US008045935B2

(12) **United States Patent**
Lakkis et al.

(10) **Patent No.: US 8,045,935 B2**
(45) **Date of Patent: Oct. 25, 2011**

(54) **HIGH DATA RATE TRANSMITTER AND RECEIVER**

(75) Inventors: **Ismail Lakkis**, San Diego, CA (US); **Yasaman Bahreini**, San Diego, CA (US); **John Santhoff**, Carlsbad, CA (US)

(73) Assignee: **Pulse-Link, Inc.**, Carlsbad, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1273 days.

(21) Appl. No.: **11/055,525**

(22) Filed: **Feb. 9, 2005**

(65) **Prior Publication Data**

US 2005/0233710 A1 Oct. 20, 2005

Related U.S. Application Data

(63) Continuation-in-part of application No. 10/010,601, filed on Dec. 6, 2001, now Pat. No. 7,289,494.

(51) **Int. Cl.**
H04B 1/02 (2006.01)
H04B 1/04 (2006.01)

(52) **U.S. Cl.** **455/91**; 455/102; 455/118; 370/310

(58) **Field of Classification Search** 455/102, 455/118, 258, 91, 105, 113, 141, 147, 208, 455/209, 255, 504, 506, 63.1, 114.2, 278.1, 455/296; 375/134, 135, 142, 146; 370/319, 370/320, 321, 335, 337, 344, 347, 310, 314, 370/329

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,568,148 A 3/1971 Clark, Jr.
4,001,693 A 1/1977 Stackhouse et al.
4,039,972 A 8/1977 Scholten et al.
4,278,978 A 7/1981 Easterling et al.
4,317,210 A 2/1982 Dekker et al.
4,498,174 A 2/1985 LeGresley
4,569,052 A 2/1986 Cohen et al.
4,632,124 A 12/1986 Hiller et al.
4,707,839 A 11/1987 Andren et al.
4,847,876 A 7/1989 Baumbach et al.
4,860,003 A 8/1989 DeLuca et al.
4,905,234 A 2/1990 Childress et al.
5,041,836 A 8/1991 Paschen et al.

(Continued)

OTHER PUBLICATIONS

Suk Won Kim, Dong Sam Ha & Jeong Ho Kim, Performance Gain of Smart Antennas with Diversity Combining at Handsets for the 3GPP WCDMA System, 13th International Conference on Wireless Communications (Wireless 2001), Jul. 2001, pp. 235-242, Calgary, ALberta, Canada.

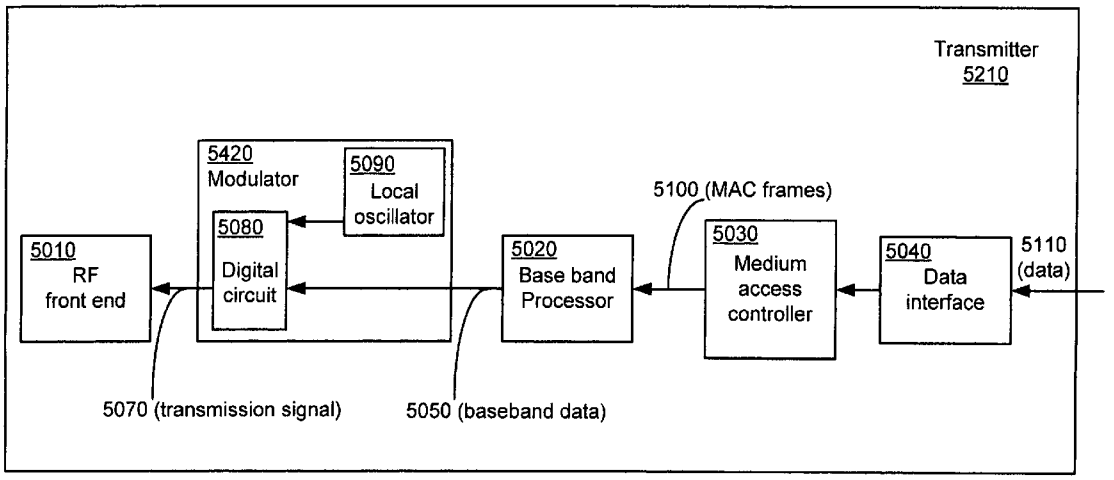
(Continued)

Primary Examiner — Sonny Trinh

(57) **ABSTRACT**

A high-speed transmitter and receiver are provided. In one embodiment, a transmitter comprises a baseband processor structured to receive data and to convert the data into a multiplicity of high and low signal values, with each high and low signal value having a first timing interval. A local oscillator generates a clock signal at a second timing interval and a digital circuit combines the high and low signal values with the clock signal to produce a transmission signal directly at a transmission frequency. A receiver is configured to receive the signal. This Abstract is provided for the sole purpose of complying with the Abstract requirement rules that allow a reader to quickly ascertain the subject matter of the disclosure contained herein. This Abstract is submitted with the explicit understanding that it will not be used to interpret or to limit the scope or the meaning of the claims.

7 Claims, 45 Drawing Sheets



U.S. PATENT DOCUMENTS

5,087,835	A	2/1992	Hattangadi	
5,109,390	A	4/1992	Gilhousen et al.	
5,170,412	A	12/1992	Massey	
5,177,765	A	1/1993	Holland et al.	
5,177,767	A	1/1993	Kato	
5,235,326	A	8/1993	Beigel et al.	
5,260,976	A	11/1993	Dolivo et al.	
5,377,223	A	12/1994	Schilling	
5,389,939	A	2/1995	Tang et al.	
5,396,110	A	3/1995	Houston	
5,416,767	A	5/1995	Koppelaar et al.	
5,422,952	A	6/1995	Kennedy et al.	
5,457,702	A	10/1995	Williams et al.	
5,467,367	A	11/1995	Izumi et al.	
5,555,183	A	9/1996	Willard et al.	
5,561,468	A	10/1996	Bryan et al.	
5,563,909	A	10/1996	Nakazawa	
5,590,399	A	12/1996	Matsumoto et al.	
5,606,729	A	2/1997	D'Amico et al.	
5,633,889	A	5/1997	Schilling et al.	
5,642,391	A	6/1997	Ewell et al.	
5,644,568	A	7/1997	Ayerst et al.	
5,675,572	A	10/1997	Hidejima et al.	
5,677,927	A	10/1997	Fullerton et al.	
5,680,419	A	10/1997	Bottomley	
5,734,963	A	3/1998	Fitzgerals et al.	
5,757,766	A	5/1998	Sugita	
5,784,387	A	7/1998	Widmer	
5,787,118	A	7/1998	Ueda	
5,790,591	A	8/1998	Gold et al.	
5,793,756	A	8/1998	Ayerst et al.	
5,809,060	A	9/1998	Cafarella et al.	
5,822,359	A	10/1998	Bruckert et al.	
5,835,498	A	11/1998	Kim et al.	
5,852,770	A	12/1998	Kasamatsu	
5,867,478	A	2/1999	Baum et al.	
5,870,432	A	2/1999	Kerckhove	
5,903,556	A	5/1999	Matui	
5,926,503	A	7/1999	Kelton et al.	
5,930,296	A	7/1999	Kot	
5,943,331	A	8/1999	Lavean	
5,949,813	A	9/1999	Hunsinger et al.	
5,974,032	A *	10/1999	Snowden et al.	370/316
5,978,412	A	11/1999	Takai	
5,982,807	A *	11/1999	Snell	375/146
5,991,308	A	11/1999	Fuhrmann et al.	
5,991,331	A	11/1999	Chennakeshu et al.	
6,005,640	A	12/1999	Strolle et al.	
6,012,161	A	1/2000	Ariyavisitakul et al.	
6,023,783	A	2/2000	Divsalar et al.	
6,032,033	A	2/2000	Mprros et al.	
6,034,987	A	3/2000	Chennakeshu et al.	
6,049,707	A	4/2000	Buer et al.	
6,064,663	A	5/2000	Honkasalo et al.	
6,067,290	A	5/2000	Paulraj et al.	
6,078,576	A	6/2000	Schilling et al.	
6,078,607	A	6/2000	Monroe et al.	
6,081,562	A	6/2000	He et al.	
6,108,317	A	8/2000	Jones et al.	
6,128,276	A	10/2000	Agee	
6,128,330	A	10/2000	Schilling	
6,128,387	A	10/2000	Hsieh	
6,128,473	A	10/2000	Ritzen et al.	
6,133,876	A	10/2000	Fullerton et al.	
6,141,374	A	10/2000	Burns	
6,142,946	A	11/2000	Hwang et al.	
6,154,486	A	11/2000	Scott et al.	
6,185,258	B1	2/2001	Alamouti et al.	
6,191,724	B1	2/2001	McEwan	
6,192,068	B1	2/2001	Fattouche et al.	
6,198,327	B1	3/2001	Takahashi	
6,198,989	B1	3/2001	Tankhilevich et al.	
6,215,777	B1	4/2001	Chen et al.	
6,215,814	B1	4/2001	Ylitalo et al.	
6,222,832	B1	4/2001	Proctor	
6,240,081	B1 *	5/2001	Khalona	370/342
6,256,508	B1	7/2001	Nakagawa et al.	
6,259,687	B1	7/2001	Lomp et al.	
6,298,092	B1	10/2001	Heath, Jr. et al.	
6,298,726	B1	10/2001	Adachi et al.	
6,300,903	B1	10/2001	Richards et al.	
6,301,681	B1	10/2001	Chen et al.	
6,307,868	B1	10/2001	Rakib et al.	
6,310,866	B1	10/2001	Kronestedt et al.	
6,317,411	B1	11/2001	Whinnett et al.	
6,320,897	B1	11/2001	Fattouche et al.	
6,337,878	B1	1/2002	Endres et al.	
6,351,499	B1	2/2002	Paulraj et al.	
6,356,607	B1	3/2002	Scott et al.	
6,363,107	B1	3/2002	Scott	
6,366,613	B2	4/2002	Sommer et al.	
6,373,827	B1	4/2002	Tayebi et al.	
6,388,997	B1	5/2002	Scott	
6,392,500	B1	5/2002	McCallister et al.	
RE37,802	E	7/2002	Fattouche et al.	
6,430,170	B1	8/2002	Saints et al.	
6,459,246	B1	10/2002	Roo	
6,459,728	B1	10/2002	Bar-David et al.	
6,466,610	B1	10/2002	Schilling	
6,473,467	B1	10/2002	Wallace et al.	
6,501,841	B1	12/2002	Horne	
6,526,090	B1	2/2003	Papasakellariou	
6,549,567	B1	4/2003	Fullerton	
6,549,588	B2	4/2003	Neubauer	
6,560,463	B1	5/2003	Santhoff	
6,563,856	B1	5/2003	O 'Shea et al.	
6,570,912	B1	5/2003	Mirfakhraei	
6,570,919	B1	5/2003	Lee	
6,603,818	B1	8/2003	Dress, Jr.	
6,606,350	B2	8/2003	Dress, Jr.	
6,630,897	B2	10/2003	Low et al.	
6,650,910	B1	11/2003	Mazur et al.	
6,657,551	B2	12/2003	Huckaba et al.	
6,683,955	B1	1/2004	Horne	
6,690,723	B1	2/2004	Gosse et al.	
6,690,741	B1	2/2004	Larrick et al.	
6,700,881	B1	3/2004	Kong et al.	
6,700,939	B1	3/2004	McCorkle et al.	
6,701,129	B1	3/2004	Hashem et al.	
6,728,323	B1 *	4/2004	Chen et al.	375/340
6,748,025	B1	6/2004	Hickling	
6,754,294	B1	6/2004	Adireddy et al.	
6,757,543	B2	6/2004	Moran et al.	
6,763,075	B2	7/2004	Zhengdi et al.	
6,765,517	B1	7/2004	Ali	
6,771,976	B1	8/2004	Koyama	
6,775,324	B1 *	8/2004	Mohan et al.	375/238
6,781,470	B2	8/2004	Rogerson	
6,785,323	B1	8/2004	Proctor, Jr.	
6,795,424	B1	9/2004	Kooper et al.	
6,810,259	B1	10/2004	Zhang	
6,833,767	B1	12/2004	Huff et al.	
6,842,495	B1	1/2005	Jaffe et al.	
6,851,086	B2	2/2005	Szymanski	
6,856,652	B2	2/2005	West et al.	
6,865,174	B1 *	3/2005	Tsubouchi et al.	370/342
6,904,110	B2	6/2005	Trans et al.	
6,931,078	B2	8/2005	McCorkle et al.	
6,937,667	B1	8/2005	Fullerton et al.	
6,941,378	B2	9/2005	Apostolopoulos et al.	
6,944,232	B2	9/2005	Koga et al.	
6,946,981	B1	9/2005	Zilberman et al.	
6,947,498	B2	9/2005	Reed	
6,952,444	B1	10/2005	Segal et al.	
6,952,456	B1	10/2005	Aiello et al.	
6,980,083	B2 *	12/2005	Sako et al.	340/7.52
6,990,348	B1	1/2006	Benveniste	
7,046,187	B2	5/2006	Fullerton et al.	
7,050,508	B2	5/2006	Sorrells	
7,072,413	B2	7/2006	Walton et al.	
7,076,168	B1	7/2006	Shattil	
7,076,927	B2	7/2006	Trpkovski	
7,088,795	B1	8/2006	Aiello et al.	
7,110,472	B2 *	9/2006	Sakoda et al.	375/316
7,123,843	B2	10/2006	Melick et al.	
7,145,954	B1	12/2006	Pendergrass et al.	
7,154,955	B2	12/2006	Suwa et al.	

7,206,349	B2	4/2007	Linnartz et al.	
7,209,523	B1	4/2007	Larrick et al.	
7,236,757	B2	6/2007	Raghavan et al.	
7,336,699	B2 *	2/2008	Mathew et al.	375/149
7,415,626	B2 *	8/2008	Lilja et al.	713/500
2001/0017563	A1	8/2001	Kim	
2001/0019282	A1	9/2001	Kuhne	
2001/0025361	A1	9/2001	Kim	
2001/0026584	A1	10/2001	Sommer et al.	
2001/0030973	A1	10/2001	Lomp et al.	
2001/0038674	A1	11/2001	Trans	
2001/0040931	A1	11/2001	Samueli et al.	
2001/0048333	A1	12/2001	Fujii	
2001/0050623	A1	12/2001	Kahlman	
2002/0034191	A1	3/2002	Shattil	
2002/0042899	A1	4/2002	Tzannes et al.	
2002/0048333	A1	4/2002	Ahmed et al.	
2002/0064245	A1	5/2002	McCorkle	
2002/0097790	A1	7/2002	Dress et al.	
2002/0114355	A1	8/2002	Kim et al.	
2002/0118783	A1	8/2002	Cripps et al.	
2002/0126769	A1	9/2002	Jett et al.	
2002/0146994	A1	10/2002	Marrah et al.	
2002/0159425	A1	10/2002	Uesugi et al.	
2002/0176511	A1	11/2002	Fullerton et al.	
2002/0190786	A1	12/2002	Yoon et al.	
2003/0033575	A1	2/2003	Richardson et al.	
2003/0040329	A1	2/2003	Yona et al.	
2003/0048840	A1	3/2003	Ling et al.	
2003/0053555	A1	3/2003	McCorkle et al.	
2003/0067963	A1	4/2003	Miller et al.	
2003/0079171	A1	4/2003	Coe	
2003/0091100	A1	5/2003	El Nahas El Homsi et al.	
2003/0104788	A1	6/2003	Kim	
2003/0108133	A1	6/2003	Richards	
2003/0128776	A1	7/2003	Rawlins et al.	
2003/0147655	A1	8/2003	Shattil	
2003/0161411	A1	8/2003	McCorkle et al.	
2003/0193970	A1	10/2003	Kim et al.	
2003/0202537	A1	10/2003	Rogerson et al.	
2003/0227980	A1	12/2003	Batra et al.	
2004/0001540	A1	1/2004	Jones	
2004/0008617	A1	1/2004	Dabak et al.	
2004/0034828	A1	2/2004	Hocevar	
2004/0047430	A1	3/2004	McCarty	
2004/0052312	A1	3/2004	Matero	
2004/0057575	A1	3/2004	Zhang et al.	
2004/0070433	A1	4/2004	Adams et al.	
2004/0071249	A1	4/2004	Schell	
2004/0086027	A1	5/2004	Shattil	
2004/0087291	A1	5/2004	Wada	
2004/0098659	A1	5/2004	Bjerke et al.	
2004/0141561	A1	7/2004	Santhoff et al.	
2004/0151269	A1	8/2004	Balakrishnan	
2004/0153938	A1	8/2004	Okamura et al.	
2004/0161064	A1	8/2004	Brethour et al.	
2004/0174928	A1	9/2004	Siwiak et al.	
2004/0176121	A1	9/2004	Zhou	
2004/0213351	A1	10/2004	Shattil	
2004/0213352	A1	10/2004	Lundby et al.	
2004/0218581	A1	11/2004	Cattaneo	
2004/0218683	A1	11/2004	Batra et al.	
2004/0224656	A1	11/2004	Mimura et al.	
2004/0234147	A1	11/2004	Malvar	
2004/0239562	A1	12/2004	Schantz	
2005/0031045	A1	2/2005	Mayor et al.	
2005/0053121	A1	3/2005	Lakkis	
2005/0084031	A1	4/2005	Rosen et al.	
2005/0105594	A1	5/2005	Giannakis et al.	
2005/0157782	A1	7/2005	Lakkis	
2005/0159129	A1	7/2005	Iida	
2005/0207351	A1	9/2005	Inagawa	
2006/0039317	A1	2/2006	Subramanian et al.	
2006/0193373	A1	8/2006	Agee et al.	
2006/0291537	A1	12/2006	Fullerton et al.	
2007/0053471	A1	3/2007	Feher	

OTHER PUBLICATIONS

Yiannia Argyropoulos, Scott Jordan & Srikanta P. R. Kumar, Dynamic Channel Allocation in Interference-Limited Cellular SYstems with Uneven Traffic Distribution, IEEE Transactions on Vehicular Technology, vol. 48, No. 1, Jan. 1999, pp. 224-232.

Anurag Kahol, Sumit Khurana, Sandeep K Gupta & Pradip K Srimani, Adaptive Distributed Dynamic Channel Allocation for Wireless Networks, Computer Science Technical Report CS-98-105, May 27, 1998, 11 pages.

Shengli Zhou & Georgios B. Giannakis, Space-Time Coding with Maximum Diversity Gains Over Frequency-Selective Fading Channels, IEEE Signal Processing Letters, vol. 8, No. 10, Oct. 2001, pp. 269-272.

Zhiqiang Liu, Yan Xin & Georios B. Giannakis, Space-Time-Frequency Coded OFDM Over Frequency-Selective Fading Channels, IEEE Transactions on Signal Processing, vol. 50, No. 10, Oct. 2002, pp. 2465-2476.

Hector Salgado, Marvin Sirbu & Jon Peha, Spectrum Sharing Through Dynamic Channel Assignment for Open Access to Personal Communications Services, Proc. of IEEE Intl. COmmunications Comference (ICC), Jun. 1995, pp. 417-422.

Helmut Bolcskei, Moritz Borgmann & Arogyaswami J. Paulraj, Space-Frequency Coded MIMO-OFDM with Variable Multiplexing-Diversity Tradeoff, IEEE 2003, pp. 2837-2841.

Timothy X Brown, Dynamic Channel Assignment in Shotgun Cellular Systems, Proc. of 1999 IEEE Radio and Wireless Conference (RAWCON99), pp. 147-150.

Chi Chung Lam, An Efficient Distributed Channel Allocation Algorithm Based on Dynamic Channel Boundaries, IEEE 1996, pp. 236-243.

Azzedine Boukerche, Sungbum Hong & Tom Jacob, Distributed Dynamic Channel Allocation for Mobile Communication System, Proceedings of the Eighth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunications Systems, IEEE 2000.

Jonathan S. Blogh, Peter J. Cherriman & Lajos Hanzo, Dynamic Channel Allocation Techniques Using Adaptive Modulation and Adaptive Antennas, IEEE Journal on Selected Areas in Communication, vol. 19, No. 2, Feb. 2001, pp. 312-321.

Sergio Verdu, Wireless Bandwidth in the Making, Multiple Access for Broadband Wireless Networks, IEEE Communications Magazine, Jul. 2000, pp. 53-58.

Jeff Foerster et al., Intel CFP Presentation for a UWB PHY, IEEE P802.15 Working Group for Wireless Personal Area Networks, IEEE 802.15-03/109, Mar. 3, 2003, 53 pages.

Joy Kelly et al., Multi-band OFDM Physical Layer Proposal Update, IEEE P802.15 Working Group for Wireless Personal Area Networks, IEEE 802.15-04/0122r4, Mar. 5, 2004, 26 pages.

David G. Leeper, Wireless Data Blaster, Scientific American, May 2002, pp. 64-69.

Bruno Pattan, A Brief Exposure to Ultra-Wideband Signaling, Microwace Journal, Dec. 2003, 5 pages.

Zhong Ye, Gary J. Saulnier, Kenneth S. Vastola and Michael J. Medley, Throuput Analysis for a Packet Radio Network Using Rate Adaptive OFDM Signaling, IEEE International Conference on Communications, Jun. 1999, 7 pages.

Patrick Vandenameele, Liesbet Van Der Perre, Marc Engels, Bert Gyselinckx and Hugo De Man, A Combined OFDM/SDMA Approach, IEEE Journal on Selected Areas in Communications, Nov. 2000, 27 pages.

Justin Chuang and Nelson Sollenberger, Beyond 3G: Wideband Wireless Data Access Based on OFDM and Dynamic Packet Assignment, Multiple Access for Broadband Wireless Networks, IEEE Communications Magazine, Jul. 2000, pp. 78-87.

Ove Edfords, Magnus Sandell, Jan-Jaap Van De Beek, Daniel Lamdstrom and Frank Sjoberg, An Introduction to orthogonal frequency-division multiplexing, Sep. 1996, 50 pages.

Rick S. Blum, Ye Li, Jack H. Winters and Qing Yan, Improved Space—Time Coding for MIMO-OFDM Wireless Communications, IEEE Transactions on Communications, vol. 49, No. 11, Nov. 2001, pp. 1873-1878.

Cheong Yui Wong, Roger S. Cheng, Khaled Ben Letaief and Ross D. Murch, Multiuser OFDM with Adaptive Subcarrier, Bit, and Power Allocation, IEEE Journal on Selected Areas in Communications, vol. 17, No. 10, Oct. 1999, pp. 1747-1758.
John A C. Bingham, Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come, IEEE Communications Magazine, May 1990, pp. 5-14.
Jinho Choi, Seong Rag Kim & Cheng-Chew Lim, Receivers With Chip-Level Decision Feedback Equalizer for CDMA Downlink Channels, IEEE Transactions on Wireless Communications, vol. 3, No. 1, Jan. 2004, pp. 300-314.
Masaaki Fujii & Takeshi Toda, Implementation of Space-Time Equalizer Using Multiple Single-Constrained SMI Array Processors and MLSE, IEEE Transactions on Wireless Communications, vol. 1, No. 2, Apr. 2002, pp. 333-341.
Robert Schober & Wolfgang H Gerstacker, Adaptive Noncoherent DFE for MDPSK Signals Transmitted over ISI Channels, IEEE Transactions on Communications, vol. 48, No. 7, Jul. 2000, pp. 1128-1140.
Michael Reuter, Jeffery C Allen, James R Zeidler & Richard C North, Mitigating Error Propagation Effects in a Decision Feedback Equalizer, IEEE Transactions on Communications, vol. 49, No. 11, Nov. 2001, pp. 2028-2041.
Christian Drewes, Ralph Hasholzner & Joachim S Hammerschmidt, Adaptive Equalization for Wireless ATM, 13th International Conference on Digital Signaling Processing (DSP '97), Santorini, Greec, Jul. 1997, 4 pages.
Marco Chiani, Introducing Erasures in Decision-Feedback Equalization to Reduce Error Propagation, IEEE Transactions on Communications, vol. 45, No. 7, Jul. 1997, pp. 757-760.
Turgay Celik, A New Method to Reduce Error Propagation In Decision Feedback Equalizer, IJCI Proceedings of International Conference on Signal Processing, ISSN 1304-2386, vol. 1, No. 2, Sep. 2003, pp. 213-216.
Roberto Lopez-Valcarce, Realizable Linear and Decision Feedback Equalizers: Properties and Connections, IEEE Transactions on Signal Processing vol. 52, No. 3, Mar. 2004, pp. 757-773.
Ramon Schlagenhaufer, Equalizer Structures for Spread Spectrum Multiuser Systems, Calgary, Alberta, Apr. 2001, Background Section 1.1 pp. 2-6 & Chapter 4 pp. 134-179.
Jaiganesh Balakrishnan, Bidirectional Decision Feedback Equalization and MIMO Channel Training, A Dissertation presented to the faculty of Graduate School of Cornell University, Aug. 2002, 159 pages.
Robert G Gallager, Low-Density Parity-Check Codes, Cambridge, Massachussets, Jul. 1963, 90 pages.
Marjan Karkooti, Semi-Parallel Architectures for Real-Time LDPC Coding, A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree Master of Science, Rice University, Houston, Texas, May 2004, 78 pages.
Khiam-Boon Png, Xiaoming Peng & Francois Chin, Performance Studies of a Multi-band OFDM System Using a Simplified LDPC Code, 0-7803-8373-7/04 IEEE 2004, pp. 376-380, Singapore.
William E. Ryan, An Introduction to LDPC Codes, University of Arizona, Tucson, Arizona, Aug. 19, 2003, 23 pages.
Frederic Guilloud, Generic Architecture for LDPC Codes Decoding, Telecom Paris, Jul. 2004, 174 pages.
Ben Lu, Xiaodong Wang & Krishna R. Narayanan, LDPC-Based Space-Time Coded OFDM Systems Over Correlated Fading Channels: Performance Analysis and Receiver Design, IEEE Transactions on Communications, vol. 50, No. 1, Jan. 2002, pp. 74-88.
Sachio Lida et al., IEEE 2005 Digest of Technical Papers, "A 3.1 to 5GHz CMOS DSSS UWB Transceiver for WPANs", vol. 48, Section 11.8, Feb. 6, 2005, pp. 214-215.
Alamouti, A Simple Transmit Diversity Technique for Wireless Communications, vol. 16, No. 8, Oct. 1998.
Chaudhury, Prodip et al. "The 3GPP Proposal for IMT-2000" IEEE Communications Magazine, Dec. 1999, pp. 72-81.
Cherubini, Giovanni "Hybrid TDMA/CDMA Based on Filtered Multitone Modulation for Uplink Transmission in HFC Networks" IEEE Communications Magazine, Oct. 2003, pp. 108-115.
Compton, Jr., "A New Algorithm to Optimize Barker Code Sidelobe SuppressionFilters', IEEE Log No. 36738, Dec. 13, 1989, pp. 673-677 and p. 921."
Baier, Paul Walter et al. "Taking the Challenge of Multiple Access for Third-Generation Cellular Mobile Radio Systems—A European View" IEEE Communications Magazine, Feb. 1996, pp. 82-89.
Cramer, et al., Impulse Radio Multipath Characteristics and Diversity Reception, IEEE 1998, 5 page.
CWTS WG1, Method and Principles of Uplink Synchronization, TSG-RAN Working Group 1 (Radio) meeting #5, Cheju, South Korea, Jun. 1-4, 1999, 6 pages.
Edfords, Ove et al. "An introduction to orthogonal frequency-division multiplexing" Sep. 1996, pp. 1-50.
Ertel, et al., Overview of Spatial Channel Models for Antenna Array Communication Systems, IEEE Personal Communications, Feb. 1998, pp. 10-22.
Fleming et al., CMOS Ultra-Wideband Localizers for Networking in the Extreme, Sep. 10, 2001, 28 pages.
Haas, Harald "A Dynamic Channel Assignment Algorithm for a Hybrid TDMA/CDMA—TDD Interface Using the Novel TS-Operating Technique" IEEE Journal on Selected Areas in Communications, vol. 19, No. 10, Oct. 2001, pp. 1831-1846.
Krishnamachari et al., Ultra High Speed Digital Processing for Wireless Systems using Passive Microwave Logic, The Cooper Union for the Advancement of Science and Art, RAWCON'98 Proceedings, 1998 IEEE, New York, New York, 4 pages.
Le Martret , et al., All-Digital Impulse Radio with Multiuser Detection for Wireless Cellular Systems, IEEE Transactions on Communications, vol. 50, No. 9, Sep. 2002, pp. 1440-1450.
Lu, et al., A Practical Implementation of Broadband Multiple Antenna Systems, 0-7803-4788-9/98 IEEE 1998, 5 pages.
O'Donnell, et al., An Integrated, Low-Power, Ultra-Wideband Transceiver Architecture for Low-Rate Indoor Wireless Systems, IEEE CAS Workshop on Wireless Communications and Networking, Sep. 4-5, 2002, 23 pages, Pasadena.
Olson, Examining and Modifying Marker Generator Output, Jackson Harbor Press, Oct. 26, 1999, 5 pages.
Prasad , et al., An Overview of CDMA Evolution Toward Wideband CDMA, IEEE Communications Survey, Fourth Quarter 1998, vol. 1 No. 1, 28 pages.
Richardson, K. W. "UMTS Overview" , Electronics & Communication Engineering Journal, Jun. 2000, pp. 93-101.
Rihacczek et al., "Range Sidelobe Suppression for Barker Codes" IEEE Transactions on Aerospace and Electronic Systems, vol. 7, No. 6, Nov. 1971, pp. 1087-1092.
Rohling et al., "Mismatched-Filter Design for Periodical Binary Phased Signals", IEEE Transactions on Aerospace and Electronic Systedms, vol. 25, No. 6, Nov. 1989, pp. 890-896. C. Cantoni, R.T. Winters, On the Capacity of Radio Communication Systems with Diversity in a Rayleigh Fading Environment, IEEE Journal on Selected Areas in Communication, vol. SAC-5, No. 5, Jun. 1987, pp. 871-878.
Winters, Optimum Combining for Indoor Radio Systems with Multiple Users, IEEE Transactions on Communications, vol. COM-35, No. 11, Nov. 1987, pp. 1222-1230.
Winters, Optimum Combining in Digital Mobile Radio with Cochannel Interference, IEEE Journal on Selected Areas in Communications, vol. SAC-2, No. 4, Jul. 1984, pp. 528-539.
Winters, Smart Antennas for Wireless Systems, IEEE Personal Communications, Feb. 1998, pp. 23-27.
Wornell et al., Efficient Signal Processing Techniques for Exploiting Transmit Antenna Diversity on Fading Channels, vol. 45, No. 1, Jan. 1997.
US 5,987,079, 11/1999, Scott et al. (withdrawn)

* cited by examiner

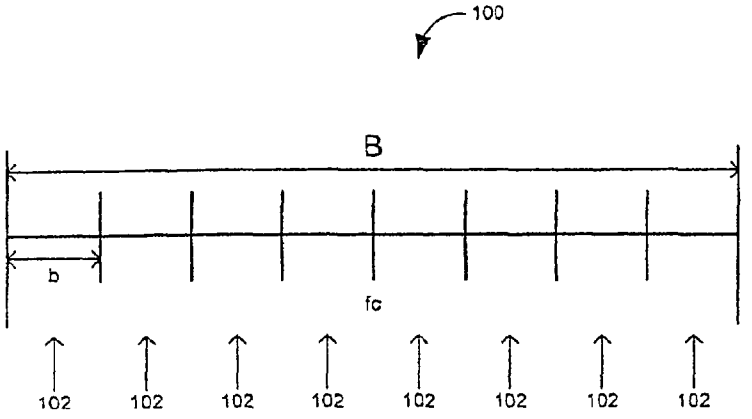


Figure 1A

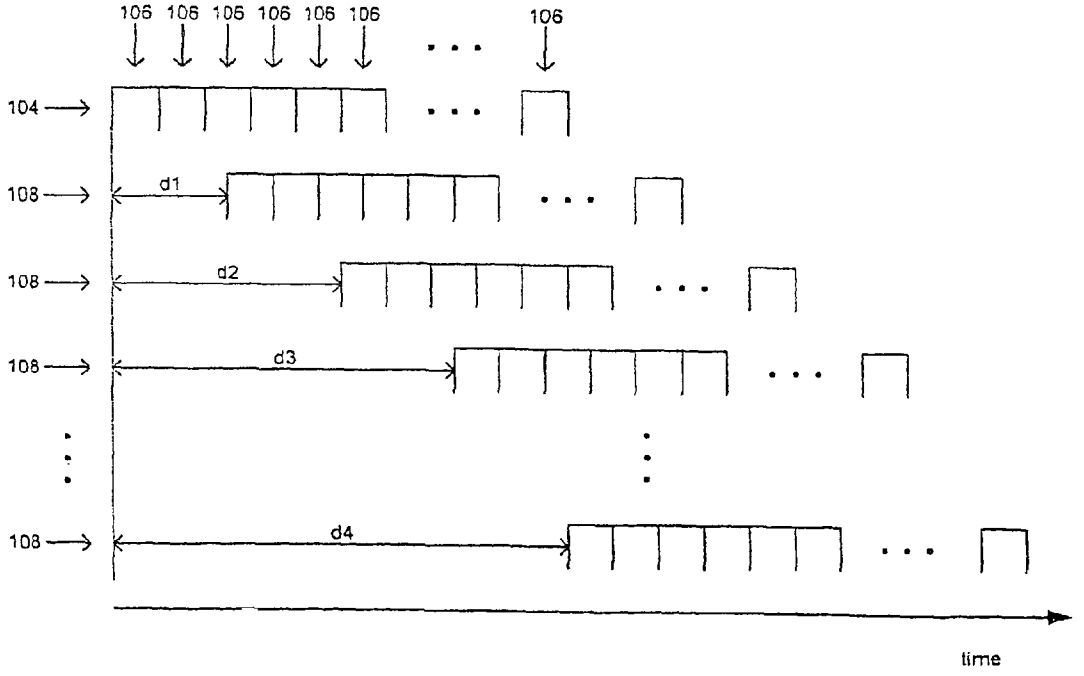


Figure 1B

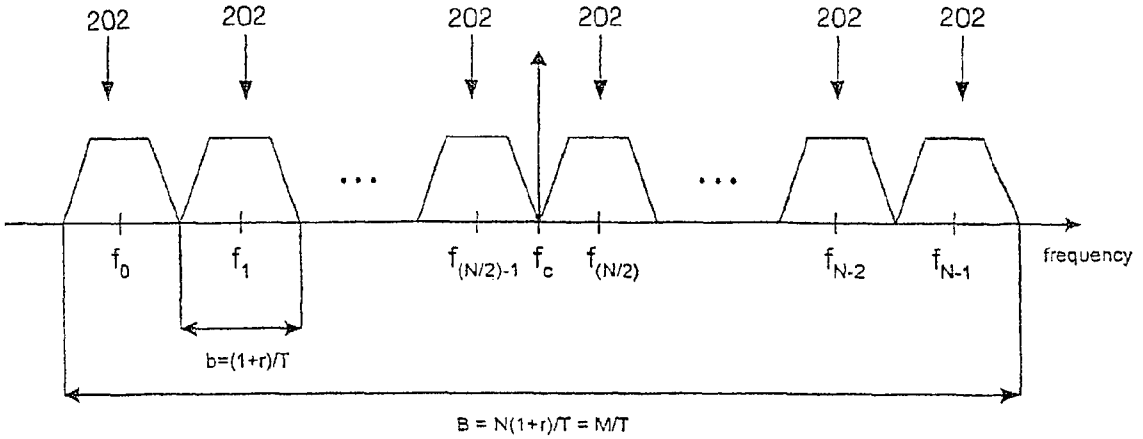

202
202
202
202
202
202
f0
f1
f(N/2)-1
fc
f(N/2)
fN-2
fN-1
frequency
b=(1+r)/T
B = N(1+r)/T = M/T

Figure 2

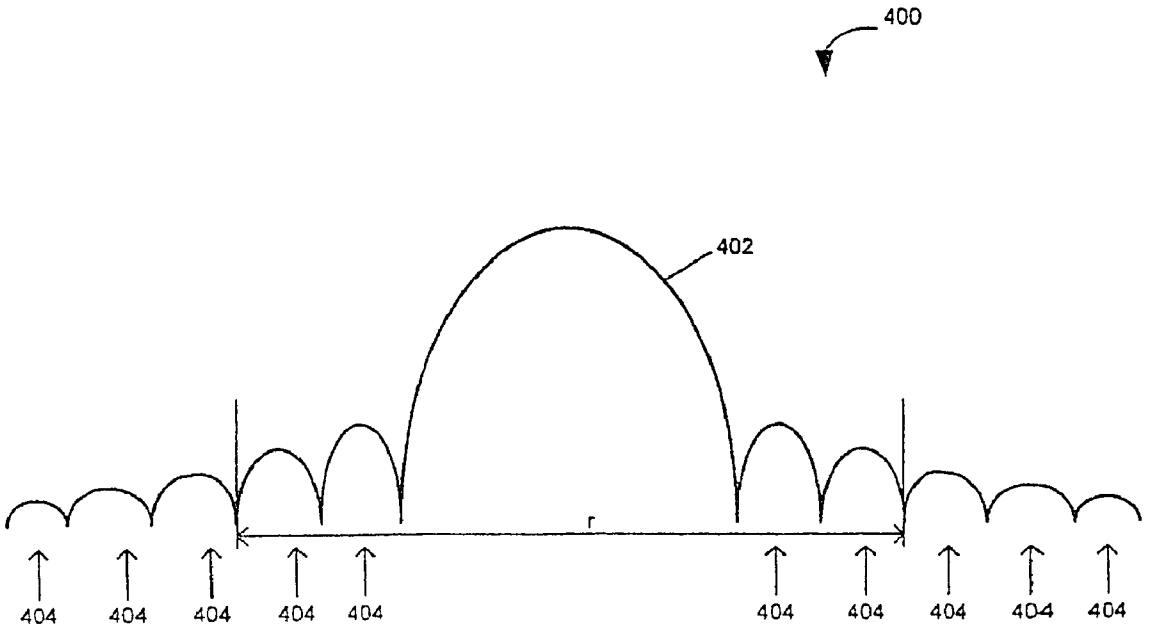


Figure 3

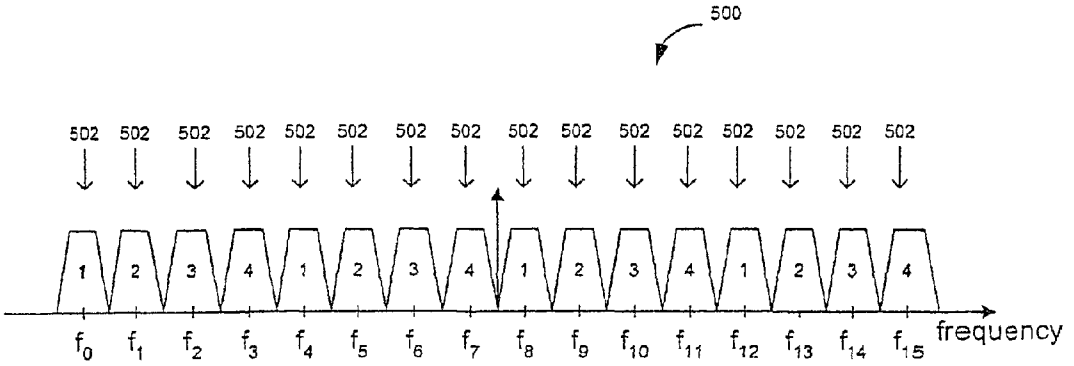


Figure 4A

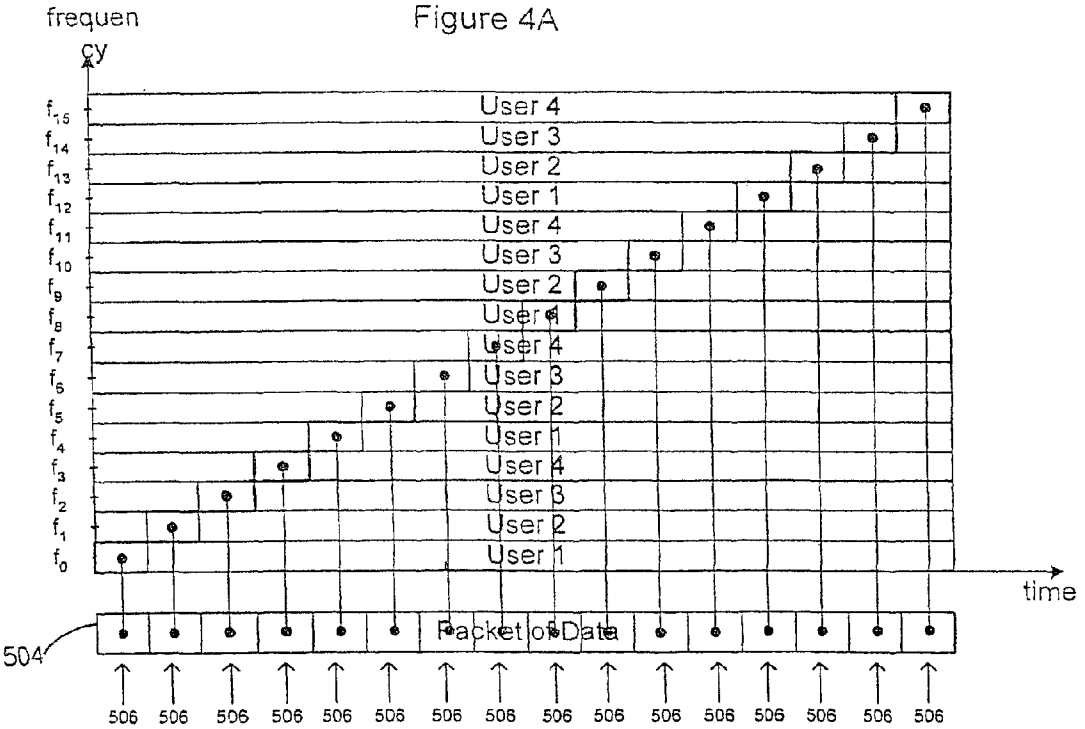


Figure 4B

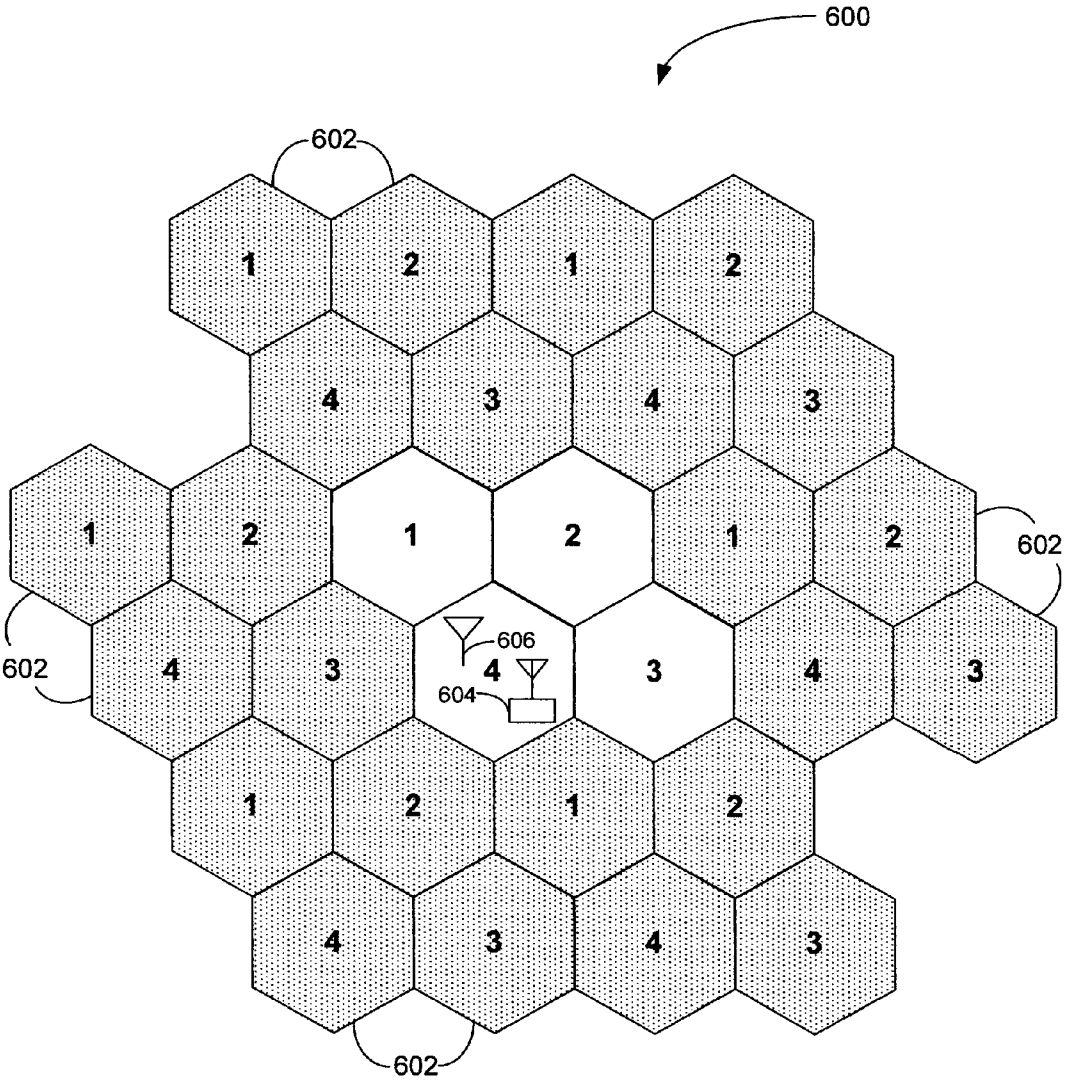


FIG. 5

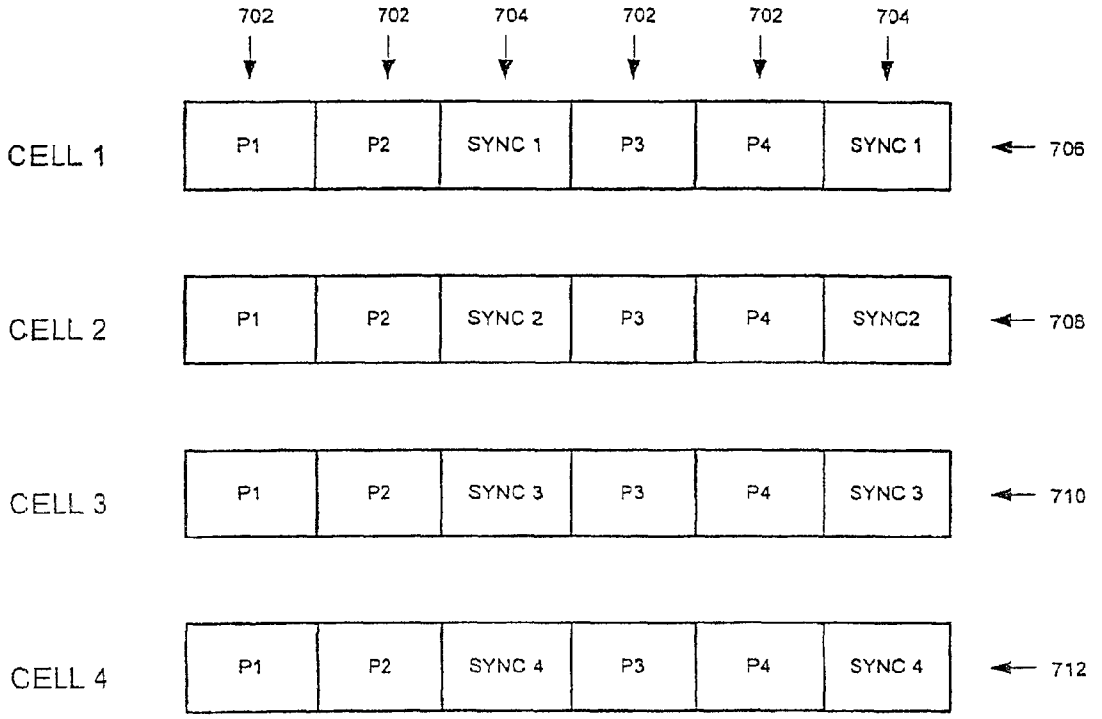

702
702
704
702
702
704
CELL 1
P1
P2
SYNC 1
P3
P4
SYNC 1
706
CELL 2
P1
P2
SYNC 2
P3
P4
SYNC2
708
CELL 3
P1
P2
SYNC 3
P3
P4
SYNC 3
710
CELL 4
P1
P2
SYNC 4
P3
P4
SYNC 4
712

Figure 6

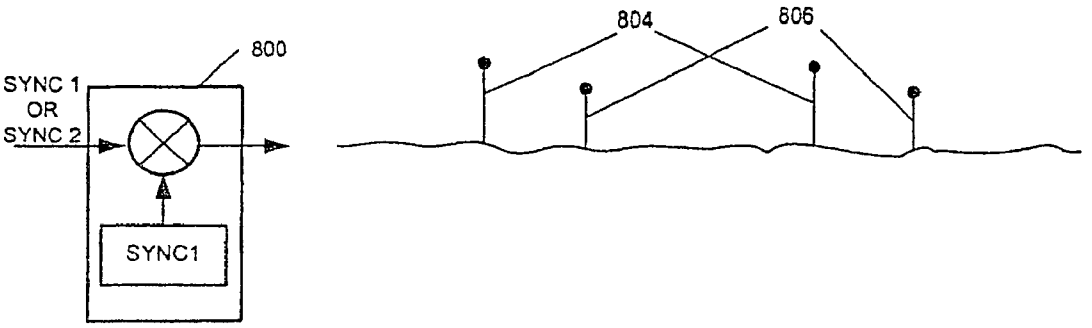


Figure 7

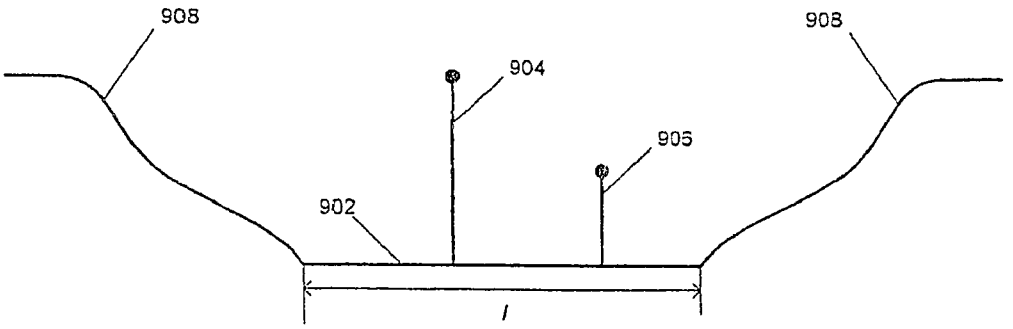


Figure 8

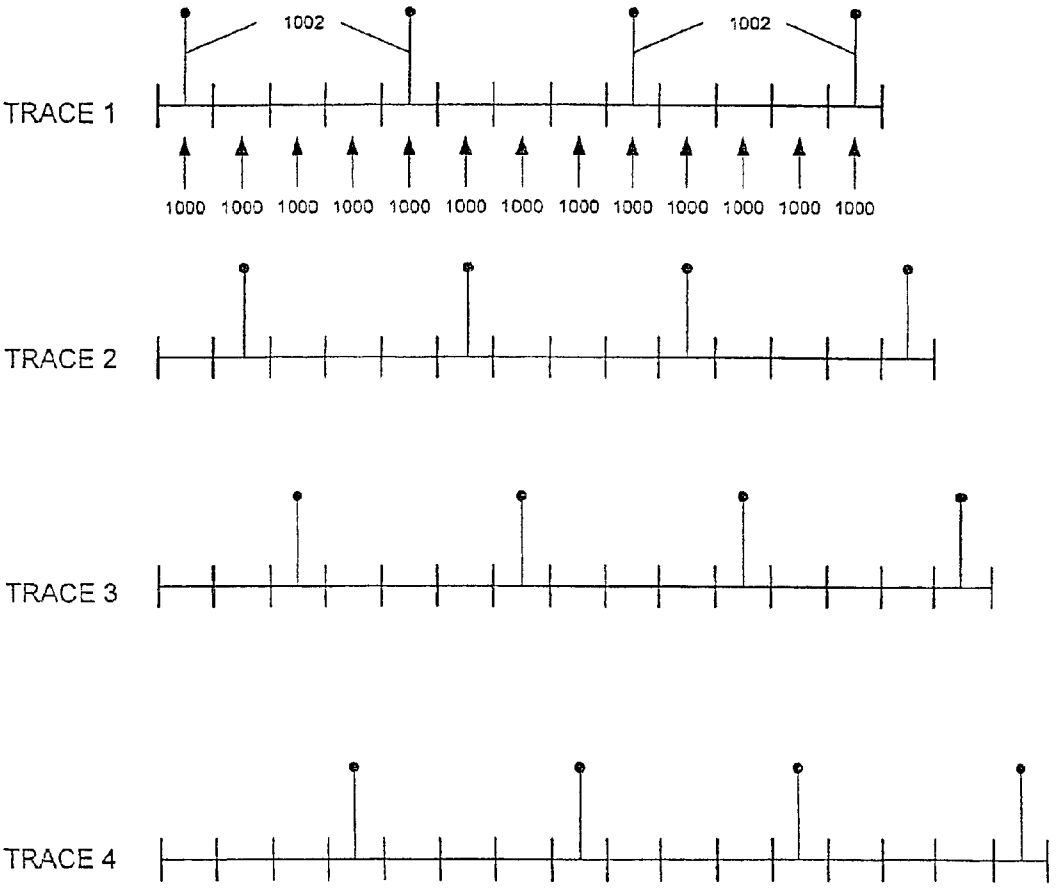

1002
1002
TRACE 1
1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000
TRACE 2
TRACE 3
TRACE 4

Figure 9

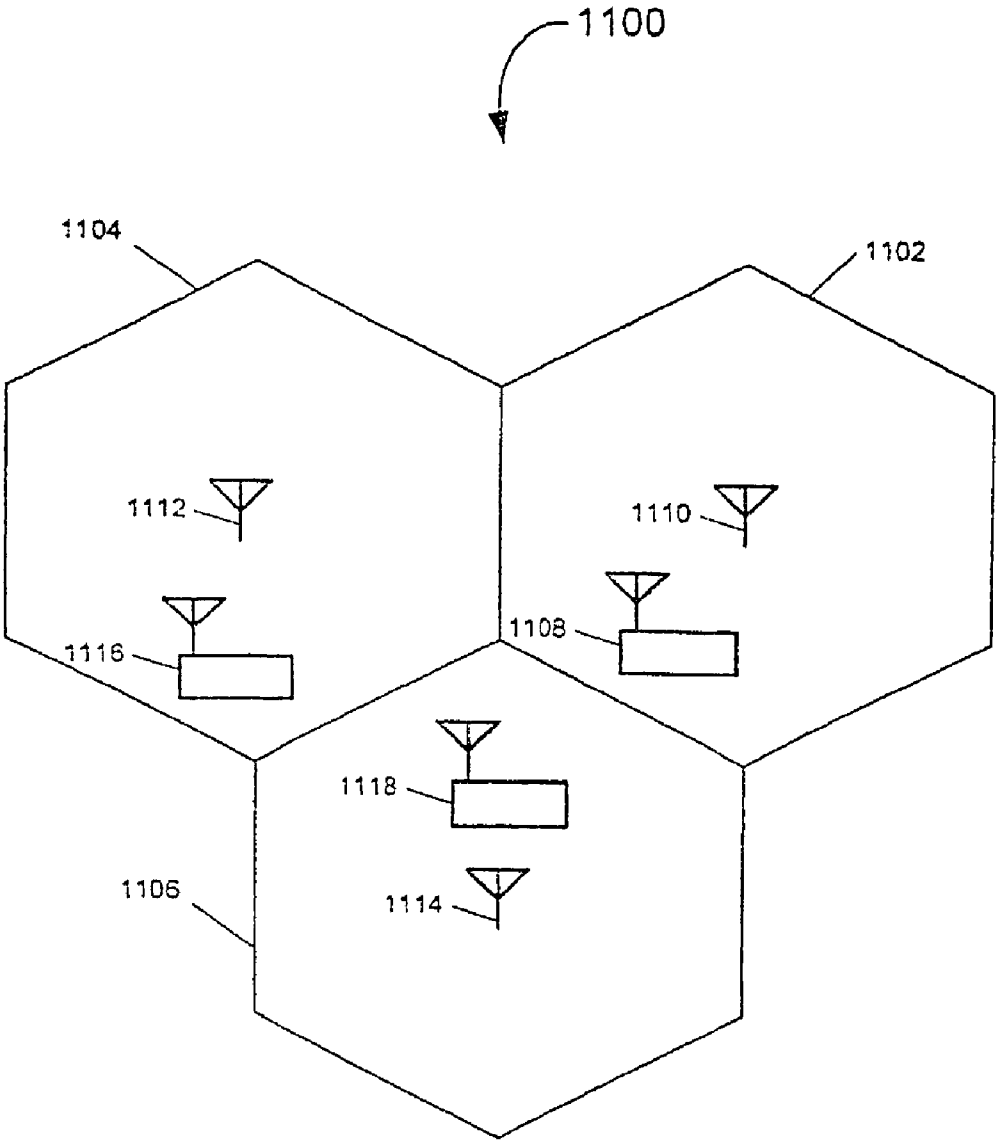


Figure 10

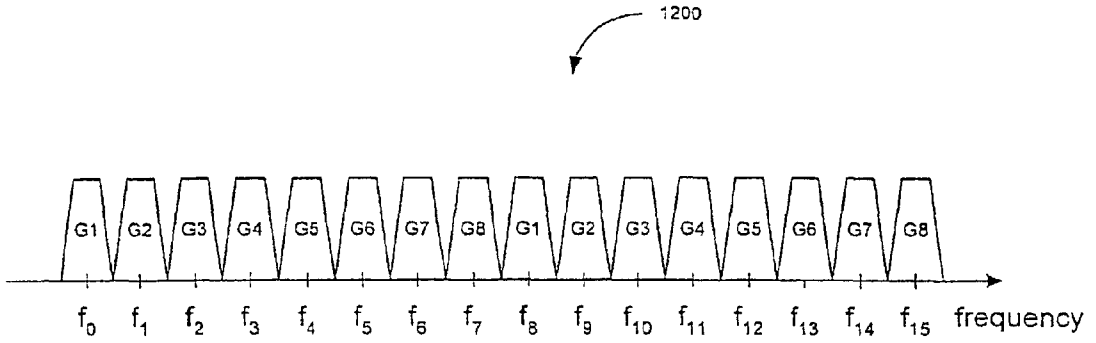


Figure 11A

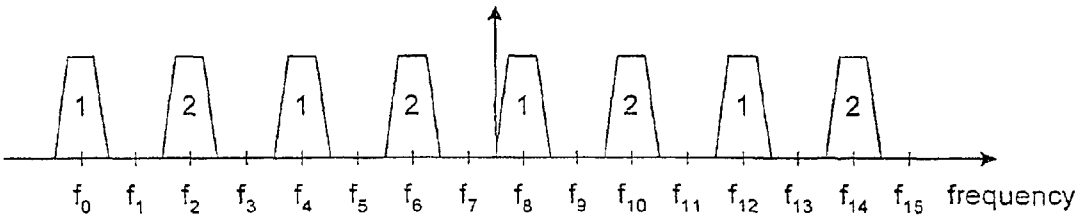


Figure 11B

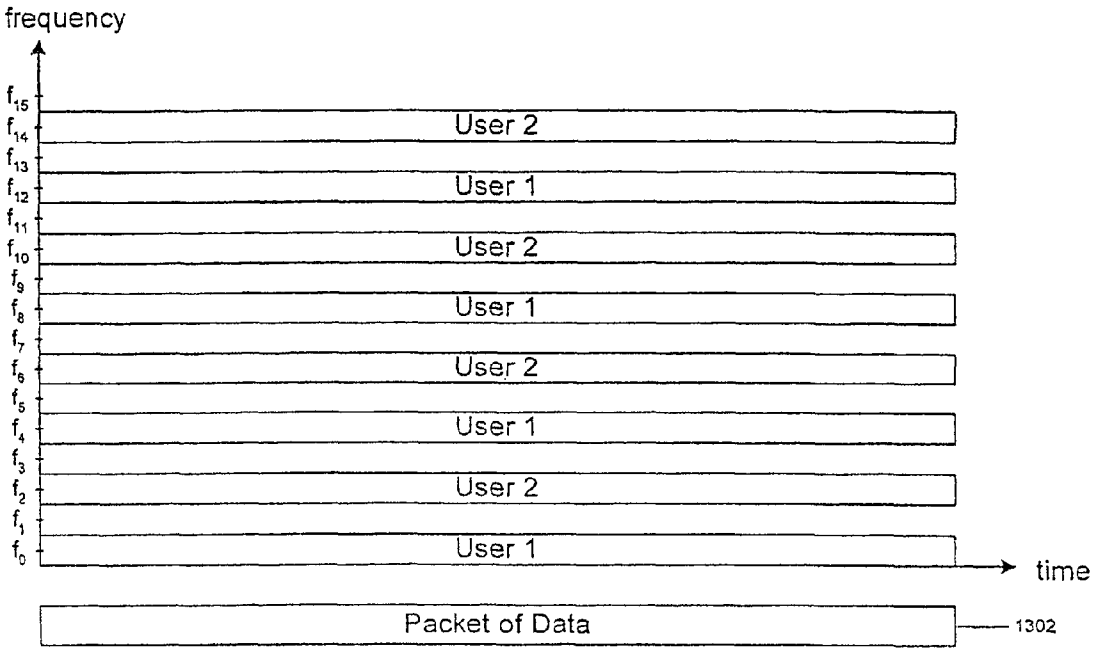

frequency
f_{15}
f_{14}
f_{13}
f_{12}
f_{11}
f_{10}
f_9
f_8
f_7
f_6
f_5
f_4
f_3
f_2
f_1
f_0
User 2
User 1
User 2
User 1
User 2
User 1
User 2
User 1
time
Packet of Data
1302

Figure 12

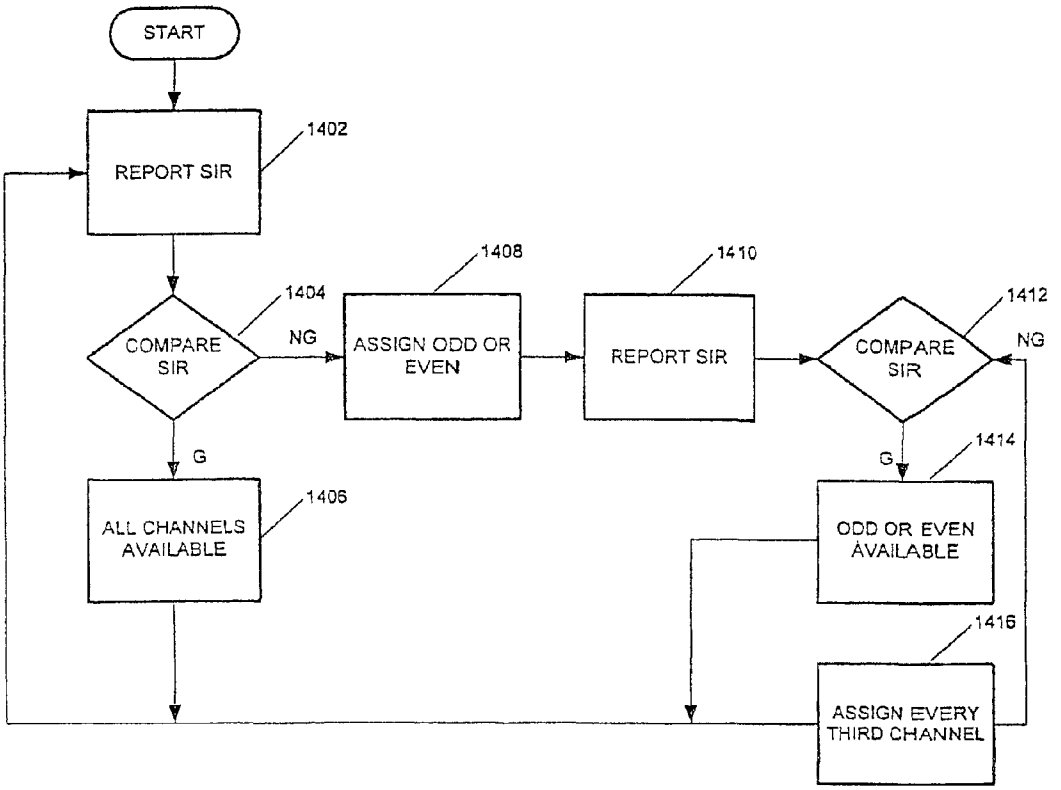

START
REPORT SIR
1402
COMPARE SIR
1404
NG
G
ASSIGN ODD OR EVEN
1408
REPORT SIR
1410
COMPARE SIR
1412
NG
G
1414
ALL CHANNELS AVAILABLE
1406
ODD OR EVEN AVAILABLE
1416
ASSIGN EVERY THIRD CHANNEL

Figure 13

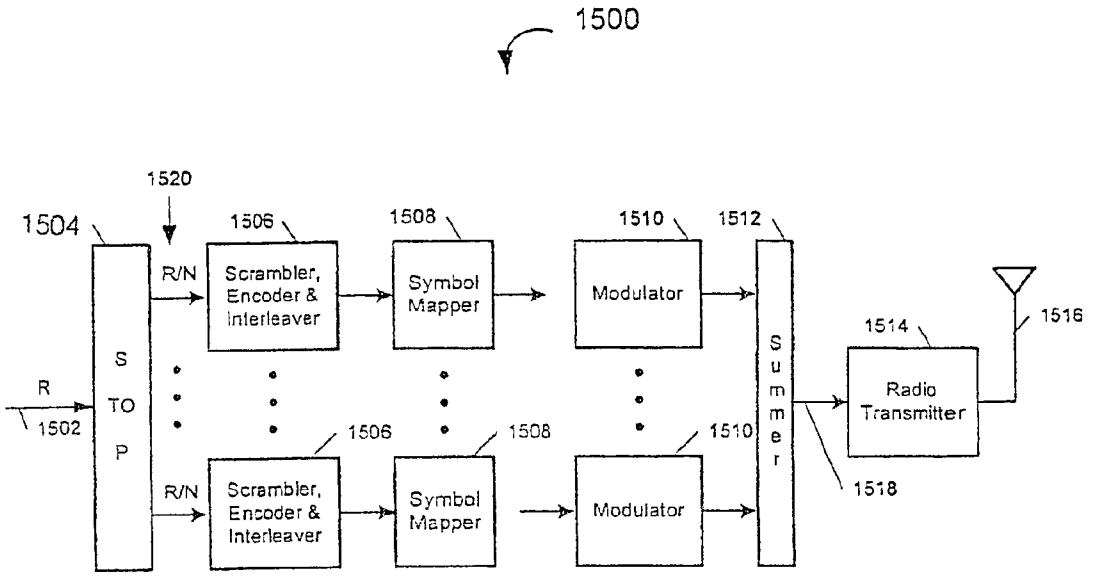

1500
1520
1504
1506
1508
1510
1512
R/N
Scrambler, Encoder & Interleaver
Symbol Mapper
Modulator
S TO P
Summer
1514
Radio Transmitter
1516
R
1502
1506
1508
1510
R/N
Scrambler, Encoder & Interleaver
Symbol Mapper
Modulator
1518

Figure 14

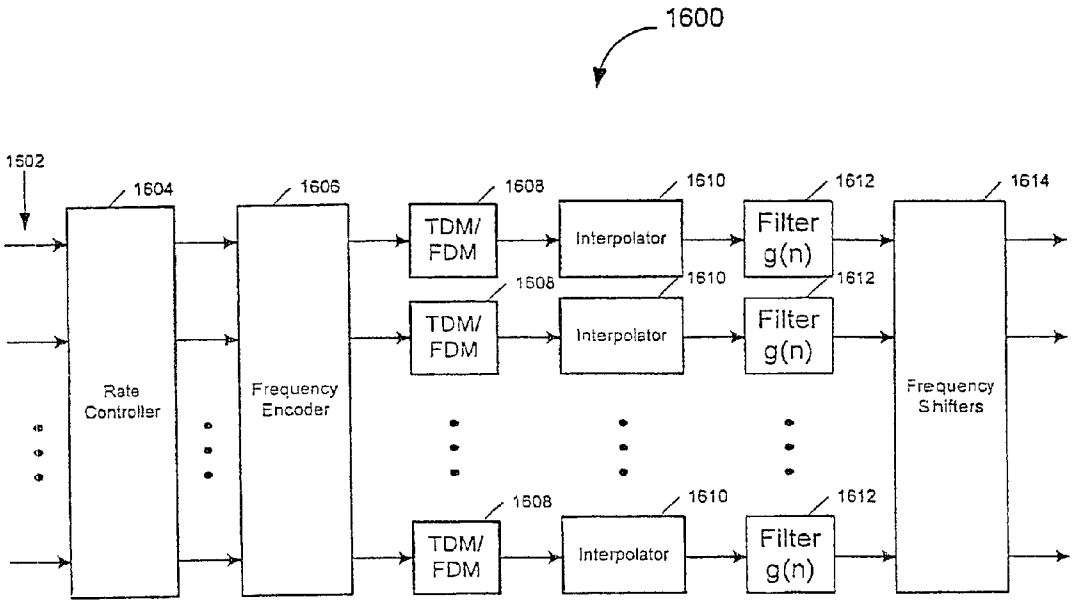

1600
1602
1604
Rate Controller
1606
Frequency Encoder
1608
TDM/ FDM
1610
Interpolator
1612
Filter g(n)
1614
Frequency Shifters

Figure 15

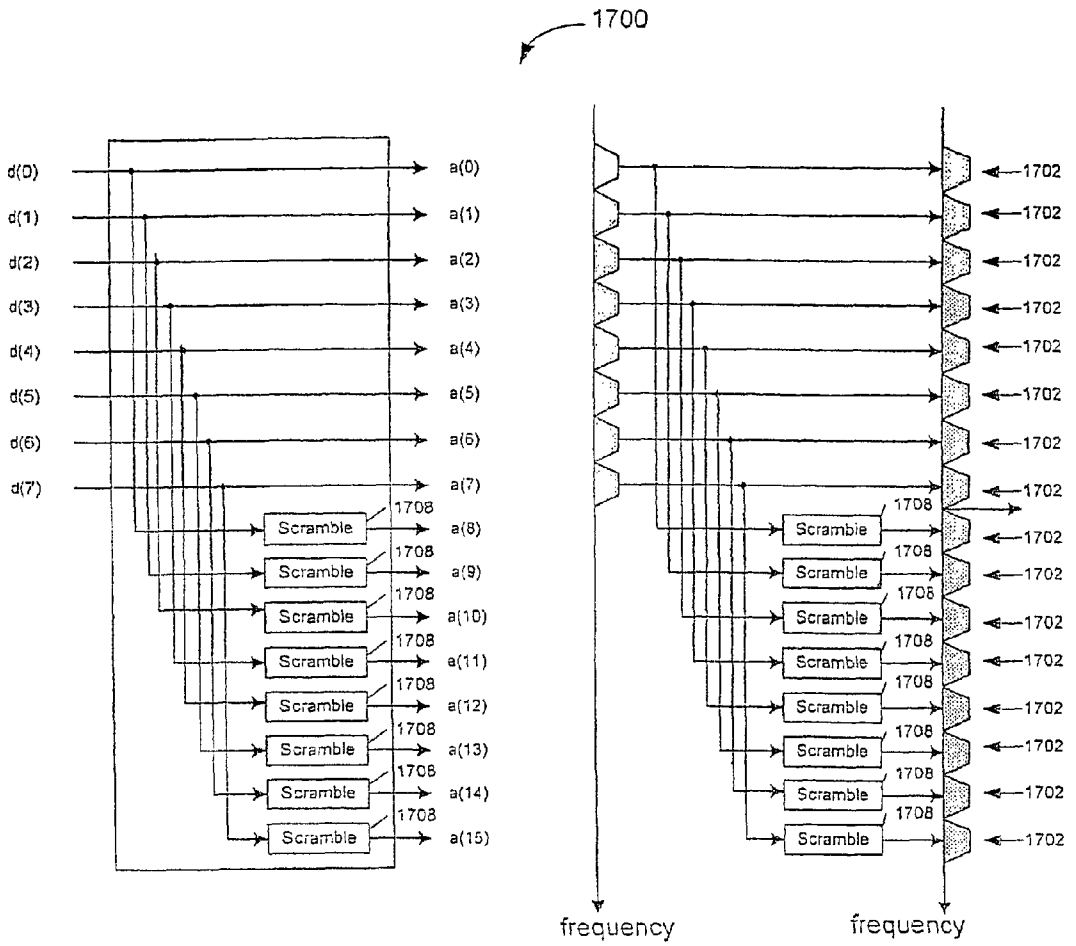

1700
d(0)
d(1)
d(2)
d(3)
d(4)
d(5)
d(6)
d(7)
a(0)
a(1)
a(2)
a(3)
a(4)
a(5)
a(6)
a(7)
a(8)
a(9)
a(10)
a(11)
a(12)
a(13)
a(14)
a(15)
Scramble
1708
1702
frequency
frequency

Figure 16

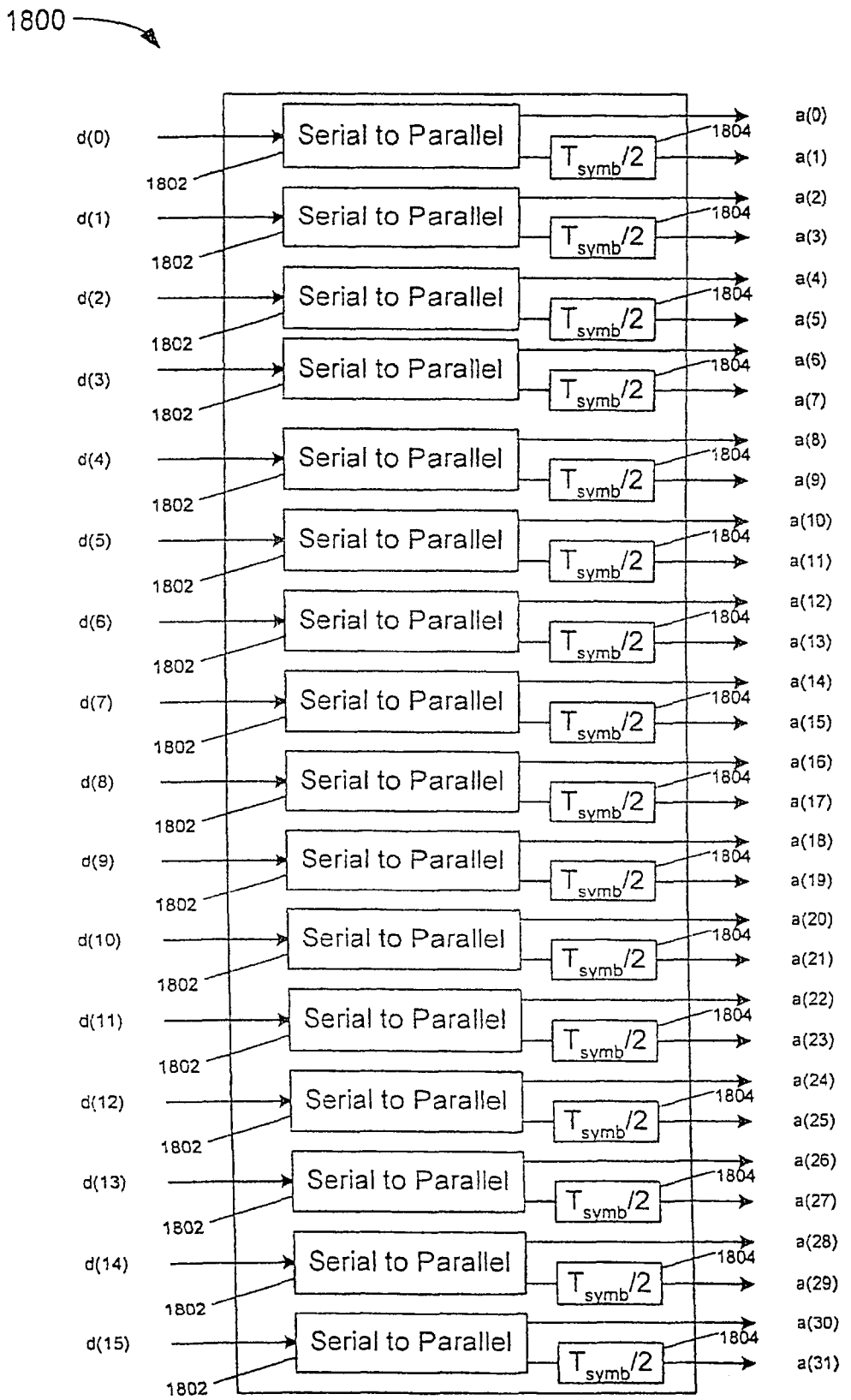

1800
d(0)
d(1)
d(2)
d(3)
d(4)
d(5)
d(6)
d(7)
d(8)
d(9)
d(10)
d(11)
d(12)
d(13)
d(14)
d(15)
1802
Serial to Parallel
$T_{symb}/2$
1804
a(0)
a(1)
a(2)
a(3)
a(4)
a(5)
a(6)
a(7)
a(8)
a(9)
a(10)
a(11)
a(12)
a(13)
a(14)
a(15)
a(16)
a(17)
a(18)
a(19)
a(20)
a(21)
a(22)
a(23)
a(24)
a(25)
a(26)
a(27)
a(28)
a(29)
a(30)
a(31)

Figure 17

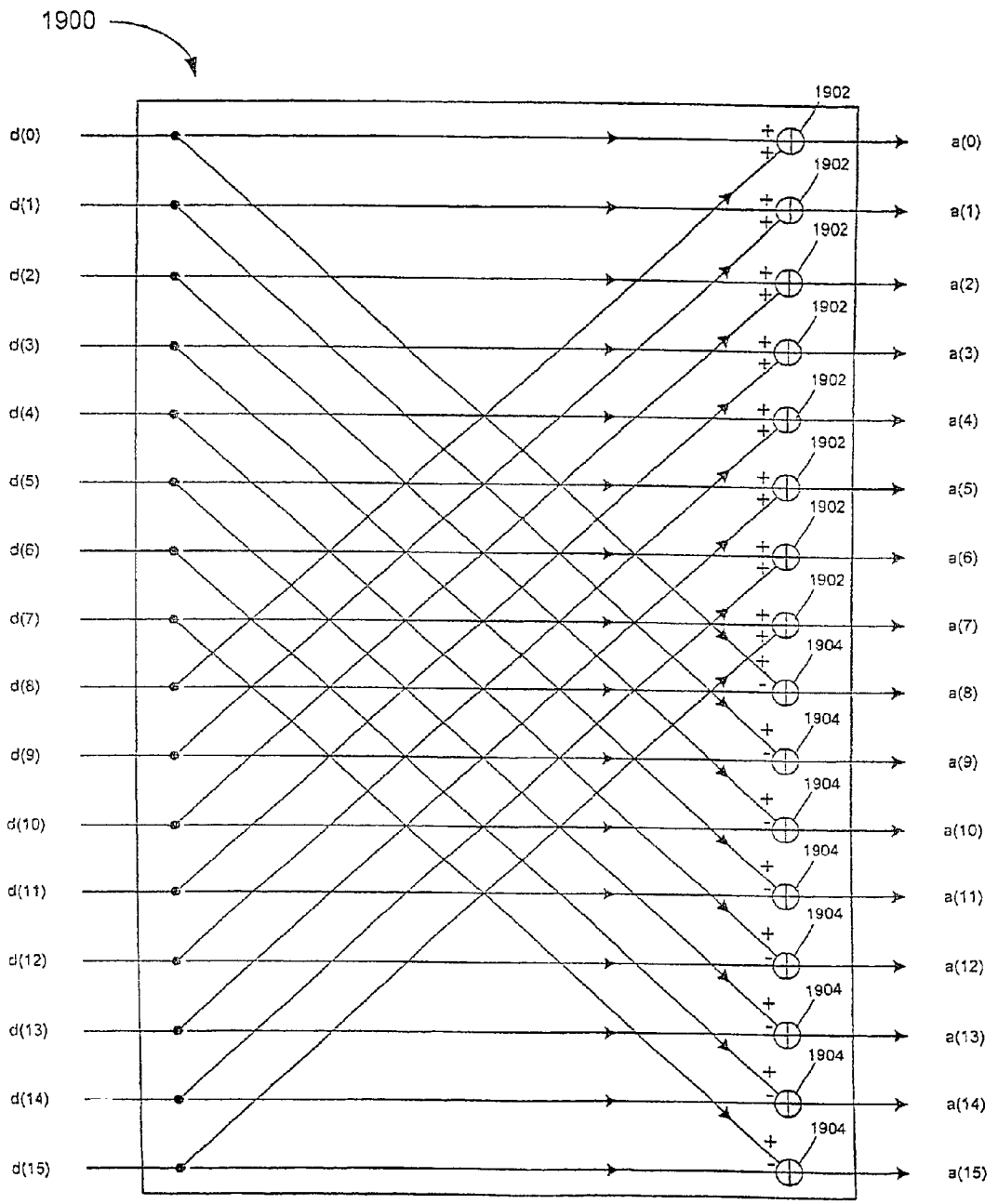

1900
d(0)
d(1)
d(2)
d(3)
d(4)
d(5)
d(6)
d(7)
d(8)
d(9)
d(10)
d(11)
d(12)
d(13)
d(14)
d(15)
1902
1904
a(0)
a(1)
a(2)
a(3)
a(4)
a(5)
a(6)
a(7)
a(8)
a(9)
a(10)
a(11)
a(12)
a(13)
a(14)
a(15)

Figure 18

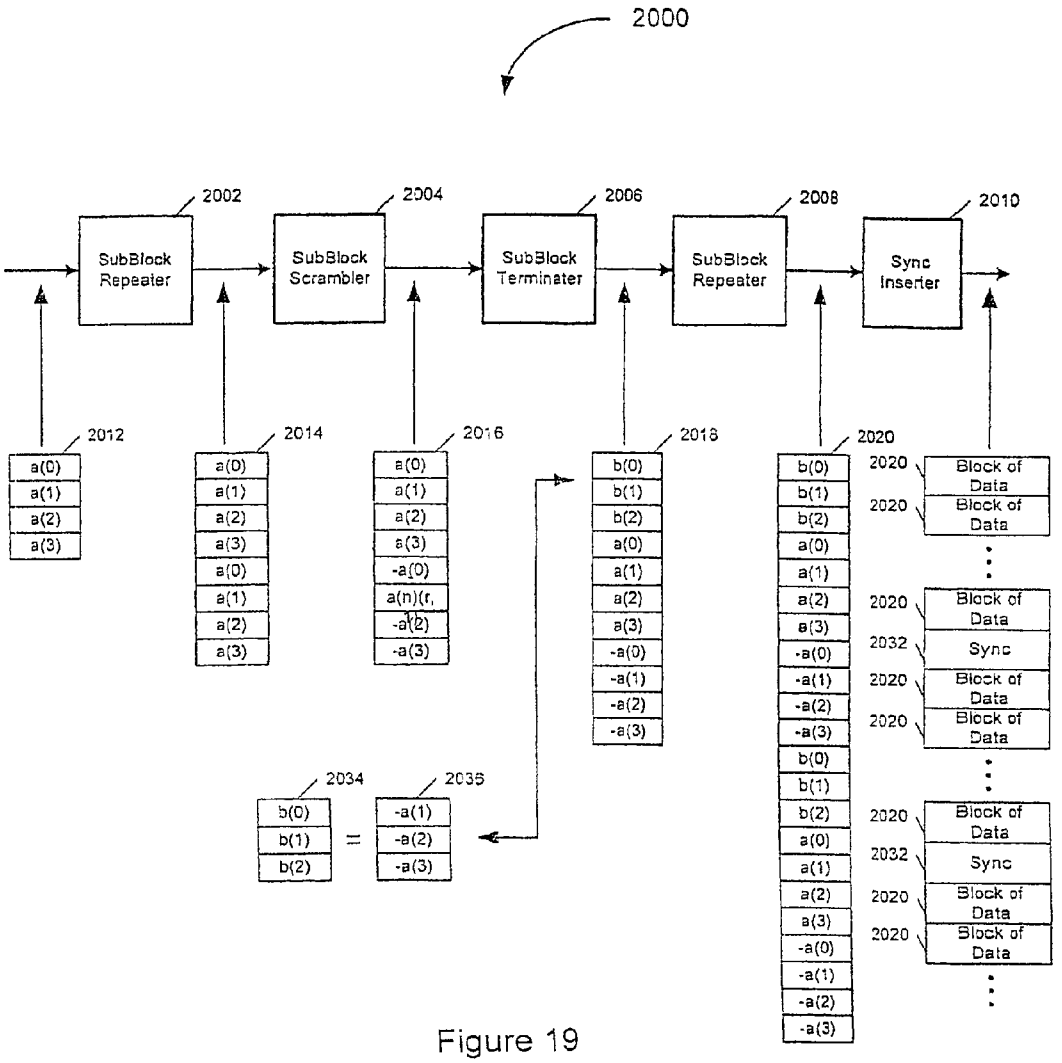

2000
2002
SubBlock Repeater
2004
SubBlock Scrambler
2006
SubBlock Terminater
2008
SubBlock Repeater
2010
Sync Inserter
2012
a(0)
a(1)
a(2)
a(3)
2014
a(0)
a(1)
a(2)
a(3)
a(0)
a(1)
a(2)
a(3)
2016
a(0)
a(1)
a(2)
a(3)
-a(0)
a(n)(r,
-a(2)
-a(3)
2018
b(0)
b(1)
b(2)
a(0)
a(1)
a(2)
a(3)
-a(0)
-a(1)
-a(2)
-a(3)
2020
b(0)
b(1)
b(2)
a(0)
a(1)
a(2)
a(3)
-a(0)
-a(1)
-a(2)
-a(3)
b(0)
b(1)
b(2)
a(0)
a(1)
a(2)
a(3)
-a(0)
-a(1)
-a(2)
-a(3)
2020 Block of Data
2020 Block of Data
2020 Block of Data
2032 Sync
2020 Block of Data
2020 Block of Data
2020 Block of Data
2032 Sync
2020 Block of Data
2020 Block of Data
2034
b(0)
b(1)
b(2)
=
2036
-a(1)
-a(2)
-a(3)

Figure 19

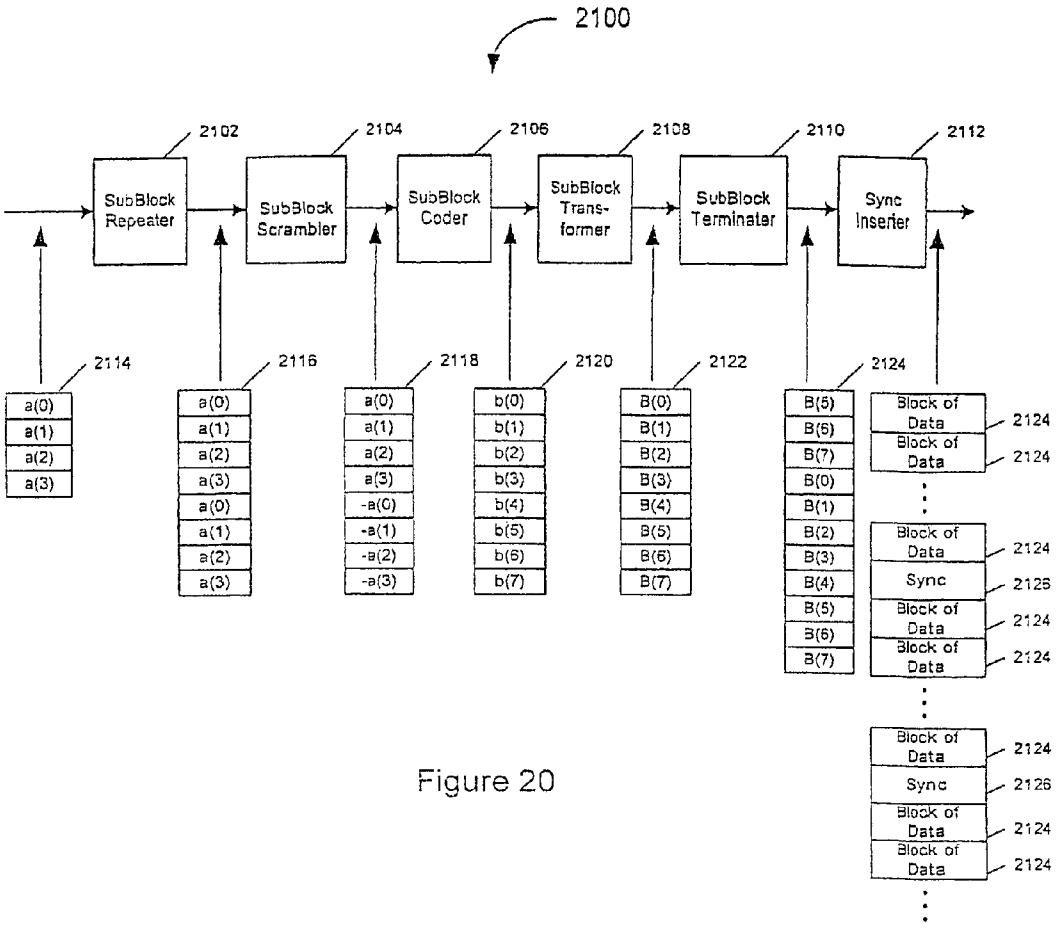

2100
2102
SubBlock Repeater
2104
SubBlock Scrambler
2106
SubBlock Coder
2108
SubBlock Trans-former
2110
SubBlock Terminator
2112
Sync Inserter
2114
a(0) a(1) a(2) a(3)
2116
a(0) a(1) a(2) a(3) a(0) a(1) a(2) a(3)
2118
a(0) a(1) a(2) a(3) -a(0) -a(1) -a(2) -a(3)
2120
b(0) b(1) b(2) b(3) b(4) b(5) b(6) b(7)
2122
B(0) B(1) B(2) B(3) B(4) B(5) B(6) B(7)
2124
B(5) B(6) B(7) B(0) B(1) B(2) B(3) B(4) B(5) B(6) B(7)
Block of Data
Sync
2126

Figure 20

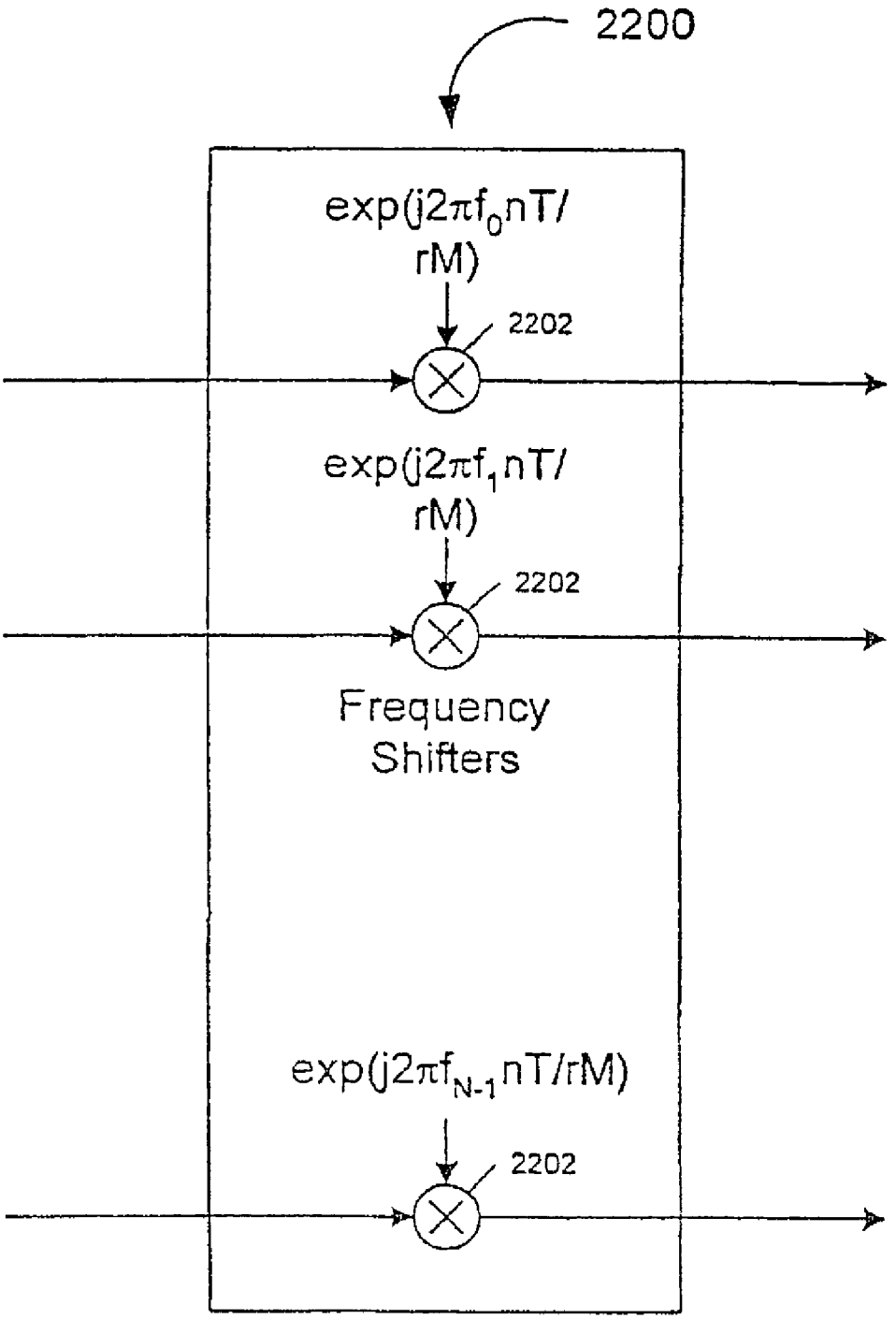

2200
exp(j2πf0nT/rM)
2202
exp(j2πf1nT/rM)
2202
Frequency Shifters
exp(j2πfN-1nT/rM)
2202

Figure 21

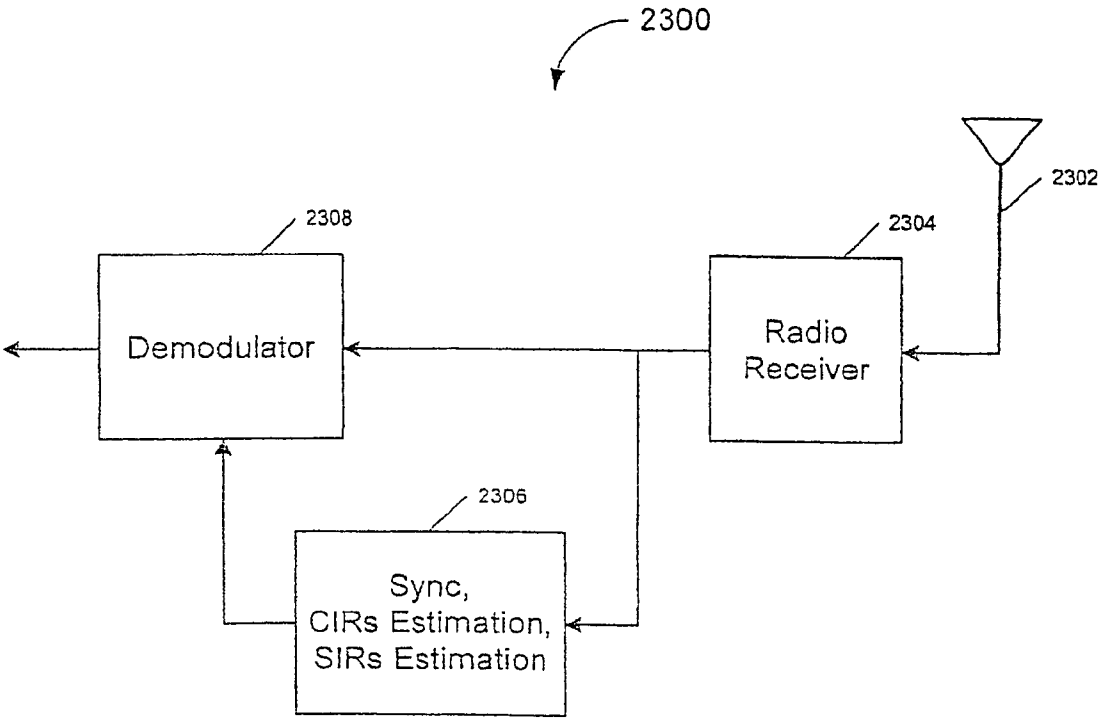

2300
2302
2308
2304
Demodulator
Radio
Receiver
2306
Sync,
CIRs Estimation,
SIRs Estimation

Figure 22

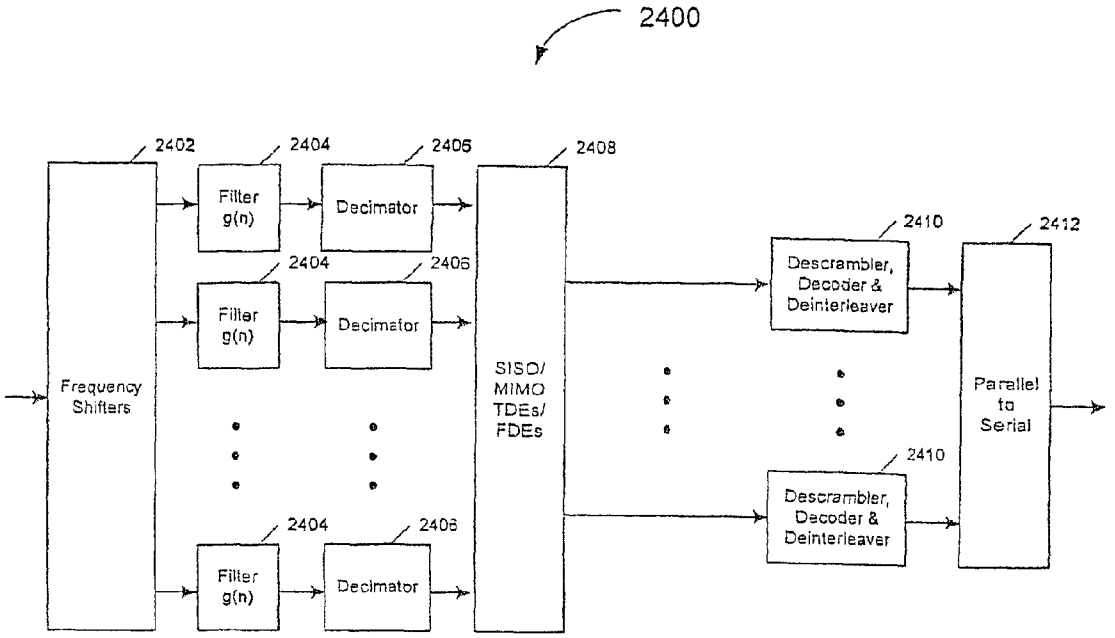

2400
2402
2404
2406
2408
2410
2412
Frequency Shifters
Filter g(n)
Decimator
SISO/ MIMO TDEs/ FDEs
Descrambler, Decoder & Deinterleaver
Parallel to Serial

Figure 23

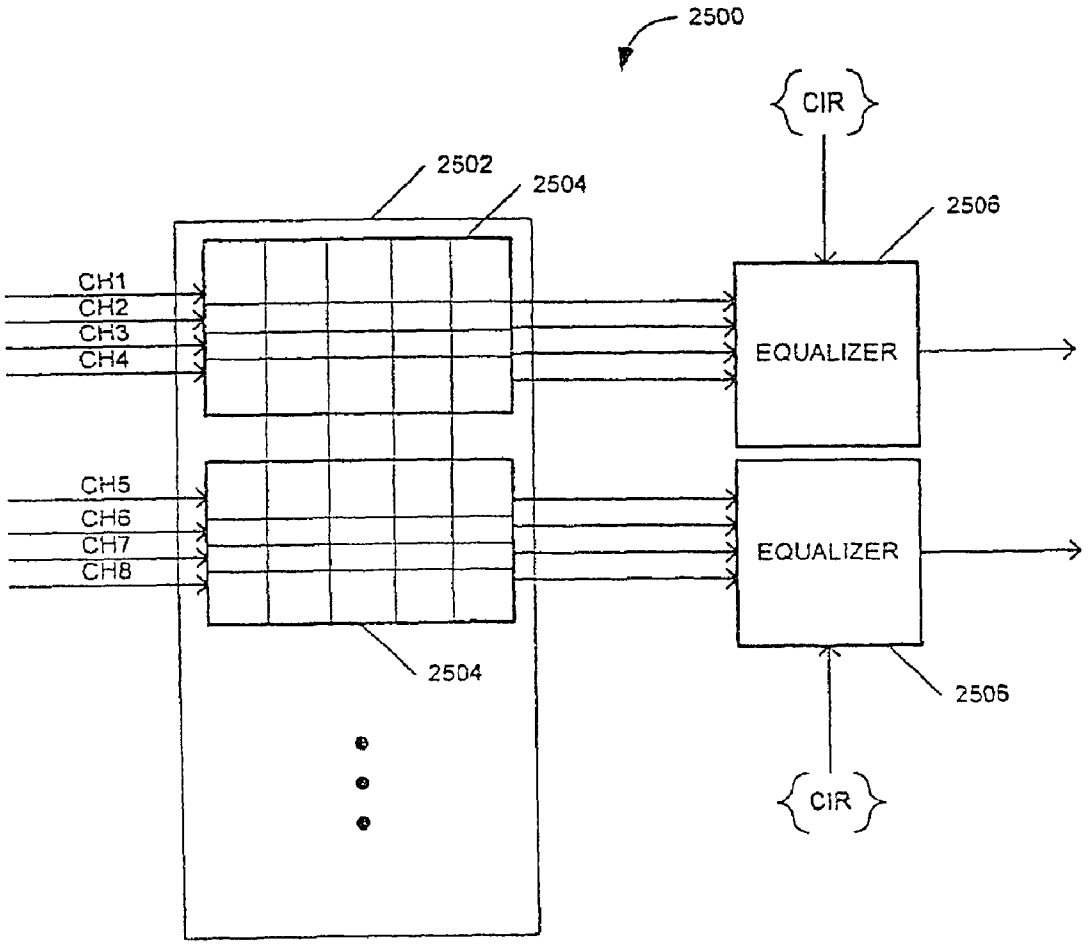

2500
CIR
2502
2504
2506
CH1
CH2
CH3
CH4
EQUALIZER
CH5
CH6
CH7
CH8
EQUALIZER
2504
2506
CIR

Figure 24

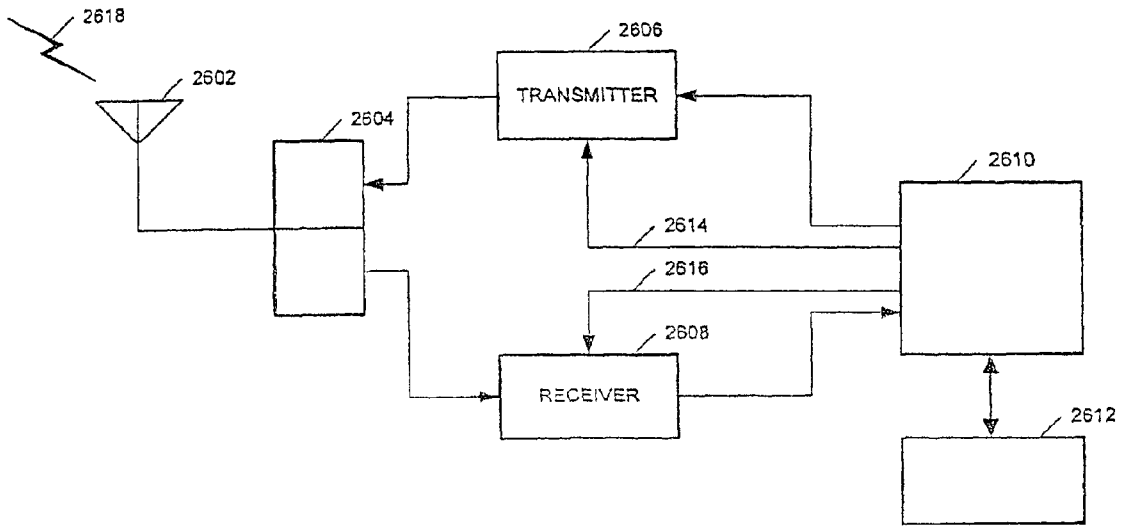

2618
2602
2604
2606
TRANSMITTER
2610
2614
2616
2608
RECEIVER
2612

Figure 25

PRIOR ART

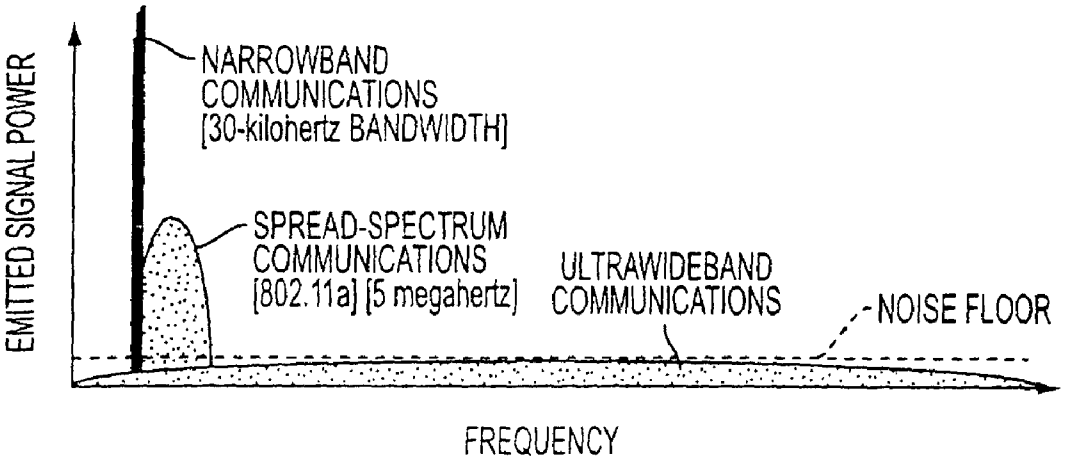


FIG. 26

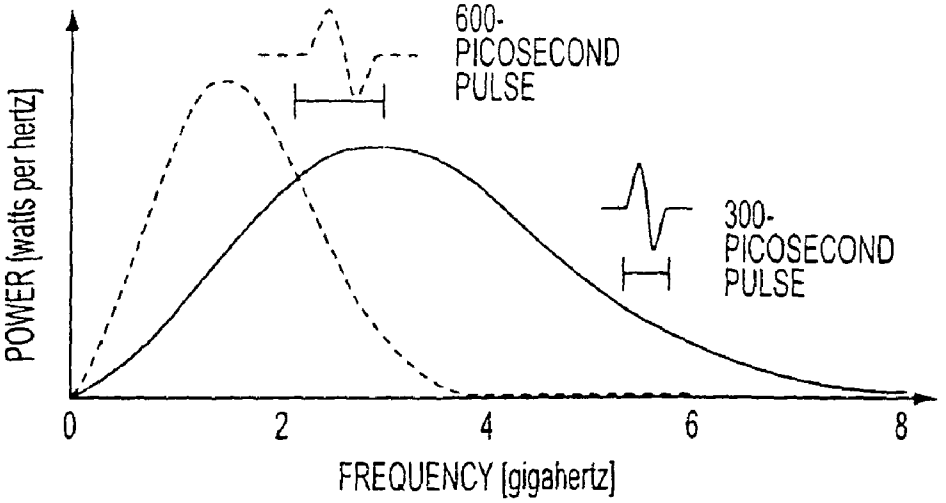
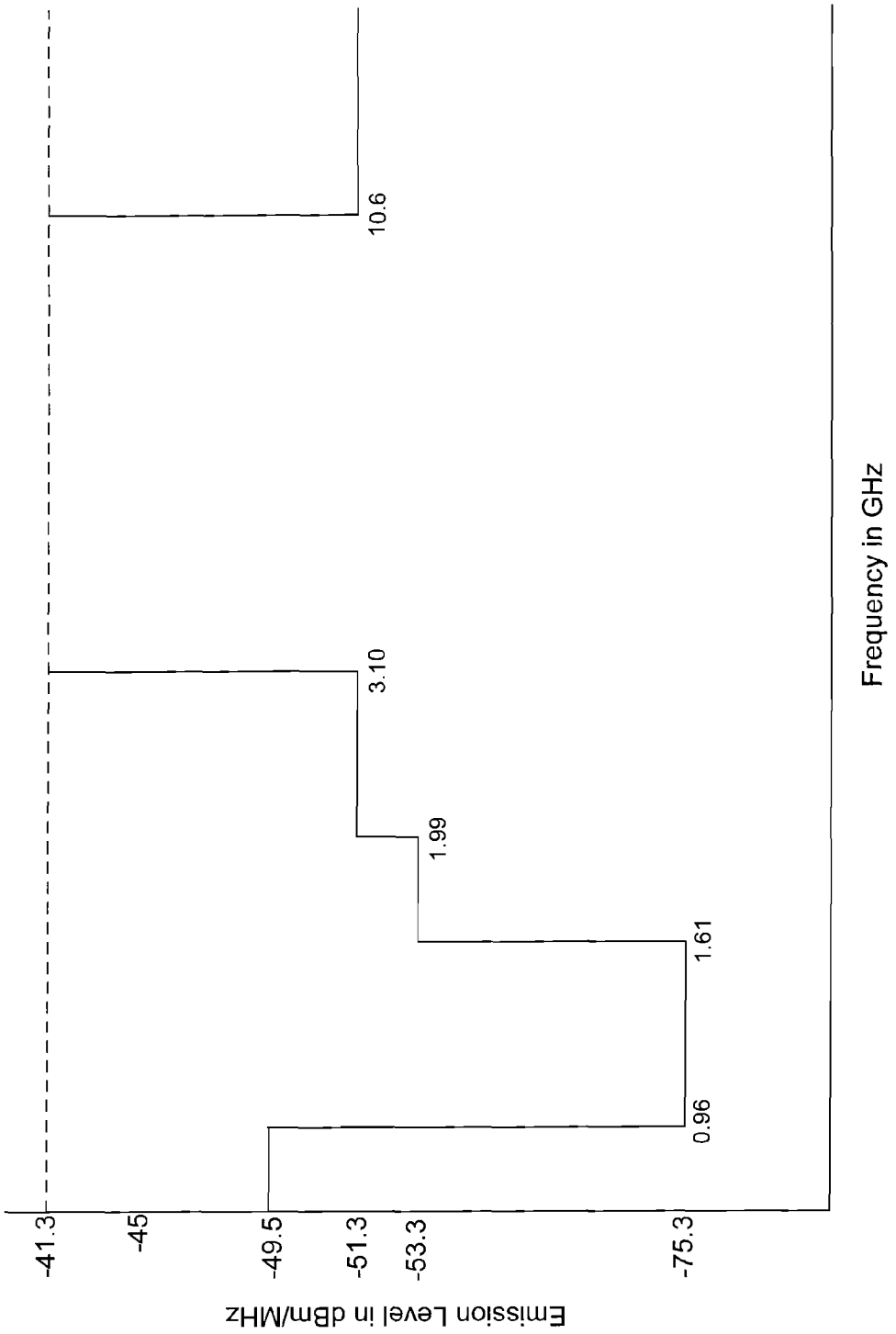


FIG. 27



PRIOR ART
FIG. 28

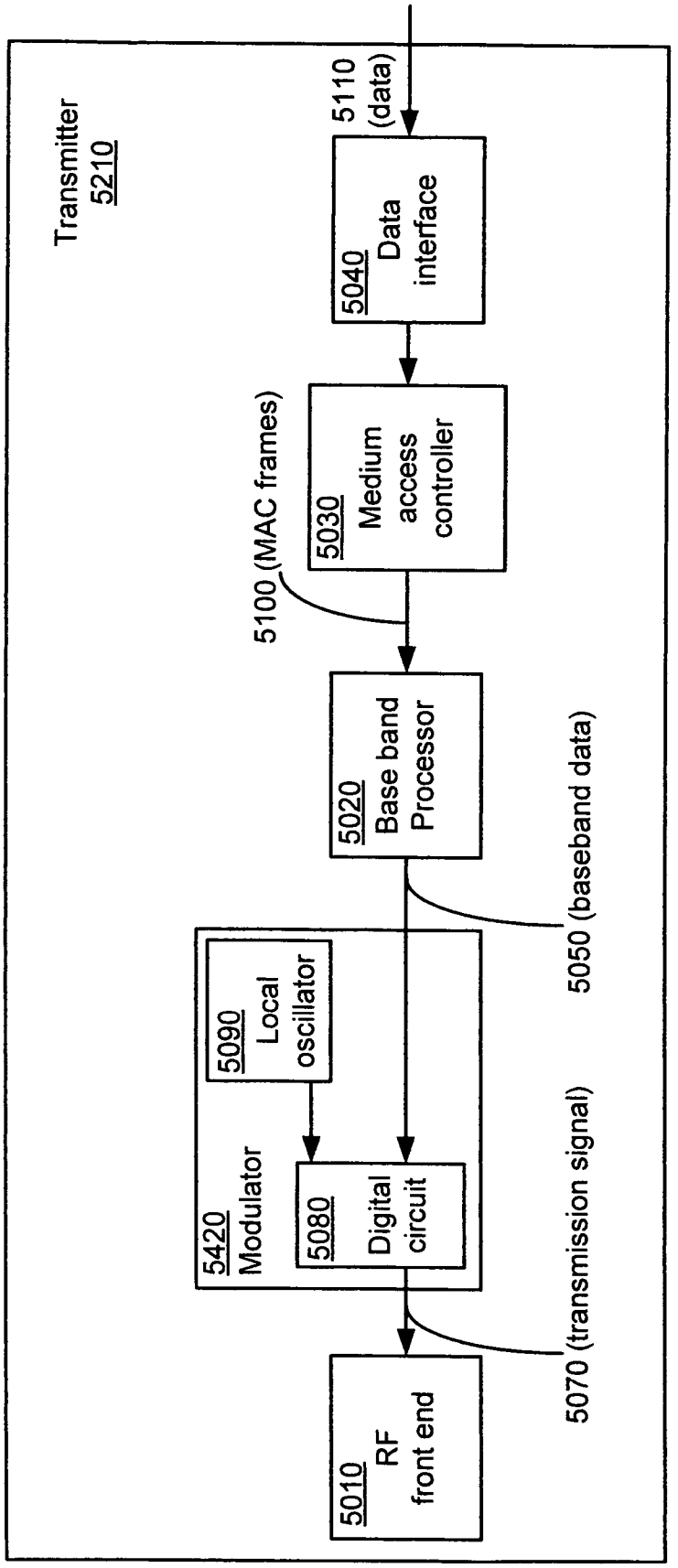

Transmitter
5210
5110
(data)
5040
Data
interface
5100 (MAC frames)
5030
Medium
access
controller
5020
Base band
Processor
5050 (baseband data)
5420
Modulator
5090
Local
oscillator
5080
Digital
circuit
5070 (transmission signal)
5010
RF
front end

FIG. 29

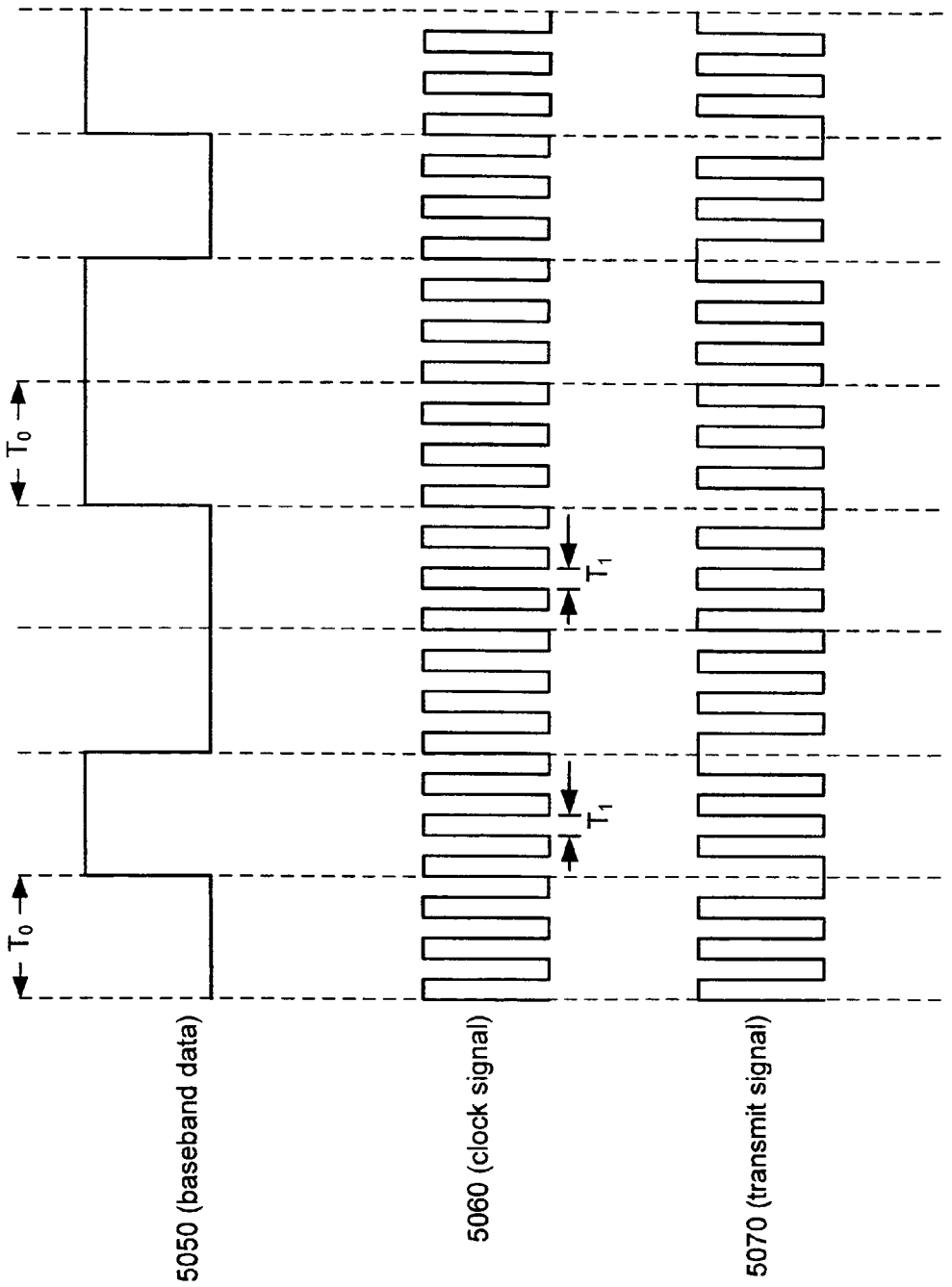

T_0
T_0
T_1
T_1
5050 (baseband data)
5060 (clock signal)
5070 (transmit signal)

FIG. 30

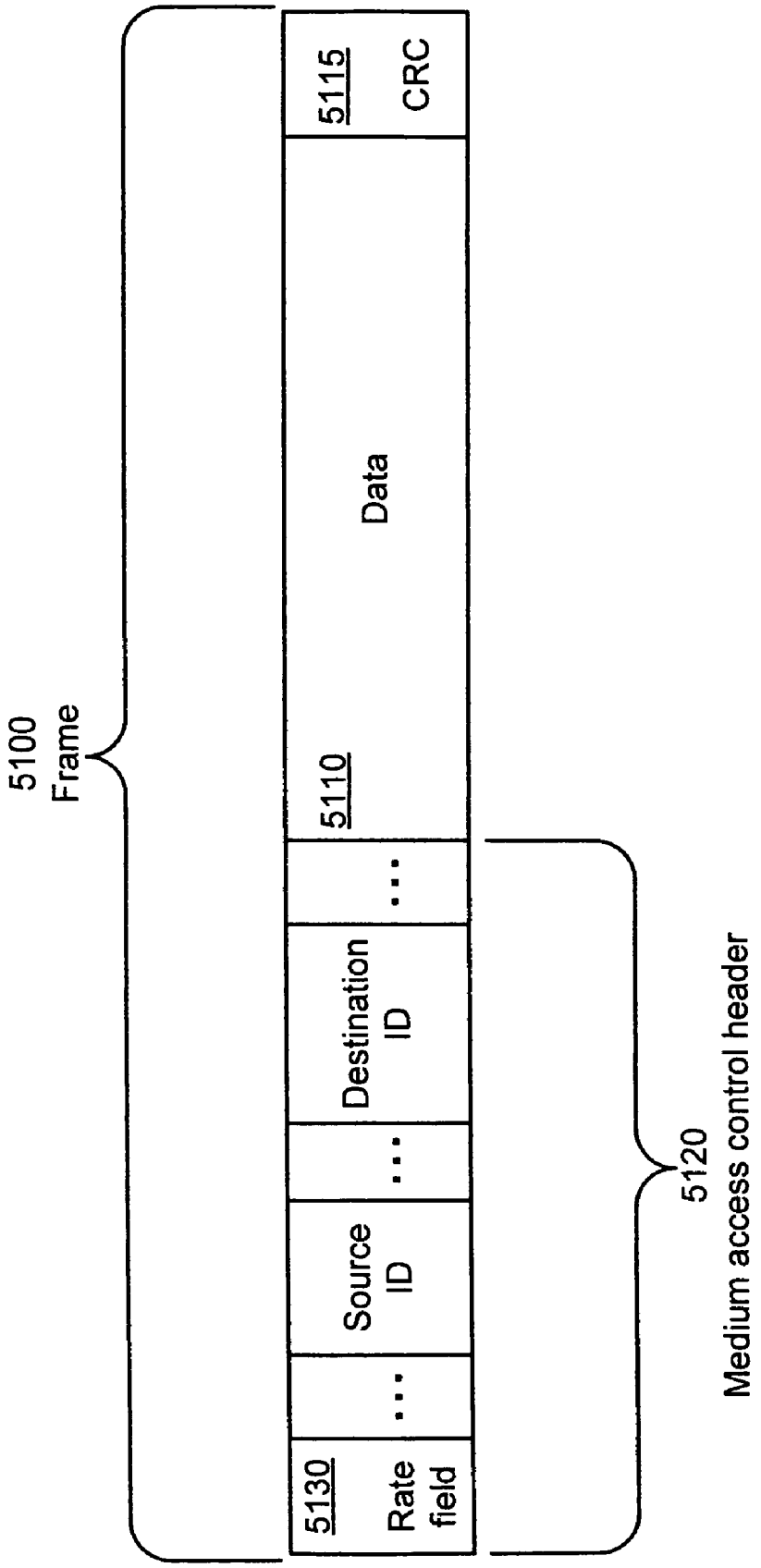

5100
Frame
5130
Rate field
...
Source ID
...
Destination ID
...
5110
Data
5115
CRC
5120
Medium access control header

FIG. 31

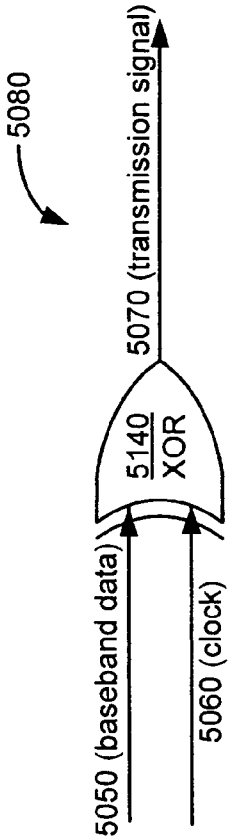


FIG. 32a

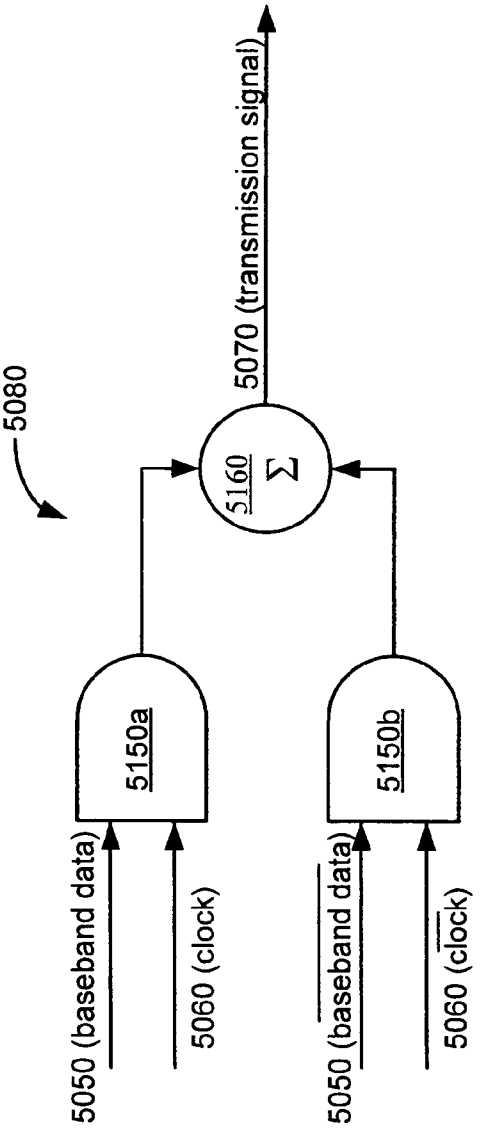


FIG. 32b

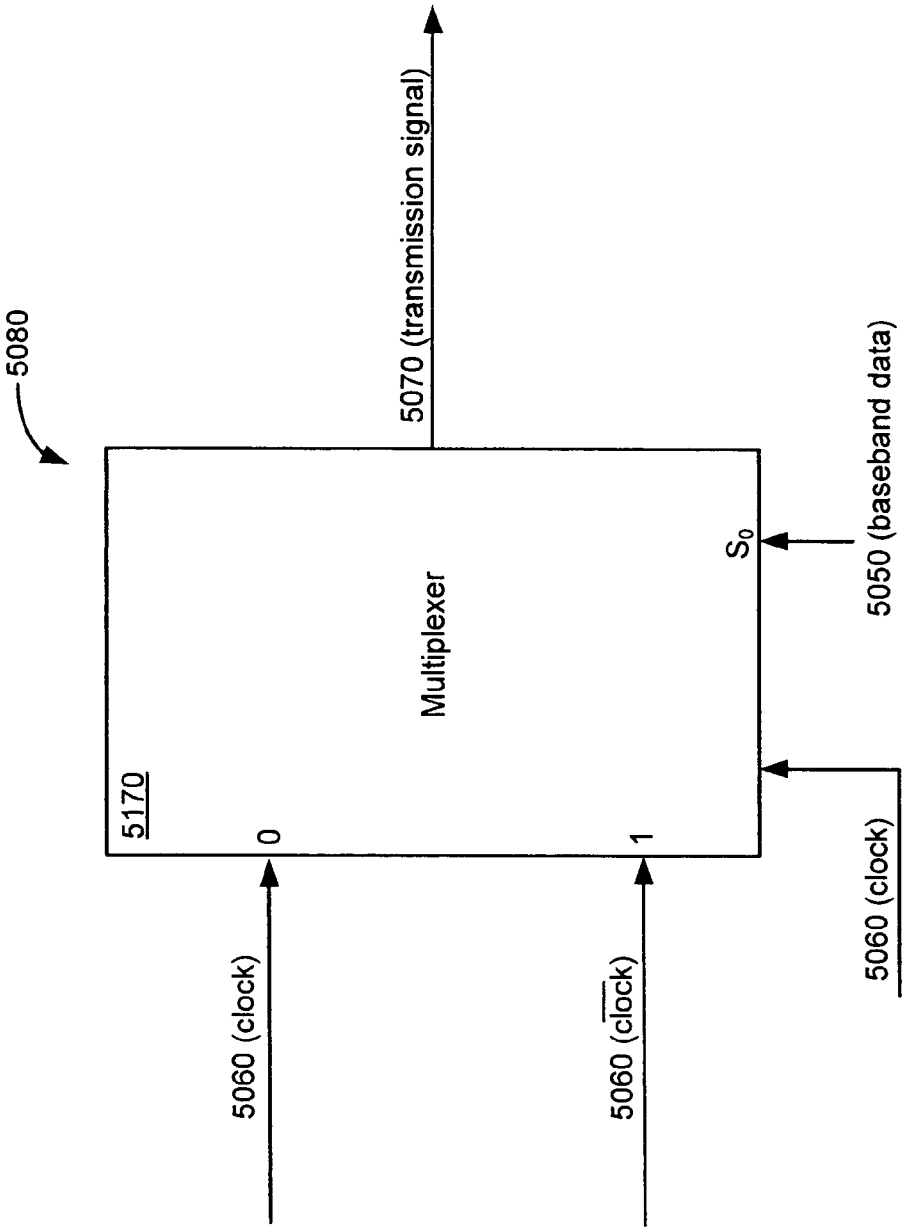

5080
5170
Multiplexer
0
1
S_0
5060 (clock)
5060 ($\overline{\text{clock}}$)
5060 (clock)
5050 (baseband data)
5070 (transmission signal)

FIG. 32c

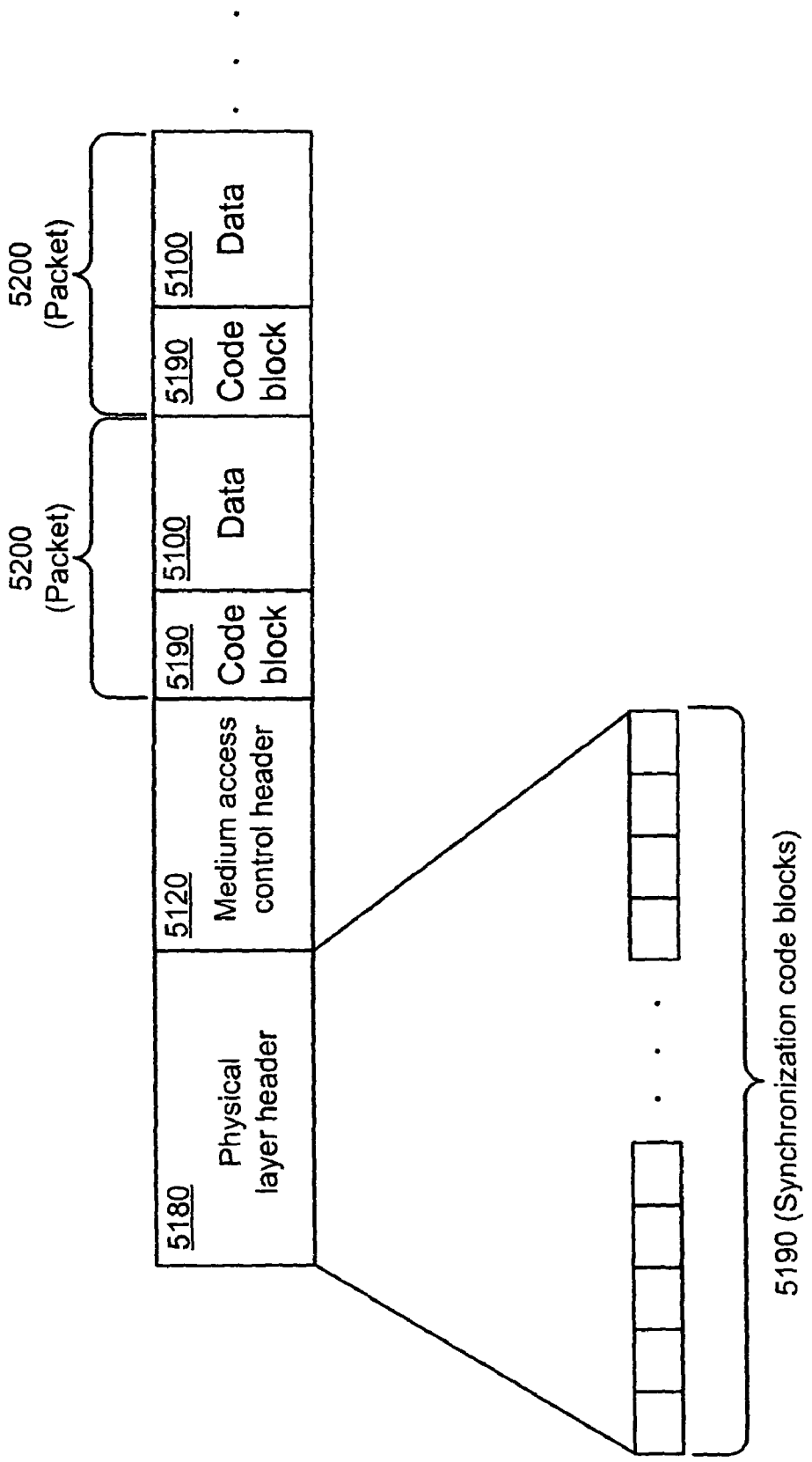

5200
(Packet)
5200
(Packet)
5180
Physical
layer header
5120
Medium access
control header
5190
Code
block
5100
Data
5190
Code
block
5100
Data
5190 (Synchronization code blocks)

FIG. 33

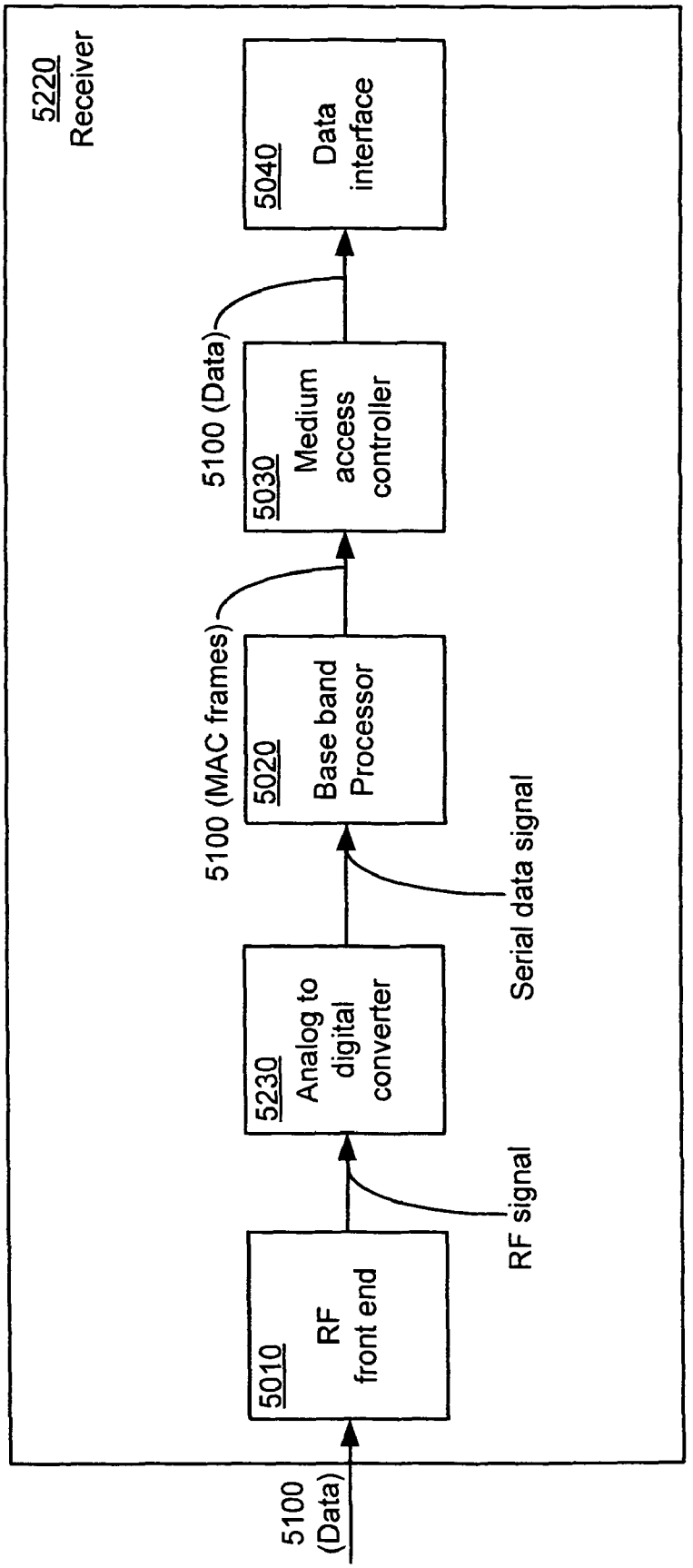

5220
Receiver
5100
(Data)
5010
RF
front end
RF signal
5230
Analog to
digital
converter
Serial data signal
5100 (MAC frames)
5020
Base band
Processor
5100 (Data)
5030
Medium
access
controller
5040
Data
interface

FIG. 34

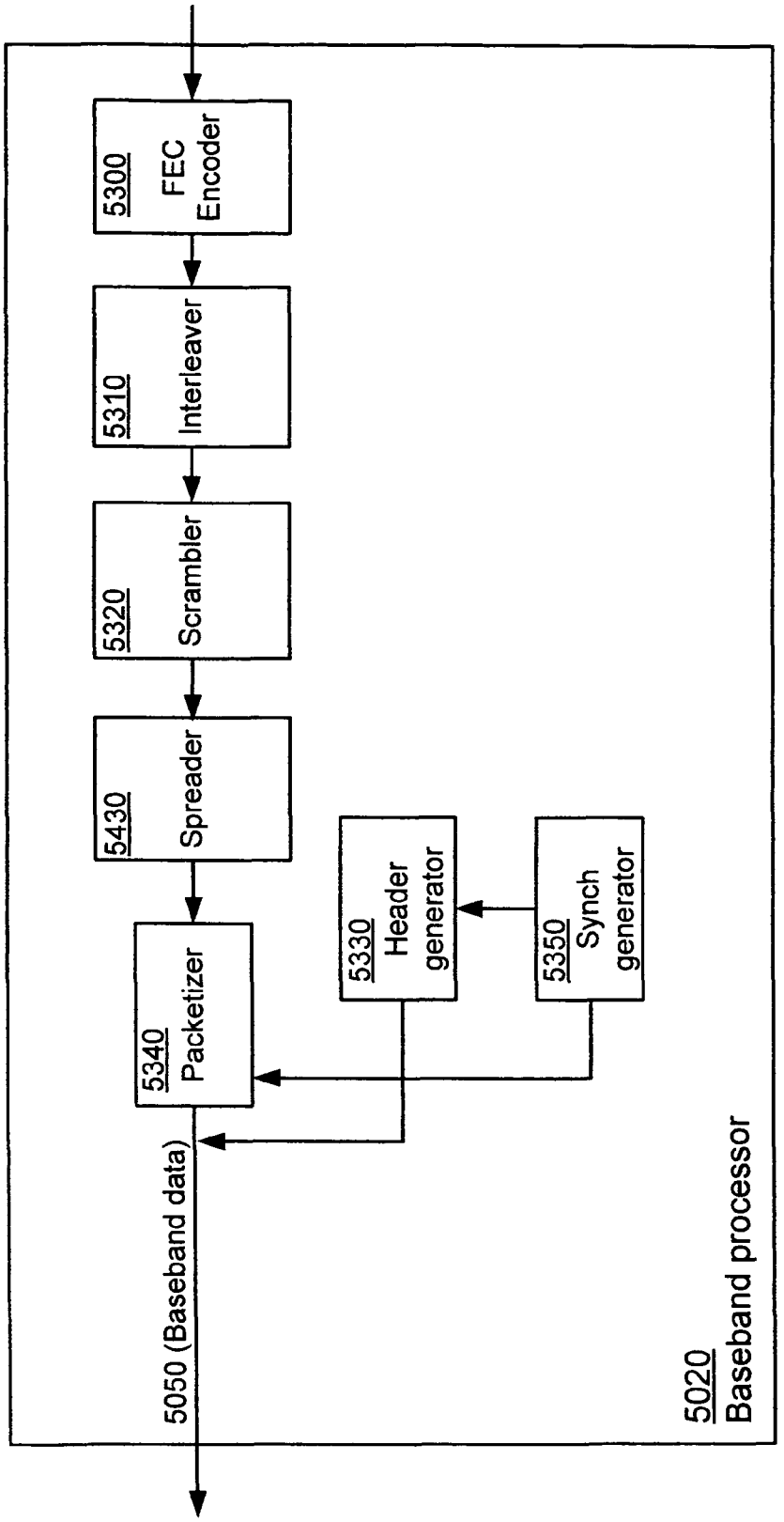

5300
FEC
Encoder
5310
Interleaver
5320
Scrambler
5430
Spreader
5340
Packetizer
5330
Header
generator
5350
Synch
generator
5050 (Baseband data)
5020
Baseband processor

FIG. 35

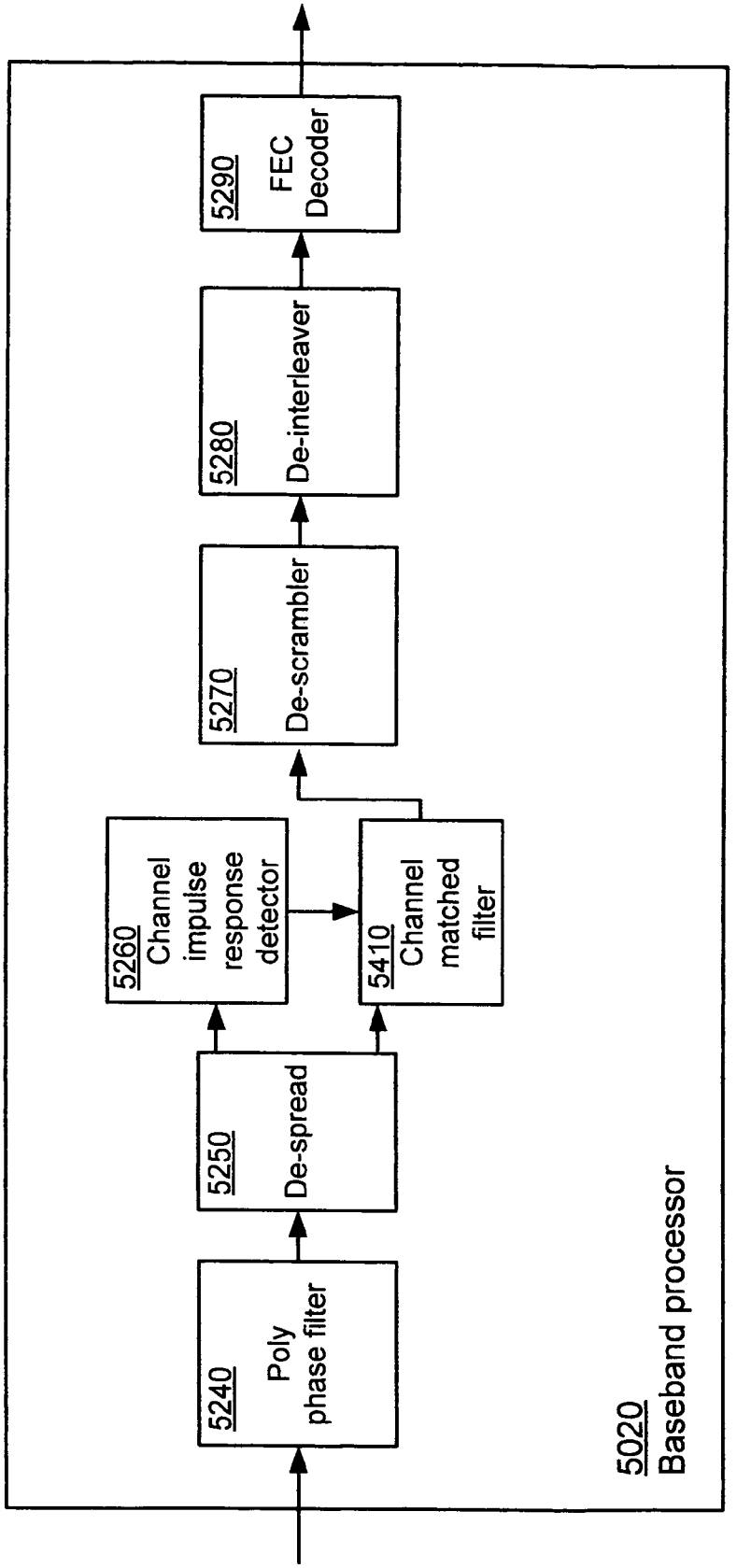

5240
Poly phase filter
5250
De-spread
5260
Channel impulse response detector
5410
Channel matched filter
5270
De-scrambler
5280
De-interleaver
5290
FEC Decoder
5020
Baseband processor

FIG. 36

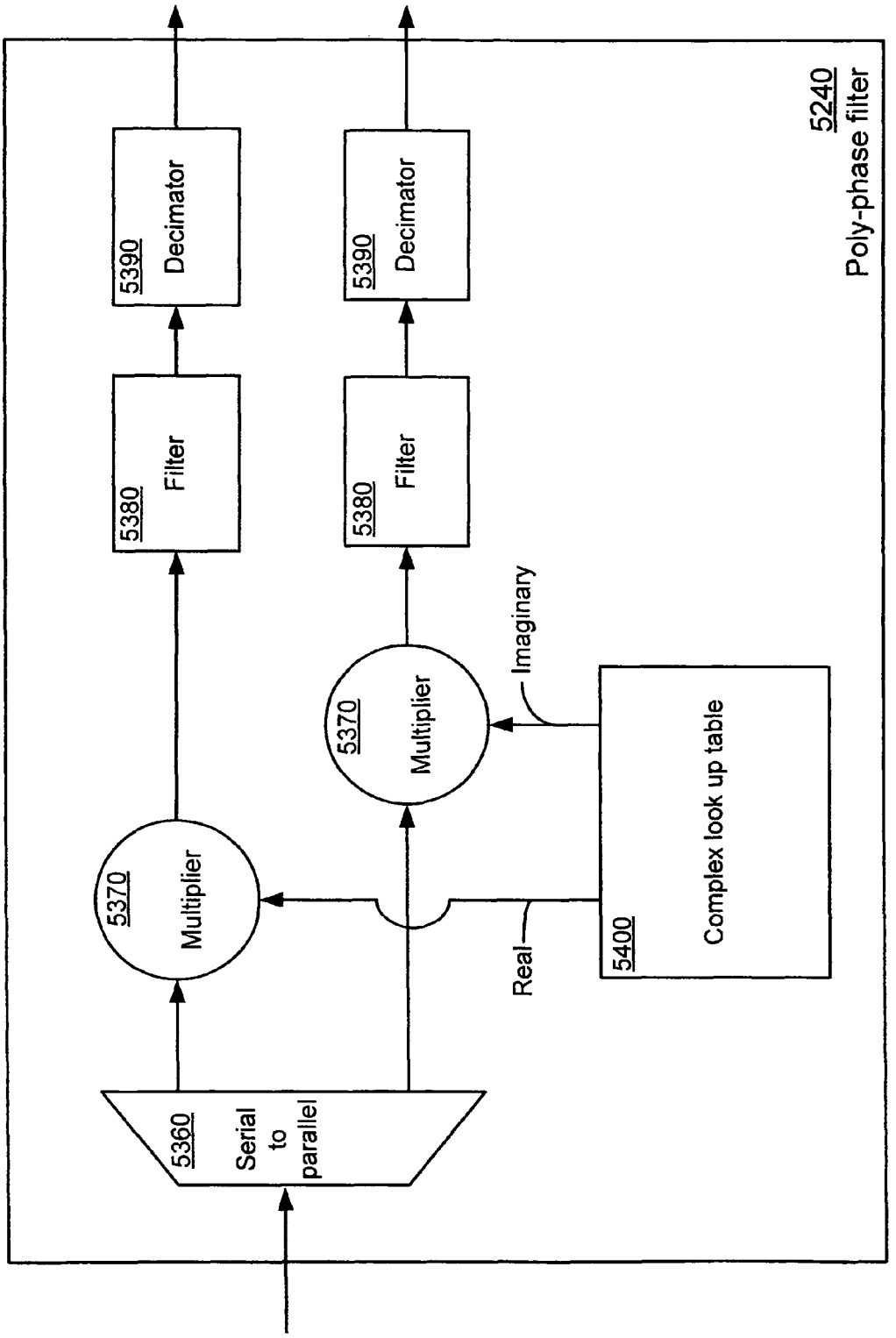

5240
Poly-phase filter
5360
Serial to parallel
5370
Multiplier
5370
Multiplier
5380
Filter
5380
Filter
5390
Decimator
5390
Decimator
Real
Imaginary
5400
Complex look up table

FIG. 37

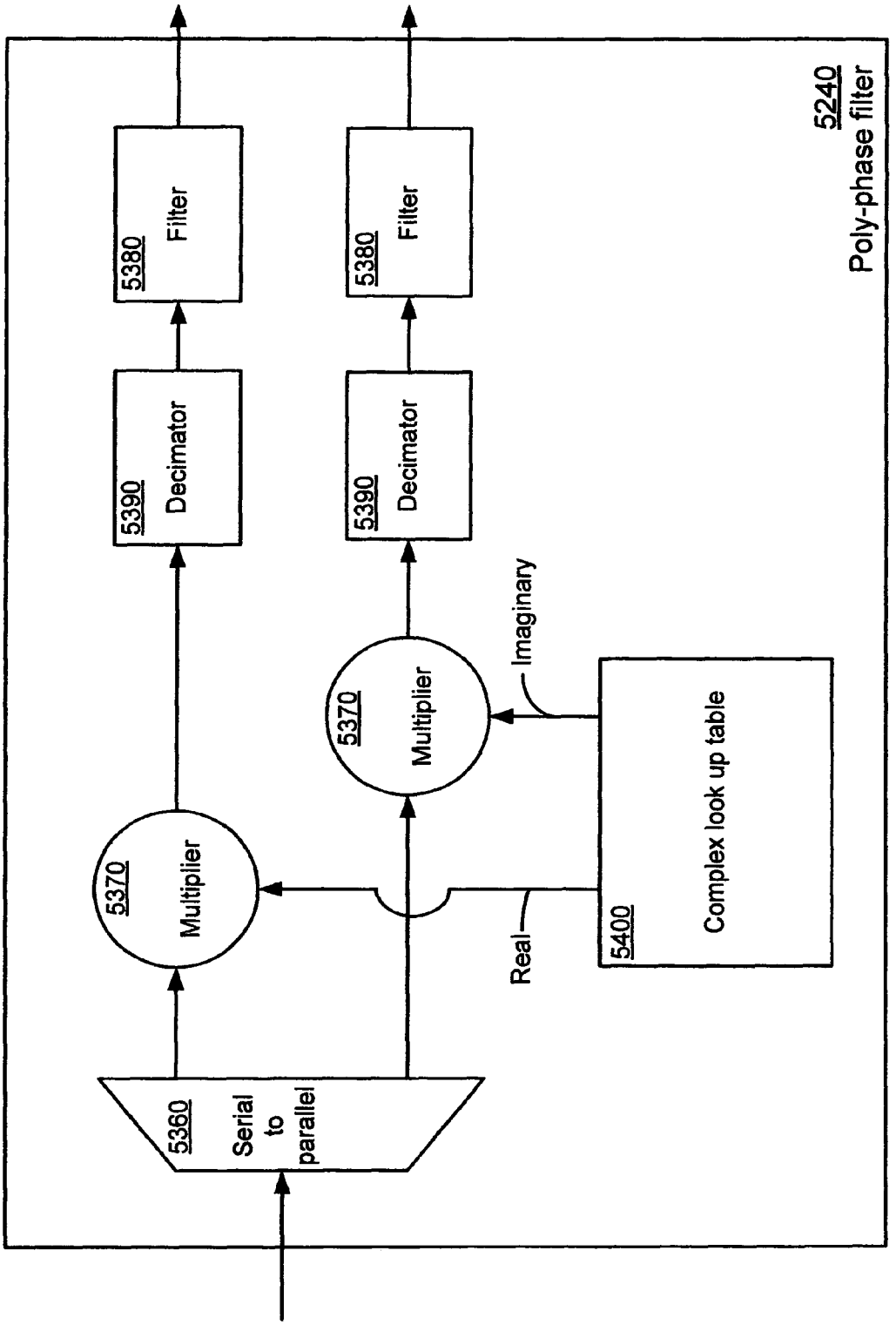

5240
Poly-phase filter
5360
Serial to parallel
5370
Multiplier
5370
Multiplier
5390
Decimator
5390
Decimator
5380
Filter
5380
Filter
Real
Imaginary
5400
Complex look up table

FIG. 38

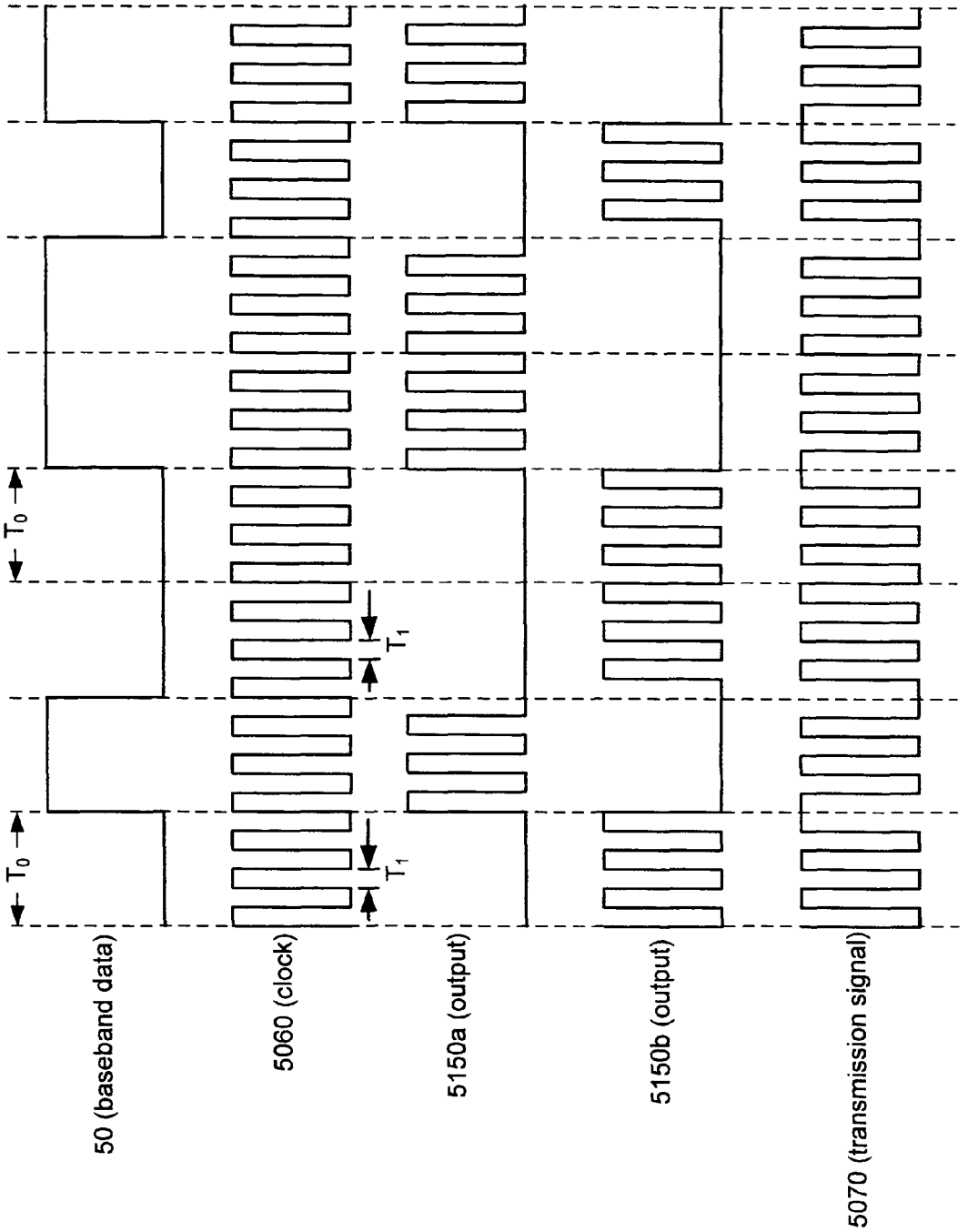

50 (baseband data)
5060 (clock)
5150a (output)
5150b (output)
5070 (transmission signal)
T_0
T_0
T_1
T_1

FIG. 39

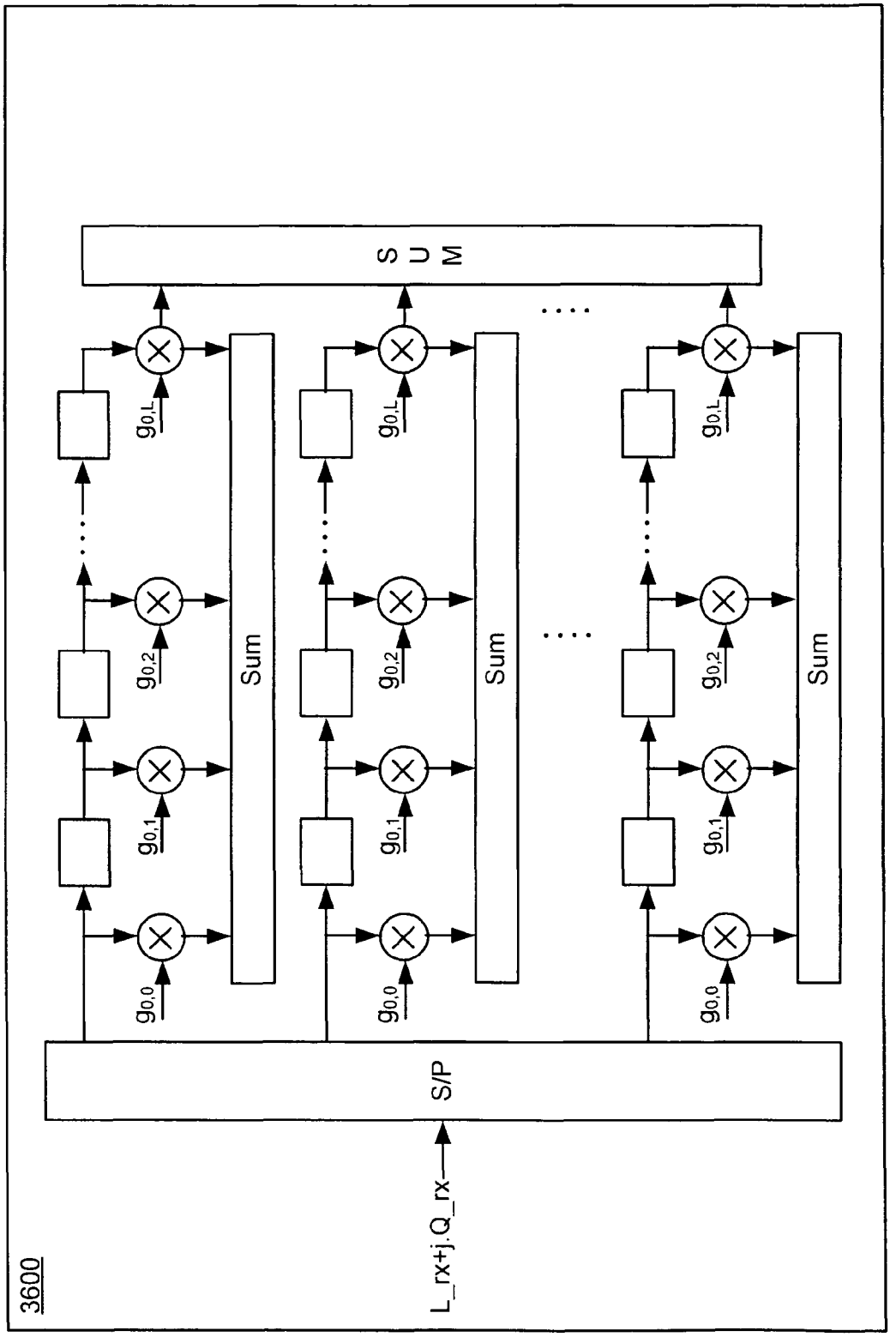


FIG. 40

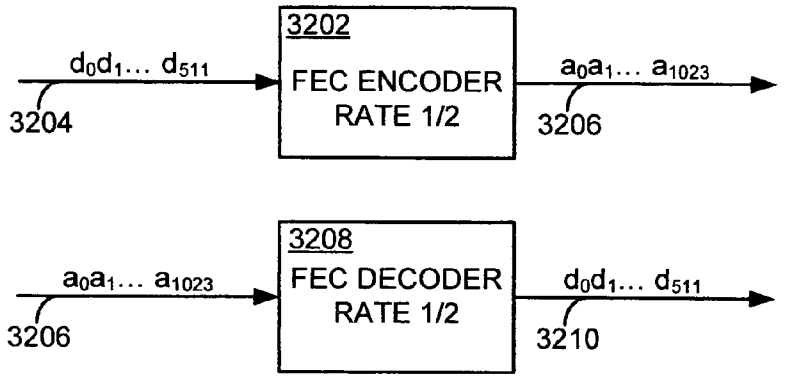

3202
FEC ENCODER
RATE 1/2
$d_0d_1 \ldots d_{511}$
3204
$a_0a_1 \ldots a_{1023}$
3206
3208
FEC DECODER
RATE 1/2
$a_0a_1 \ldots a_{1023}$
3206
$d_0d_1 \ldots d_{511}$
3210

FIG. 41

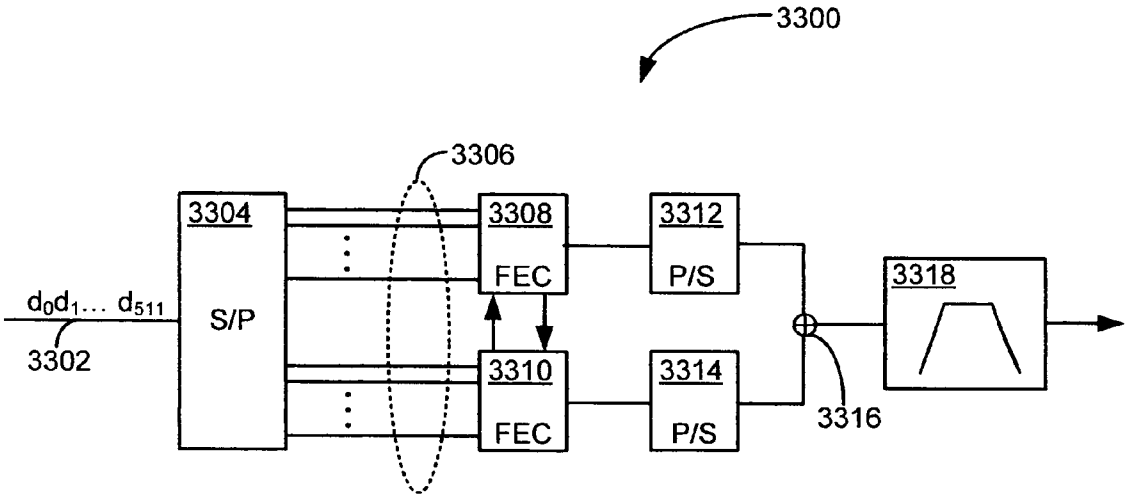

3300
3306
3304
S/P
$d_0d_1 \ldots d_{511}$
3302
3308
FEC
3310
FEC
3312
P/S
3314
P/S
3316
3318

FIG. 42

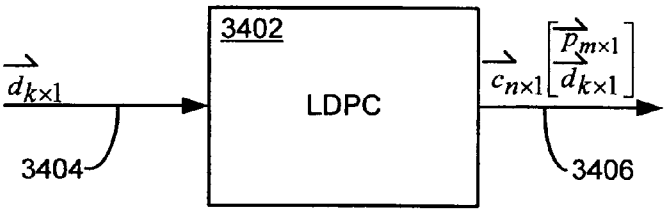

3402
LDPC
$\vec{d}_{k\times1}$
3404
$\vec{c}_{n\times1}$
$\begin{bmatrix}\vec{p}_{m\times1}\\ \vec{d}_{k\times1}\end{bmatrix}$
3406

FIG. 43

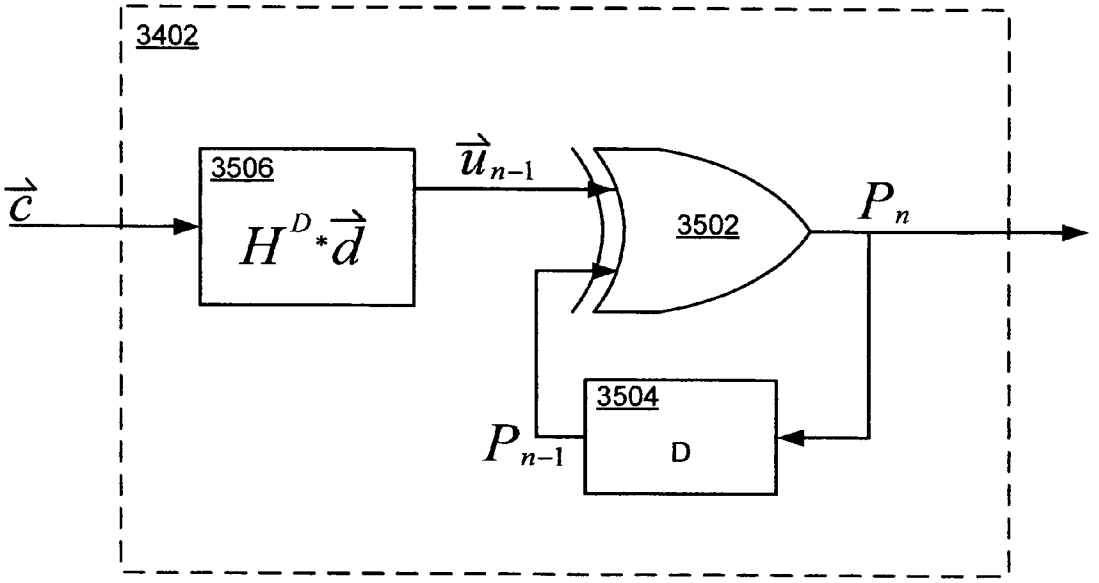

3402
$\vec{c}$
3506
$H^D * \vec{d}$
$\vec{u}_{n-1}$
3502
P_n
3504
D
P_{n-1}

FIG. 44

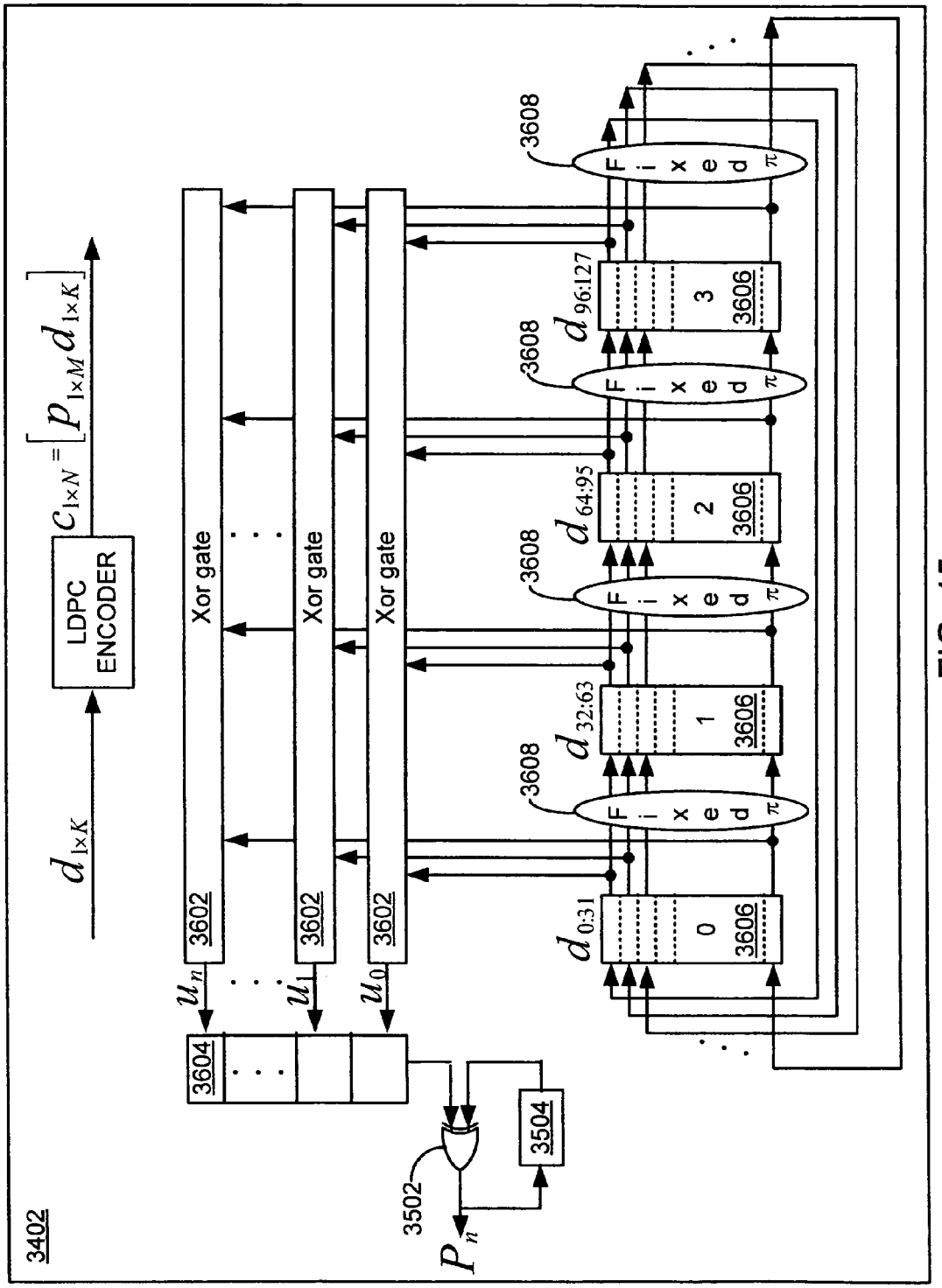

3402
$d_{1\times K}$
LDPC ENCODER
$c_{1\times N} = \left[p_{1\times M} d_{1\times K} \right]$
Xor gate
Xor gate
Xor gate
3602
3602
3602
u_n
u_1
u_0
3604
3502
3504
P_n
$d_{0:31}$
$d_{32:63}$
$d_{64:95}$
$d_{96:127}$
0
1
2
3
3606
3608
Fixed
π

FIG. 45

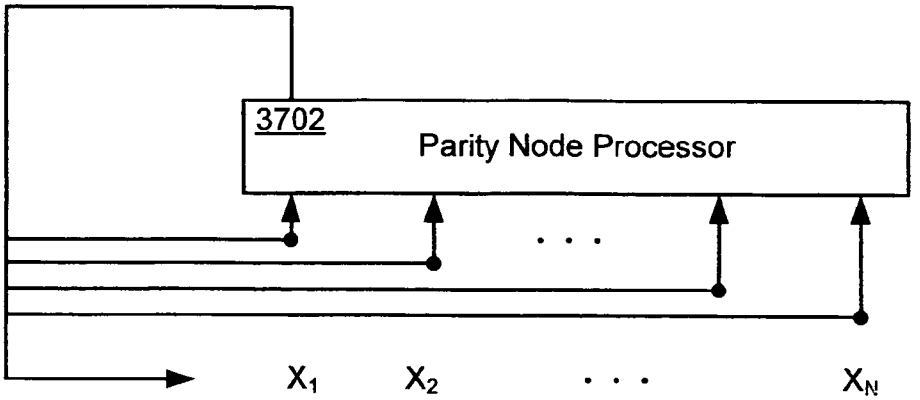

3702
Parity Node Processor
. . .
x_1
x_2
. . .
x_N

FIG. 46

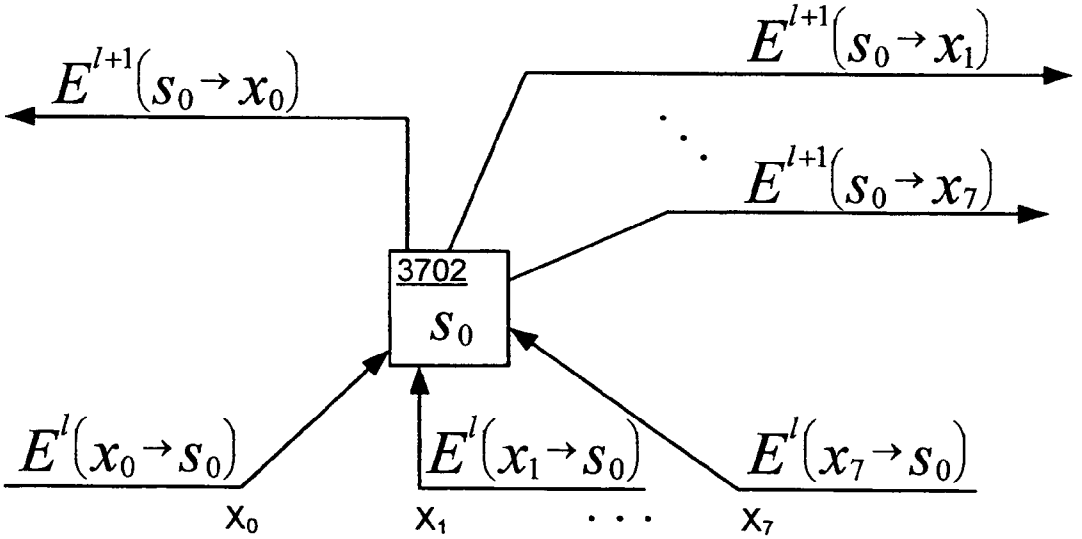

$E^{l+1}(s_0 \to x_0)$
$E^{l+1}(s_0 \to x_1)$
$E^{l+1}(s_0 \to x_7)$
3702
s_0
$E^{l}(x_0 \to s_0)$
$E^{l}(x_1 \to s_0)$
$E^{l}(x_7 \to s_0)$
x_0
x_1
. . .
x_7

FIG. 47

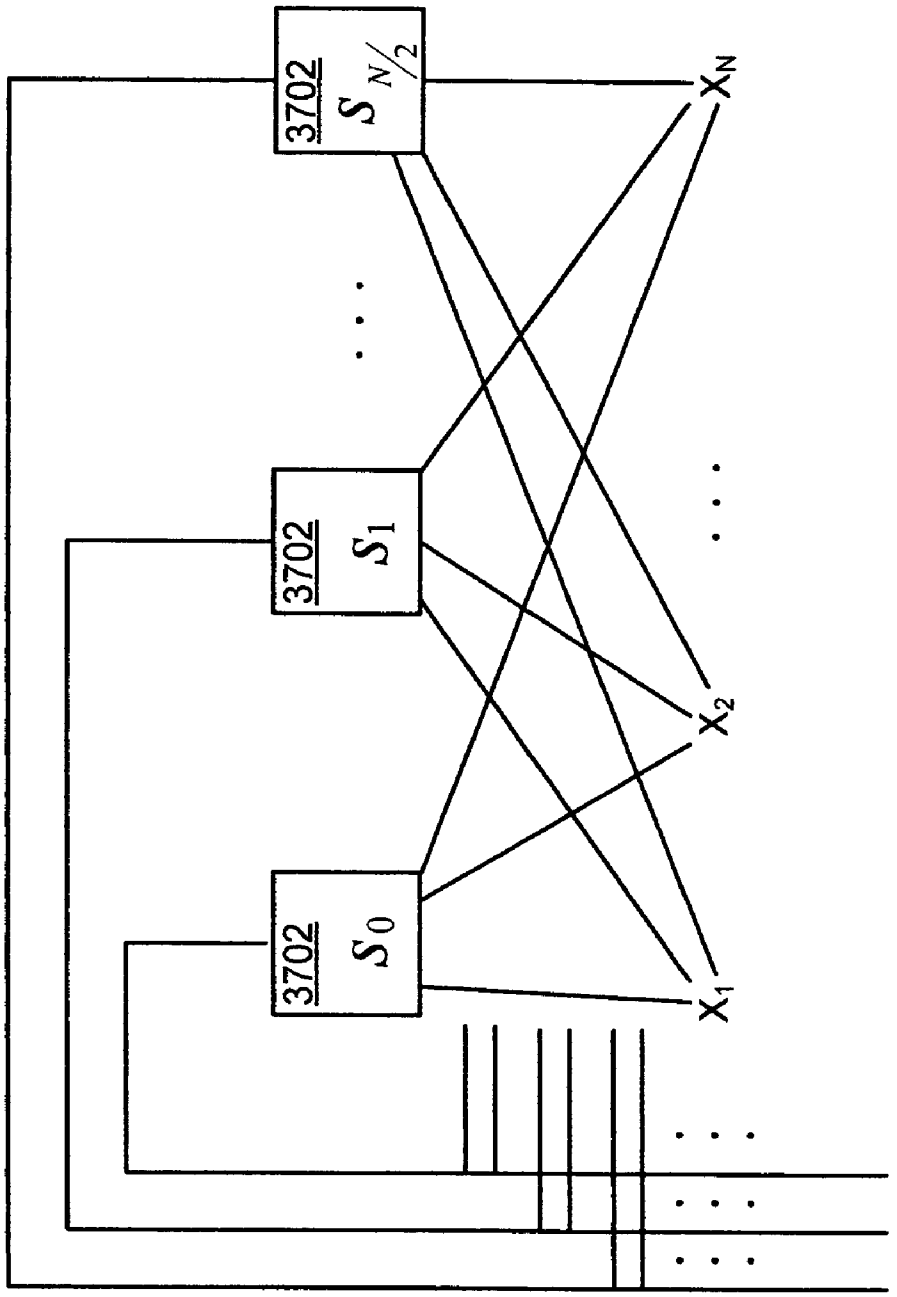


FIG. 48

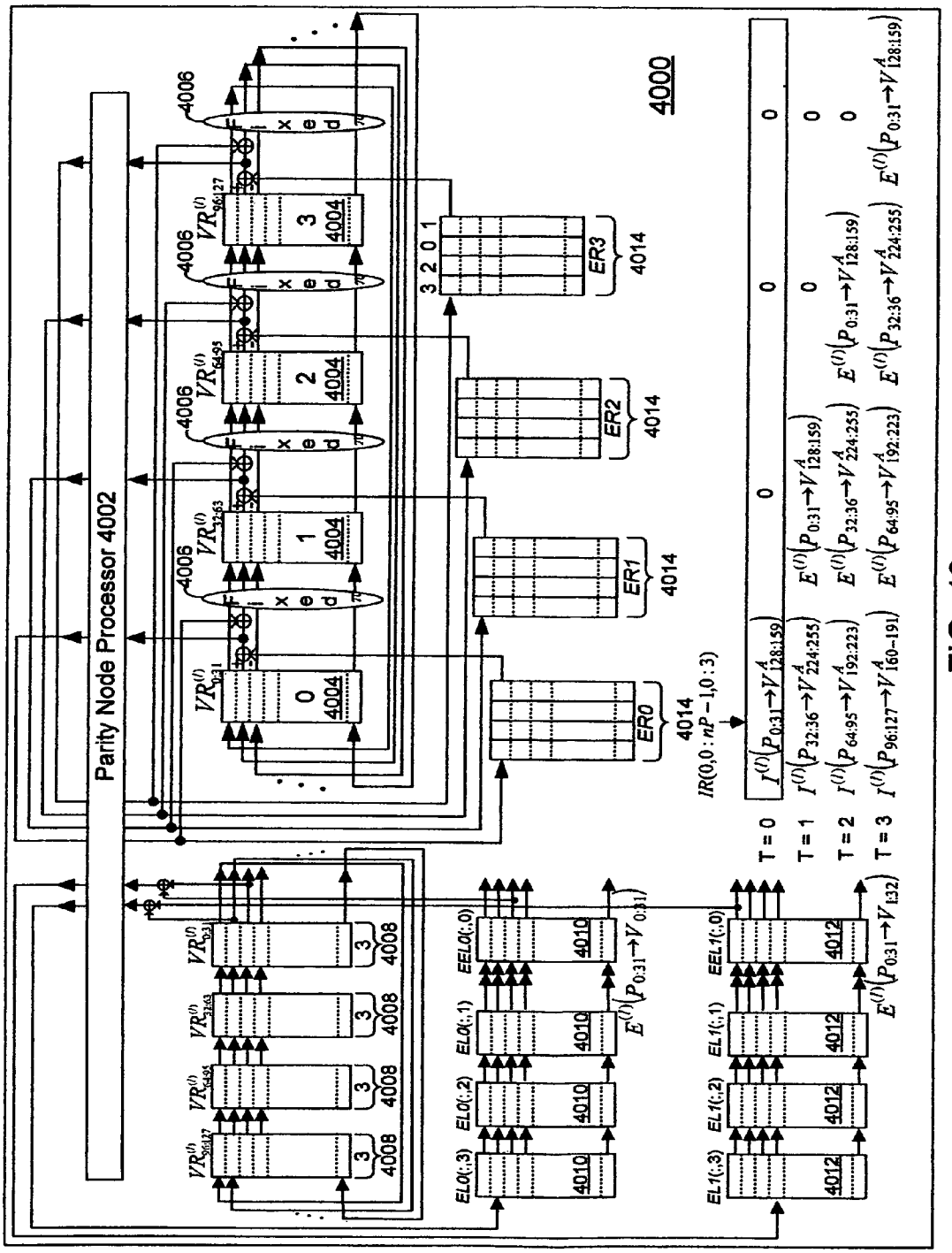

4000
Parity Node Processor 4002
4004
4006
4008
4010
4012
4014
ER0
ER1
ER2
ER3
EEL0(:,0)
EL0(:,1)
EL0(:,2)
EL0(:,3)
EEL1(:,0)
EL1(:,1)
EL1(:,2)
EL1(:,3)
T = 0
T = 1
T = 2
T = 3

FIG. 49

HIGH DATA RATE TRANSMITTER AND RECEIVER

This is a continuation-in-part of U.S. patent application Ser. No. 10/010,601, filed Dec. 6, 2001 now U.S. Pat. No. 7,289,494, entitled: SYSTEMS AND METHODS FOR WIRELESS COMMUNICATION OVER A WIDE BANDWIDTH CHANNEL USING A PLURALITY OF SUBCHANNELS.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates generally to communications, and more particularly to systems and methods for high data rate communications.

2. Background

Wireless communication systems are proliferating at the Wide Area Network (WAN), Local Area Network (LAN), and Personal Area Network (PAN) levels. These wireless communication systems use a variety of techniques to allow simultaneous access to multiple users. The most common of these techniques are Frequency Division Multiple Access (FDMA), which assigns specific frequencies to each user, Time Division Multiple Access (TDMA), which assigns particular time slots to each user, and Code Division Multiple Access (CDMA), which assigns specific codes to each user. But these wireless communication systems and various modulation techniques are afflicted by a host of problems that limit the capacity and the quality of service provided to the users. The following paragraphs briefly describe a few of these problems for the purpose of illustration.

One problem that can exist in a wireless communication system is multipath interference. Multipath interference, or multipath, occurs because some of the energy in a transmitted wireless signal bounces off of obstacles, such as buildings or mountains, as it travels from source to destination. The obstacles in effect create reflections of the transmitted signal and the more obstacles there are, the more reflections they generate. The reflections then travel along their own transmission paths to the destination (or receiver). The reflections will contain the same information as the original signal; however, because of the differing transmission path lengths, the reflected signals will be out of phase with the original signal. As a result, they will often combine destructively with the original signal in the receiver. This is referred to as fading. To combat fading, current systems typically try to estimate the multipath effects and then compensate for them in the receiver using an equalizer. In practice, however, it is very difficult to achieve effective multipath compensation.

A second problem that can affect the operation of wireless communication systems is interference from adjacent communication cells within the system. In FDMA/TDMA systems, this type of interference is prevent through a frequency reuse plan. Under a frequency reuse plan, available communication frequencies are allocated to communication cells within the communication system such that the same frequency will not be used in adjacent cells. Essentially, the available frequencies are split into groups. The number of groups is termed the reuse factor. Then the communication cells are grouped into clusters, each cluster containing the same number of cells as there are frequency groups. Each frequency group is then assigned to a cell in each cluster. Thus, if a frequency reuse factor of 7 is used, for example, then a particular communication frequency will be used only once in every seven communication cells. Thus, in any group

of seven communication cells, each cell can only use $1/7^{th}$ of the available frequencies, i.e., each cell is only able to use $1/7^{th}$ of the available bandwidth.

In a CDMA communication system, each cell uses the same wideband communication channel. In order to avoid interference with adjacent cells, each communication cell uses a particular set of spread spectrum codes to differentiate communications within the cell from those originating outside of the cell. Thus, CDMA systems preserve the bandwidth in the sense that they avoid reuse planning. But as will be discussed, there are other issues that limit the bandwidth in CDMA systems as well.

Thus, in overcoming interference, system bandwidth is often sacrificed. Bandwidth is becoming a very valuable commodity as wireless communication systems continue to expand by adding more and more users. Therefore, trading off bandwidth for system performance is a costly, albeit necessary, proposition that is inherent in all wireless communication systems.

The foregoing are just two examples of the types of problems that can affect conventional wireless communication systems. The examples also illustrate that there are many aspects of wireless communication system performance that can be improved through systems and methods that, for example, reduce interference, increase bandwidth, or both.

Not only are conventional wireless communication systems effected by problems, such as those described in the preceding paragraphs, but also different types of systems are effected in different ways and to different degrees. Wireless communication systems can be split into three types: 1) line-of-sight systems, which can include point-to-point or point-to-multipoint systems; 2) indoor non-line of sight systems; and 3) outdoor systems such as wireless WANs. Line-of-sight systems are least affected by the problems described above, while indoor systems are more affected, due for example to signals bouncing off of building walls. Outdoor systems are by far the most affected of the three systems. Because these types of problems are limiting factors in the design of wireless transmitters and receivers, such designs must be tailored to the specific types of system in which it will operate. In practice, each type of system implements unique communication standards that address the issues unique to the particular type of system. Even if an indoor system used the same communication protocols and modulation techniques as an outdoor system, for example, the receiver designs would still be different because multipath and other problems are unique to a given type of system and must be addressed with unique solutions. This would not necessarily be the case if cost efficient and effective methodologies can be developed to combat such problems as described above that build in programmability so that a device can be reconfigured for different types of systems and still maintain superior performance.

SUMMARY OF THE INVENTION

In order to combat the above problems, a high data rate transmitter and receiver are provided. In one embodiment, a transmitter comprises a baseband processor structured to receive data and to convert the data into a multiplicity of high and low signal values, with each high and low signal value having a first timing interval. A local oscillator generates a clock signal at a second timing interval and a digital circuit combines the high and low signal values with the clock signal to produce a transmission signal directly at a transmission frequency.

The radio frequency used for transmission may range up to 11 Giga-Hertz, and production of the transmission signal directly at the transmission frequency is possible by use of a high-speed oscillator.

A receiver is structured to receive the communication signal, which in one embodiment, may have a fractional bandwidth that may range between approximately 20 percent and approximately 200 percent. The receiver includes a high-speed analog to digital converter configured to directly convert the radio frequency signal into a data signal.

These and other features and advantages of the present invention will be appreciated from review of the following Detailed Description of the Preferred Embodiments, along with the accompanying figures in which like reference numerals are used to describe the same, similar or corresponding parts in the several views of the drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Preferred embodiments of the present inventions taught herein are illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings, in which:

FIG. **1**A is a diagram illustrating an example embodiment of a wideband channel divided into a plurality of sub-channels in accordance with the invention;

FIG. **1**B is a diagram illustrating the effects of multipath in a wireless communication system;

FIG. **2** is a diagram illustrating another example embodiment of a wideband communication channel divided into a plurality of sub-channels in accordance with the invention;

FIG. **3** is a diagram illustrating the application of a roll-off factor to the sub-channels of FIGS. **1** and **2**;

FIG. **4**A is a diagram illustrating the assignment of sub-channels for a wideband communication channel in accordance with the invention;

FIG. **4**B is a diagram illustrating the assignment of time slots for a wideband communication channel in accordance with the invention;

FIG. **5** is a diagram illustrating an example embodiment of a wireless communication in accordance with the invention;

FIG. **6** is a diagram illustrating the use of synchronization codes in the wireless communication system of FIG. **5** in accordance with the invention;

FIG. **7** is a diagram illustrating a correlator that can be used to correlate synchronization codes in the wireless communication system of FIG. **5**;

FIG. **8** is a diagram illustrating synchronization code correlation in accordance with the invention;

FIG. **9** is a diagram illustrating the cross-correlation properties of synchronization codes configured in accordance with the invention;

FIG. **10** is a diagram illustrating another example embodiment of a wireless communication system in accordance with the invention;

FIG. **11**A is a diagram illustrating how sub-channels of a wideband communication channel according to the present invention can be grouped in accordance with the present invention;

FIG. **11**B is a diagram illustrating the assignment of the groups of sub-channels of FIG. **11**A in accordance with the invention;

FIG. **12** is a diagram illustrating the group assignments of FIG. **11**B in the time domain;

FIG. **13** is a flow chart illustrating the assignment of sub-channels based on SIR measurements in the wireless communication system of FIG. **10** in accordance with the invention;

FIG. **14** is a logical block diagram of an example embodiment of transmitter configured in accordance with the invention;

FIG. **15** is a logical block diagram of an example embodiment of a modulator configured in accordance with the present invention for use in the transmitter of FIG. **14**;

FIG. **16** is a diagram illustrating an example embodiment of a rate controller configured in accordance with the invention for use in the modulator of FIG. **15**;

FIG. **17** is a diagram illustrating another example embodiment of a rate controller configured in accordance with the invention for use in the modulator of FIG. **15**;

FIG. **18** is a diagram illustrating an example embodiment of a frequency encoder configured in accordance with the invention for use in the modulator of FIG. **15**;

FIG. **19** is a logical block diagram of an example embodiment of a TDM/FDM block configured in accordance with the invention for use in the modulator of FIG. **15**;

FIG. **20** is a logical block diagram of another example embodiment of a TDM/FDM block configured in accordance with the invention for use in the modulator of FIG. **15**;

FIG. **21** is a logical block diagram of an example embodiment of a frequency shifter configured in accordance with the invention for use in the modulator of FIG. **15**;

FIG. **22** is a logical block diagram of a receiver configured in accordance with the invention;

FIG. **23** is a logical block diagram of an example embodiment of a demodulator configured in accordance with the invention for use in the receiver of FIG. **22**;

FIG. **24** is a logical block diagram of an example embodiment of an equalizer configured in accordance with the present invention for use in the demodulator of FIG. **23**;

FIG. **25** is a logical block diagram of an example embodiment of a wireless communication device configured in accordance with the invention;

FIG. **26** is an illustration of different communication methods;

FIG. **27** is an illustration of two ultra-wideband pulses;

FIG. **28** is a chart of ultra-wideband emission limits as established by the Federal Communications Commission on Apr. 22, 2002;

FIG. **29** illustrates a transmitter consistent with one embodiment of the present invention;

FIG. **30** illustrates a timing diagram of various signals;

FIG. **31** illustrates a frame consistent with one embodiment of the present invention;

FIG. **32***a* illustrates one embodiment of a digital circuit employed in the transmitter of FIG. **29**;

FIG. **32***b* illustrates a second embodiment of a digital circuit employed in the transmitter of FIG. **29**;

FIG. **32***c* illustrates a third embodiment of a digital circuit employed in the transmitter of FIG. **29**;

FIG. **33** illustrates a data stream consistent with one embodiment of the present invention;

FIG. **34** illustrates a receiver consistent with one embodiment of the present invention;

FIG. **35** illustrates a schematic of a first portion of a baseband processor employed in the transmitter of FIG. **29**;

FIG. **36** illustrates a schematic of a second portion of a baseband processor employed in the receiver of FIG. **34**;

FIG. **37** illustrates one embodiment of a poly-phase filter employed in the baseband processor of FIG. **36**;

FIG. **38** illustrates another embodiment of a poly-phase filter employed in the baseband processor of FIG. **36**;

FIG. **39** illustrates another timing diagram of signals consistent with the present invention.

FIG. **40** illustrates one embodiment of an equalizer consistent with the present invention;

FIG. **41** illustrates an exemplary FEC encoder and exemplary FEC decoder;

FIG. **42** illustrates an example FEC encoder configured in accordance with one embodiment of the present invention;

FIG. **43** illustrates a FEC encoder configured to generate a code word from input data in accordance with one embodiment;

FIG. **44** illustrates the encoder of FIG. **42** in more detail;

FIG. **45** illustrates further detail for the encoder of FIG. **42**;

FIG. **46** illustrates an example parity node processor that can be included in a decoder in accordance with one embodiment;

FIG. **47** illustrates one node of the parity node processor of FIG. **45**;

FIG. **48** illustrates the parity node processor of FIG. **45** in more detail; and

FIG. **49** illustrates a parity node processor configured in accordance with one embodiment.

It will be recognized that some or all of the Figures are schematic representations for purposes of illustration and do not necessarily depict the actual relative sizes or locations of the elements shown. The Figures are provided for the purpose of illustrating one or more embodiments of the invention with the explicit understanding that they will not be used to limit the scope or the meaning of the claims.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

1. Introduction

In the following paragraphs, the present invention will be described in detail by way of example with reference to the attached drawings. While this invention is capable of embodiment in many different forms, there is shown in the drawings and will herein be described in detail specific embodiments, with the understanding that the present disclosure is to be considered as an example of the principles of the invention and not intended to limit the invention to the specific embodiments shown and described. That is, throughout this description, the embodiments and examples shown should be considered as exemplars, rather than as limitations on the present invention. As used herein, the "present invention" refers to any one of the embodiments of the invention described herein, and any equivalents. Furthermore, reference to various feature(s) of the "present invention" throughout this document does not mean that all claimed embodiments or methods must include the referenced feature(s).

In order to improve wireless communication system performance and allow a single device to move from one type of system to another, while still maintaining superior performance, the systems and methods described herein provide various communication methodologies that enhance performance of transmitters and receivers with regard to various common problems that afflict such systems and that allow the transmitters and/or receivers to be reconfigured for optimal performance in a variety of systems. Accordingly, the systems and methods described herein define a channel access protocol that uses a common wideband communication channel for all communication cells. The wideband channel, however, is then divided into a plurality of sub-channels. Different sub-channels are then assigned to one or more users within each cell. But the base station, or service access point, within each cell transmits one message that occupies the entire bandwidth of the wideband channel. Each user's communication device receives the entire message, but only decodes those portions of the message that reside in sub-channels assigned to the user. For a point-to-point system, for example, a single user may be assigned all sub-channels and, therefore, has the full wide band channel available to them. In a wireless WAN, on the other hand, the sub-channels may be divided among a plurality of users.

In the descriptions of example embodiments that follow, implementation differences, or unique concerns, relating to different types of systems will be pointed out to the extent possible. But it should be understood that the systems and methods described herein are applicable to any type of communication systems. In addition, terms such as communication cell, base station, service access point, etc. are used interchangeably to refer to the common aspects of networks at these different levels.

To begin illustrating the advantages of the systems and methods described herein, one can start by looking at the multipath effects for a single wideband communication channel **100** of bandwidth B as shown in FIG. **1**A. Communications sent over channel **100** in a traditional wireless communication system will comprise digital data bits, or symbols, that are encoded and modulated onto a RF carrier that is centered at frequency f_c and occupies bandwidth B. Generally, the width of the symbols (or the symbol duration) T is defined as 1/B. Thus, if the bandwidth B is equal to 100 MHz, then the symbol duration T is defined by the following equation:

$$T=1/B=1/100 \text{ megahertz (MHZ)}=10 \text{ nanoseconds (ns)}. \quad (1)$$

When a receiver receives the communication, demodulates it, and then decodes it, it will recreate a stream **104** of data symbols **106** as illustrated in FIG. **1**B. But the receiver will also receive multipath versions **108** of the same data stream. Because multipath data streams **108** are delayed in time relative to the data stream **104** by delays d**1**, d**2**, d**3**, and d**4**, for example, they may combine destructively with data stream **104**.

A delay spread d_s is defined as the delay from reception of data stream **104** to the reception of the last multipath data stream **108** that interferes with the reception of data stream **104**. Thus, in the example illustrated in FIG. **1**B, the delay spread d_s is equal to delay d**4**. The delay spread d_s will vary for different environments. An environment with a lot of obstacles will create a lot of multipath reflections. Thus, the delay spread d_s will be longer. Experiments have shown that for outdoor WAN type environments, the delay spread d_s can be as long as 20 microseconds. Using the 10 ns symbol duration of equation (1), this translates to 2000 symbols. Thus, with a very large bandwidth, such as 100 MHz, multipath interference can cause a significant amount of interference at the symbol level for which adequate compensation is difficult to achieve. This is true even for indoor environments. For indoor LAN type systems, the delay spread d_s is significantly shorter, typically about 1 microsecond. For a 10 ns symbol duration, this is equivalent to 100 symbols, which is more manageable but still significant.

By segmenting the bandwidth B into a plurality of sub-channels **202**, as illustrated in FIG. **2**, and generating a distinct data stream for each sub-channel, the multipath effect can be reduced to a much more manageable level. For example, if the bandwidth b of each sub-channel **202** is 500

KHz, then the symbol duration is 2 microseconds. Thus, the delay spread d_s for each sub-channel is equivalent to only 10 symbols (outdoor) or half a symbol (indoor). Thus, by breaking up a message that occupies the entire bandwidth B into discrete messages, each occupying the bandwidth b of sub-channels **202**, a very wideband signal that suffers from relatively minor multipath effects is created.

Before discussing further features and advantages of using a wideband communication channel segmented into a plurality of sub-channels as described, certain aspects of the sub-channels will be explained in more detail. Referring back to FIG. **2**, the overall bandwidth B is segmented into N sub-channels center at frequencies f_o to f_{N-1}. Thus, the sub-channel **202** that is immediately to the right of fc is offset from fc by b/2, where b is the bandwidth of each sub-channel **202**. The next sub-channel **202** is offset by 3b/2, the next by 5b/2, and so on. To the left of fc, each sub-channel **202** is offset by −b/2, −3b/2, −5b/2, etc.

Preferably, sub-channels **202** are non-overlapping as this allows each sub-channel to be processed independently in the receiver. To accomplish this, a roll-off factor is preferably applied to the signals in each sub-channel in a pulse-shaping step. The effect of such a pulse-shaping step is illustrated in FIG. **2** by the non-rectangular shape of the pulses in each sub-channel **202**. Thus, the bandwidth b of each sub-channel can be represented by an equation such as the following:

$$b=(1+r)/T; \tag{2}$$

Where r=the roll-off factor; and

T=the symbol duration.

Without the roll-off factor, i.e., b=1/T, the pulse shape would be rectangular in the frequency domain, which corresponds to a (sin x)/x function in the time domain. The time domain signal for a (sin x)/x signal **400** is shown in FIG. **3** in order to illustrate the problems associated with a rectangular pulse shape and the need to use a roll-off factor.

As can be seen, main lobe **402** comprises almost all of signal **400**. But some of the signal also resides in side lobes **404**, which stretch out indefinitely in both directions from main lobe **402**. Side lobes **404** make processing signal **400** much more difficult, which increases the complexity of the receiver. Applying a roll-off factor r, as in equation (2), causes signal **400** to decay faster, reducing the number of side lobes **404**. Thus, increasing the roll-off factor decreases the length of signal **400**, i.e., signal **400** becomes shorter in time. But including the roll-off factor also decreases the available bandwidth in each sub-channel **202**. Therefore, r must be selected so as to reduce the number of side lobes **404** to a sufficient number, e.g., 15, while still maximizing the available bandwidth in each sub-channel **202**.

Thus, the overall bandwidth B for communication channel **200** is given by the following equation:

$$B=N(1+r)/T, \tag{3}$$

or

$$B=M/T; \tag{4}$$

Where

$$M=(1+r)N. \tag{5}$$

For efficiency purposes related to transmitter design, it is preferable that r is chosen so that M in equation (5) is an integer. Choosing r so that M is an integer allows for more efficient transmitters designs using, for example, Inverse Fast Fourier Transform (IFFT) techniques. Since M=N+N(r), and N is always an integer, this means that r must be chosen so that N(r) is an integer. Generally, it is preferable for r to be between 0.1 and 0.5. Therefore, if N is 16, for example, then 0.5 could be selected for r so that N(r) is an integer. Alternatively, if a value for r is chosen in the above example so that N(r) is not an integer, B can be made slightly wider than M/T to compensate. In this case, it is still preferable that r be chosen so that N(r) is approximately an integer.

2. Example Embodiment of a Wireless Communication System

With the above in mind, FIG. **5** illustrates an example communication system **600** comprising a plurality of cells **602** that each use a common wideband communication channel to communicate with communication devices **604** within each cell **602**. The common communication channel is a wideband communication channel as described above. Each communication cell **602** is defined as the coverage area of a base station, or service access point, **606** within the cell. One such base station **606** is shown for illustration in FIG. **5**. For purposes of this specification and the claims that follow, the term base station will be used generically to refer to a device that provides wireless access to the wireless communication system for a plurality of communication devices, whether the system is a line of sight, indoor, or outdoor system.

Because each cell **602** uses the same communication channel, signals in one cell **602** must be distinguishable from signals in adjacent cells **602**. To differentiate signals from one cell **602** to another, adjacent base stations **606** use different synchronization codes according to a code reuse plan. In FIG. **6**, system **600** uses a synchronization code reuse factor of 4, although the reuse factor can vary depending on the application.

Preferably, the synchronization code is periodically inserted into a communication from a base station **606** to a communication device **604** as illustrated in FIG. **6**. After a predetermined number of data packets **702**, in this case two, the particular synchronization code **704** is inserted into the information being transmitted by each base station **606**. A synchronization code is a sequence of data bits known to both the base station **606** and any communication devices **604** with which it is communicating. The synchronization code allows such a communication device **604** to synchronize its timing to that of base station **606**, which, in turn, allows device **604** to decode the data properly. Thus, in cell **1** (see lightly shaded cells **602** in FIG. **6**), for example, synchronization code **1** (SYNC **1**) is inserted into data stream **706**, which is generated by base station **606** in cell **1**, after every two packets **702**; in cell **2** SYNC**2** is inserted after every two packets **702**; in cell **3** SYNC**3** is inserted; and in cell **4** SYNC**4** is inserted. Use of the synchronization codes is discussed in more detail below.

In FIG. **4**A, an example wideband communication channel **500** for use in communication system **600** is divided into 16 sub-channels **502**, centered at frequencies fo to f_{15}. A base station **606** at the center of each communication cell **602** transmits a single packet occupying the whole bandwidth B of wideband channel **500**. Such a packet is illustrated by packet **504** in FIG. **4**B. Packet **504** comprises sub-packets **506** that are encoded with a frequency offset corresponding to one of sub-channels **502**. Sub-packets **506** in effect define available time slots in packet **504**. Similarly, sub-channels **502** can be said to define available frequency bins in communication channel **500**. Therefore, the resources available in communication cell **602** are time slots **506** and frequency bins **502**, which can be assigned to different communication devices **604** within each cell **602**.

Thus, for example, frequency bins **502** and time slots **506** can be assigned to 4 different communication devices **604** within a cell **602** as shown in FIG. **5**. Each communication device **604** receives the entire packet **504**, but only processes those frequency bins **502** and/or timeslots **506** that are assigned to it. Preferably, each device **604** is assigned non-adjacent frequency bins **502**, as in FIG. **4**A. This way, if interference corrupts the information in a portion of communication channel **500**, then the effects are spread across all devices **604** within a cell **602**. Hopefully, by spreading out the effects of interference in this manner the effects are minimized and the entire information sent to each device **604** can still be recreated from the unaffected information received in other frequency bins. For example, if interference, such as fading, corrupted the information in bins f_o-f_4, then each user **1**-**4** loses one packet of data. But each user potentially receives three unaffected packets from the other bins assigned to them. Hopefully, the unaffected data in the other three bins provides enough information to recreate the entire message for each user. Thus, frequency diversity can be achieved by assigning non-adjacent bins to each of multiple users.

Ensuring that the bins assigned to one user are separated by more than the coherence bandwidth ensures frequency diversity. As discussed above, the coherence bandwidth is approximately equal to $1/d_s$. For outdoor systems, where ds is typically 1 microsecond, $1/d_s$=1/1 microsecond=1 Mega Hertz (MHz). Thus, the non-adjacent frequency bands assigned to a user are preferably separated by at least 1 MHz. It is even more preferable, however, if the coherence bandwidth plus some guard band to ensure sufficient frequency diversity separate the non-adjacent bins assigned to each user. For example, it is preferable in certain implementations to ensure that at least 5 times the coherence bandwidth, or 5 MHz in the above example, separates the non-adjacent bins.

Another way to provide frequency diversity is to repeat blocks of data in frequency bins assigned to a particular user that are separated by more than the coherence bandwidth. In other words, if 4 sub-channels **202** are assigned to a user, then data block a can be repeated in the first and third sub-channels **202** and data block b can be repeated in the second and fourth sub-channels **202**, provided the sub-channels are sufficiently separated in frequency. In this case, the system can be said to be using a diversity length factor of 2. The system can similarly be configured to implement other diversity lengths, e.g., 3, 4, . . . , l.

It should be noted that spatial diversity can also be included depending on the embodiment. Spatial diversity can comprise transmit spatial diversity, receive spatial diversity, or both. In transmit spatial diversity, the transmitter uses a plurality of separate transmitters and a plurality of separate antennas to transmit each message. In other words, each transmitter transmits the same message in parallel. The messages are then received from the transmitters and combined in the receiver. Because the parallel transmissions travel different paths, if one is affected by fading, the others will likely not be affected. Thus, when they are combined in the receiver, the message should be recoverable even if one or more of the other transmission paths experienced severe fading.

Receive spatial diversity uses a plurality of separate receivers and a plurality of separate antennas to receive a single message. If an adequate distance separates the antennas, then the transmission path for the signals received by the antennas will be different. Again, this difference in the transmission paths will provide imperviousness to fading when the signals from the receivers are combined.

Transmit and receive spatial diversity can also be combined within a system such as system **600** so that two antennas are used to transmit and two antennas are used to receive. Thus, each base station **606** transmitter can include two antennas, for transmit spatial diversity, and each communication device **604** receiver can include two antennas, for receive spatial diversity. If only transmit spatial diversity is implemented in system **600**, then it can be implemented in base stations **606** or in communication devices **604**. Similarly, if only receive spatial diversity is included in system **600**, then it can be implemented in base stations **606** or communication devices **604**.

The number of communication devices **604** assigned frequency bins **502** and/or time slots **506** in each cell **602** is preferably programmable in real time. In other words, the resource allocation within a communication cell **602** is preferably programmable in the face of varying external conditions, i.e., multipath or adjacent cell interference, and varying requirements, i.e., bandwidth requirements for various users within the cell. Thus, if user **1** requires the whole bandwidth to download a large video file, for example, then the allocation of bins **502** can be adjust to provide user **1** with more, or even all, of bins **502**. Once user **1** no longer requires such large amounts of bandwidth, the allocation of bins **502** can be readjusted among all of users **1**-**4**.

It should also be noted that all of the bins assigned to a particular user can be used for both the forward and reverse link. Alternatively, some bins **502** can be assigned as the forward link and some can be assigned for use on the reverse link, depending on the implementation.

To increase capacity, the entire bandwidth B is preferably reused in each communication cell **602**, with each cell **602** being differentiated by a unique synchronization code (see discussion below). Thus, system **600** provides increased immunity to multipath and fading as well as increased bandwidth due to the elimination of frequency reuse requirements.

3. Synchronization

FIG. **6** illustrates an example embodiment of a synchronization code correlator **800** (shown in FIG. **7**). When a device **604** in cell **1** (see FIG. **5**), for example, receives an incoming communication from the cell **1** base station **606**, it compares the incoming data with SYNC**1** in correlator **800**. Essentially, the device scans the incoming data trying to correlate the data with the known synchronization code, in this case SYNC**1**. Once correlator **800** matches the incoming data to SYNC**1** it generates a correlation peak **804** at the output. Multipath versions of the data will also generate correlation peaks **806**, although these peaks **806** are generally smaller than correlation peak **804**. The device can then use the correlation peaks to perform channel estimation, which allows the device to adjust for the multipath using an equalizer. Thus, in cell **1**, if correlator **800** receives a data stream comprising SYNC**1**, it will generate correlation peaks **804** and **806**. If, on the other hand, the data stream comprises SYNC**2**, for example, then no peaks will be generated and the device will essentially ignore the incoming communication.

Even though a data stream that comprises SYNC**2** will not create any correlation peaks, it can create noise in correlator **800** that can prevent detection of correlation peaks **804** and **806**. Several steps can be taken to prevent this from occurring. One way to minimize the noise created in correlator **800** by signals from adjacent cells **602**, is to configure system **600** so that each base station **606** transmits at the same time. This way, the synchronization codes can preferably be generated in such a manner that only the synchronization codes **704** of adjacent cell data streams, e.g., streams **708**, **710**, and **712**, as opposed to packets **702** within those streams, will interfere

with detection of the correct synchronization code **704**, e.g., SYNC**1**. The synchronization codes can then be further configured to eliminate or reduce the interference.

For example, the noise or interference caused by an incorrect synchronization code is a function of the cross correlation of that synchronization code with respect to the correct code. The better the cross correlation between the two, the lower the noise level. When the cross correlation is ideal, then the noise level will be virtually zero as illustrated in FIG. **8** by noise level **902**. Therefore, a preferred embodiment of system **600** uses synchronization codes that exhibit ideal cross correlation, i.e., zero. Preferably, the ideal cross correlation of the synchronization codes covers a period l that is sufficient to allow accurate detection of multipath **906** as well as multipath correlation peaks **904**. This is important so that accurate channel estimation and equalization can take place. Outside of period l, the noise level **908** goes up, because the data in packets **702** is random and will exhibit low cross correlation with the synchronization code, e.g., SYNC**1**. Preferably, period l is actually slightly longer then the multipath length in order to ensure that the multipath can be detected.

a. Synchronization Code Generation

Conventional systems use orthogonal codes to achieve cross correlation in correlator **800**. In system **600** for example, SYNC**1**, SYNC**2**, SYNC**3**, and SYNC**4**, corresponding to cells **1-4** (see lightly shaded cells **602** of FIG. **5**) respectively, will all need to be generated in such a manner that they will have ideal cross correlation with each other. In one embodiment, if the data streams involved comprise high and low data bits, then the value "1" can be assigned to the high data bits and "−1" to the low data bits. Orthogonal data sequences are then those that produce a "0" output when they are exclusively ORed (XORed) together in correlator **800**. The following example illustrates this point for orthogonal sequences 1 and 2:

```
sequence 1:  1  1 −1  1
sequence 2:  1  1  1 −1
             ----------
             11 −1 −1 = 0
```

Thus, when the results of XORing each bit pair are added, the result is "0".

But in system **600**, for example, each code must have ideal, or zero, cross correlation with each of the other codes used in adjacent cells **602**. Therefore, in one example embodiment of a method for generating synchronization codes exhibiting the properties described above, the process begins by selecting a "perfect sequence" to be used as the basis for the codes. A perfect sequence is one that when correlated with itself produces a number equal to the number of bits in the sequence. For example:

```
Perfect sequence 1:  11 −11
                     11 −11
                     ------
                     1 1 1 1 = 4
```

But each time a perfect sequence is cyclically shifted by one bit, the new sequence is orthogonal with the original sequence. Thus, for example, if perfect sequence 1 is cyclically shifted by one bit and then correlated with the original, the correlation produces a "0" as in the following example;

```
Perfect sequence 1:  1 1 −1 1
                     1 1 1 −1
                     ---------
                     11 −1 −1 = 0
```

If the perfect sequence 1 is again cyclically shifted by one bit, and again correlated with the original, then it will produce a "0". In general, you can cyclically shift a perfect sequence by any number of bits up to its length and correlate the shifted sequence with the original to obtain a "0".

Once a perfect sequence of the correct length is selected, the first synchronization code is preferably generated in one embodiment by repeating the sequence 4 times. Thus, if perfect sequence 1 is being used, then a first synchronization code y would be the following:

$$y=1\ 1\ {-1}\ 1\ 1\ 1\ {-1}\ 1\ 1\ 1\ {-1}\ 1\ 1\ 1\ {-1}\ 1\ 1.$$

Or in generic form:

$$y=x(0)x(1)x(2)x(3)x(0)x(1)x(2)x(3)x(0)x(1)x(2)x(3)x(0)x(1)x(2)x(3).$$

For a sequence of length L:

$$y=x(0)x(1)\ldots x(L)x(0)x(1)\ldots x(L)x(0)x(1)\ldots x(L)x(0)x(1)\ldots x(L)$$

Repeating the perfect sequence allows correlator **800** a better opportunity to detect the synchronization code and allows generation of other uncorrelated frequencies as well. Repeating has the effect of sampling in the frequency domain. This effect is illustrated by the graphs in FIG. **9**. Thus, in TRACE 1, which corresponds to synchronization code y, a sample **1002** is generated every fourth sample bin **1000**. Each sample bin is separated by 1/(4L×T), where T is the symbol duration. Thus, in the above example, where L=4, each sample bin is separated by 1/(16×T) in the frequency domain. TRACES 2-4 illustrate the next three synchronization codes. As can be seen, the samples for each subsequent synchronization code are shifted by one sample bin relative to the samples for the previous sequence. Therefore, none of the sequences interfere with each other.

To generate the subsequent sequences, corresponding to TRACES 2-4, sequence y must be shifted in frequency. This can be accomplished using the following equation:

$$z^r(m)=y(m)*\exp(j*2*\pi*r*m/(n*L)), \tag{5}$$

for r=1 to L (# of sequences) and m=0 to 4*L−1 (time); and where:

$z^r(m)$=each subsequent sequence;

y(m)=the first sequence; and

n=the number of times the sequence is repeated.

It will be understood that multiplying by an exp(j2π(r*m/N)) factor, where N is equal to the number of times the sequence is repeated n multiplied by the length of the underlying perfect sequence L, in the time domain results in a shift in the frequency domain. Equation (5) results in the desired shift as illustrated in FIG. **9** for each of synchronization codes **2-4**, relative to synchronization code **1**. The final step in generating each synchronization code is to append the copies of the last M samples, where M is the length of the multipath, to the front of each code. This is done to make the convolution with the multipath cyclic and to allow easier detection of the multipath.

It should be noted that synchronization codes can be generated from more than one perfect sequence using the same methodology. For example, a perfect sequence can be generated and repeated four times and then a second perfect sequence can be generated and repeated four times to get a n

factor equal to eight. The resulting sequence can then be shifted as described above to create the synchronization codes.

b. Signal Measurements Using Synchronization Codes

Therefore, when a communication device is at the edge of a cell, it will receive signals from multiple base stations and, therefore, will be decoding several synchronization codes at the same time. This can be illustrated with the help of FIG. **10**, which illustrates another example embodiment of a wireless communication system **1100** comprising communication cells **1102**, **1104**, and **1106** as well as communication device **1108**, which is in communication with base station **1110** of cell **1102** but also receiving communication from base stations **1112** and **1114** of cells **1104** and **1106**, respectively.

If communications from base station **1110** comprise synchronization code SYNC**1** and communications from base station **1112** and **1114** comprise SYNC**2** and SYNC**3** respectively, then device **1108** will effectively receive the sum of these three synchronization codes. This is because, as explained above, base stations **1110**, **1112**, and **1114** are configured to transmit at the same time. Also, the synchronization codes arrive at device **1108** at almost the same time because they are generated in accordance with the description above.

Again as described above, the synchronization codes SYNC**1**, SYNC**2**, and SYNC**3** exhibit ideal cross correlation. Therefore, when device **1108** correlates the sum x of codes SYNC**1**, SYNC**2**, and SYNC**3**, the latter two will not interfere with proper detection of SYNC**1** by device **1108**. Importantly, the sum x can also be used to determine important signal characteristics, because the sum x is equal to the sum of the synchronization code signal in accordance with the following equation:

$$x=SYNC1+SYNC2+SYNC3. \quad (6)$$

Therefore, when SYNC**1** is removed, the sum of SYNC**2** and SYNC**3** is left, as shown in the following:

$$x-SYNC1=SYNC2+SYNC3. \quad (7)$$

The energy computed from the sum (SYNC**2**+SYNC**3**) is equal to the noise or interference seen by device **1108**. Since the purpose of correlating the synchronization code in device **1106** is to extract the energy in SYNC **1**, device **1108** also has the energy in the signal from base station **1110**, i.e., the energy represented by SYNC**1**. Therefore, device **1106** can use the energy of SYNC**1** and of (SYNC**2**+SYNC**3**) to perform a signal-to-interference measurement for the communication channel over which it is communicating with base station **1110**. The result of the measurement is preferably a signal-to-interference ratio (SIR). The SIR measurement can then be communicated back to base station **1110** for purposes that will be discussed below.

The ideal cross correlation of the synchronization codes, also allows device **1108** to perform extremely accurate determinations of the Channel Impulse Response (CIR), or channel estimation, from the correlation produced by correlator **800**. This allows for highly accurate equalization using low cost, low complexity equalizers, thus overcoming a significant draw back of conventional systems.

4. Sub-Channel Assignments

As mentioned, the SIR as determined by device **1108** can be communicated back to base station **1110** for use in the assignment of channels **502**. In one embodiment, due to the fact that each sub-channel **502** is processed independently, the SIR for each sub-channel **502** can be measured and communicated back to base station **1110**. In such an embodiment, therefore, sub-channels **502** can be divided into groups and a SIR measurement for each group can be sent to base station **1110**. This is illustrated in FIG. **11**A, which shows a wide-band communication channel **1200** segmented into sub-channels fo to f_{15}. Sub-channels fo to f_{15} are then grouped into 8 groups G**1** to G**8**. Thus, in one embodiment, device **1108** and base station **1110** communicate over a channel such as channel **1200**.

Sub-channels in the same group are preferably separated by as many sub-channels as possible to ensure diversity. In FIG. **11**A for example, sub-channels within the same group are 7 sub-channels apart, e.g., group G**1** comprises f_0 and f_8. Device **1102** reports a SIR measurement for each of the groups G**1** to G**8**. These SIR measurements are preferably compared with a threshold value to determine which sub-channels groups are useable by device **1108**. This comparison can occur in device **1108** or base station **1110**. If it occurs in device **1108**, then device **1108** can simply report to base station **1110** which sub-channels groups are useable by device **1108**.

SIR reporting will be simultaneously occurring for a plurality of devices within cell **1102**. Thus, FIG. **11**B illustrates the situation where two communication devices corresponding to User **1** and User **2** report SIR levels above the threshold for groups G**1**, G**3**, G**5**, and G**7**. Base station **1110** preferably then assigns sub-channel groups to User **1** and User **2** based on the SIR reporting as illustrated in FIG. **11**B. When assigning the "good" sub-channel groups to User **1** and User **2**, base station **1110** also preferably assigns them based on the principles of frequency diversity. In FIG. **11**B, therefore, User **1** and User **2** are alternately assigned every other "good" sub-channel.

The assignment of sub-channels in the frequency domain is equivalent to the assignment of time slots in the time domain. Therefore, as illustrated in FIG. **12**, two users, User **1** and User **2**, receive packet **1302** transmitted over communication channel **1200**. FIG. **12** also illustrated the sub-channel assignment of FIG. **11**B. While FIGS. **11** and **12** illustrate sub-channel/time slot assignment based on SIR for two users, the principles illustrated can be extended for any number of users. Thus, a packet within cell **1102** can be received by 3 or more users. Although, as the number of available sub-channels is reduced due to high SIR, so is the available bandwidth. In other words, as available channels are reduced, the number of users that can gain access to communication channel **1200** is also reduced.

Poor SIR can be caused for a variety of reasons, but frequently it results from a device at the edge of a cell receiving communication signals from adjacent cells. Because each cell is using the same bandwidth B, the adjacent cell signals will eventually raise the noise level and degrade SIR for certain sub-channels. In certain embodiments, therefore, sub-channel assignment can be coordinated between cells, such as cells **1102**, **1104**, and **1106** in FIG. **10**, in order to prevent interference from adjacent cells.

Thus, if communication device **1108** is near the edge of cell **1102**, and device **1118** is near the edge of cell **1106**, then the two can interfere with each other. As a result, the SIR measurements that device **1108** and **1118** report back to base stations **1110** and **1114**, respectively, will indicate that the interference level is too high. Base station **1110** can then be configured to assign only the odd groups, i.e., G**1**, G**3**, G**5**, etc., to device **1108**, while base station **1114** can be configured to assign the even groups to device **1118**. The two devices **1108** and **1118** will then not interfere with each other due to the coordinated assignment of sub-channel groups.

Assigning the sub-channels in this manner reduces the overall bandwidth available to devices **1108** and **1118**, respectively. In this case the bandwidth is reduced by a factor of two. But it should be remembered that devices operating closer to each base station **1110** and **1114**, respectively, will still be able to use all channels if needed. Thus, it is only devices, such as device **1108**, that are near the edge of a cell that will have the available bandwidth reduced. Contrast this with a CDMA system, for example, in which the bandwidth for all users is reduced, due to the spreading techniques used in such systems, by approximately a factor of 10 at all times. It can be seen, therefore, that the systems and methods for wireless communication over a wide bandwidth channel using a plurality of sub-channels not only improves the quality of service, but can also increase the available bandwidth significantly.

When there are three devices **1108**, **1118**, and **1116** near the edge of their respective adjacent cells **1102**, **1104**, and **1106**, the sub-channels can be divided by three. Thus, device **1108**, for example, can be assigned groups G**1**, G**4**, etc., device **1118** can be assigned groups G**2**, G**5**, etc., and device **1116** can be assigned groups G**3**, G**6**, etc. In this case the available bandwidth for these devices, i.e., devices near the edges of cells **1102**, **1104**, and **1106**, is reduced by a factor of 3, but this is still better than a CDMA system, for example.

The manner in which such a coordinated assignment of sub-channels can work is illustrated by the flow chart in FIG. **13**. First in step **1402**, a communication device, such as device **1108**, reports the SIR for all sub-channel groups G**1** to G**8**. The SIRs reported are then compared, in step **1404**, to a threshold to determine if the SIR is sufficiently low for each group. Alternatively, device **1108** can make the determination and simply report which groups are above or below the SIR threshold. If the SIR levels are good for each group, then base station **1110** can make each group available to device **1108**, in step **1406**. Periodically, device **1108** preferably measures the SIR level and updates base station **1110** in case the SIR as deteriorated. For example, device **1108** may move from near the center of cell **1102** toward the edge, where interference from an adjacent cell may affect the SIR for device **1108**.

If the comparison in step **1404** reveals that the SIR levels are not good, then base station **1110** can be preprogrammed to assign either the odd groups or the even groups only to device **1108**, which it will do in step **1408**. Device **1108** then reports the SIR measurements for the odd or even groups it is assigned in step **1410**, and they are again compared to a SIR threshold in step **1412**.

It is assumed that the poor SIR level is due to the fact that device **1108** is operating at the edge of cell **1102** and is therefore being interfered with by a device such as device **1118**. But device **1108** will be interfering with device **1118** at the same time. Therefore, the assignment of odd or even groups in step **1408** preferably corresponds with the assignment of the opposite groups to device **1118**, by base station **1114**. Accordingly, when device **1108** reports the SIR measurements for whichever groups, odd or even, are assigned to it, the comparison in step **1410** should reveal that the SIR levels are now below the threshold level. Thus, base station **1110** makes the assigned groups available to device **1108** in step **1414**. Again, device **1108** preferably periodically updates the SIR measurements by returning to step **1402**.

It is possible for the comparison of step **1410** to reveal that the SIR levels are still above the threshold, which should indicate that a third device, e.g., device **1116** is still interfering with device **1108**. In this case, base station **1110** can be preprogrammed to assign every third group to device **1108** in step **1416**. This should correspond with the corresponding assignments of non-interfering channels to devices **1118** and **1116** by base stations **1114** and **1112**, respectively. Thus, device **1108** should be able to operate on the sub-channel groups assigned, i.e., G**1**, G**4**, etc., without undue interference. Again, device **1108** preferably periodically updates the SIR measurements by returning to step **1402**. Optionally, a third comparison step (not shown) can be implemented after step **1416**, to ensure that the groups assigned to device **1408** posses an adequate SIR level for proper operation. Moreover, if there are more adjacent cells, i.e., if it is possible for devices in a 4th or even a 5th adjacent cell to interfere with device **1108**, then the process of FIG. **13** would continue and the sub-channel groups would be divided even further to ensure adequate SIR levels on the sub-channels assigned to device **1108**.

Even though the process of FIG. **13** reduces the bandwidth available to devices at the edge of cells **1102**, **1104**, and **1106**, the SIR measurements can be used in such a manner as to increase the data rate and therefore restore or even increase bandwidth. To accomplish this, the transmitters and receivers used in base stations **1102**, **1104**, and **1106**, and in devices in communication therewith, e.g., devices **1108**, **1114**, and **1116** respectively, must be capable of dynamically changing the symbol mapping schemes used for some or all of the sub-channel. For example, in some embodiments, the symbol mapping scheme can be dynamically changed among BPSK, QPSK, 8PSK, 16QAM, 32QAM, etc. As the symbol mapping scheme moves higher, i.e., toward 32QAM, the SIR level required for proper operation moves higher, i.e., less and less interference can be withstood. Therefore, once the SIR levels are determined for each group, the base station, e.g., base station **1110**, can then determine what symbol mapping scheme can be supported for each sub-channel group and can change the modulation scheme accordingly. Device **1108** must also change the symbol mapping scheme to correspond to that of the base stations. The change can be effected for all groups uniformly, or it can be effected for individual groups. Moreover, the symbol mapping scheme can be changed on just the forward link, just the reverse link, or both, depending on the embodiment.

Thus, by maintaining the capability to dynamically assign sub-channels and to dynamically change the symbol mapping scheme used for assigned sub-channels, the systems and methods described herein provide the ability to maintain higher available bandwidths with higher performance levels than conventional systems. To fully realize the benefits described, however, the systems and methods described thus far must be capable of implementation in a cost effect and convenient manner. Moreover, the implementation must include reconfigurability so that a single device can move between different types of communication systems and still maintain optimum performance in accordance with the systems and methods described herein. The following descriptions detail example high level embodiments of hardware implementations configured to operate in accordance with the systems and methods described herein in such a manner as to provide the capability just described above.

5. Sample Transmitter Embodiments

FIG. **14** is logical block diagram illustrating an example embodiment of a transmitter **1500** configured for wireless communication in accordance with the systems and methods described above. The transmitter could, for example be within a base station, e.g., base station **606**, or within a communication device, such as device **604**. Transmitter **1500** is provided to illustrate logical components that can be included

in a transmitter configured in accordance with the systems and methods described herein. It is not intended to limit the systems and methods for wireless communication over a wide bandwidth channel using a plurality of sub-channels to any particular transmitter configuration or any particular wireless communication system.

With this in mind, it can be seen that transmitter **1500** comprises a serial-to-parallel converter **1504** configured to receive a serial data stream **1502** comprising a data rate R. Serial-to-parallel converter **1504** converts data stream **1502** into N parallel data streams **1504**, where N is the number of sub-channels **202**. It should be noted that while the discussion that follows assumes that a single serial data stream is used, more than one serial data stream can also be used if required or desired. In any case, the data rate of each parallel data stream **1504** is then R/N. Each data stream **1504** is then sent to a scrambler, encoder, and interleaver block **1506**. Scrambling, encoding, and interleaving are common techniques implemented in many wireless communication transmitters and help to provide robust, secure communication. Examples of these techniques will be briefly explained for illustrative purposes.

Scrambling breaks up the data to be transmitted in an effort to smooth out the spectral density of the transmitted data. For example, if the data comprises a long string of "1"s, there will be a spike in the spectral density. This spike can cause greater interference within the wireless communication system. By breaking up the data, the spectral density can be smoothed out to avoid any such peaks. Often, scrambling is achieved by XORing the data with a random sequence.

Encoding, or coding, the parallel bit streams **1504** can, for example, provide Forward Error Correction (FEC). The purpose of FEC is to improve the capacity of a communication channel by adding some carefully designed redundant information to the data being transmitted through the channel. The process of adding this redundant information is known as channel coding. Convolutional coding and block coding are the two major forms of channel coding. Convolutional codes operate on serial data, one or a few bits at a time. Block codes operate on relatively large (typically, up to a couple of hundred bytes) message blocks. There are a variety of useful convolutional and block codes, and a variety of algorithms for decoding the received coded information sequences to recover the original data. For example, convolutional encoding or turbo coding with Viterbi decoding is a FEC technique that is particularly suited to a channel in which the transmitted signal is corrupted mainly by additive white gaussian noise (AWGN) or even a channel that simply experiences fading.

Convolutional codes are usually described using two parameters: the code rate and the constraint length. The code rate, k/n, is expressed as a ratio of the number of bits into the convolutional encoder (k) to the number of channel symbols (n) output by the convolutional encoder in a given encoder cycle. A common code rate is ½, which means that 2 symbols are produced for every 1-bit input into the coder. The constraint length parameter, K, denotes the "length" of the convolutional encoder, i.e. how many k-bit stages are available to feed the combinatorial logic that produces the output symbols. Closely related to K is the parameter m, which indicates how many encoder cycles an input bit is retained and used for encoding after it first appears at the input to the convolutional encoder. The m parameter can be thought of as the memory length of the encoder.

Interleaving is used to reduce the effects of fading. Interleaving mixes up the order of the data so that if a fade interferes with a portion of the transmitted signal, the overall message will not be effected. This is because once the message is de-interleaved and decoded in the receiver, the data lost will comprise non-contiguous portions of the overall message. In other words, the fade will interfere with a contiguous portion of the interleaved message, but when the message is de-interleaved, the interfered with portion is spread throughout the overall message. Using techniques such as FEC, the missing information can then be filled in, or the impact of the lost data may just be negligible.

After blocks **1506**, each parallel data stream **1504** is sent to symbol mappers **1508**. Symbol mappers **1508** apply the requisite symbol mapping, e.g., BPSK, QPSK, etc., to each parallel data stream **1504**. Symbol mappers **1508** are preferably programmable so that the modulation applied to parallel data streams can be changed, for example, in response to the SIR reported for each sub-channel **202**. It is also preferable, that each symbol mapper **1508** be separately programmable so that the optimum symbol mapping scheme for each sub-channel can be selected and applied to each parallel data stream **1504**.

After symbol mappers **1508**, parallel data streams **1504** are sent to modulators **1510**. Important aspects and features of example embodiments of modulators **1510** are described below. After modulators **1510**, parallel data streams **1504** are sent to summer **1512**, which is configured to sum the parallel data streams and thereby generate a single serial data stream **1518** comprising each of the individually processed parallel data streams **1504**. Serial data stream **1518** is then sent to radio module **1512**, where it is modulated with an RF carrier, amplified, and transmitted via antenna **1516** according to known techniques.

The transmitted signal occupies the entire bandwidth B of communication channel **100** and comprises each of the discrete parallel data streams **1504** encoded onto their respective sub-channels **102** within bandwidth B. Encoding parallel data streams **1504** onto the appropriate sub-channels **102** requires that each parallel data stream **1504** be shifted in frequency by an appropriate offset. This is achieved in modulator **1510**.

FIG. **15** is a logical block diagram of an example embodiment of a modulator **1600** in accordance with the systems and methods described herein. Importantly, modulator **1600** takes parallel data streams **1602** performs Time Division Modulation (TDM) or Frequency Division Modulation (FDM) on each data stream **1602**, filters them using filters **1612**, and then shifts each data stream in frequency using frequency shifter **1614** so that they occupy the appropriate sub-channel. Filters **1612** apply the required pulse shaping, i.e., they apply the roll-off factor described in section 1. The frequency shifted parallel data streams **1602** are then summed and transmitted. Modulator **1600** can also include rate controller **1604**, frequency encoder **1606**, and interpolators **1610**. All of the components shown in FIG. **15** are described in more detail in the following paragraphs and in conjunction with FIGS. **16-22**.

FIG. **16** illustrates one example embodiment of a rate controller **1700** in accordance with the systems and methods described herein. Rate control **1700** is used to control the data rate of each parallel data stream **1602**. In rate controller **1700**, the data rate is halved by repeating data streams d(**0**) to d(**7**), for example, producing streams a(**0**) to a(**15**) in which a(**0**) is the same as a(**8**), a(**1**) is the same as a(**9**), etc. FIG. **16** also illustrates that the effect of repeating the data streams in this manner is to take the data streams that are encoded onto the first 8 sub-channels **1702**, and duplicate them on the next 8 sub-channels **1702**. As can be seen, 7 sub-channels separate sub-channels **1702** comprising the same, or duplicate, data streams. Thus, if fading effects one sub-channel **1702**, for example, the other sub-channels **1702** carrying the same data

will likely not be effected, i.e., there is frequency diversity between the duplicate data streams. So by sacrificing data rate, in this case half the data rate, more robust transmission is achieved. Moreover, the robustness provided by duplicating the data streams d(**0**) to d(**7**) can be further enhanced by applying scrambling to the duplicated data streams via scramblers **1708**.

It should be noted that the data rate can be reduced by more than half, e.g., by four or more. Alternatively, the data rate can also be reduced by an amount other than half. For example if information from n data stream is encoded onto m sub-channels, where m>n. Thus, to decrease the rate by ⅔, information from one data stream can be encoded on a first sub-channel, information from a second data stream can be encoded on a second data channel, and the sum or difference of the two data streams can be encoded on a third channel. In which case, proper scaling will need to be applied to the power in the third channel. Otherwise, for example, the power in the third channel can be twice the power in the first two.

Preferably, rate controller **1700** is programmable so that the data rate can be changed responsive to certain operational factors. For example, if the SIR reported for sub-channels **1702** is low, then rate controller **1700** can be programmed to provide more robust transmission via repetition to ensure that no data is lost due to interference. Additionally, different types of wireless communication system, e.g., indoor, outdoor, line-of-sight, may require varying degrees of robustness. Thus, rate controller **1700** can be adjusted to provide the minimum required robustness for the particular type of communication system. This type of programmability not only ensures robust communication, it can also be used to allow a single device to move between communication systems and maintain superior performance.

FIG. **17** illustrates an alternative example embodiment of a rate controller **1800** in accordance with the systems and methods described. In rate controller **1800** the data rate is increased instead of decreased. This is accomplished using serial-to-parallel converters **1802** to convert each data streams d(**0**) to d(**15**), for example, into two data streams. Delay circuits **1804** then delay one of the two data streams generated by each serial-to-parallel converter **1802** by ½ a symbol. Thus, data streams d(**0**) to d(**15**) are transformed into data streams a(**0**) to a(**31**). The data streams generated by a particular serial-to-parallel converter **1802** and associate delay circuit **1804** must then be summed and encoded onto the appropriate sub-channel. For example, data streams a(**0**) and a(**1**) must be summed and encoded onto the first sub-channel. Preferably, the data streams are summed subsequent to each data stream being pulsed shaped by a filter **1612**.

Thus, rate controller **1604** is preferably programmable so that the data rate can be increased, as in rate controller **1800**, or decreased, as in rate controller **1700**, as required by a particular type of wireless communication system, or as required by the communication channel conditions or sub-channel conditions. In the event that the data rate is increased, filters **1612** are also preferably programmable so that they can be configured to apply pulse shapping to data streams a(**0**) to a(**31**), for example, and then sum the appropriate streams to generate the appropriate number of parallel data streams to send to frequency shifter **1614**.

The advantage of increasing the data rate in the manner illustrated in FIG. **17** is that higher symbol mapping rates can essentially be achieved, without changing the symbol mapping used in symbol mappers **1508**. Once the data streams are summed, the summed streams are shifted in frequency so that they reside in the appropriate sub-channel. But because the number of bits per each symbol has been doubled, the symbol mapping rate has been doubled. Thus, for example, a 4QAM symbol mapping can be converted to a 16QAM symbol mapping, even if the SIR is too high for 16QAM symbol mapping to otherwise be applied. In other words, programming rate controller **1800** to increase the data rate in the manner illustrated in FIG. **17** can increase the symbol mapping even when channel conditions would otherwise not allow it, which in turn can allow a communication device to maintain adequate or even superior performance regardless of the type of communication system.

The draw back to increasing the data rate as illustrated in FIG. **17** is that interference is increased, as is receiver complexity. The former is due to the increased amount of data. The latter is due to the fact that each symbol cannot be processed independently because of the ½ symbol overlap. Thus, these concerns must be balanced against the increase symbol mapping ability when implementing a rate controller such as rate controller **1800**.

FIG. **18** illustrates one example embodiment of a frequency encoder **1900** in accordance with the systems and methods described herein. Similar to rate encoding, frequency encoding is preferably used to provide increased communication robustness. In frequency encoder **1900** the sum or difference of multiple data streams are encoded onto each sub-channel. This is accomplished using adders **1902** to sum data streams d(**0**) to d(**7**) with data streams d(**8**) to d(**15**), respectively, while adders **1904** subtract data streams d(**0**) to d(**7**) from data streams d(**8**) to d(**15**), respectively, as shown. Thus, data streams a(**0**) to a(**15**) generated by adders **1902** and **1904** comprise information related to more than one data streams d(**0**) to d(**15**). For example, a(**0**) comprises the sum of d(**0**) and d(**8**), i.e., d(**0**)+d(**8**), while a(**8**) comprises d(**8**)−d(**0**). Therefore, if either a(**0**) or a(**8**) is not received due to fading, for example, then both of data streams d(**0**) and d(**8**) can still be retrieved from data stream a(**8**).

Essentially, the relationship between data stream d(**0**) to d(**15**) and a(**0**) to a(**15**) is a matrix relationship. Thus, if the receiver knows the correct matrix to apply, it can recover the sums and differences of d(**0**) to d(**15**) from a(**0**) to a(**15**). Preferably, frequency encoder **1900** is programmable, so that it can be enabled and disabled in order to provided robustness when required. Preferable, adders **1902** and **1904** are programmable also so that different matrices can be applied to d(**0**) to d(**15**).

After frequency encoding, if it is included, data streams **1602** are sent to TDM/FDM blocks **1608**. TDM/FDM blocks **1608** perform TDM or FDM on the data streams as required by the particular embodiment. FIG. **19** illustrates an example embodiment of a TDM/FDM block **2000** configured to perform TDM on a data stream. TDM/FDM block **2000** is provided to illustrate the logical components that can be included in a TDM/FDM block configured to perform TDM on a data stream. Depending on the actual implementation, some of the logical components may or may not be included. TDM/FDM block **2000** comprises a sub-block repeater **2002**, a sub-block scrambler **2004**, a sub-block terminator **2006**, a sub-block repeater **2008**, and a sync inserter **2010**.

Sub-block repeater **2002** is configured to receive a sub-block of data, such as block **2012** comprising bits a(**0**) to a(**3**) for example. Sub-block repeater is then configured to repeat block **2012** to provide repetition, which in turn leads to more robust communication. Thus, sub-block repeater **2002** generates block **2014**, which comprises 2 blocks **2012**. Sub-block scrambler **2004** is then configured to receive block **2014** and to scramble it, thus generating block **2016**. One method of scrambling can be to invert half of block **2014** as illustrated in

block **2016**. But other scrambling methods can also be implemented depending on the embodiment.

Sub-block terminator **2006** takes block **2016** generated by sub-block scrambler **2004** and adds a termination block **2034** to the front of block **2016** to form block **2018**. Termination block **2034** ensures that each block can be processed independently in the receiver. Without termination block **2034**, some blocks may be delayed due to multipath, for example, and they would therefore overlap part of the next block of data. But by including termination block **2034**, the delayed block can be prevented from overlapping any of the actual data in the next block.

Termination block **2034** can be a cyclic prefix termination **2036**. A cyclic prefix termination **2036** simply repeats the last few symbols of block **2018**. Thus, for example, if cyclic prefix termination **2036** is three symbols long, then it would simply repeat the last three symbols of block **2018**. Alternatively, termination block **2034** can comprise a sequence of symbols that are known to both the transmitter and receiver. The selection of what type of block termination **2034** to use can impact what type of equalizer is used in the receiver. Therefore, receiver complexity and choice of equalizers must be considered when determining what type of termination block **2034** to use in TDM/FDM block **2000**.

After sub-block terminator **2006**, TDM/FDM block **2000** can include a sub-block repeater **2008** configured to perform a second block repetition step in which block **2018** is repeated to form block **2020**. In certain embodiments, sub-block repeater can be configured to perform a second block scrambling step as well. After sub-block repeater **2008**, if included, TDM/FDM block **2000** comprises a sync inserter **210** configured to periodically insert an appropriate synchronization code **2032** after a predetermined number of blocks **2020** and/or to insert known symbols into each block. The purpose of synchronization code **2032** is discussed in section 3.

FIG. **20**, on the other hand, illustrates an example embodiment of a TDM/FDM block **2100** configured for FDM, which comprises sub-block repeater **2102**, sub-block scrambler **2104**, block coder **2106**, sub-block transformer **2108**, sub-block terminator **2110**, and sync inserter **2112**. As with TDM/FDM block **2000**, sub-block repeater **2102** repeats block **2114** and generates block **2116**. Sub-block scrambler then scrambles block **2116**, generating block **2118**. Sub-block coder **2106** takes block **2118** and codes it, generating block **2120**. Coding block correlates the data symbols together and generates symbols b. This requires joint demodulation in the receiver, which is more robust but also more complex. Sub-block transformer **2108** then performs a transformation on block **2120**, generating block **2122**. Preferably, the transformation is an IFFT of block **2120**, which allows for more efficient equalizers to be used in the receiver. Next, sub-block terminator **2110** terminates block **2122**, generating block **2124** and sync inserter **2112** periodically inserts a synchronization code **2126** after a certain number of blocks **2124** and/or insert known symbols into each block. Preferably, sub-block terminator **2110** only uses cyclic prefix termination as described above. Again this allows for more efficient receiver designs.

TDM/FDM block **2100** is provided to illustrate the logical components that can be included in a TDM/FDM block configured to perform FDM on a data stream. Depending on the actual implementation, some of the logical components may or may not be included. Moreover, TDM/FDM block **2000** and **2100** are preferably programmable so that the appropriate logical components can be included as required by a particular implementation. This allows a device that incorporates one of blocks **2000** or **2100** to move between different systems with different requirements. Further, it is preferable that TDM/FDM block **1608** in FIG. **15** be programmable so that it can be programmed to perform TDM, such as described in conjunction with block **2000**, or FDM, such as described in conjunction with block **2100**, as required by a particular communication system.

After TDM/FDM blocks **1608**, in FIG. **15**, the parallel data streams are preferably passed to interpolators **1610**.

After Interpolators **1610**, the parallel data streams are passed to filters **1612**, which apply the pulse shapping described in conjunction with the roll-off factor of equation (2) in section 1. Then the parallel data streams are sent to frequency shifter **1614**, which is configured to shift each parallel data stream by the frequency offset associated with the sub-channel to which the particular parallel data stream is associated.

FIG. **21** illustrates an example embodiment of a frequency shifter **2200** in accordance with the systems and methods described herein. As can be seen, frequency shifter **2200** comprises multipliers **2202** configured to multiply each parallel data stream by the appropriate exponential to achieve the required frequency shift. Each exponential is of the form: $\exp(j2\pi_c nT/rM)$, where c is the corresponding sub-channel, e.g., c=0 to N−1, and n is time. Preferably, frequency shifter **1614** in FIG. **5** is programmable so that various channel/sub-channel configurations can be accommodated for various different systems. Alternatively, an IFFT block can replace shifter **1614** and filtering can be done after the IFFT block. This type of implementation can be more efficient depending on the implementation.

After the parallel data streams are shifted, they are summed, e.g., in summer **1512** of FIG. **14**. The summed data stream is then transmitted using the entire bandwidth B of the communication channel being used. But the transmitted data stream also comprises each of the parallel data streams shifted in frequency such that they occupy the appropriate sub-channel. Thus, each sub-channel may be assigned to one user, or each sub-channel may carry a data stream intended for different users. The assignment of sub-channels is described in section 3b. Regardless of how the sub-channels are assigned, however, each user will receive the entire bandwidth, comprising all the sub-channels, but will only decode those sub-channels assigned to the user.

6. Sample Receiver Embodiments

FIG. **22** illustrates an example embodiment of a receiver **2300** that can be configured in accordance with the present invention. Receiver **2300** comprises an antenna **2302** configured to receive a message transmitted by a transmitter, such as transmitter **1500**. Thus, antenna **2302** is configured to receive a wide band message comprising the entire bandwidth B of a wide band channel that is divided into sub-channels of bandwidth b. As described above, the wide band message comprises a plurality of messages each encoded onto each of a corresponding sub-channel. All of the sub-channels may or may not be assigned to a device that includes receiver **2300**; Therefore, receiver **2300** may or may not be required to decode all of the sub-channels.

After the message is received by antenna **2300**, it is sent to radio receiver **2304**, which is configured to remove the carrier associated with the wide band communication channel and extract a baseband signal comprising the data stream transmitted by the transmitter. The baseband signal is then sent to correlator **2306** and demodulator **2308**. Correlator **2306** is configured to correlated with a synchronization code inserted in the data stream as described in section 3. It is also prefer-

ably configured to perform SIR and multipath estimations as described in section 3(b). Demodulator **2308** is configured to extract the parallel data streams from each sub-channel assigned to the device comprising receiver **2300** and to generate a single data stream therefrom.

FIG. **23** illustrates an example embodiment of a demodulator **2400** in accordance with the systems and methods described herein. Demodulator **2402** comprises a frequency shifter **2402**, which is configured to apply a frequency offset to the baseband data stream so that parallel data streams comprising the baseband data stream can be independently processed in receiver **2400**. Thus, the output of frequency shifter **2402** is a plurality of parallel data streams, which are then preferably filtered by filters **2404**. Filters **2404** apply a filter to each parallel data stream that corresponds to the pulse shape applied in the transmitter, e.g., transmitter **1500**. Alternatively, an IFFT block can replace shifter **1614** and filtering can be done after the IFFT block. This type of implementation can be more efficient depending on the implementation.

Next, receiver **2400** preferably includes decimators **2406** configured to decimate the data rate of the parallel bit streams. Sampling at higher rates helps to ensure accurate recreation of the data. But the higher the data rate, the larger and more complex equalizer **2408** becomes. Thus, the sampling rate, and therefore the number of samples, can be reduced by decimators **2406** to an adequate level that allows for a smaller and less costly equalizer **2408**.

Equalizer **2408** is configured to reduce the effects of multipath in receiver **2300**. Its operation will be discussed more fully below. After equalizer **2408**, the parallel data streams are sent to de-scrambler, decoder, and de-interleaver **2410**, which perform the opposite operations of scrambler, encoder, and interleaver **1506** so as to reproduce the original data generated in the transmitter. The parallel data streams are then sent to parallel to serial converter **2412**, which generates a single serial data stream from the parallel data streams.

Equalizer **2408** uses the multipath estimates provided by correlator **2306** to equalize the effects of multipath in receiver **2300**. In one embodiment, equalizer **2408** comprises Single-In Single-Out (SISO) equalizers operating on each parallel data stream in demodulator **2400**. In this case, each SISO equalizer comprising equalizer **2408** receives a single input and generates a single equalized output. Alternatively, each equalizer can be a Multiple-In Multiple-Out (MIMO) or a Multiple-In Single-Out (MISO) equalizer. Multiple inputs can be required for example, when a frequency encoder or rate controller, such as frequency encoder **1900**, is included in the transmitter. Because frequency encoder **1900** encodes information from more than one parallel data stream onto each sub-channel, each equalizers comprising equalizer **2408** need to equalize more than one sub-channel. Thus, for example, if a parallel data stream in demodulator **2400** comprises d(**1**)+d(**8**), then equalizer **2408** will need to equalize both d(**1**) and d(**8**) together. Equalizer **2408** can then generate a single output corresponding to d(**1**) or d(**8**) (MISO) or it can generate both d(**1**) and d(**8**) (MIMO).

Equalizer **2408** can also be a time domain equalizer (TDE) or a frequency domain equalizer (FDE) depending on the embodiment. Generally, equalizer **2408** is a TDE if the modulator in the transmitter performs TDM on the parallel data streams, and a FDE if the modulator performs FDM. But equalizer **2408** can be an FDE even if TDM is used in the transmitter. Therefore, the preferred equalizer type should be taken into consideration when deciding what type of block termination to use in the transmitter. Because of power requirements, it is often preferable to use FDM on the forward link and TDM on the reverse link in a wireless communication system.

As with transmitter **1500**, the various components comprising demodulator **2400** are preferably programmable, so that a single device can operate in a plurality of different systems and still maintain superior performance, which is a primary advantage of the systems and methods described herein. Accordingly, the above discussion provides systems and methods for implementing a channel access protocol that allows the transmitter and receiver hardware to be reprogrammed slightly depending on the communication system.

Thus, when a device moves from one system to another, it preferably reconfigures the hardware, i.e. transmitter and receiver, as required and switches to a protocol stack corresponding to the new system. An important part of reconfiguring the receiver is reconfiguring, or programming, the equalizer because multipath is a main problem for each type of system. The multipath, however, varies depending on the type of system, which previously has meant that a different equalizer is required for different types of communication systems. The channel access protocol described in the preceding sections, however, allows for equalizers to be used that need only be reconfigured slightly for operation in various systems.

a. Sample Equalizer Embodiment

FIG. **24** illustrates an example embodiment of a receiver **2500** illustrating one way to configure equalizers **2506** in accordance with the systems and methods described herein. Before discussing the configuration of receiver **2500**, it should be noted that one way to configure equalizers **2506** is to simply include one equalizer per channel (for the systems and methods described herein, a channel is the equivalent of a sub-channel as described above). A correlator, such as correlator **2306** (FIG. **22**), can then provide equalizers **2506** with an estimate of the number, amplitude, and phase of any multipaths present, up to some maximum number. This is also known as the Channel Impulse Response (CIR). The maximum number of multipaths is determined based on design criteria for a particular implementation. The more multipaths included in the CIR the more path diversity the receiver has and the more robust communication in the system will be. Path diversity is discussed a little more fully below.

If there is one equalizer **2506** per channel, the CIR is preferably provided directly to equalizers **2506** from the correlator (not shown). If such a correlator configuration is used, then equalizers **2506** can be run at a slow rate, but the overall equalization process is relatively fast. For systems with a relatively small number of channels, such a configuration is therefore preferable. The problem, however, is that there is large variances in the number of channels used in different types of communication systems. For example, an outdoor system can have has many as 256 channels. This would require 256 equalizers **2506**, which would make the receiver design too complex and costly. Thus, for systems with a lot of channels, the configuration illustrated in FIG. **25** is preferable. In receiver **2500**, multiple channels share each equalizer **2506**. For example, each equalizer can be shared by 4 channels, e.g., Ch**1**-Ch**4**, Ch**5**-Ch**8**, etc., as illustrated in FIG. **25**. In which case, receiver **2500** preferably comprises a memory **2502** configured to store information arriving on each channel.

Memory **2502** is preferably divided into sub-sections **2504**, which are each configured to store information for a particular subset of channels. Information for each channel in each subset is then alternately sent to the appropriate equalizer **2506**, which equalizes the information based on the CIR

provided for that channel. In this case, each equalizer must run much faster than it would if there was simply one equalizer per channel. For example, equalizers **2506** would need to run 4 or more times as fast in order to effectively equalize 4 channels as opposed to 1. In addition, extra memory **2502** is required to buffer the channel information. But overall, the complexity of receiver **2500** is reduced, because there are fewer equalizers. This should also lower the overall cost to implement receiver **2500**.

Preferably, memory **2502** and the number of channels that are sent to a particular equalizer is programmable. In this way, receiver **2500** can be reconfigured for the most optimum operation for a given system. Thus, if receiver **2500** were moved from an outdoor system to an indoor system with fewer channels, then receiver **2500** can preferably be reconfigured so that there are fewer, even as few as 1, channel per equalizer. The rate at which equalizers **2506** are run is also preferably programmable such that equalizers **2506** can be run at the optimum rate for the number of channels being equalized.

In addition, if each equalizer **2506** is equalizing multiple channels, then the CIR for those multiple paths must alternately be provided to each equalizer **2506**. Preferably, therefore, a memory (not shown) is also included to buffer the CIR information for each channel. The appropriate CIR information is then sent to each equalizer from the CIR memory (not shown) when the corresponding channel information is being equalized. The CIR memory (not shown) is also preferably programmable to ensure optimum operation regardless of what type of system receiver **2500** is operating in.

Returning to the issue of path diversity, the number of paths used by equalizers **2506** must account for the delay spread d_s in the system. For example, if the system is an outdoor system operating in the 5 Giga Hertz (GHz) range, the communication channel can comprise a bandwidth of 125 Mega Hertz (MHz), e.g., the channel can extend from 5.725 GHz to 5.85 GHz. If the channel is divided into 512 sub-channels with a roll-off factor r of 0.125, then each subchannel will have a bandwidth of approximately 215 kilohertz (KHz), which provides approximately a 4.6 microsecond symbol duration. Since the worst case delay spread d_s is 20 microseconds, the number of paths used by equalizers **2504** can be set to a maximum of 5. Thus, there would be a first path P**1** at zero microseconds, a second path P**2** at 4.6 microseconds, a third path P**3** at 9.2 microseconds, a fourth path P**4** at 13.8 microseconds, and fifth path P**5** at 18.4 microseconds, which is close to the delay spread d_s. In another embodiment, a sixth path can be included so as to completely cover the delay spread d_s; however, 20 microseconds is the worst case. In fact, a delay spread d_s of 3 microseconds is a more typical value. In most instances, therefore, the delay spread d_s will actually be shorter and an extra path is not needed. Alternatively, fewer sub-channels can be used, thus providing a larger symbol duration, instead of using an extra path. But again, this would typically not be needed.

As explained above, equalizers **2506** are preferably configurable so that they can be reconfigured for various communication systems. Thus, for example, the number of paths used must be sufficient regardless of the type of communication system. But this is also dependent on the number of sub-channels used. If, for example, receiver **2500** went from operating in the above described outdoor system to an indoor system, where the delay spread d_s is on the order of 1 microsecond, then receiver **2500** can preferably be reconfigured for 32 sub-channels and 5 paths. Assuming the same overall bandwidth of 125 MHz, the bandwidth of each sub-channel is approximately 4 MHz and the symbol duration is approximately 250 nanoseconds.

Therefore, there will be a first path P**1** at zero microseconds and subsequent paths P**2** to P**5** at 250 ns, 500 ns, 750 ns, and 1 microsecond, respectively. Thus, the delay spread ds should be covered for the indoor environment. Again, the 1 microsecond delay spread d_s is worst case so the 1 microsecond delay spread d_s provided in the above example will often be more than is actually required. This is preferable, however, for indoor systems, because it can allow operation to extend outside of the inside environment, e.g., just outside the building in which the inside environment operates. For campus style environments, where a user is likely to be traveling between buildings, this can be advantageous.

7. Sample Embodiment of a Wireless Communication Device

FIG. **25** illustrates an example embodiment of a wireless communication device in accordance with the systems and methods described herein. Device **2600** is, for example, a portable communication device configured for operation in a plurality of indoor and outdoor communication systems. Thus, device **2600** comprises an antenna **2602** for transmitting and receiving wireless communication signals over a wireless communication channel **2618**. Duplexor **2604**, or switch, can be included so that transmitter **2606** and receiver **2608** can both use antenna **2602**, while being isolated from each other. Duplexors, or switches used for this purpose, are well known and will not be explained herein.

Transmitter **2606** is a configurable transmitter configured to implement the channel access protocol described above. Thus, transmitter **2606** is capable of transmitting and encoding a wideband communication signal comprising a plurality of sub-channels. Moreover, transmitter **2606** is configured such that the various sub-components that comprise transmitter **2606** can be reconfigured, or programmed, as described in section 5. Similarly, receiver **2608** is configured to implement the channel access protocol described above and is, therefore, also configured such that the various sub-components comprising receiver **2608** can be reconfigured, or reprogrammed, as described in section 6.

Transmitter **2606** and receiver **2608** are interfaced with processor **2610**, which can comprise various processing, controller, and/or Digital Signal Processing (DSP) circuits. Processor **2610** controls the operation of device **2600** including encoding signals to be transmitted by transmitter **2606** and decoding signals received by receiver **2608**. Device **2610** can also include memory **2612**, which can be configured to store operating instructions, e.g., firmware/software, used by processor **2610** to control the operation of device **2600**.

Processor **2610** is also preferably configured to reprogram transmitter **2606** and receiver **2608** via control interfaces **2614** and **2616**, respectively, as required by the wireless communication system in which device **2600** is operating. Thus, for example, device **2600** can be configured to periodically ascertain the availability is a preferred communication system. If the system is detected, then processor **2610** can be configured to load the corresponding operating instruction from memory **2612** and reconfigure transmitter **2606** and receiver **2608** for operation in the preferred system.

For example, it may preferable for device **2600** to switch to an indoor wireless LAN if it is available. So device **2600** may be operating in a wireless WAN where no wireless LAN is available, while periodically searching for the availability of an appropriate wireless LAN. Once the wireless LAN is

detected, processor **2610** will load the operating instructions, e.g., the appropriate protocol stack, for the wireless LAN environment and will reprogram transmitter **2606** and receiver **2608** accordingly. In this manner, device **2600** can move from one type of communication system to another, while maintaining superior performance.

It should be noted that a base station configured in accordance with the systems and methods herein will operate in a similar manner as device **2600**; however, because the base station does not move from one type of system to another, there is generally no need to configure processor **2610** to reconfigure transmitter **2606** and receiver **2608** for operation in accordance with the operating instruction for a different type of system. But processor **2610** can still be configured to reconfigure, or reprogram the sub-components of transmitter **2606** and/or receiver **2608** as required by the operating conditions within the system as reported by communication devices in communication with the base station. Moreover, such a base station can be configured in accordance with the systems and methods described herein to implement more than one mode of operation. In which case, controller **2610** can be configured to reprogram transmitter **2606** and receiver **2608** to implement the appropriate mode of operation.

8. High Data Rate Transmitter and Receiver

Referring now to FIGS. **26-49**, additional embodiments of the present invention are illustrated. The embodiments described below may contain some of the features and functionality as described above.

The embodiments of the present invention discussed below employ ultra-wideband communication technology. Referring to FIGS. **26** and **27**, impulse type ultra-wideband (UWB) communication employs discrete pulses of electromagnetic energy that are emitted at, for example, nanosecond or picosecond intervals (generally tens of picoseconds to a few nanoseconds in duration). For this reason, ultra-wideband is often called "impulse radio." That is, the UWB pulses may be transmitted without modulation onto a sine wave, or a sinusoidal carrier, in contrast with conventional carrier wave communication technology. This type of UWB generally requires neither an assigned frequency nor a power amplifier.

An example of a conventional carrier wave communication technology is illustrated in FIG. **26**. IEEE 802.11a is a wireless local area network (LAN) protocol, which transmits a sinusoidal radio frequency signal at a 5 GHz center frequency, with a radio frequency spread of about 5 MHz. As defined herein, a carrier wave is an electromagnetic wave of a specified frequency and amplitude that is emitted by a radio transmitter in order to carry information. The 802.11 protocol is an example of a carrier wave communication technology. The carrier wave comprises a substantially continuous sinusoidal waveform having a specific narrow radio frequency (5 MHz) that has a duration that may range from seconds to minutes.

In contrast, an ultra-wideband (UWB) pulse may have a 2.0 GHz center frequency, with a frequency spread of approximately 4 GHz, as shown in FIG. **27**, which illustrates two typical UWB pulses. FIG. **27** illustrates that the shorter the UWB pulse in time, the broader the spread of its frequency spectrum. This is because bandwidth is inversely proportional to the time duration of the pulse. A 600-picosecond UWB pulse can have about a 1.8 GHz center frequency, with a frequency spread of approximately 1.6 GHz and a 300-picosecond UWB pulse can have about a 3 GHz center frequency, with a frequency spread of approximately 3.3 GHz. Thus, UWB pulses generally do not operate within a specific frequency, as shown in FIG. **26**. Either of the pulses shown in FIG. **27** may be frequency shifted, for example, by using heterodyning, to have essentially the same bandwidth but centered at any desired frequency. And because UWB pulses are spread across an extremely wide frequency range, UWB communication systems allow communications at very high data rates, such as 100 megabits per second or greater.

Several different methods of ultra-wideband (UWB) communications have been proposed. For wireless UWB communications in the United States, all of these methods must meet the constraints recently established by the Federal Communications Commission (FCC) in their Report and Order issued Apr. 22, 2002. Currently, the FCC is allowing limited UWB communications, but as UWB systems are deployed, and additional experience with this new technology is gained, the FCC may revise its current limits and allow for expanded use of UWB communication technology.

The FCC April 22 Report and Order requires that UWB pulses, or signals occupy greater than 20% fractional bandwidth or 500 megahertz, whichever is smaller. Fractional bandwidth is defined as 2 times the difference between the high and low 10 dB cutoff frequencies divided by the sum of the high and low 10 dB cutoff frequencies. Specifically, the fractional bandwidth equation is:

$$\text{Fractional Bandwidth} = 2\frac{f_h - f_l}{f_h + f_l}$$

where f_h is the high 10 dB cutoff frequency, and f_l is the low 10 dB cutoff frequency.

Stated differently, fractional bandwidth is the percentage of a signal's center frequency that the signal occupies. For example, a signal having a center frequency of 10 MHz, and a bandwidth of 2 MHz (i.e., from 9 to 11 MHz), has a 20% fractional bandwidth. That is, center frequency, $f_c=(f_h+f_l)/2$

FIG. **28** illustrates the ultra-wideband emission limits for indoor systems mandated by the April 22 Report and Order. The Report and Order constrains UWB communications to the frequency spectrum between 3.1 GHz and 10.6 GHz, with intentional emissions to not exceed −41.3 dBm/MHz. The report and order also established emission limits for hand held UWB systems, vehicular radar systems, medical imaging systems, surveillance systems, through-wall imaging systems, ground penetrating radar and other UWB systems. It will be appreciated that the invention described herein may be employed indoors, and/or outdoors, and may be fixed, and/or mobile, and may employ either a wireless or wire media for a communication channel.

Generally, in the case of wireless communications, a multiplicity of UWB pulses may be transmitted at relatively low power density (milliwatts per megahertz). However, an alternative UWB communication system, located outside the United States, may transmit at a higher power density. For example, UWB pulses may be transmitted between 30 dBm to −50 dBm.

UWB pulses, however, transmitted through many wire media will not interfere with wireless radio frequency transmissions. Therefore, the power (sampled at a single frequency) of UWB pulses transmitted though wire media may range from about +30 dBm to about −140 dBm. The FCC's April 22 Report and Order does not apply to communications through wire media.

Communication standards committees associated with the International Institute of Electrical and Electronics Engineers (IEEE) are considering a number of ultra-wideband (UWB) wireless communication methods that meet the constraints

established by the FCC. One UWB communication method may transmit UWB pulses that occupy 500 MHz bands within the 7.5 GHz FCC allocation (from 3.1 GHz to 10.6 GHz). In one embodiment of this communication method, UWB pulses have about a 2-nanosecond duration, which corresponds to about a 500 MHz bandwidth. The center frequency of the UWB pulses can be varied to place them wherever desired within the 7.5 GHz allocation. In another embodiment of this communication method, an Inverse Fast Fourier Transform (IFFT) is performed on parallel data to produce 122 carriers, each approximately 4.125 MHz wide. In this embodiment, also known as Orthogonal Frequency Division Multiplexing (OFDM), the resultant UWB pulse, or signal is approximately 506 MHz wide, and has approximately 242-nanosecond duration. It meets the FCC rules for UWB communications because it is an aggregation of many relatively narrow band carriers rather than because of the duration of each pulse.

Another UWB communication method being evaluated by the IEEE standards committees comprises transmitting discrete UWB pulses that occupy greater than 500 MHz of frequency spectrum. For example, in one embodiment of this communication method, UWB pulse durations may vary from 2 nanoseconds, which occupies about 500 MHz, to about 133 picoseconds, which occupies about 7.5 GHz of bandwidth. That is, a single UWB pulse may occupy substantially all of the entire allocation for communications (from 3.1 GHz to 10.6 GHz).

Yet another UWB communication method being evaluated by the IEEE standards committees comprises transmitting a sequence of pulses that may be approximately 0.7 nanoseconds or less in duration, and at a chipping rate of approximately 1.4 giga pulses per second. The pulses are modulated using a Direct-Sequence modulation technique, and is called DS-UWB. Operation in two bands is contemplated, with one band is centered near 4 GHz with a 1.4 GHz wide signal, while the second band is centered near 8 GHz, with a 2.8 GHz wide UWB signal. Operation may occur at either or both of the UWB bands. Data rates between about 28 Megabits/second to as much as 1,320 Megabits/second are contemplated.

Another method of UWB communications comprises transmitting a modulated continuous carrier wave where the frequency occupied by the transmitted signal occupies more than the required 20 percent fractional bandwidth. In this method the continuous carrier wave may be modulated in a time period that creates the frequency band occupancy. For example, if a 4 GHz carrier is modulated using binary phase shift keying (BPSK) with data time periods of 750 picoseconds, the resultant signal may occupy 1.3 GHz of bandwidth around a center frequency of 4 GHz. In this example, the fractional bandwidth is approximately 32.5%. This signal would be considered UWB under the FCC regulation discussed above.

Thus, described above are four different methods of ultra-wideband (UWB) communication. It will be appreciated that the present invention may be employed by any of the above-described UWB methods, or others yet to be developed.

Referring now to FIG. **29**, which illustrates a block diagram of a transmitter **5210** consistent with one embodiment of the present invention. In this embodiment data **5110** of interest may be provided to data interface **5040**. A number of data interfaces **5040** are known in the art and can be used to practice the current invention. The data interface **5040** may include an industry standard such as a Universal Serial Bus (USB) standard interface, an IEEE 1394 standard interface, a Peripheral Component Interconnect standard (PCI), a Peripheral Component Interconnect Express (PCI-Express) standard, a MILSPEC-1760 standard, and a MILSPEC-1553 standard. Non-industry standard interfaces may also be employed and the present invention is not limited with respect to the type of data interface **5040** used. Data **5110** is sent from data interface **5040** to the Medium Access Controller (MAC) **5030**. The MAC **5030** performs a number of functions on the data **5110** to form a plurality of frames **5100**.

As illustrated in FIG. **31**, a data frame **5100** comprises a medium access control header **5120**, a data section **5110**, a source ID, a destination ID, a rate field **5130**, and in some embodiments may include a Cyclical Redundancy Check **5115** (CRC) appended to the end of the frame **5100**. Referring back to FIG. **29**, the data frames **5100** are then sent to a baseband processor **5020**, which performs a number of functions (described below) and produces baseband frame **5050**, illustrated in FIG. **33**.

A "frame" as defined herein, whether a data frame **5100**, a baseband frame **5050**, or another type of "frame," may include many different constructions and arrangements. Generally, a "frame" usually consists of a representation of the original data to be transmitted (generally comprising a specified number of bits, or binary digits), together with other bits that may be used for error detection or control. A "frame" may also include routing information, such as a source address, a destination address, and other information. A "frame" may be of different lengths, and contain variable amounts of data. It will be appreciated that the construction of baseband frame **5050** and data frame **5100** may vary without exceeding the scope of the present invention.

For example, additional bits in a "frame" may be used for routing (possibly in the form of an address field), synchronization, overhead information not directly associated with the original data, a frame check sequence, and a cyclic redundancy check (CRC), among others. CRC is an error detection algorithm that is known in the art of communications. One embodiment of a CRC may be described as follows. Given a data section **5110** having bits of length "k," the transmitter **5210** generates an n-bit sequence, known as the Frame Check Sequence (FCS) such that by appending the FCS to the data section **5110**, the resulting data section **5110** has a length k+n. The FCS is calculated in such a way that when a receiver divides the received resulting data section **5110** by a predetermined number there is no remainder. If no remainder is found the data section **5110** is assumed to be error free.

FIG. **33** illustrates the baseband frame **5050** produced by the baseband processor **5020**. The baseband frame **5050** comprises a physical layer header **5180**, the medium access control header **5120** and a number of data packets **5200**. Each data packet **5200** includes a code block **5190**, which is used by the receiver **5220** (shown in FIG. **34**) for synchronization of the packet **5200**. Additionally, the baseband frame **5050** may include the FCS used to decode the CRC as described above. Physical layer header **5180** may comprise a number of synchronization code blocks **5190** which are used by the receiver **5220** to synchronize its timing reference to the timing reference of the transmitter **5210**.

Generally, synchronization is used to obtain a fixed relationship among corresponding significant instants of two or more signals. Put differently, synchronization (also known as frame synchronization, frame alignment, or framing) is used by a receiver to lock onto an incoming frame so that it may receive the data contained in the frame. Generally, the receiver synchronizes its time base, or reference to the time base of the transmitter.

For example, a "frame synchronization pattern," generally comprising a recurring pattern of bits, is transmitted that

enables the receiver to align its clock, or time reference with the transmitter's time reference (i.e., synchronization). Repetition of the bit pattern helps ensure that the receiver will have an opportunity to "lock" in on the timing of the incoming signal.

In one embodiment of the present invention, the synchronization code blocks **5190** are comprised of 256 bit Golay codes. In another embodiment, one or more of the Golay codes may be inverse Golay codes. It will be appreciated that other types of synchronization codes, comprised of other bit sizes, may be employed by the present invention. One feature of the present invention is that upon reception of the synchronization sequence, the receiver may adjust its time base, its frequency base, and a setting of an automatic gain control amplifier (not shown).

Returning to FIG. **29**, the baseband frame **5050** is then sent from the baseband processor **5020** to modulator **5420**, which contains a digital circuit **5080** and local oscillator **5090**. Modulator **5420** performs modulation of the baseband frame **5050**, which includes representations of individual data bits, into a transmission signal **5070**. That is, the baseband processor **5020** outputs a signal comprised of high and low signal values, each having a time duration, or time base T_0, shown in FIG. **30**, which represent the data comprising the baseband frame **5050**.

The local oscillator **5090** generates a clock signal **5060** at a time base T_1, illustrated in FIG. **30**. In one embodiment of the present invention local oscillator **5090** may be a voltage controlled oscillator. As mentioned above, the signal values representing the baseband frame **5050** are at a time base T_0. Using the clock signal **5060**, the digital circuit **5080** modulates, or changes the signal values representing the baseband frame **5050**. In the illustrated embodiment, the type of modulation is phase modulation.

As shown in FIG. **30**, the inverse of the clock signal time base T_1 is the center frequency of the transmission signal **5070**. That is, $1/T_1$=center frequency. It will be appreciated that virtually any center frequency can be employed by the present invention. For example, the local oscillator **5090** may generate a clock signal **5060** with a time base T_1 of 250 picoseconds. In this example, the digital circuit **5080** produces a transmission signal **5070** that would be centered at 4 Giga-Hertz (GHz), which is the inverse of 250 picoseconds. The inverse of the baseband frame **5050** signal values (time base T_0) controls the amount of occupied bandwidth around the center frequency of the transmission signal **5070**. In the above example, if the time base T_0 of the baseband frame **5050** signal values is 750 picoseconds, the transmission signal would occupy 1.3 GHz of bandwidth, around a 4 GHz center frequency. In this case the bandwidth (i.e., amount of radio frequency spectrum) occupied would extend from approximately 3.33 GHz to approximately 4.66 GHz. The fractional bandwidth of this signal, calculated by the formula given above, would be approximately 33.25%. Thus, this transmission signal **5070** would be considered UWB under the current FCC definition because its fractional bandwidth exceeds 20%.

In another example, the clock signal **5060** time base T_1 may be approximately 133 picoseconds and the time base T_0 of the baseband frame **5050** signal values may be approximately 146 picoseconds. The transmission signal **5070** in this case would have a center frequency of 6.85 GHz and the signal would occupy 7.5 GHz of bandwidth around the center frequency. In this example, the transmission signal **5070** would occupy the entire available UWB spectrum from 3.1 GHz to 10.6 GHz. It would have a fractional bandwidth of approximately 110% and would be considered UWB. In yet another example, a clock signal time base T_1 of approximately 2 nanoseconds with a time base T_0 of the baseband frame **5050** signal values of approximately 5300 picoseconds yields a transmission signal **5070** that occupies 500 MHz of bandwidth located around a center frequency of 3.35 GHz. The fractional bandwidth of this exemplary transmission signal **5070** is only approximately 15%. While this signal does not meet the current UWB definition in terms of fractional bandwidth, it is still considered UWB since it occupies the required minimum of 500 MHz of bandwidth. In yet another example, a clock signal **5060** time base T_1 may be approximately 100 picoseconds and the time base T_0 of the baseband frame **5050** signal values may be 200 picoseconds. In this example, the transmission signal **5070** would occupy 10 GHz of bandwidth around a center frequency of 5 GHz. The bandwidth occupied by this transmission signal **5070** would extend from Direct Current at zero Hertz up to 10 GHz. This signal would occupy a fractional bandwidth of approximately 200%. Under the current UWB definition this signal would be a UWB signal but under the current FCC regulations would not be allowed for wireless transmission as a portion of the signal would be below the FCC mandated 3.1 GHz frequency boundary.

One feature of the present invention is that by generating a clock signal at the desired center frequency used for transmission, the present invention does not need to employ a mixer to position the signal at the transmission frequency. As discussed above, the present invention can generate a signal anywhere within (or outside of) the FCC mandated UWB radio frequency band by using a high-speed clock signal at the desired frequency. This feature reduces the overall cost and complexity of the device. In one embodiment, the high-speed clock is a 10.6 Giga-Hertz (GHz) clock, but it will be appreciated that other clocks, such as 4 GHz, 8 GHz, 12 GHz, and others may be employed by the present invention.

Several embodiments of digital circuit **5080** are illustrated in FIGS. **32***a*, **32***b*, and **32***c*. One feature of the digital circuits **5080** discussed below is that they directly generate the transmission signal, without mixing, or up-converting the signal to the radio frequency used for transmission.

Referring to FIG. **32***a*, the locally generated clock **5060** and the baseband frame **5050** signal values are the inputs to an "exclusive or" function or gate. As is known in the art, and shown in TABLE I, an "exclusive or" (XOR) gate performs the following function:

TABLE I

Input: Signal values 5050	Input: Clock 5060	Output: Transmission Signal 5070
0	0	0
0	1	1
1	0	1
1	1	0

As illustrated in FIG. **30**, during time periods where signal values, or baseband data **5050** has a "low" value, the "high" values in clock **5060** will cause the transmission signal **5070** to be "high." During time periods where the signal values **5050** are "low," the "low" values will result in a "low" in the transmission signal **5070**. Put differently, during "low" signal value **5050** time periods T_0, the transmission signal **5070** mirrors the clock **5060**. During time periods T_0 where signal values **5050** have a "high" value, the "high" values in clock **5060** result in "low" values in the transmission signal **5070**. Additionally during "high" signal value **5050** time periods,

the "low" clock **5060** values result in a "high" transmission signal **5070**. In other words, during "high" signal values **5050** time periods, the inverse of clock **5060** becomes the transmission signal **5070**. In this manner the signal values **5050** modulate the phase of the transmission signal **5070**.

In an alternate embodiment of digital circuit **5080**, illustrated in FIG. **32***b*, signal values **5050** and clock **5060** are inputs into an "and" gate. Additionally, the inverse of clock **5060** and signal values **5050** are inputs into another "and" gate. Combiner **5160** may then passively combine the outputs of the two "and" gates, or functions. As is known in the art, and shown in TABLE II, an "and" gate performs the following logical function:

TABLE II

Input: Signal values 5050	Input: Clock 5060	Output: Transmission Signal 5070
0	0	0
0	1	0
1	0	0
1	1	1

In a like manner, and as illustrated in FIG. **38**, during time periods T_0 where the signal values **5050** are high, the output of the "and" gate **5150***a* follows the clock **5060**. When the signal values **5050** are low, the output of "and" gate **5150** is "low". The inverse of signal values **5050** and the inverse of clock **5060** are inputs to "and" gate **5150***b*. During time periods where the signal values **5050** are low, the inverse of signal values are "high." During this time period the transmission signal **5070** becomes the inverse of the clock **5060**. The two outputs from "and" gates **5150***a* and **5150***b* may then be combined by combiner **5160** to produce transmission signal **5070**. In like manner to the "exclusive or" implementation described above, the phase of the clock **5060** is modulated by the signal values **5050** to become transmission signal **5070**. It should be noted that the transmission signal **5070** generated by the embodiment in FIG. **32***a* has an inverse phase relationship to the transmission signal generated by the embodiment shown in FIG. **32***b*. Either circuit may be modified by one with skill in the art to produce the other signal.

Yet another embodiment of digital circuit **5080** is illustrated in FIG. **32***c*. This embodiment can produce either of the transmission signals **5070** shown in FIG. **30** and FIG. **38**, by reversing the inputs of clock **5060** and its inverse. In this embodiment, a 2:1 multiplexer **5170** is used to generate a transmission signal **5070**. The clock **5060** and its inverse are connected to the multiplexer **5170**. The signal values **5050** from the baseband frame are connected to the control S_0. When the signal value **5050** has a low value, the signal present at input **0**, clock **5060**, is passed to the output transmission signal **5070**. When the signal value **5050** has a "high" value, the signal present at input **1**, inverse clock **5060**, is passed to the output. In this manner, the clock **5060** is phase modulated by signal values **5050** to produce transmission signal **5070**.

Many spread spectrum communications technologies are known in the art of communications. Generally, data to be transmitted is multiplied by a chipping code, where the time period of the code is referred to as a chip, or chip duration. The chipping code usually has a shorter duration time period than the signal value used to represent the data. Stated otherwise, the chip duration is usually shorter than the data symbol, or signal value duration. The resulting signal is a signal that occupies the bandwith of the chipping signal and carries the data signal. This bandwidth can be expressed as the inverse of the chip duration. The ratio of chips per data symbol is commonly referred to as the spreading factor. The process of multiplying the data signal by the chipping code is generally referred to as spreading the signal. In like manner, the process in a receiver of recovering the data signal from a spread signal may be referred to as de-spreading. In conventional spread spectrum communications systems, the spread signal is then multiplied by a carrier wave to place the signal at the radio frequency used for transmission. In some communication systems, orthogonal codes are used to enable a multiple access scheme, where multiple users can communicate simultaneously.

The spreading factor introduces generally unwanted overhead into a communications system. For example, a data symbol could be transmitted without spreading. In this case, a spreading factor of 1 is employed, implying the data has not been spread. When using a spreading factor of 256 the same data symbol would be 256 times larger than the same symbol using a spreading factor of 1. For example, if a spreading factor of 1 is used to send 1 bit of data, then 1 bit is transmitted. If a spreading factor of 256 is employed, then 256 bits are used to transmit 1 bit of data. So, as the spreading factor increases, the amount of data transmitted decreases.

One advantage of spreading the signal with a chipping code is that a receiver may use the entire chipping code to recover the signal. This process is commonly referred to as processing gain. Processing gain, expressed in dB, assists the receiver in detection of the signal, which increases communication reliability. Another advantage of spreading with a chipping code, is that when orthogonal codes are employed in different networks, the users in one network will not intercept the signals of the users in other networks.

In one embodiment of the present invention, the transmission signal **5070** is spread by a chipping code or code block **5190**, shown in FIG. **33**. In one embodiment, portions of the transmission signal **5070** have a different spreading factor. For example, the physical layer header **5180** may have a spreading factor of 256 where the medium access control header **5120** may have a spreading factor of 64. In another embodiment of the present invention, the packets **5200** may have a spreading factor that is dynamically controlled by the medium access controller **5030** that inserts the chosen spreading factor in rate field **5130** of the medium access control header **5120**.

In this fashion, the spreading factor may be dynamically adjusted to accommodate a changing communication environment. For example, if the distance the transmission signal **5070** must travel increases, the spreading factor may also increase, so that a receiver can recover the signal. Or, in a communication environment that is conducive to multipath, the spreading factor may also be increased. Alternatively, when the communication environment is favorable to communications, the spreading factor may be reduced, thereby increasing the rate at which data is transmitted.

Referring now to FIG. **35**, which illustrates some functions performed by baseband processor **5020**. Frames **5100** are received from the medium access controller **5030** by the baseband processor **5020**. The rate field **5130** in the medium access control header **5120** is evaluated to determine the data rate for the packets **5200**. Based on the rate field **5230**, FEC encoder **5300** applies a FEC (forward error correction, discussed below) encoding level to the data frame **5100**. For example, in one embodiment shown in TABLE III, the baseband processor **5020** uses the rate field **5130** to set the FEC encoding and/or the spreading factor. It will be appreciated that different spreading factors, and/or FEC encoding levels may be employed by the present invention.

TABLE III

Rate Field 5130 Value	FEC Encoding Level	Spreading Factor
0	1	1
1	¾	1
2	½	1
3	1	2
4	¾	2
5	½	2
6	1	4
7	¾	4
8	½	4
9	1	8
10	¾	8
11	½	8
12	1	16
13	¾	16
14	½	16
15	1	32
16	¾	32
17	½	32
18	1	64
19	¾	64
20	½	64
21	1	128
22	¾	128
23	½	128
24	1	256
25	¾	256
26	½	256

Encoding for Forward Error Correction (FEC) is a process by which redundancy is added too the data to be transmitted. With the additional redundancy the receiver may then attempt to detect and correct errors in the received data. An initial step in a FEC algorithm is to encode the data with additional bits. There are a number of FEC encoding algorithms. Of significant importance in communications are block codes and convolutional codes. Both types of encoding algorithms transform the original data set into a coded sequence of larger size. This increased size can yield a decrease in performance of information throughput for a particular data rate but may enable a more robust communication link. In convolutional encoding the coded sequence depends not only on the current data bits being encoded but also on one or more previous data bits. In convolutional coding the encoding is performed on a continuous basis. In block encoding a distinct block of data bits is encoded by a code. The FEC encoding level, otherwise known as the coding efficiency is a ratio of the original data to the encoded data. In other words, a FEC encoding level of ½ implies a 50% overhead or redundancy has been added to the data (50% more bits). Likewise, a ¾ FEC encoding level includes a 25% overhead. A FEC encoding level of 1 means that no additional bits have been added to the data. Other encoding rates are known in the art of communications and may be used. Those encoding levels include $\frac{1}{8}^{th}$ rate encoding, ¼ rate encoding, $\frac{3}{8}^{th}$ rate encoding ½ rate encoding, $\frac{5}{8}^{th}$ rate encoding, $\frac{7}{8}^{th}$ rate encoding, and ¾ rate encoding.

Referring again to FIG. **35**, after the FEC encoder **5300** has encoded the data, the data is then passed on to the interleaver **5310**. Interleaving is a process by which the order of the bits to be transmitted is changed. One purpose of interleaving bits or a block of bits is to improve a communications systems' noise immunity. For example, if bits from different portions of the data frame **5100** are interleaved, or mixed into a packet **5200** and that packet is corrupted by noise, or other factors during transmission, the impact of the corrupted packet is distributed across multiple areas of the data. This reduces the number of potential errors in any contiguous block of data, thereby increasing the probability that a receiver can correct the corrupted data.

After the data is been interleaved, the data is forwarded to scrambler **5320**. Scrambling the data reduces the probability of having long strings of similar data bits. Long strings of similar data bits may alter the distribution of transmitted power, known as the Power Spectral Density (PSD), within the spectrum. In many cases it is advantageous to have the effect of the data on the PSD be minimal. In those instances the effect of data should be random, or white, within the spectrum. A number of scrambling algorithms are known in the art and may be used to practice this embodiment of the invention.

The data is then sent to the spreader **5430**. Depending on the information in the rate field **5130** a spreading factor is applied to the data. As discussed above, the spreading factor may change based on the contents of the rate field **5130**.

The spread data is then sent to the packetizer **5340** where it is broken into discrete blocks appropriate for each packet **5200**. The synchronization generator **5350** generates synchronization code blocks **5190** for each packet. The header generator **5330** generates and forms the physical layer header **5180**. The physical layer header **5180** is then appended onto the medium access control header **5120**. A completed baseband frame **5050** is then forwarded to the modulator **5420**. It will be appreciated that the data processing order described above may be changed, and that other processing steps may be added or subtracted.

An exemplary receiver **5220** is depicted in FIG. **34**. In one embodiment, an RF signal is received from the communication media (wire or wireless) by the RF front end **5010**. The RF front end **5010** sends the received signal to an analog-to-digital converter (ADC) **5230**. The ADC **5230** may be a 1-bit ADC, a 2-bit ADC, a 3-bit ADC, a 4-bit ADC, a 5-bit ADC, a 6-bit ADC, a 7-bit ADC or an 8-bit ADC. Other bit densities for ADCs are known in the art of communications and may be used to practice the invention. Additionally, a number of ADC architectures are known in the art and may be used to practice the invention but will not be discussed here. In one embodiment of the present invention ADC **5230** is a 1-bit sigma delta ADC. In this embodiment, ADC **5230** samples the RF signal and creates a serial data signal. The serial data signal is sent to baseband processor **5020** which converts, or reassembles the packets **5200** from the serial data signal into data frames **5110** which are sent to the medium access controller **5030**. The medium access controller converts the data frames **5110** into data **5100**, which is sent to a data interface **5040**. Data interface **5040** may comprise a number of different data interfaces as described above.

RF front end **5010** may comprise a number of components including one or more antennas for communications in a wireless media, or coupling circuits for communication using wire media. The baseband processor **5020**, as illustrated in FIG. **36**, may comprise a poly-phase filter **5240**, a de-spreader **5250**, a channel impulse response detector **5260**, a channel matched filter **5410**, a de-scrambler **5270**, a de-interleaver **5280** and a FEC decoder **5290**.

In one embodiment RF front end **5010** may comprise two or more receive antennas (not shown). In this embodiment the receive antennas are separated by a physical distance from each other that approximates at least one wavelength of the center frequency of the signal the receiver is designed to receive. The wavelength is calculated by dividing the speed of light by the frequency. For example, a communication signal with a 4 GHz center frequency has a wavelength of approximately 7.5 cm. By separating multiple receive antennas by this distance, the receiver has a better chance of determining which received signal is a direct path signal and which is a multi-path signal. Additionally, the multiple receive antennas

provide additional energy collection which may be used to detect the communication signal.

Two embodiments of poly-phase filter **5240** are illustrated in FIGS. **37** and **38**. One function of the poly-phase filter **5240** is to down-convert the serial data signal into two lower frequency signals. The two signals are commonly referred to as In-phase (I) and Quadrature (Q). This conversion is accomplished by multiplying the serial data signal by a complex sinusoid. Since the serial data signal is discrete (having been sampled) the complex sinusoid is discrete samples of a sinusoid. The real and imaginary parts of a complex sinusoid may be calculated and stored in a look-up table **5400**. The serial data signal is split into two parallel signals by serial to parallel converter **5360**. Serial to parallel converter **5360** merely outputs alternate samples onto each output. Multipliers **5370** multiply the samples by real and imaginary coefficients that represent the complex sinusoid. In the embodiment illustrated in FIG. **37**, the resultant product signals are filtered by filters **5380**. In one embodiment, filters **5380** are low-pass finite impulse response (FIR) filters. FIR filters are known in the art of signal processing and will not be discussed herein. It will be appreciated that other types of filters may be employed by the present invention. Decimators **5390** then decimate the filtered signals. Decimation is a process by which a number of samples are discarded. In the embodiment illustrated in FIG. **38**, decimation occurs prior to filtering the signals.

Returning to FIG. **36**, the poly-phase filter sends the resultant signal to the de-spreader **5250**. The process of de-spreading the signal involves correlating the signal with a synchronization code block. If the received signal contains the same, or an inverse of the synchronization code block, the de-spreader finds a strong correlation, either positive or negative. The synchronization code block may then be removed and replaced by a value. The de-spreader **5250** then sends the signal to the channel impulse response detector **5260** and the channel matched filter **5410**.

One feature of the present invention is that it provides an adaptive matched filter system that can rapidly adjust to changing communication conditions. A wireless communication channel is generally characterized as a multipath fading channel, which includes multipath signals that cause intersymbol interference. A conventional matched filter includes an estimated model of the communication channel, which is used to aid the matched filter in obtaining the strongest possible signal. However, when the estimated model does not accurately reflect the actual communication channel, the signal may be poorly recovered leading to a poor signal-to-noise ratio (SNR). A degraded SNR may result in an increased bit-error-rate (BER), or may reduce the effective range of the communication system.

In the present invention, a channel impulse response detector **5260** is employed to provide a real-time analysis of the actual communication channel to the channel matched filter **5410**. During the detection of the physical layer header **5180**, the channel impulse response detector **5260** measures the communication channel impulse response by "listening" for correlations at a number of time periods. Generally, the impulse response is detected from the time period in which strong correlations are found with the codes contained within the physical layer header **5180** and code blocks **5190** (in packets **5200**). A number of codes are known in the art, but exemplary codes may include Golay, Walsh and perfect code sequences.

In the presence of multipath signal components, the de-spreader **5250** may correlate on delayed or multipath copies of the intended signal. Because of different propagation path lengths, multipath copies may show up at the receiver **5220** at a different time period than the intended signal. In this situation, the channel impulse response detector **5260** provides the time of arrival and strength of the multipath copies to the channel matched filter **5410**. The channel matched filter **5410** may then sum the received energy within the multipath copies to provide a stronger signal strength. In this way, the actual communication channel characteristics are determined and used to obtain the strongest possible signal.

Referring to FIG. **36**, the channel matched filter **5410** may also include an equalization capability, or function. Generally, the goal of equalization is to provide as accurate an estimate of the original data as possible. This "estimated" data can then be forwarded to subsequent processing blocks, such as the forward error correction (FEC) decoder **5290**. One function of the FEC is to detect and correct errors in the estimated data. If errors are detected, various remedial measures are performed. These measures will be discussed below in connection with the FEC decoder **5290**.

The channel matched filter **5410** includes a simplified decision feedback equalization (DFE) function. The multipath copies of previous signals may arrive at a time when receiver **5220** is processing a current signal. In this case the received signal may be made up of the intended signal and a number of multipath copies of previous signals. For example, the data signal may comprise the sum of 2 or more autocorrelation functions. In one case, the data signal may comprise the sum of 64 autocorrelation functions, or alternatively, the data signal may comprise the sum of 32, 128 or other sums of autocorrelation functions. As a result:

$$Z_n=(A_0\times d_n)+(A_1\times d_2)+(A_2\times d_{n-1})+(A_1{}^*\times d_{n+1})+(A_2{}^*\times d_{n+2}) \quad (8)$$

Where: A**1*** and A**2*** are the complex conjugates of A**1** and A**2**.

In this example the last four terms in equation (8) are what is known as Inter Symbol Interference (ISI) or Inter Chip Interference (ICI). In some cases where the symbol is substantially longer than the chip duration the interference may be intra-symbol. The first term is the data. So, in this exaple, a current data sample, or decision, actually depends on the current sample as well as two past samples and two future samples. Obtaining the two past samples should not drive the complexity of the equalizer; however, obtaining two future samples does increase complexity.

Accordingly, in one embodiment of equalizer of the present invention, the two future samples are ignored. In this embodiment, a "hard" estimation is employed. This is in contrast with most conventional equalizers, which often depend on "soft" decisions. The output from the hard decision is used to obtain the past two samples, which are multiplied by the associated amplitude factors A**1** and A**2** and combined. As shown in FIG. **40**, the A**1** and A**2** and other amplitude factors are represented by $g_{L,L}$. The coefficients $g_{L,L}$ are determined from the channel estimations. In one embodiment, the amplitude factors are supplied to the channel matched filter **5410** from the channel impulse response detector **5260** during processing of the physical layer header **5180** and during the processing of each packet **5200**.

Thus, the DFE converts the following:

$$Z_0=(A0\times d0)+(A1\times d1)+(A2\times d2) \quad (9)$$

Which becomes:

$$Z_0-[(A1\times d1)+(A2\times d2)]=(A0\times d0) \quad (10)$$

The output of the DFE can then be passed onward for further processing. A general implementation of DFE is an iterative process that significantly reduces the ISI or ICI.

Further, the DFE may include a parity check, or the like, in order to detect errors. If there are no errors, then there is no reason to feedback the data and perform the iteration.

As shown in FIG. **36**, in one embodiment, the channel matched filter **5410** then sends the signal to de-scrambler **5270**. In an embodiment where the transmitter **5210** scrambled the data, de-scrambler **5270** de-scrambles the data. The de-scrambler sends the de-scrambled data to the de-interleaver **5280**. In an embodiment where transmitter **5210** interleaved the data, de-interleaver **280** de-interleaves the data. FEC decoder **5290** detects and corrects errors in the recovered data **5110**. A number of decoding algorithms are known in the art and may be used to practice the invention. In one embodiment the FEC decoding algorithm is a low density parity check (LDPC) algorithm.

There are a number of error control methods known in the art of communications. Generally speaking, error control comprises two methods, error detection and error correction. In most error detection algorithms, the received data is merely checked for error. If errors are found, the transmitter may be notified and the data may be retransmitted. In error correction algorithms, the receiver attempts to correct detected errors. In one class of algorithms, known as Forward Error Correction (FEC), extra bits are transmitted with the data that can be used by the receiver to detect and correct errors in the data that was received. Depending on the implementation, the receiver can then ask that the data be re-sent if too many errors are detected.

Accordingly, as can be seen in FIG. **41**, an FEC encoder **3202** adds bits to an input data stream **3204** to create an output data stream **3206** that necessarily requires a higher data rate due to the added bits. In the example of FIG. **32**, FEC encoder **3202** is a ½ rate FEC encoder, which means that for every input bit d_n, FEC encoder **3202** adds a bit that can be used to detect errors when data stream **3206** is decoded. As discussed above other rate encoders, such as full rate, or ¾ rate encoders may be employed by the present invention. Thus, in the case of a ½ rate FEC encoder, data rate of output **3206** is twice that of input **3204**. Data stream **3206** can then be modulated and transmitted to a receiver. In the receiver, an FEC decoder **3208** can be used to remove the extra bits and detect errors in the original data. Thus, FEC decoder **3208** should match FEC encoder **3202**, i.e., FEC decoder **3208** should be a ½ rate FEC decoder, in the above example.

A problem with conventional FEC encoders and decoders is that the data rates can be too high for conventional technology. This can be especially true, for example, in an ultra-wideband application, where the data rates can be extremely high. One way to overcome this problem in accordance with the systems and methods described herein is illustrated in FIG. **42**, which depicts a portion of a transmitter chain **3300**. In the example of FIG. **33**, a data stream **3302**, with a data rate (R) is first split into a plurality of parallel data streams **3306** in serial to parallel converter **3304**, each with a lower data rate (R/n) where n is the number of parallel data streams **3306**. The parallel data streams **3306** can then be encoded using a plurality of FEC encoders. Here two encoders **3308** and **3310** are illustrated. Thus, each of FEC encoders **3308** and **3310** can, depending on the implementation, encode half as much data and operate at a lower speed than required in a conventional system. More generally, FEC encoders **3308** and **3310** can be configured to assist each other with FEC encoding and reduce the overall load on each FEC encoder in the system. This, of course, requires some coordination, or message passing, between the two FEC encoders.

The outputs of FEC encoders **3308** and **3310** can then, for example, be passed through parallel to serial converters **3312** and **3314** and combined via combiner **3316** into a single data stream with FEC encoding. The single data stream can then be optional filtered and/or pulse shaped before being modulated an transmitted, e.g., via optional block **3318**,

In another example embodiment, of an FEC encoder configured in accordance with the systems and methods described herein, a code word is generated from an input data word by adding parity bits to the data word as illustrated in FIG. **43**. In this example embodiment, FEC encoder **3402**, referred to as a Low Density Parity Check (LDPC) encoder, takes data word **3404** and generates output code word **3406**. As can be seen, the data word and code word are illustrated in matrix form. Thus, for example, the data word is a matrix comprising p_m+d_k rows and 1 column.

LDPC is an error correction algorithm where the data to be sent is encoded by a generator matrix and decoded by a parity matrix. Derivation of the two matrices is seen below in equation 12. The FEC encoder **3402**, a "K" length block of data $\vec{d}_{K\times 1}$ is multiplied by the generator matrix $G_{N\times K}$ which produces a "N" length block $\vec{C}_{N\times 1}$ where N>K. The additional length is attributed to the overhead described above. The parity matrix may represent a connection of two types of nodes in the decoder. The locations of 1's in the matrix represent the connection of the two types of nodes.

The decoding of the block on receipt is usually an iterative process by which the first type of node may calculate information related to the probability of the bit under consideration being a 1 or a 0. In some cases this probability may be expressed as a "log likelihood ratio" or mathematically:

$$l(c) = \ln\left[\frac{Pr(c_i = 0 \mid y)}{Pr(c_i = 1 \mid y)}\right]$$

where ln is the natural log, the numerator is the probability that the bit c_i is a zero and the denominator is the probability it is a 1. This information is passed to the other type of nodes specified by the parity check matrix, who perform a similar calculation based on the information received from each of the first type of node. The second type of node then sends its calculation to each of the first type of nodes it is connected to. This process continues until it is stopped or reaches some figure of merit in its result. Since each node is connected to a number of nodes of the other type, each iteration improves the probability calculation at each node.

In one embodiment of an LDPC **3402**, the code word can be generated using a generator matrix as illustrated by the following equation:

$$\vec{C}_{N\times 1} = G_{N\times K} * \vec{d}_{K\times 1} \tag{11}$$

where:

$G_{N\times K}$ is the Generator Matrix

N=M+k;

R=k/N;

if R=½, then M=k.

The generator matrix can, in turn, be generated from an identify matrix and a parity check matrix as illustrated in the following equation:

$$G=[I;P] \tag{12}$$

where:

I=the identity matrix; and

P=the parity matrix.

Alternatively, a parity matrix H can be used to generate the code word C according to the following:

$$H_{M\times N}*\vec{C}_{N\times 1}=\vec{0}_{M\times 1} \tag{13}$$

The parity matrix H can then be defined as:

$$H_{M\times N}=[H^P{}_{M\times M}H^D{}_{M\times K}] \tag{14}$$

Accordingly, and dropping the subscripts for simplicity:

$$[H^P H^D]*\begin{bmatrix}\vec{P}\\ \vec{d}\end{bmatrix}=0,\text{ or} \tag{15}$$

$$\left(H^P*\vec{P}\right)+\left(H^D*\vec{d}\right)=0 \tag{16}$$

The goal now is to solve for $\vec{P}$, since $\vec{d}$ is known, i.e., it is the input data. To facilitate finding $\vec{P}$ in one embodiment, H^P is configured as a dual diagonal matrix with M rows and M columns. Dual diagonal matrices are well known and will not be described here; however and exemplary one is illustrated by the following:

$$H^P=\begin{bmatrix}1&1&0&0&\cdots&0\\0&1&1&0&\cdots&0\\0&0&1&1&\cdots&0\\0&0&0&1&\cdots&\\&&&&&1\\0&0&0&\cdots&0&1\end{bmatrix} \tag{17}$$

Further, H^D can, depending on the embodiment, be formed from a matrix of matrices. In one embodiment, this matrix of matrices is itself block cyclic. For example, in one embodiment, 4 matrices A, B, C and D can be used as in the following:

$$H^D=\begin{bmatrix}A&B&C&D\\B&C&D&A\\C&D&A&B\\D&A&B&C\end{bmatrix} \tag{18}$$

Here, each of the matrices A, B, C, and D will have k/4 rows and k/4 columns.

Thus, an encoder and decoder configured in accordance with the systems and methods described herein can be optimized for a dual diagonal H^P and a block cyclic H^D, an explained below. Many methods can be used to generate matrices A, B, C, and D consistent with the systems and methods described herein. One example method, however, will be described in the following paragraphs. This example method will assume, for the time being, that k=16 and therefore k/4=4. Then an identity matrix I can be used, such as the following:

$$I=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \tag{19}$$

Each of matrices A, B, C, and D can then be generated from this identity matrix L For example, a permutation vector, in this example of length 4, can then be used to generate A. Of course, other methods for generating matrices A, B, C, and D can be used consistently with the systems and methods described herein. Thus, the matrix A can, e.g., have the following form, once an appropriate permutation vector is used to modify identity matrix I:

$$A=\begin{bmatrix}0&0&0&1\\1&0&0&0\\0&0&1&0\\0&1&0&0\end{bmatrix} \tag{20}$$

Basically, as can be seen, a permutation vector can be used to shift the positions of the 1's in identity matrix I. In one embodiment, a single permutation matrix can be required. Once the first matrix A is generated using the single permutation vector, then the other matrices B, C, and D can be generated by shifting matrix A. For example, in one embodiment, each subsequent matrices B, C, and D is generated by shifting the previous matrix, starting with A, by 90°. Thus, B would be as follows:

$$B=\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&0&0&1\\0&1&0&0\end{bmatrix} \tag{21}$$

But as can be seen, in the example embodiment for generating matrices A, B, C, and D described above, each row has only a single 1.

In one embodiment, Galois Field algebra (GF(2)) can be used to define the following equations for use in solving for $\vec{P}$:

$$1+1=0;$$

$$1-1=0;$$

$$1+0=1;$$

$$0+1=1;$$

$$0+1=1;\text{ and}$$

$$0-1=-1=1. \tag{22}$$

Thus, even results are equal to 0, while odd results are equal to 1.

Now returning to the equation at issue:

$$(H^P*\vec{P})+(H^D*d)=0 \tag{23}$$

This can be rewritten as:

$$(H^P*\vec{P})=(H^D*d) \tag{24}$$

But using the equations (22), −1=1, therefore:

$$(H^P*\vec{P})=(H^D*d) \tag{25}$$

In one embodiment, the following equation can be used:

$$(H^D*\vec{d})=\vec{u} \tag{26}$$

Accordingly:

$$(H^P*\vec{P})=\vec{u} \tag{27}$$

Equation (27) can be implemented effectively if $\vec{u}$ can be generated efficiently. In one embodiment, based on the examples above, if k=6, then $\vec{u}$ can be determined as follows:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} p_5 \\ p_4 \\ p_3 \\ p_2 \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} u_5 \\ u_4 \\ u_3 \\ u_2 \\ u_1 \\ u_0 \end{bmatrix} \tag{28}$$

This will result in the following equations:

$$p_0 - u_0;$$

$$p_1 + p_0 = u_1;$$

$$p_2 + p_1 = u_2;$$

$$p_3 + p_2 = u_3;$$

$$p_4 + p_3 = u_4;$$

$$p_5 + p_4 = u_5; \tag{29}$$

The equations of (29) define the following general equation:

$$p_n = u_n - p_{n-1} \tag{30}$$

This equation then suggests a configuration for an LDPC encoder **3402**, such as that illustrated in FIG. **44**. As can be seen, the $\vec{u}$ values are fed into Exclusive-OR (XOR) **3502**, the output of which is fed through a delay **3504** and back to the other input of XOR **3502**. A remaining issue, however, is the generation of the U terms. In other words, the equation $(H^P * \vec{P}) = \vec{u}$ as implemented by block **3506** should also be done in the most efficient manner possible.

In the example above, H^D was partitioned in 4 s, therefore $\vec{d}$ should also be partitioned by 4 as illustrated in the following:

$$\vec{u} = \begin{bmatrix} A & B & C & D \\ D & A & B & C \\ C & D & A & B \\ B & C & D & A \end{bmatrix} * \begin{bmatrix} \vec{d}_A \\ \vec{d}_B \\ \vec{d}_C \\ \vec{d}_D \end{bmatrix} \tag{31}$$

The above equation can be implemented efficiently, for example, using a circuit such as the example circuit illustrated in FIG. **45**. The circuit of FIG. **45** is generalized for the situation where k=128; however, it will be appreciated that the example embodiments described herein are not limited to any particular lengths or configurations. As can be seen, the circuit of FIG. **45** uses a bank of cyclic shift registers **3606** to implement $\vec{d}$. The outputs of shift registers **3606** can then be passed to a plurality XORs **3602** as shown. Thus, XORs **3602** collect the appropriate outputs from shift registers **3606** in order to generate the U terms.

But since it is known, in the examples above, that the output of each cyclic shift register will only have one 1, due to the fact that A, B, C, and D have only one 1 in each row, the outputs of cyclic shift registers **3606** can be rearranged and fixed so that, e.g., the first outputs of each go to the first XOR **3602**, the second outputs go to the second XOR **3602**, etc. Accordingly, efficient fixed connections **3608** can be used to reduce the complexity of LDPC **3402**. The U terms can then be registered and fed to XOR **3502** as illustrated.

Accordingly, if everything is segmented by 4's as illustrated in the above examples, then the cyclic shift registers **3606** can be shifted k/4 times. On each clock cycle, k/4 of the solution would be generated, such that it takes k/4 cycles to get the entire solution. This can result in a highly parallel encoder, such as that illustrated in FIG. **36**, for high-speed operation. The result can also be a low cost encoder, because the hardware can be reduced to ¼th that required by conventional circuits through the reuse of the components.

The LDPC encoder of FIG. **44** can, therefore, be used to generate code word C, which can be modulated and transmitted. But the receiver will receive C corrupted by noise as illustrated in the following:

$$\vec{x} = (1 - 2\vec{C}) + (\text{noise}) \tag{32}$$

The job of the decoder is then to extract $\vec{d}$ from the signal represented by equation 32. In one embodiment, this can be accomplished by making soft decisions as to the value of $\vec{x}$ and combining it with hard decisions related to the sign $\vec{x}$ such that $\vec{d}$ can then be accurately determined. The soft decisions can be based on a multilevel possibility. For example, if 4 bits are used in 2's complement, then you can have up to 16 levels. In one embodiment, the levels can, for example, be from −8 to 7. Alternatively, using offset 2's complement, the levels can be from −7.5 to 7.5. An advantage of the later is that the levels are not biased, e.g., toward the negative. An advantage of the former, however, is that it includes the level 0. Of course, any level scheme can be used as long as it allows for accurate determinations of $\vec{d}$.

The levels can be used to determine the probabilities of the value of $\vec{x}$ and ultimately $\vec{d}$. For example, if the level determined for $\vec{x}$ is 7 or 7.5, then the decoder can be configured to see this as a high probability that the value is 1. If the level is −8 or −7.5, then this can be seen as a high probability that the value is −1.

Parity check equations can then be generated from the following:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 1 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & \\ & & & & & 1 \\ 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} * \begin{bmatrix} A & B & C & D \\ B & C & D & A \\ C & D & A & B \\ D & A & B & C \end{bmatrix} * \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \\ x_{n-1} \end{bmatrix} \tag{33}$$

This will produce a set of parity equations in which, based on the examples above, there will be 6 terms, except in the last one, because there is exactly one 1 in each row of A, B, C, and D. The first of these parity equations would then, e.g., look like the following, based on the above examples:

$$S_0 = x_0 + x_1 + x_2 + x_3 + x_4 + x_5 \tag{34}$$

Then, if $S_o = +1$, then the operation can be viewed as passing. If, on the other hand, it is −1, then it can be viewed as a failure. A parity node processor **3702** can be used to implement equation 34, as illustrated by the example embodiment depicted in FIG. **46**. Message passing algorithms can be used to allow each such node **3702** to make final estimations.

FIG. **48** is a diagram illustrating and example embodiment in which a plurality of parity node processors **3702** are configured in accordance with the systems and methods described herein. Thus, each node **3702** receives information as to what the values $x_0, x_1, \ldots x_N$ are believed to be. A given node **3702** can then process this information and produce estimates as to what the node believes the output of the other nodes should be and feed this information back in such a manner that the subsequent input to the other nodes is modified. It should be noted, therefore, that in such an embodiment, a node does not produce information to be fedback to its own input related to what it believes its own output should be.

This is illustrated in FIG. **47** for a single node processor at time=0. As can be seen, information for each bit is provided to node **3702**, which processes the information and produces information related to what it determines each bit should be. These inputs and outputs can be referred to as edges (E). Each output edge is fedback to the relevant input bit. The node processors **3702** will, therefore, comprise storage to store the information being fed to it and processed as required. As a result, both storage and routing overheads can become excessive.

For example, when information related to bit x_0 is fed to node S_0, the information from each other node related to x_0 is also added into the information provided to s_0. This is illustrated by the following:

$$x_0+E^{1+1}(s_1\rightarrow x_0)+E^1+1(S_2\rightarrow x_0)+\ldots \quad (35)$$

Again, as mentioned above, in this embodiment, the edge produce by node S_0 is not fed back to bit x_0.

FIG. **49** is a diagram illustrating an example decoder **4000** that can be configured to reduce storage and routing overhead in accordance with one embodiment of the systems and methods described herein. The basic premise behind decoder **4000** is that all the edges produced form parity node processor **4002** can be added and then the last edge for each node, produced by that node, can be subtracted out. Thus, on the right hand side of FIG. **49**, a given row can be updated for all edges and then shifted in shift registers **4004**. The appropriate edge can then be subtracted out for each row using the data provided from registers **4014**, as opposed to doing each row, storing the result and updating it with information from other nodes.

It should be noted that the output of shift registers **4004** can be rearranged and fixed to reduce routing overhead. It should also be noted that this process provides an approximation of the correct data; however, the results converge and ultimately provide the same answer. On the left hand side of decoder **4000**, each shift register **4008** gets information from only two nodes **3702**, e.g., via registers **4010** and **4012**.

One feature of the present invention is that it may be used to increase the bandwidth of wireless networks or networks that employ wired media. The present invention can be used to transmit ultra-wideband signals across any type of wired media. For example, the wired media can include optical fiber ribbon, fiber optic cable, single mode fiber optic cable, multimode fiber optic cable, plenum wire, PVC wire, and coaxial cable.

In addition, the wired media can include twisted-pair wiring, whether shielded or unshielded. Twisted-pair wire may consist of "pairs" of color-coded wires. Common sizes of twisted-pair wire are 2 pair, 3 pair, 4 pair, 25 pair, 50 pair and 100 pair. Twisted-pair wire is commonly used for telephone and computer networks. It comes in ratings ranging from category 1 to category 7. Twisted-pair wiring also is available unshielded. That is, the wiring does not have a foil or other type of wrapping around the group of conductors within the jacket. This type of wiring is most commonly used for wiring for voice and data networks. The foregoing list of wired media is meant to be exemplary, and not exclusive.

As described above, the present invention can provide additional bandwidth to enable the transmission of large amounts of data over an existing wired media network, whether the wired media network is a Internet service provider, cable television provider, or a computer network located in a business or university. The additional bandwidth can allow consumers to receive the high speed Internet access, interactive video and other features that are bandwidth intensive.

The present invention may be employed in any type of network, be it wireless, wire, or a mix of wire media and wireless components. That is, a network may use both wire media, such as coaxial cable, and wireless devices, such as satellites, or cellular antennas. As defined herein, a network is a group of points or nodes connected by communication paths. The communication paths may use wires or they may be wireless. A network as defined herein can interconnect with other networks and contain sub-networks. A network as defined herein can be characterized in terms of a spatial distance, for example, such as a local area network (LAN), a personal area network (PAN), a metropolitan area network (MAN), a wide area network (WAN), and a wireless personal area network (WPAN), among others. A network as defined herein can also be characterized by the type of data transmission technology used by the network, such as, for example, a Transmission Control Protocol/Internet Protocol (TCP/IP) network, a Systems Network Architecture network, among others. A network as defined herein can also be characterized by whether it carries voice, data, or both kinds of signals. A network as defined herein may also be characterized by users of the network, such as, for example, users of a public switched telephone network (PSTN) or other type of public network, and private networks (such as within a single room or home), among others. A network as defined herein can also be characterized by the usual nature of its connections, for example, a dial-up network, a switched network, a dedicated network, and a non-switched network, among others. A network as defined herein can also be characterized by the types of physical links that it employs, for example, optical fiber, coaxial cable, a mix of both, unshielded twisted pair, and shielded twisted pair, among others. The present invention may be employed in any type of wireless network, such as a wireless PAN, LAN, MAN, or WAN. In addition, the present invention may be employed in wire media, as the present invention dramatically increases the bandwidth of conventional networks that employ wire media, yet it can be inexpensively deployed without extensive modification to the existing wire media network.

One feature of the present invention is that it has a data rate and quality of service high enough to support multiple video streams. For example, one embodiment of the present invention may provide a communication channel having a data rate of 1.3 gigabits per second. This high data rate is particularly useful in hand held security devices. Such systems can provide dramatically improved national security. For example, current airport security systems involve large, stationary equipment that scans luggage and passengers. However, an individual may pass through a security checkpoint without being scanned or checked for identification. At most commercial airports it may be exceedingly difficult to locate the individual using current methods and equipment. In most cases the security personnel are relying on a verbal description of the individual, which may be inaccurate. Under current regulatory guidelines the terminal must be closed, emptied of passengers and manually searched.

With the data rates provided by the present invention, security camera access points throughout the airport may transmit one or more channels of streaming video directly to video viewers carried by security personnel, thereby allowing the search to be conducted in a more efficient manner. The data rates of conventional wireless communication systems cannot support multiple video streams, and therefore cannot provide the features and functionality of the present invention.

Thus, it is seen that systems and methods of providing a high speed transmitter and receiver are provided. One skilled in the art will appreciate that the present invention can be practiced by other than the above-described embodiments, which are presented in this description for purposes of illustration and not of limitation. The specification and drawings are not intended to limit the exclusionary scope of this patent document. It is noted that various equivalents for the particular embodiments discussed in this description may practice the invention as well. That is, while the present invention has been described in conjunction with specific embodiments, it is evident that many alternatives, modifications, permutations and variations will become apparent to those of ordinary skill in the art in light of the foregoing description. Accordingly, it is intended that the present invention embrace all such alternatives, modifications and variations as fall within the scope of the appended claims. The fact that a product, process or method exhibits differences from one or more of the above-described exemplary embodiments does not mean that the product or process is outside the scope (literal scope and/or other legally-recognized scope) of the following claims.

What is claimed is:

1. A transmitter comprising:

a data interface;

a medium access controller configured to receive data from the data interface and arrange the data into a plurality of frames;

a baseband processor configured to receive the plurality of frames and configured to produce a multiplicity of high and low signal values representing the data, with each high and low signal value having a first timing interval;

a local oscillator generating a clock signal at a second timing interval; and

a digital circuit configured to combine the high and low signal values with the clock signal to produce a transmission signal directly at a transmission frequency.

2. The transmitter of claim **1**, wherein the baseband processor segments the data from the plurality of frames into a plurality of data packets.

3. The transmitter of claim **2**, wherein the baseband processors adds a synchronization code to each of the plurality of data packets.

4. The transmitter of claim **2**, wherein the baseband processor adds a single physical layer header to the plurality of data packets.

5. The transmitter of claim **4**, wherein the physical layer header comprises a plurality of synchronization code blocks.

6. The transmitter of claim **1**, further comprising a forward error correction encoder that encodes the data with a forward error correction algorithm.

7. The transmitter of claim **6**, wherein the forward error correction encoder is a low density parity check algorithm.

* * * * *