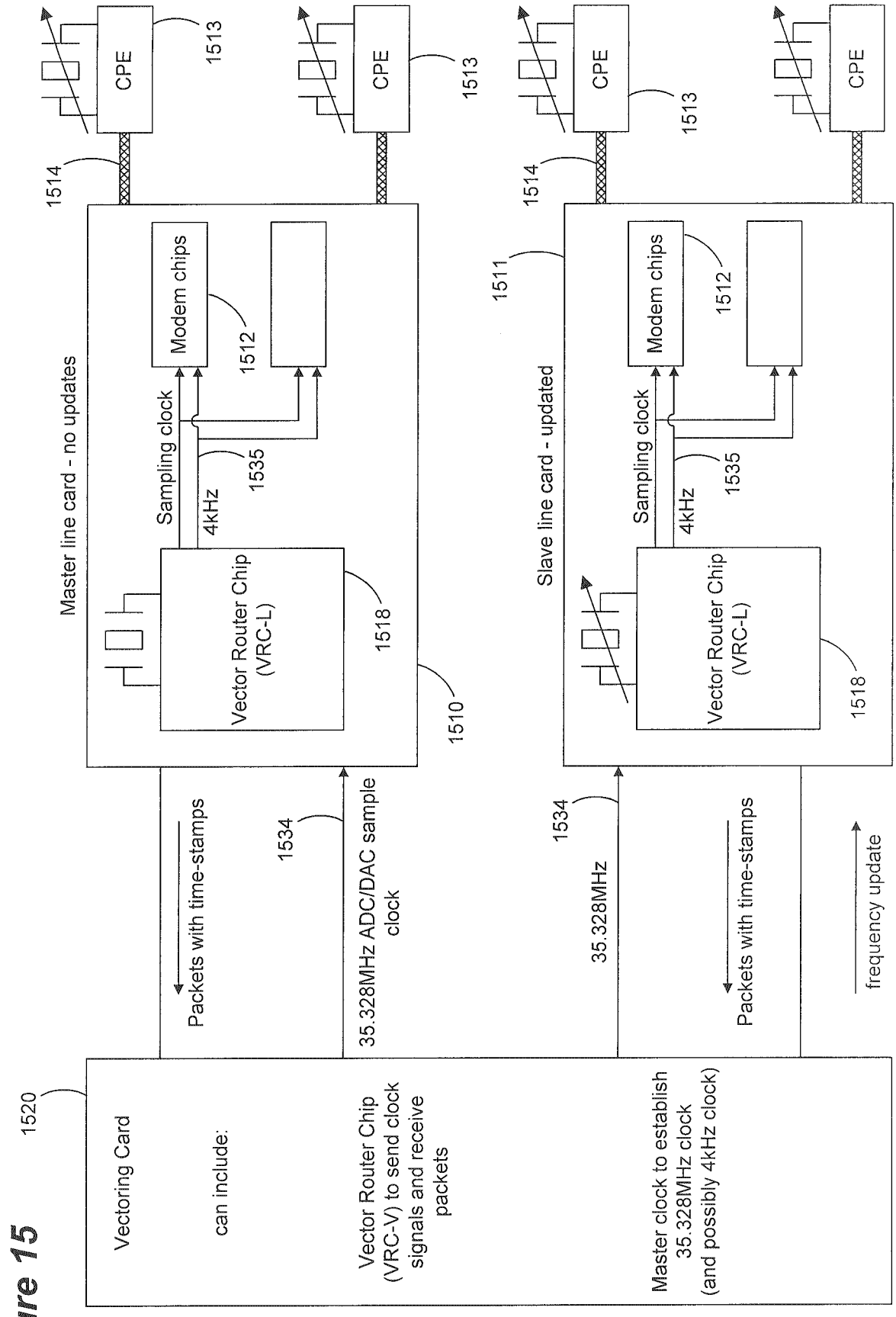

Vectoring Card
can include:
Vector Router Chip (VRC-V) to send clock signals and receive packets
Master clock to establish 35.328MHz clock (and possibly 4kHz clock)
1520
Packets with time-stamps
35.328MHz ADC/DAC sample clock
1534
35.328MHz
Packets with time-stamps
frequency update
Master line card - no updates
Slave line card - updated
1510
1511
Vector Router Chip (VRC-L)
1518
Sampling clock
4kHz
1535
Modem chips
1512
1514
CPE
1513

Figure 15

Figure 16

Sub-Field Names	Sub-Field Length (bits)	Parameter Description	Field Length (bits)	Field Name
Ports In Packet	16	16-bit map of ports included in packet. LSB=port 0	**16**	**Ports in Packet**
Superframe Index	15	this allows pilot sequence lengths up to 1024. Index resets to zero at start of Pilot Symbol Sequence.	**24**	**Index**
Symbol Index	9	0:255=data frames, 256= sync frame, 257-511 undefined		
Band StartBin Index	16	first tone of band	**32**	**Bin Range**
Band StopBin Index	16	last tone of band		
Port 0 Data	2*b	b=configured FFT/IFFT data precision, port 0 first	**2*N*b**	**StartBin Payload**
Port 1 Data	2*b			
...	...			
Port N-1 Data	2*b			
Port 0 Data	2*b	b=configured FFT/IFFT data precision	**2*N*b**	**StartBin+1 Payload**
Port 1 Data	2*b			
...	...			
Port N-1 Data	2*b			
...	**...**	**...**	**...**	**...**
Port 0 Data	2*b	b=configured FFT/IFFT data precision	**2*N*b**	**StopBin Payload**
Port 1 Data	2*b			
...	...			
Port N-1 Data	2*b			
CRC	16	Not required if included in Serdes protocol. Can pad Payload to make entire payload 16 bit multiple.	**16**	**CRC**