



US007444531B2

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

(10) **Patent No.:** **US 7,444,531 B2**  
(45) **Date of Patent:** **Oct. 28, 2008**

(54) **METHODS AND DEVICES FOR TREATING  
AND PROCESSING DATA**

(75) Inventors: **Martin Vorbach**, München (DE);  
**Volker Baumgarte**, München (DE)

(73) Assignee: **Pact XPP Technologies AG**, Munich  
(DE)

(\*) Notice: Subject to any disclaimer, the term of this  
patent is extended or adjusted under 35  
U.S.C. 154(b) by 307 days.

(21) Appl. No.: **10/469,909**

(22) PCT Filed: **Mar. 5, 2002**

(86) PCT No.: **PCT/EP02/02402**

§ 371 (c)(1),  
(2), (4) Date: **Sep. 21, 2004**

(87) PCT Pub. No.: **WO02/071196**

PCT Pub. Date: **Sep. 12, 2002**

(65) **Prior Publication Data**

US 2005/0066213 A1 Mar. 24, 2005

(30) **Foreign Application Priority Data**

Mar. 5, 2001	(DE)	101 10 530
Mar. 7, 2001	(DE)	101 11 014
Jun. 20, 2001	(DE)	101 29 237
Jun. 20, 2001	(EP)	01115021
Jun. 24, 2001	(DE)	101 35 210
Jul. 24, 2001	(DE)	101 35 211
Aug. 16, 2001	(DE)	101 39 170
Aug. 29, 2001	(DE)	101 42 231
Sep. 3, 2001	(DE)	101 42 894
Sep. 3, 2001	(DE)	101 42 903
Sep. 3, 2001	(DE)	101 42 904
Sep. 11, 2001	(DE)	101 44 732
Sep. 11, 2001	(DE)	101 44 733
Sep. 17, 2001	(DE)	101 45 792

Sep. 17, 2001	(DE)	101 45 795
Sep. 19, 2001	(DE)	101 46 132
Nov. 5, 2001	(DE)	101 54 259
Nov. 5, 2001	(DE)	101 54 260
Dec. 14, 2001	(EP)	01129923
Jan. 18, 2002	(EP)	02001331
Jan. 19, 2002	(DE)	102 02 044
Jan. 20, 2002	(DE)	102 02 175
Feb. 15, 2002	(DE)	102 06 653
Feb. 18, 2002	(DE)	102 06 856
Feb. 18, 2002	(DE)	102 06 857
Feb. 21, 2002	(DE)	102 07 224
Feb. 21, 2002	(DE)	102 07 225
Feb. 21, 2002	(DE)	102 07 226
Feb. 27, 2002	(DE)	102 08 434
Feb. 27, 2002	(DE)	102 08 435

(51) **Int. Cl.**

**G06F 1/00** (2006.01)

(52) **U.S. Cl.** **713/323; 713/300; 713/310;**  
**713/320; 713/321; 713/322; 713/324; 713/330;**  
**713/340; 713/400; 713/401**

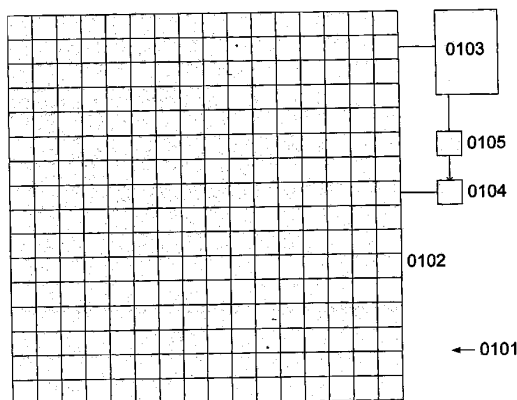
(58) **Field of Classification Search** **713/300,**  
**713/310, 320-324, 330, 340, 400-401, 500-503,**  
**713/600-601**

See application file for complete search history.

(56) **References Cited**

**U.S. PATENT DOCUMENTS**

2,067,477	A	1/1937	Cooper
3,242,998	A	3/1966	Gubbins
3,681,578	A	8/1972	Stevens
3,757,608	A	9/1973	Willner
3,855,577	A	12/1974	Vandierendonck
4,233,667	A	11/1980	Devine et al.
4,414,547	A *	11/1983	Knapp et al. 326/40
4,498,134	A	2/1985	Hansen et al.
4,498,172	A	2/1985	Bhavsar
4,566,102	A	1/1986	Hefner
4,591,979	A	5/1986	Iwashita
4,663,706	A	5/1987	Allen et al.
4,682,284	A	7/1987	Schrofer
4,706,216	A	11/1987	Carter
4,720,780	A	1/1988	Dolecek



# US 7,444,531 B2

Page 2

4,739,474	A	4/1988	Holsztynski	5,485,103	A	1/1996	Pedersen et al.
4,761,755	A	8/1988	Ardini et al.	5,485,104	A	1/1996	Agrawal et al.
4,811,214	A	3/1989	Nosenchuck et al.	5,489,857	A	2/1996	Agrawal et al.
4,852,043	A	7/1989	Guest	5,491,353	A	2/1996	Kean
4,852,048	A	7/1989	Morton	5,493,239	A	2/1996	Zlotnick
4,860,201	A	8/1989	Miranker et al.	5,497,498	A	3/1996	Taylor
4,870,302	A	9/1989	Freeman	5,506,998	A	4/1996	Kato et al.
4,891,810	A	1/1990	de Corlieu et al.	5,510,730	A	4/1996	El Gamal et al.
4,901,268	A	2/1990	Judd	5,511,173	A	4/1996	Yamaura et al.
4,910,665	A	3/1990	Mattheyses et al.	5,513,366	A	4/1996	Agarwal et al.
4,967,340	A	10/1990	Dawes	5,521,837	A	5/1996	Frankle et al.
5,014,193	A	5/1991	Garner et al.	5,522,083	A	5/1996	Gove et al.
5,015,884	A	5/1991	Agrawal et al.	5,530,873	A	6/1996	Takano
5,021,947	A	6/1991	Campbell et al.	5,530,946	A	6/1996	Bouvier et al.
5,023,775	A	6/1991	Poret	5,532,693	A	7/1996	Winters et al.
5,043,978	A	8/1991	Nagler et al.	5,532,957	A	7/1996	Malhi
5,047,924	A	9/1991	Matsubara et al.	5,535,406	A	7/1996	Kolchinsky
5,065,308	A	11/1991	Evans	5,537,057	A	7/1996	Leong et al.
5,072,178	A	12/1991	Matsumoto	5,537,601	A	7/1996	Kimura et al.
5,081,375	A	1/1992	Pickett et al.	5,541,530	A	7/1996	Cliff et al.
5,109,503	A	4/1992	Cruikshank et al.	5,544,336	A	8/1996	Kato et al.
5,113,498	A	5/1992	Evan et al.	5,548,773	A	8/1996	Kemeny et al.
5,115,510	A	5/1992	Okamoto et al.	5,555,434	A	9/1996	Carlstedt
5,123,109	A	6/1992	Hillis	5,559,450	A	9/1996	Ngai et al.
5,125,801	A	6/1992	Nabity et al.	5,561,738	A	10/1996	Kinerk et al.
5,128,559	A	7/1992	Steele	5,570,040	A	10/1996	Lytle et al.
5,142,469	A	8/1992	Weisenborn	5,574,930	A	11/1996	Halverson, Jr. et al.
5,144,166	A	9/1992	Camarota et al.	5,583,450	A	12/1996	Trimberger et al.
5,193,202	A	3/1993	Lee et al.	5,586,044	A	12/1996	Agrawal et al.
5,203,005	A	4/1993	Horst	5,587,921	A	12/1996	Agrawal et al.
5,204,935	A	4/1993	Mihara et al.	5,588,152	A	12/1996	Dapp et al.
5,208,491	A	5/1993	Ebeling et al.	5,590,345	A	12/1996	Barker et al.
5,212,716	A *	5/1993	Ferraiolo et al. .... 375/373	5,590,348	A	12/1996	Phillips et al.
5,226,122	A	7/1993	Thayer et al.	5,596,742	A	1/1997	Agarwal et al.
RE34,363	E	8/1993	Freeman	5,600,265	A	2/1997	El Gamal et al.
5,233,539	A	8/1993	Agrawal et al.	5,600,845	A	2/1997	Gilson
5,247,689	A	9/1993	Ewert	5,611,049	A	3/1997	Pitts
5,274,593	A	12/1993	Proebsting	5,617,547	A	4/1997	Feeney et al.
5,287,472	A	2/1994	Horst	5,625,806	A	4/1997	Kromer
5,294,119	A	3/1994	Vincent et al.	5,634,131	A	5/1997	Matter et al.
5,301,284	A	4/1994	Estes et al.	5,649,176	A	7/1997	Selvidge et al.
5,301,344	A	4/1994	Kolchinsky	5,649,179	A	7/1997	Steenstra et al.
5,303,172	A	4/1994	Magar et al.	5,652,529	A *	7/1997	Gould et al. .... 326/93
5,311,079	A	5/1994	Ditlow et al.	5,652,894	A	7/1997	Hu et al.
5,336,950	A	8/1994	Popli et al.	5,655,069	A	8/1997	Ogawara et al.
5,347,639	A	9/1994	Rechtschaffen et al.	5,655,124	A	8/1997	Lin
5,349,193	A	9/1994	Mott et al.	5,657,330	A	8/1997	Matsumoto
5,353,432	A	10/1994	Richek et al.	5,659,797	A	8/1997	Zandveld et al.
5,361,373	A	11/1994	Gilson	5,675,743	A	10/1997	Mavity
5,379,444	A	1/1995	Mumme	5,680,583	A	10/1997	Kuijsten
5,392,437	A	2/1995	Matter et al.	4,720,778	A	1/1998	Hall et al.
5,410,723	A	4/1995	Schmidt et al.	5,713,037	A	1/1998	Wilkinson et al.
5,418,952	A	5/1995	Morley et al.	5,717,943	A	2/1998	Barker et al.
5,421,019	A	5/1995	Holsztynski et al.	5,732,209	A	3/1998	Vigil et al.
5,422,823	A	6/1995	Agrawal et al.	5,734,921	A	3/1998	Dapp et al.
5,425,036	A	6/1995	Liu et al.	5,737,516	A	4/1998	Circello et al.
5,426,378	A	6/1995	Ong	5,742,180	A	4/1998	Detton et al.
5,428,526	A	6/1995	Flood et al.	5,748,872	A	5/1998	Norman
5,430,687	A	7/1995	Hung et al.	5,754,827	A	5/1998	Barbier et al.
5,440,245	A	8/1995	Galbraith et al.	5,754,871	A	5/1998	Wilkinson et al.
5,440,538	A	8/1995	Olsen et al.	5,760,602	A	6/1998	Tan
5,442,790	A	8/1995	Nosenchuck	5,761,484	A	6/1998	Agarwal et al.
5,444,394	A	8/1995	Watson et al.	5,773,994	A	6/1998	Jones
5,448,186	A	9/1995	Kawata	5,778,439	A	7/1998	Trimberger et al.
5,455,525	A	10/1995	Ho et al.	5,784,636	A	7/1998	Rupp
5,457,644	A	10/1995	McCollum	5,794,059	A	8/1998	Barker et al.
5,465,375	A	11/1995	Thepaut et al.	5,794,062	A	8/1998	Baxter
5,473,266	A	12/1995	Ahanin et al.	5,801,715	A	9/1998	Norman
5,473,267	A	12/1995	Stansfield	5,802,290	A	9/1998	Casselman
5,475,583	A	12/1995	Bock et al.	5,828,229	A	10/1998	Cliff et al.
5,475,803	A	12/1995	Stearns et al.	5,828,858	A	10/1998	Athanas et al.
5,475,856	A	12/1995	Kogge	5,838,165	A	11/1998	Chatter
5,483,620	A	1/1996	Pechanek et al.	5,841,973	A	11/1998	Cooke et al.

# US 7,444,531 B2

Page 3

5,844,888	A	12/1998	Narjyka	6,289,440	B1	9/2001	Casselmann
5,848,238	A	12/1998	Shimomura et al.	6,298,472	B1	10/2001	Phillips et al.
5,854,918	A	12/1998	Baxter	6,301,706	B1	10/2001	Maslennikov et al.
5,859,544	A	1/1999	Norman	6,311,200	B1	10/2001	Hanrahan et al.
5,865,239	A	2/1999	Carr	6,321,366	B1	11/2001	Tseng et al.
5,867,691	A	2/1999	Shiraishi	6,321,373	B1	11/2001	Ekanadham et al.
5,867,723	A	2/1999	Peters, Jr. et al.	6,338,106	B1	1/2002	Vorbach et al.
5,884,075	A	3/1999	Hester et al.	6,341,318	B1	1/2002	Dakhil
5,887,162	A	3/1999	Williams et al.	6,347,346	B1	2/2002	Taylor
5,887,165	A	3/1999	Martel et al.	6,349,346	B1	2/2002	Hanrahan et al.
5,889,982	A	3/1999	Rodgers et al.	6,370,596	B1	4/2002	Dakhil
5,892,370	A	4/1999	Eaton et al.	6,378,068	B1	4/2002	Foster et al.
5,892,961	A	4/1999	Trimberger	6,389,379	B1	5/2002	Lin et al.
5,892,962	A	4/1999	Cloutier	6,389,579	B1	5/2002	Phillips et al.
5,901,279	A	5/1999	Davis, III	6,392,912	B1	5/2002	Hanrahan et al.
5,915,123	A	6/1999	Mirsky et al.	6,398,383	B1	6/2002	Huang
5,924,119	A	7/1999	Sindhu et al.	6,404,224	B1	6/2002	Azegami et al.
5,926,638	A	7/1999	Inoue	6,405,299	B1	6/2002	Vorbach et al.
5,927,423	A	7/1999	Wada et al.	6,421,809	B1	7/2002	Wuytack et al.
5,933,642	A	8/1999	Baxter et al.	6,421,817	B1	7/2002	Mohan et al.
5,936,424	A	8/1999	Young et al.	6,425,068	B1	7/2002	Vorbach et al.
5,943,242	A	8/1999	Vorbach et al.	6,434,695	B1	8/2002	Esfahani et al.
5,956,518	A	9/1999	DeHon et al.	6,434,699	B1	8/2002	Jones et al.
5,960,200	A	9/1999	Eager et al.	6,437,441	B1	8/2002	Yamamoto
5,966,534	A	10/1999	Cooke et al.	6,457,116	B1	9/2002	Mirsky et al.
5,970,254	A	10/1999	Cooke et al.	6,477,643	B1	11/2002	Vorbach et al.
5,978,260	A	11/1999	Trimberger et al.	6,480,937	B1	11/2002	Vorbach et al.
5,996,083	A	11/1999	Gupta et al.	6,480,954	B2	11/2002	Trimberger et al.
6,003,143	A	12/1999	Kim et al.	6,490,695	B1	12/2002	Zagorski et al.
6,011,407	A	1/2000	New	6,496,971	B1	12/2002	Lesca et al.
6,014,509	A	1/2000	Furtek et al.	6,504,398	B1	1/2003	Lien et al.
6,020,758	A	2/2000	Patel et al.	6,513,077	B2	1/2003	Vorbach et al.
6,021,490	A	2/2000	Vorbach et al.	6,519,674	B1	2/2003	Lam et al.
6,023,564	A	2/2000	Trimberger	6,526,520	B1	2/2003	Vorbach et al.
6,023,742	A	2/2000	Ebeling et al.	6,538,468	B1	3/2003	Moore
6,034,538	A	3/2000	Abramovici	6,539,477	B1	3/2003	Seawright
6,038,650	A	3/2000	Vorbach et al.	6,542,844	B1	4/2003	Hanna
6,038,656	A	3/2000	Cummings et al.	6,542,998	B1	4/2003	Vorbach et al.
6,047,115	A	4/2000	Mohan et al.	6,571,381	B1	5/2003	Vorbach et al.
6,049,222	A	4/2000	Lawman	6,657,457	B1	12/2003	Hanrahan et al.
6,052,773	A	4/2000	DeHon et al.	6,687,788	B2	2/2004	Vorbach et al.
6,054,873	A	4/2000	Laramie	6,697,979	B1	2/2004	Vorbach et al.
6,058,469	A	5/2000	Baxter	6,704,816	B1	3/2004	Burke
6,081,903	A	6/2000	Vorbach et al.	6,717,436	B2	4/2004	Kress et al.
6,085,317	A	7/2000	Smith	6,721,830	B2	4/2004	Vorbach et al.
6,086,628	A	7/2000	Dave et al.	6,757,847	B1	6/2004	Farkash et al.
6,088,795	A	7/2000	Vorbach et al.	6,785,826	B1	8/2004	Durham et al.
6,092,174	A	7/2000	Roussakov	6,961,924	B2	11/2005	Bates et al.
6,105,105	A	8/2000	Trimberger et al.	2001/0010074	A1	7/2001	Nishihara et al.
6,108,760	A	8/2000	Mirsky et al.	2002/0038414	A1	3/2002	Taylor et al.
6,119,181	A	9/2000	Vorbach et al.	2002/0143505	A1	10/2002	Drusinsky
6,122,719	A	9/2000	Mirsky et al.	2002/0144229	A1	10/2002	Hanrahan
6,125,408	A	9/2000	McGee et al.	2002/0165886	A1	11/2002	Lam
6,127,908	A	10/2000	Bozler et al.	2003/0014743	A1	1/2003	Cooke et al.
6,150,837	A	11/2000	Beal et al.	2003/0046607	A1	3/2003	Vorbach
6,150,839	A	11/2000	New et al.	2003/0052711	A1	3/2003	Taylor et al.
6,170,051	B1	1/2001	Dowling	2003/0055861	A1	3/2003	Lai et al.
6,172,520	B1	1/2001	Lawman et al.	2003/0056085	A1	3/2003	Vorbach
6,173,434	B1	1/2001	Wirthlin et al.	2003/0056091	A1	3/2003	Greenberg
6,202,182	B1	3/2001	Abramovici et al.	2003/0056202	A1	3/2003	Vorbach
6,211,697	B1	4/2001	Lien et al.	2003/0061542	A1	3/2003	Bates et al.
6,212,650	B1	4/2001	Guccione	2003/0093662	A1	5/2003	Vorbach et al.
6,219,833	B1	4/2001	Solomon et al.	2003/0097513	A1	5/2003	Vorbach et al.
6,230,307	B1	5/2001	Davis et al.	2003/0123579	A1	7/2003	Safavi et al.
6,240,502	B1	5/2001	Panwar et al.	2003/0135686	A1	7/2003	Vorbach et al.
6,243,808	B1	6/2001	Wang	2003/0192032	A1	10/2003	Andrade et al.
6,260,179	B1	7/2001	Ohsawa et al.	2004/0015899	A1	1/2004	May et al.
6,263,430	B1	7/2001	Trimberger et al.	2004/0025005	A1	2/2004	Vorbach et al.
6,279,077	B1	8/2001	Nasserbakht et al.	2004/0168099	A1	8/2004	Vorbach et al.
6,282,627	B1	8/2001	Wong et al.				
6,282,701	B1	8/2001	Wygodny et al.				
6,286,134	B1	9/2001	Click, Jr. et al.				
6,288,566	B1	9/2001	Hanrahan et al.				

2004/0199688 A1 10/2004 Vorbach et al.

## FOREIGN PATENT DOCUMENTS

DE	42 21 278	1/1994
DE	44 16 881	11/1994
DE	196 51 075	6/1998
DE	196 54 593	7/1998
DE	196 54 595	7/1998
DE	196 54 846	7/1998
DE	197 04 044	8/1998
DE	197 04 728	8/1998
DE	197 04 742	9/1998
DE	198 07 872	8/1999
DE	198 61 088	2/2000
DE	100 28 397	12/2001
DE	100 36 627	2/2002
DE	101 29 237	4/2002
DE	102 04 044	8/2003
EP	0 208 457	1/1987
EP	0 221 360	5/1987
EP	0 428 327	5/1991
EP	0 463 721	1/1992
EP	0 477 809	4/1992
EP	0 485 690	5/1992
EP	0 427 029	8/1992
EP	0 539 595	5/1993
EP	0 628 917	12/1994
EP	0 678 985	10/1995
EP	0 686 915	12/1995
EP	0 707 269	4/1996
EP	0 735 685	10/1996
EP	0 835 685	10/1996
EP	0 748 051	12/1996
EP	0 726 532	7/1998
EP	0 926 594	6/1999
EP	1 102 674	7/1999
EP	0 696 001	12/2001
JP	58-58672	4/1983
JP	2-130023	5/1990
JP	2-226423	9/1990
JP	5-276007	10/1993
JP	7-154242	6/1995
JP	8-44581	2/1996
JP	8-250685	9/1996
JP	9-27745	1/1997
JP	11-307725	11/1999
JP	2000-181566	6/2000
WO	WO90/04835	5/1990
WO	WO90/11648	10/1990
WO	WO93/11503	6/1993
WO	WO94/06077	3/1994
WO	WO94/08399	4/1994
WO	WO95/00161	1/1995
WO	WO95/26001	9/1995
WO	WO98/26356	6/1998
WO	WO98/28697	7/1998
WO	WO98/29952	7/1998
WO	WO98/31102	7/1998
WO	WO98/35294	8/1998
WO	WO98/35299	8/1998
WO	WO99/00731	1/1999
WO	WO99/00739	1/1999
WO	WO99/12111	3/1999
WO	WO99/32975	7/1999
WO	WO99/40522	8/1999
WO	WO99/44120	9/1999
WO	WO99/44147	9/1999
WO	WO00/17771	3/2000
WO	WO00/38087	6/2000
WO	WO00/77652	12/2000
WO	WO01/55917	8/2001
WO	WO02/13000	2/2002

WO	WO02/21010	3/2002
WO	WO02/29600	4/2002
WO	WO02/071248	9/2002
WO	WO02/071249	9/2002
WO	WO02/103532	12/2002
WO	WO03/017095	2/2003
WO	WO03/023616	3/2003
WO	WO03/025781	3/2003
WO	WO03/032975	4/2003
WO	WO03/036507	5/2003
WO	WO03/103532	12/2003

## OTHER PUBLICATIONS

Arabi et al., "PLD Integrates Dedicated High-speed Data Buffering, Complex State Machine, and Fast Decode Array," conference record on WESCON '93, Sep. 28, 1993, pp. 432-436.

Ade et al., "Minimum Memory Buffers in DSP Applications," *Electronics Letters*, vol. 30, No. 6, Mar. 17, 1994, pp. 469-471.

Villasenor, John et al., "Configurable Computing," *Scientific American*, vol. 276, No. 6, Jun. 1997, pp. 66-71.

Villasenor, John et al., "Configurable Computing Solutions for Automatic Target Recognition," *IEEE*, 1996 pp. 70-79.

Tau, Edward et al., "A First Generation DPGA Implementation," *FPD'95*, pp. 138-143.

Athanas, Peter et al., "Quantitative Analysis of Floating Point Arithmetic on FPGA Based Custom Computing Machines", from IEEE Symposium on FPGAs for Custom Computing Machines, IEEE Computer Society Press, Apr. 19-21, 1995, pp. 155-162.

Athanas, Peter et al., "An Adaptive Hardware Machine Architecture and Compiler for Dynamic Processor Reconfiguration", IEEE, Laboratory for Engineering Man/Machine Systems Division of Engineering, Box D, Brown University Providence, Rhode Island, 1991, pp. 397-400.

Bittner, Ray A. Jr., "Wormhole Run-time Reconfiguration: Conceptualization and VLSI Design of a High Performance Computing System," *Dissertation*, Jan. 23, 1997, pp. I-XX, 1-415.

Myers, G. "Advances in Computer Architecture," Wiley-Interscience Publication, 2nd ed., John Wiley & Sons, Inc. pp. 463-494, 1978.

M. Saleeba, "A Self-Contained Dynamically Reconfigurable Processor Architecture," Sixteenth Australian Computer Science Conference, ASCS-16, QLD, Australia, Feb. 1993.

M. Morris Mano, "Digital Design," by Prentice Hall, Inc., Englewood Cliffs, New Jersey 07632, 1984, pp. 119-125, 154-161.

Maxfield, C. "Logic that Mutates While-U-Wait" *EDN* (Bur. Ed) (USA), *EDN* (European Edition), Nov. 7, 1996, Cahners Publishing, USA.

Norman, Richard S., "Hyperchip Business Summary, The Opportunity," Jan. 31, 2000, pp. 1-3.

Ferrante J. et al., "The Program Dependence Graph and its Use in Optimization ACM Transactions on Programming Languages and Systems," Jul. 1987, USA, [online] Bd. 9, Nr. 3, pp. 319-349, XP002156651 ISSN: 0164-0935 ACM Digital Library.

Hwang L. et al., "Min-cut Replication in Partitioned Networks" *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, [online] Bd. 14, Nr. 1, Jan. 1995, pp. 96-106, XP00053228 USA ISSN: 0278-0070 IEEE Xplore.

Baumgarte, V., et al., PACT XPP "A Self-reconfigurable Data Processing Architecture," PACT Info. GMBH, Munchen, Germany 2001.

Jantsch, Axel et al., "A Case Study on Hardware/software Partitioning," Royal Institute of Technology, Kista, Sweden, Apr. 10, 1994 *IEEE*, pp. 111-118.

Becker, J. et al., "Parallelization in Co-compilation for Configurable Accelerators—a Host/accelerator Partitioning Compilation Method," proceedings of Asia and South Pacific Design Automation Conference, Yokohama, Japan, Feb. 10-13, 1998.

Isshiki, Tsuyoshi et al., "Bit-Serial Pipeline Synthesis for Multi-FPGA Systems with C++ Design Capture," 1996 *IEEE*, pp. 38-47.

Dissertation for Doktors der Ingenieurwissenschaften der Universität Karlsruhe: Weinhardt, Markus, Übersetzungsmethoden für strukturprogrammierbare rechner, Jul. 1, 1997.

- Hammes, Jeff et al., "Cameron: High Level Language Compilation for Reconfigurable Systems," Department of Computer Science, Colorado State University, Conference on Parallel Architectures and Compilation Techniques, Oct. 12-16, 1999.
- K. Wada et al., "A Performance Evaluation of Tree-based Coherent Distributed Shared Memory" Proceedings of the Pacific RIM Conference on Communications, Comput and Signal Processing, Victoria, May 19-21, 1993.
- Nilsson et al., "The Scalable Tree Protocol —A Cache Coherence Approaches for Large-Scale Multiprocessors" IEEE, pp. 498-506 Dec. 1992.
- Wu et al., "A New Cache Directory Scheme", IEEE, pp. 466-472, Jun. 1996.
- Hauck "The Roles of FPGA's in Reprogrammable Systems," IEEE, Apr. 1998, pp. 615-638.
- Wittig et al., "OneChip: An FPGA Processor with Reconfigurable Logic" IEEE, 1996 pp. 126-135.
- Cadambi et al., "Managing Pipeline-reconfigurable FPGAs," ACM, 1998, pp. 55-64.
- Mirsky, E. DeHon, "MATRIX: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources," Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, 1996, pp. 157-166.
- Cardoso, J.M.P., "Compilation of Java™ Algorithms onto Reconfigurable Computing Systems with Exploitation of Operation-Level Parallelism," Ph.D. Thesis, Universidade Tecnica de Lisboa (UTL), Lisbon, Portugal Oct. 2000, (English Abstract).
- Kung, "Deadlock Avoidance for Systolic Communication", 1988 Conference Proceedings of 15<sup>th</sup> Annual International Symposium on Computer Architecture, May 30, 1988, pp. 252-260.
- TMS320C54X DSP: CPU and Peripherals, Texas Instruments, 1996, pp. 6-26 to 6-46.
- TMS320C54x DSP: Mnemonic Instruction Set, Texas Instruments, 1996, p. 4-64.
- XLINX, "Logic Cell Array Families: XC4000, XC4000A and XC4000H", product description, pp. 2-7 to 2-15 Additional XC3000, XC31000 and XC3100A Data, pp. 8-16 and 9-14.
- Miller, Michael J. et al., "High-Speed FIFOs Contend with Widely Differing Data Rates: Dual-port RAM Buffer and Dual-pointer System Provide Rapid, High-density Data Storage and Reduce Overhead", Computer Design, Sep. 1, 1985, pp. 83-86.
- John R. Hauser et al., "Garp: A MIPS Processor with a Reconfigurable Coprocessor", University of California, Berkeley, IEEE, 1997, pp. 12-21.
- Jorg Donandt, "Improving Response Time of Programmable Logic Controllers by Use of a Boolean Coprocessor", AEG Research Institute Berlin, IEEE, 1989, pp. 4-167-4-169.
- Alexandre F. Tenca et al., "A Variable Long-Precision Arithmetic Unit Design for Reconfigurable Coprocessor Architectures", University of California, Los Angeles, 1998, pp. 216-225.
- Andreas Koch et al., "Practical Experiences with the SPARXIL Co-Processor", IEEE, pp. 394-398.
- Beck et al., "From control flow to data flow," TR 89-1050, Oct. 1989, Dept. of Computer Science, Cornell University, Ithaca, NY, pp. 1-25.
- Dutt, et al., "If Software is King for Systems-in-Silicon, What's New in Compiler?," IEEE, 1997, pp. 322-325.
- Fineberg, S. et al., "Experimental Analysis of a Mixed-Mode Parallel Architecture Using Bitonic Sequence Sorting," Journal of Parallel and Distributed Computing, vol. 11, No. 3, Mar. 1991, pp. 239-251.
- Fornaciari, et al., "System-level power evaluation metrics, 1997 Proceedings of the 2<sup>nd</sup> Annual IEEE International Conference on Innovative Systems in Silicon, New York, NY, Oct. 1997, pp. 323-330.
- Forstner, "Wer Zuerst Kommt, Mahlt Zuerst! Teil 3: Einsatzgebiete und Anwendungsbeispiele von FIFO-Speichern," Elektronik, Aug. 2000, pp. 104-109.
- Gokhale, M.B. et al., "Automatic Allocation of Arrays to Memories in FPGA processors with Multiple Memory Banks," Field-Programmable Custom Computing Machines, 1999, IEEE, pp. 63-67.
- Hauck, "The Roles of FPGAs in Reprogrammable Systems," IEEE, Apr. 1998, pp. 615-638.
- Hedge, S.J., "3D WASP Devices for On-line Signal and Data Processing," 1994, International Conference on Wafer Scale Integration, pp. 11-21.
- Ling, X., "WASMII: An MPLD with Data-Driven Control on a Virtual Hardware," Journal of Supercomputing, Kluwer Academic Publishers, Dordrecht, Netherlands, 1995, pp. 253-276.
- Ling et al., "WASMII: A Multifunction Programmable Logic Device (MPLD) with Data Driven Control," The Transactions of the Institute of Electronics, Information and Communication Engineers, Apr. 25, 1994, vol. J77-D-1, Nr. 4, pp. 309-317. [This references is in Chinese, but should be comparable in content to the Ling et al. reference above].
- Miller, M.J., et al., "High-Speed FIFOs Contend with Widely Differing Data Rates: Dual-port RAM Buffer and Dual-pointer System Provide Rapid, High-density Data Storage and Reduce Overhead," Computer Design, Sep. 1, 1985, pp. 83-86.
- Piotrowski, A., "IEC-BUS, Die Funktionsweise des IEC-Bus und seine Anwendung in Geräten und Systemen," 1987, Franzis-Verlag GmbH, München, pp. 20-25.
- Schmit, et al., Hidden Markov Modeling and Fuzzy Controllers in FPGAs, FPGAs for Custom Computing machines, 1995; Proceedings, IEEE Symposium in Napa Valley, CA, Apr. 1995, pp. 214-221.
- Shirazi, et al., "Quantitative analysis of floating point arithmetic on FPGA based custom computing machines," IEEE Symposium on FPGAs for Custom Computing Machines, IEEE Computer Society Press, Apr. 19-21, 1995, pp. 155-162.
- Sueyoshi, T., "Present Status and Problems of the Reconfigurable Computing Systems Toward the Computer Evolution," Department of Artificial Intelligence, Kyushi Institute of Technology, Fukuoka, Japan; Institute of Electronics, Information and Communication Engineers, vol. 96, No. 426, IEICE Technical Report (1996), pp. 111-119 [English Abstract Only].
- Siemers, C., "Rechenfabrik Ansatz für Extrem Parallele Prozessoren," Verlag Heinze Heise GmbH., Hannover, DE No. 15, Jul. 16, 2001, pp. 170-179.
- Simunic, et al., Source Code Optimization and Profiling of Energy Consumption in Embedded Systems, Proceedings of the 13<sup>th</sup> International Symposium on System Synthesis, Sep. 2000, pp. 193-198.
- The XPP White Paper, Release 2.1, PACT—A Technical Perspective, Mar. 27, 2002, pp. 1-27.
- Villasenor, et al., "Configurable Computing Solutions for Automatic Target Recognition," IEEE, 1996 pp. 70-79.
- Villasenor, et al., "Express Letters Video Communications Using Rapidly Reconfigurable Hardware," IEEE Transactions on Circuits and Systems for Video Technology, IEEE, Inc., NY, Dec. 1995, pp. 565-567.
- Weinhardt, Markus et al., "Pipeline Vectorization for Reconfigurable Systems," 1999, IEEE, pp. 52-62.
- Weinhardt, Markus et al., "Pipeline Vectorization," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 20, No. 2, Feb. 2001, pp. 234-248.
- Xu, H.Y. et al., "Parallel QR Factorization on a Block Data Flow Architecture," Conference Proceeding Article, Mar. 1, 1992, pp. 332-336 XPO10255276, p. 333, Abstract 2.2, 2.3, 2.4-p. 334.
- Abnous, A., et al., "The Pleiades Architecture," Chapter I of *The Application of Programmable DSPs in Mobile Communications*, A. Gatherer and A. Auslander, Ed., Wiley, 2002, pp. 1-33.
- Chen et al., "A reconfigurable multiprocessor IC for rapid prototyping of algorithmic-specific high-speed DSP data paths," *IEEE Journal of Solid-State Circuits*, vol. 27, No. 12, Dec. 1992, pp. 1895-1904.
- Hastie et al., "The implementation of hardware subroutines on field programmable gate arrays," Custom Integrated Circuits Conference, 1990, Proceedings of the IEEE 1990, May 16, 1990, pp. 31.3.1-31.4.3 (3 pages).
- Kastrup, B., "Automatic Hardware Synthesis for a Hybrid Reconfigurable CPU Featuring Philips CPLDs," Proceedings of the PACT Workshop on Reconfigurable Computing, 1998, pp. 5-10.
- Razdan et al., A High-Performance Microarchitecture with Hardware-Programmable Functional Units, Micro-27, Proceedings of the 27<sup>th</sup> Annual International Symposium on Microarchitecture, IEEE Computer Society and Association for Computing Machinery, Nov. 30-Dec. 2, 1994, pp. 172-180.
- Siemers et al., "The >S<puter: A Novel Micoarchitecture Mode for Execution inside Superscalar and VLIW Processors Using Reconfigurable Hardware," Australian Computer Science Communications, vol. 20, No. 4, Computer Architecture, Proceedings of the

3<sup>rd</sup> Australian Computer Architecture Conference, Perth, John Morris, Ed., Feb. 2-3, 1998, pp. 169-178.

Skokan, Z.E., "Programmable logic machine (A programmable cell array)," IEEE Journal of Solid-State Circuits, vol. SC-18, No. 5, Oct. 1983, pp. 572-578.

Yeung, A. et al., "A data-driven architecture for rapid prototyping of high throughput DSP algorithms," Dept. of Electrical Engineering and Computer Sciences, Univ. of California, Berkeley, USA, *Proceedings VLSI Signal Processing Workshop, IEEE Press*, pp. 225-234, Napa, Oct. 1992.

Yeung, A. et al., "A reconfigurable data-driven multiprocessor architecture for rapid prototyping of high throughput DSP algorithms," Dept. of Electrical Engineering and Computer Sciences, Univ. of California, Berkeley, USA, pp. 169-178, *IEEE* 1993.

\* cited by examiner

*Primary Examiner*—Abdelmoniem Elamin

(74) *Attorney, Agent, or Firm*—Kenyon & Kenyon LLP

(57)

#### ABSTRACT

A data processing unit having a field of clocked logic cells (PAEs) which is operable in different configuration states and a clock preselecting means for preselecting logic cell clocking. The clock preselecting means is designed in such a way that, depending on the state, a first clock is preselected at least at a first cell (PAE) and an additional clock is preselected at least at an additional cell.

**10 Claims, 7 Drawing Sheets**

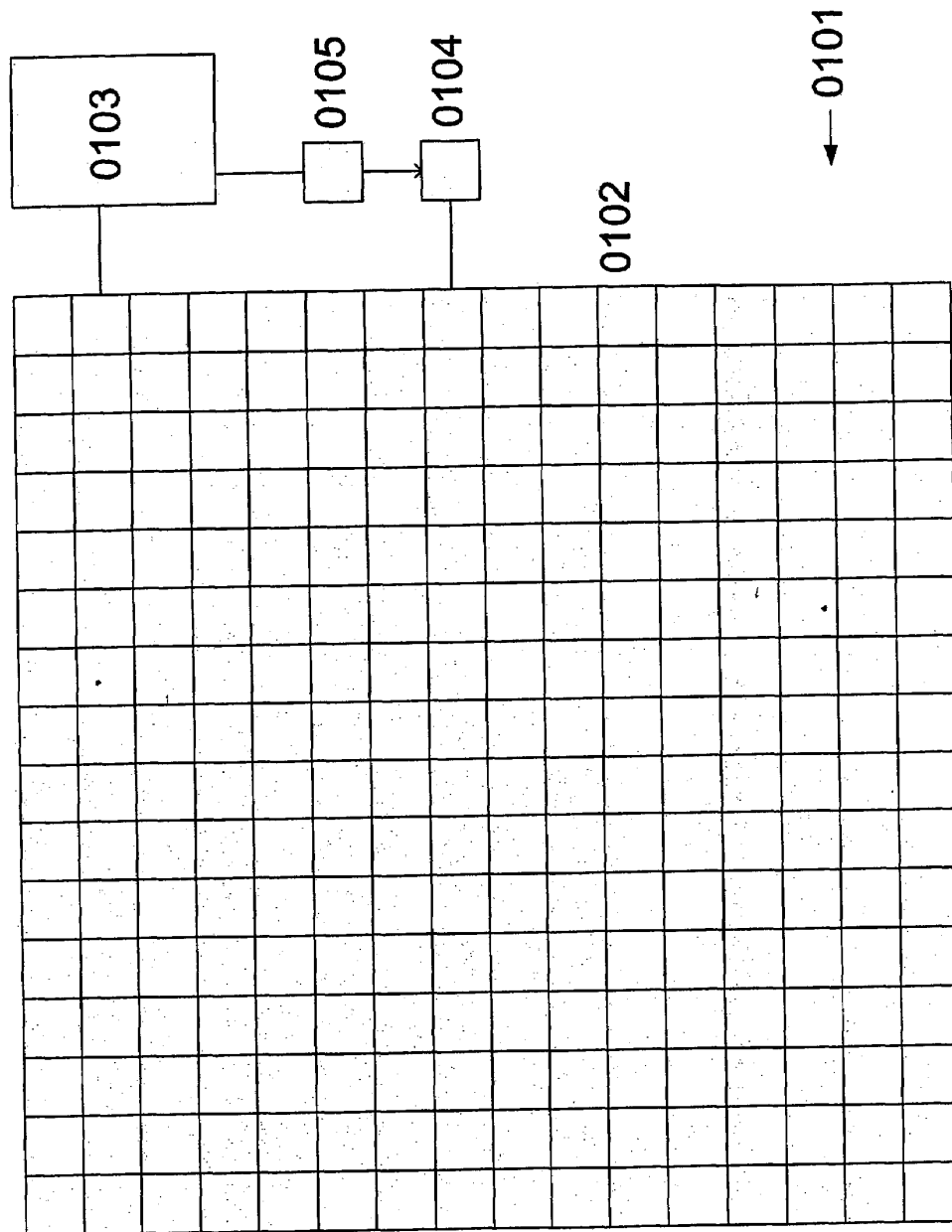
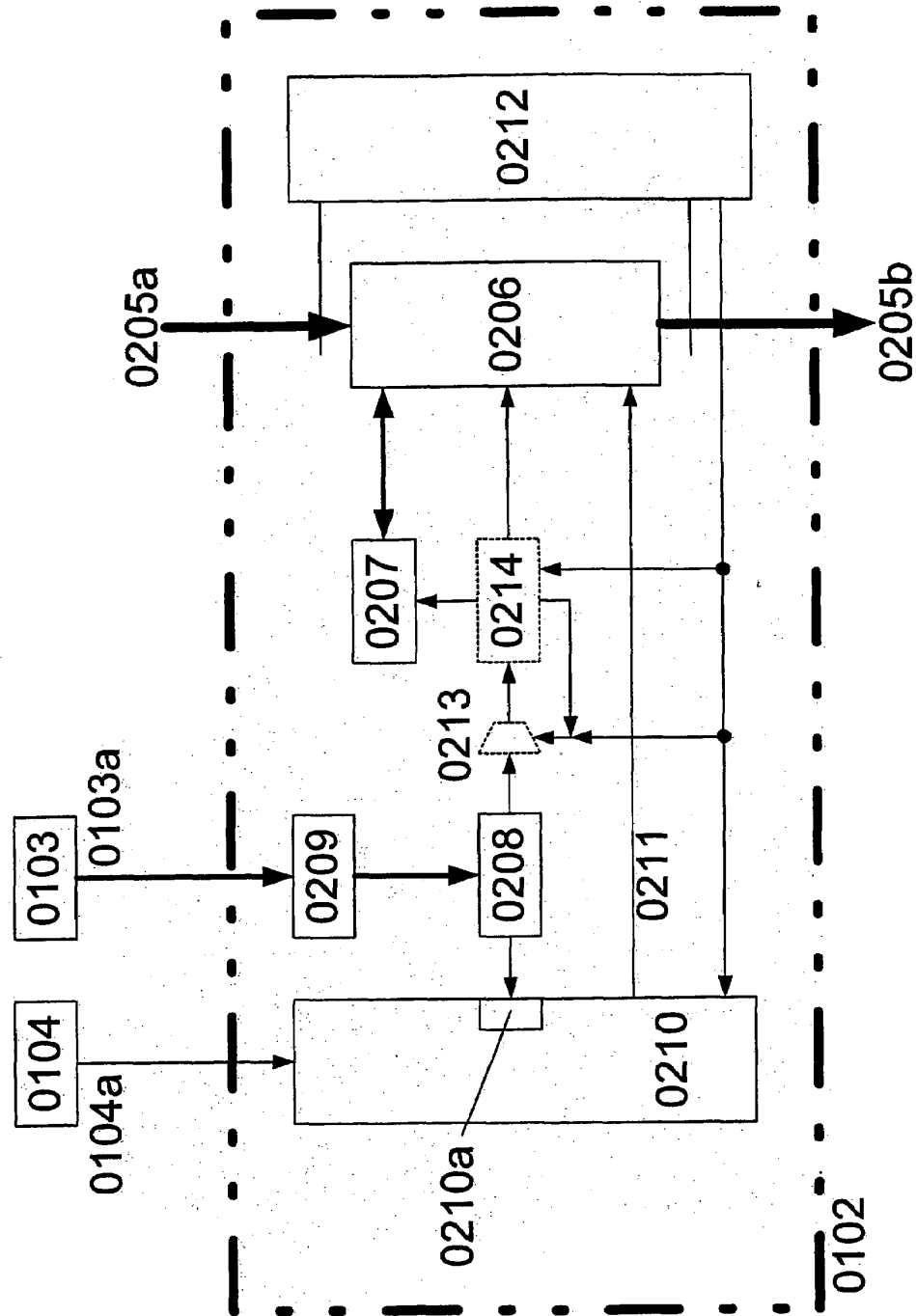


Fig. 1

**Fig. 2**



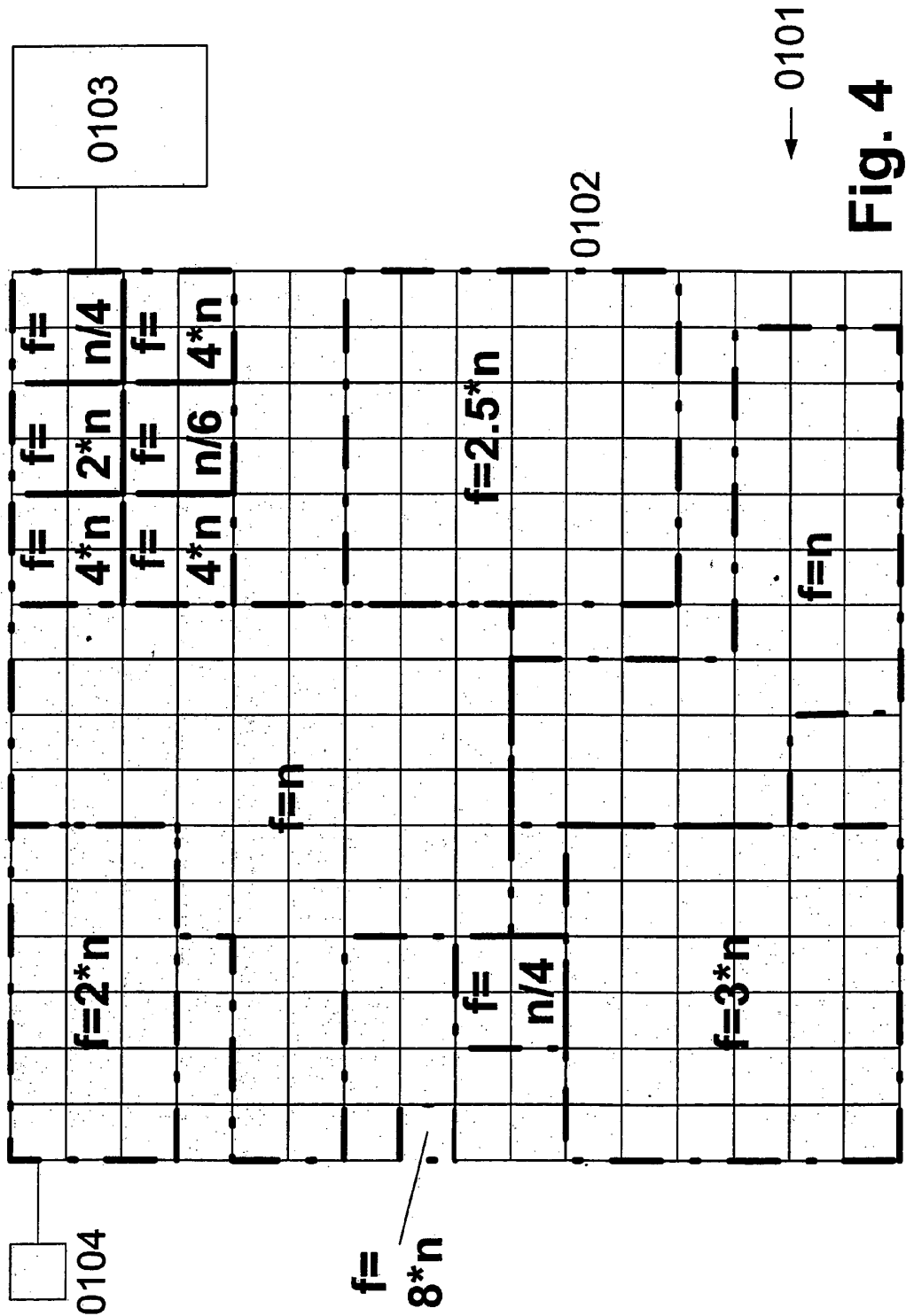
0301

0102a I/O	0102d +	0102g loop+
0102b n.c.	0102e -	0102h /
0102c n.c.	0102f *	0102i +

**Fig. 3a**

+	-	*	loop 16	/	+	I/O
clk	clk	2 clk	16 clk	32 clk	clk	
32	32	16	2	1	32	
8 Mhz	8 Mhz	16 Mhz	128 Mhz	256 Mhz	8 Mhz	
4	4	8	64	128	4	2
1	1	32	16	32	1	1

**Fig. 3b**



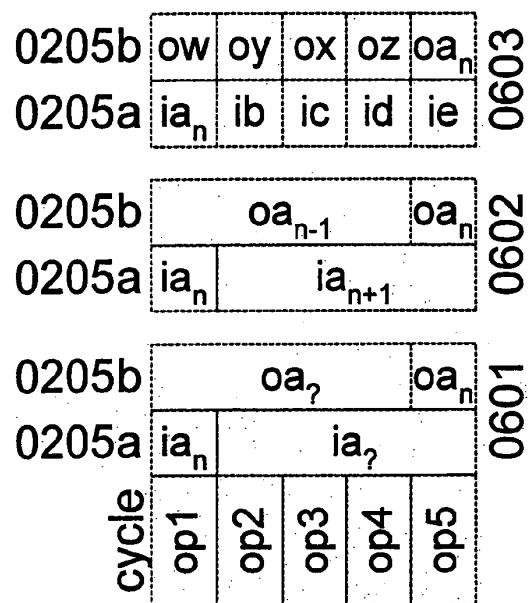


Fig. 6a

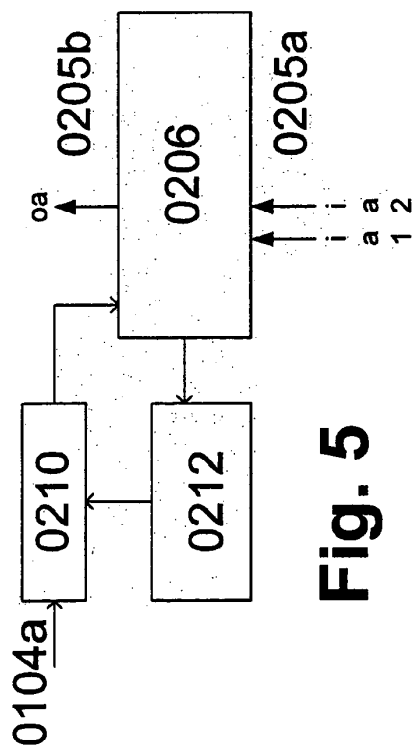


Fig. 5

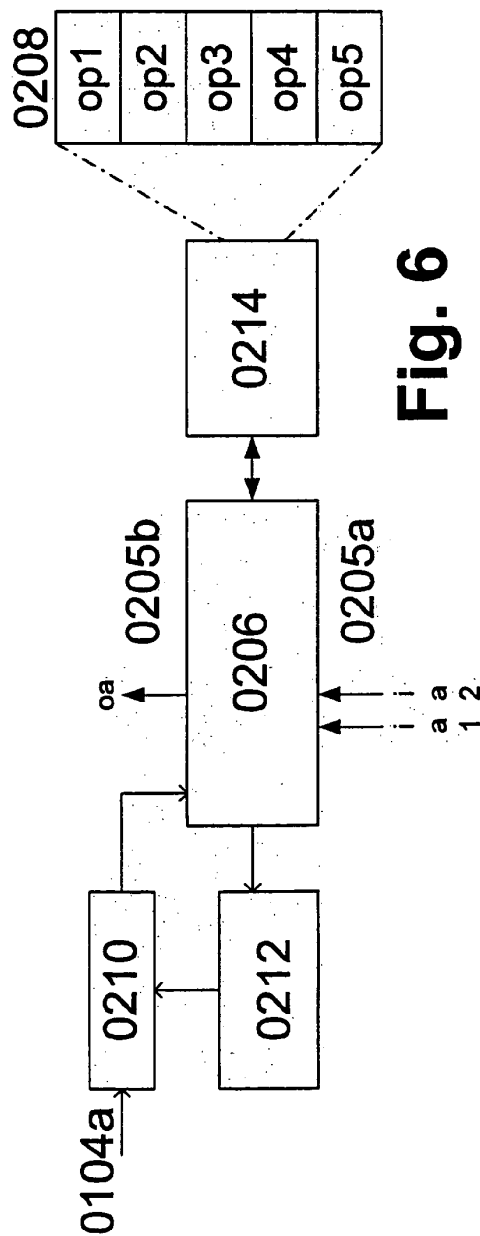
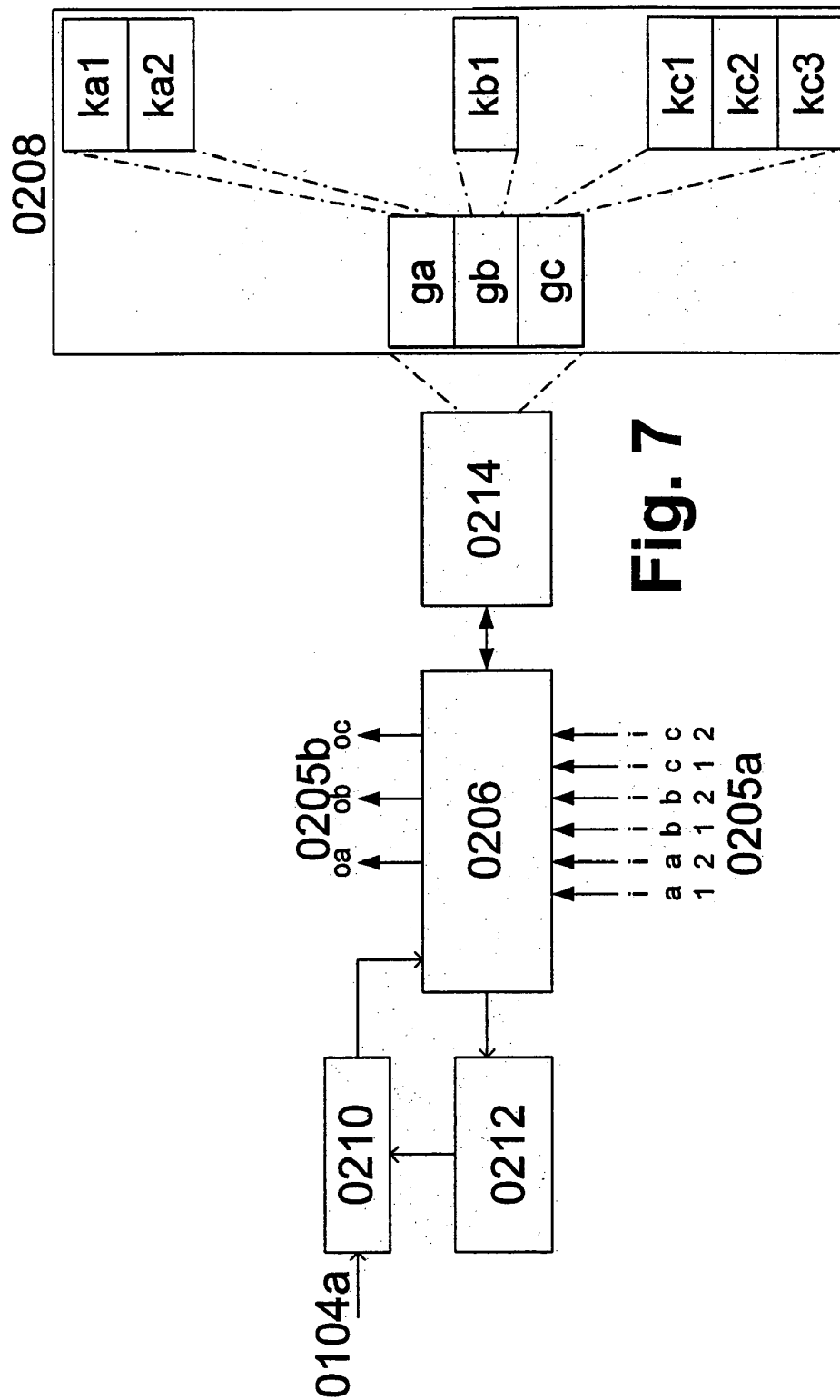


Fig. 6



**Fig. 7**

0205b		oa <sub>n</sub>	ob <sub>n</sub>			oc <sub>n</sub>	0704	
0205a	ia <sub>n</sub>		ib <sub>n</sub>	ic <sub>n</sub>				
0205b	ov	ow	oy	ox	oz	oc <sub>n</sub>	0703	
0205a	in	io	ip	ic <sub>n</sub>	iq	ir		
0205b	oq	or	ob <sub>n</sub>	os	ot	ou		
0205a	ii	ij	ib <sub>n</sub>	ik	il	im		
0205b	ol	oa <sub>n</sub>	om	on	oo	op		
0205a	ia <sub>n</sub>	id	ie	if	ig	ih		
0205b	oc <sub>n-1</sub>					oc <sub>n</sub>	0702	
0205a	ic <sub>n</sub>				ic <sub>n+1</sub>			
0205b	ob <sub>n-1</sub>		ob <sub>n</sub>					
0205a	ib <sub>n</sub>			ib <sub>n+1</sub>				
0205b	oa <sub>n-1</sub>	oa <sub>n</sub>						
0205a		ia <sub>n+1</sub>						
0205b	oc <sub>?</sub>					oc <sub>n</sub>	0701	
0205a	ic <sub>?</sub>			ic <sub>n</sub>	ic <sub>?</sub>			
0205b	ob <sub>?</sub>		ob <sub>n</sub>	ob <sub>?</sub>				
0205a	ib <sub>?</sub>		ib <sub>n</sub>	ib <sub>?</sub>				
0205b	oa <sub>?</sub>	oa <sub>n</sub>	oa <sub>?</sub>					
0205a	ia <sub>n</sub>	ia <sub>?</sub>						
cycle	ka1	ka2	kb1	kc1	kc2	kc3		

Fig. 7a

# METHODS AND DEVICES FOR TREATING AND PROCESSING DATA

## FIELD OF THE INVENTION

The present invention relates to optimization of hardware used in data processing.

## BACKGROUND INFORMATION

Data processing requires the optimization of the available resources, as well as the power consumption of the circuits involved in data processing. This is the case in particular when reconfigurable processors are used.

Reconfigurable architecture includes modules (VPU) having a configurable function and/or interconnection, in particular integrated modules having a plurality of unidimensionally or multidimensionally positioned arithmetic and/or logic and/or analog and/or storage and/or internally/externally interconnecting modules, which are connected to one another either directly or via a bus system.

These generic modules include in particular systolic arrays, neural networks, multiprocessor systems, processors having a plurality of arithmetic units and/or logic cells and/or communication/peripheral cells (IO), interconnecting and networking modules such as crossbar switches, as well as known modules of the type FPGA, DPGA, Chameleon, XPUTER, etc. Reference is also made in particular in this context to the following patents and patent applications of the same applicant:

P 44 16 881.0-53, DE 197 81 412.3, DE 197 81 483.2, DE 196 54 846.2-53, DE 196 54 593.5-53, DE 197 04 044.6-53, DE 198 80 129.7, DE 198 61 088.2-53, DE 199 80 312.9, PCT/DE 00/01869, DE 100 36 627.9-33, DE 100 28 397.7, DE 101 10 530.4, DE 101 11 014.6, PCT/EP 00/10516, EP 01 102 674.7, PCT/DE 97/02949 (PACT02/PCT), PCT/DE 97/02998 (PACT04/PCT), PCT/DE 97/02999 (PACT05/PCT), PCT/DE 98/00334 (PACT08/PCT), PCT/DE 99/00504 (PACT10b/PCT), PCT/DE 99/00505 (PACT10c/PCT), DE 101 39 170.6 (PACT11), DE 101 42 903.7 (PACT11a), DE 101 44 732.9 (PACT11b), DE 101 45 792.8, (PACT11c), DE 101 54 260.7 (PACT11d), DE 102 07 225.6 (PACT11e), PCT/DE 00/01869 (PACT13/PCT), DE 101 42 904.5 (PACT21), DE 101 44 733.7 (PACT21a), DE 101 54 259.3 (PACT21b), DE 102 07 226.4 (PACT21c), PCT/DE 00/01869 (PACT13/PCT), DE 101 10 530.4 (PACT18), DE 101 11 014.6 (PACT18a), DE 101 46 132.1 (PACT18II), DE 102 02 044.2 (PACT19), DE 102 02 175.9 (PACT19a), DE 101 35 210.7 (PACT25), DE 101 35 211.5 (PACT25a), DE 101 42 231.8 (PACT25aII), (PACT25b). The entire contents of these documents are hereby included for the purpose of disclosure.

The above-mentioned architecture is used as an example to illustrate the present invention and is referred to hereinafter as VPU. The architecture includes an arbitrary number of arithmetic, logic (including memory) and/or memory cells and/or networking cells and/or communication/peripheral (IO) cells (PAEs—Processing Array Elements) which may be positioned to form a unidimensional or multidimensional matrix (PA); the matrix may have different cells of any desired configuration. Bus systems are also understood here as cells. A configuration unit (CT) which affects the interconnection and function of the PA through configuration is assigned to the entire matrix or parts thereof. The configuration of a VPU is determined by writing configuration words into configuration registers. Each configuration word determines a subfunction. PAEs may require a plurality of configuration words for their

configuration, e.g., one/or more words for the interconnection of the PAE, one/or more words for the clock determination and one/or more words for the selection of an ALU function, etc.

Generally, a processor which is operated at a higher clock frequency requires more power. Thus, the cooling requirements in modern processors increase substantially as the clock frequency increases. Moreover, additional power must be supplied which is critical in mobile applications in particular.

To determine the clock frequency for a microprocessor based on the state is known. Such technologies are known from the area of mobile computers. However, problems arise in the overall speed with which certain applications are carried out.

## SUMMARY

An object of the present invention is to provide a novel method for commercial application.

In an example embodiment of the present invention, the power consumption may be reduced and/or optimized in VPU technology. As far as different methods are addressed in the following, it should be pointed out that they provide advantages, either individually or in combination.

In a data processing unit (VPU) according to a first aspect of the present invention, by using a field of clocked logic cells (PAEs) which is operable in different configuration states and a clock preselecting means for preselecting logic cell clocking, the clock preselecting means is designed in such a way that, depending on the state, a first clock is preselected at least at a first cell (PAE) and an additional clock is preselected at least at an additional cell (PAE).

It is therefore suggested to operate different cells using different clocking. As a rule, the additional clock corresponds to the first clock; the former is thus situated in a defined phase angle to the latter. In order to achieve optimum data processing results, in particular with regard to the required data processing time, as well as the power consumption of the entire data processing unit, it is suggested that clocking takes place depending on the state, which means that no clock is preselected jointly for all cells based on a certain state, but rather an appropriate clock is assigned to each cell based on the state.

Furthermore, it is suggested that the clocking be designed to be totally configurable, so that one calibration (configuration) mutually influences the clocking of the total number of cells.

It is possible and desired that the clock preselecting means is designed in such a way that it receives the setpoint clock for at least one first cell from a unit which preselects configuration states. This makes it possible to select the clocking of the cell based on its configuration as soon as this configuration is determined. This has the advantage that configuration may take place free of problems.

The unit preselecting configuration states may be a compiling unit, which means that required or desired clocking of the cell is already determined during the compiling of the program. If the compiling unit preselects the configuration states, then the cell configuration preselecting unit may convey clocking for cell configuration to a cell to be configured. This is advantageous since it is possible to merely add clock-determining information to the configuration word or the configuration instruction with which the configuration of a cell is determined, without additional measures being required such as the implementation of clock-assigning buses

which separately transmit the clock-determining signals, or the like; it should be noted that this is possible in principle.

It may also be provided that the clock preselecting means is designed in such a way that it receives the setpoint clock or a clock-influencing signal from one of the other logic cells, in particular a configurable logic cell. This is particularly advantageous if a first logic cell awaits an input signal from an external unit and not until arrival of such signals are the cells to be activated which process subsequently arriving signals. This makes it possible to implement a logic field sleeping mode in which only one or a plurality of cells are activated, if necessary, on a very low level, i.e., very slow clocking, and the remaining field is clocked extremely slowly. The clock frequencies required in the remaining field are dependent on physically necessary clocking which is required for the preservation of memory contents or the like.

It is also advantageous to receive a clock-influencing signal from another logic cell if, using one logic cell, one or a series of a plurality of different arithmetic and/or logical operations may be executed which, at least in part, require a different number of clock cycles, but this may not be determined in advance by the compiling unit. Also in such a case, the subsequent cells do not need to be operated at a high clock frequency if they are appropriately clocked down by corresponding signals which indicate the state of the cell participating in a processing sequence.

In a preferred variant, the clock preselecting means includes a central clock preselecting unit, e.g., a central clock generator, whose clock is transmitted to the individual cells via a clock line, as well as a local clock-generating unit for generating a local clock from and/or in response to the central clock transmitted via the clock line. In a possible embodiment, clocking of the central clock preselecting unit may be set or influenced by a configuration. The local clock-generating unit is preferably implemented by using a frequency divider and/or a frequency multiplier, and the frequency divider ratio is preferably determined by the preselections of the clock preselecting means according to the clock determination based on the state.

In a preferred variant, the logic cells or at least some of the logic cells include at least one ALU and/or are formed by such. It is possible and preferred if some of the logic cells contain at least one memory unit and/or register unit which may be assigned to the remaining logic cells. In particular, this unit may be provided for data to be processed and/or for configurations of the cell.

It is possible that a plurality of logic cells are identical and are operated using different clocking corresponding to their particular configuration. It is possible in particular that all logic cells are identical.

A method is also provided for operating a field of clocked logic cells which may be set into different configuration states, a first state being determined, at least temporarily, for at least one first cell, a clock which is to be assigned to the first cell being determined dependent on the first state and the cell being operated using this clock; a second state is determined for at least one additional cell, a second clock which is to be assigned to the second cell being determined dependent on the second state and the second cell being operated using the second clock which differs from the first clock.

As mentioned above, clocking may be preselected together with the configuration. The state is then the configuration state and/or is at least determined by it.

In known and configurable logic cells, cells are typically combined in groups for executing complex operations. If individual cells execute suboperations which run in fewer clock cycles as is the case with those cells which are [en-

gaged] in particularly drawn-out suboperations of the complex total operations executed by the group, it is preferred if these cells are operated at different clock rates, namely in such a way that the cells for less complex operations, thus operations which run in fewer clock cycles, are clocked slower than the other cells; it is preferred in particular if the cells of one group are clocked collectively in such a way that the number of blank cycles within the group is minimized. An alternative and/or an addition to this lies in the fact of temporarily changing the use of cells burdened with less complex tasks for a certain number of clock cycles, thus changing the use during a fixed number of clock cycles.

In particular, the case may occur that the maximum clock cycle rate of PAEs and/or PAE groups is limited by their function and in particular by their interconnection. The propagation time of signals via bus systems plays an increasingly frequency-limiting role, in particular in advancing semiconductor technology. Henceforth, the method allows slower clocking of such PAEs and/or PAE groups, while other PAEs and/or PAE groups operate at a different and, if needed, higher frequency. It is suggested in a simplified embodiment to make the clock rate of the entire reconfigurable module (VPU) dependent on the maximum clock rate of the slowest PAE and/or PAE group. In other words, the central clock preselecting unit may be configured in such a way that the highest mutual operating clock of all PAEs and/or PAE groups (in other words the smallest common denominator of all maximum clock rates) is globally generated for all PAEs.

The above-described method is particularly advantageous if the cells of the group process data sequentially, i.e., the result determined by one cell is passed on to one or multiple cells which are subsequently processing data.

It should be noted that in addition to prioritizing tasks within the cell field for clock preselection, the condition of a power source may also be included in cell clocking determination. Clocking may be reduced overall in the case of a drop in supply voltage, in particular in mobile applications. Clocking-down for preventing an overtemperature by responding to a temperature sensor signal or the like is equally possible. It is also possible for the user to preset the clock preselection. Different parameters may jointly establish the clock-determining state.

It was mentioned above that it is possible to perform time division multiplexing for carrying out multiple configurations on the same PAE. A preferred and enhanced design makes particularly resource-saving time division multiplexing for carrying out multiple configurations on the same PAE possible; the design may have advantages independently from the different clocking of individual cells, e.g., when latencies have to be taken into account which occur in the signal transmission of digital data via a bus, such as configuration data, data to be processed, or the like. These problems are particularly serious when reconfigurable modules, having reconfigurable units which are located in part comparatively far apart from one another, are to be operated at high clock frequencies. The problem arises here that due to the special configuration of VPUs, a plurality of arbitrary PAEs is connected via buses and considerable data transmission traffic exists via the buses. The switching frequency of transistors is expected to further increase in modern and above all in future silicon technologies, while the signal transmission via buses is to increasingly become a performance-limiting factor. It is therefore suggested to decouple the data rate or frequency on the buses vis-a-vis the operating frequency of the data-processing PAEs.

A particularly simple embodiment, preferred for simple implementations, operates in such a way that the clock rate of

5

a VPU is only globally settable. In other words, a settable clock may be preselected for all PAEs or it may be configured by a higher-level configuration unit (CT). All parameters which have an effect on clocking determine this one global clock. Such parameters may be, for example, a temperature determination, a power reserve measurement of batteries, etc.

A determining parameter may be in particular the maximum operating frequency of the slowest configuration which results as a function of a PAE configuration or a configuration of a group of PAEs. Since different configurations may include different numbers of PAEs over stretches of bus connections of different lengths, it was realized, in particular in bus signal transmission-limiting applications, that configurations may have different maximum frequencies. Configurations may have different maximum frequencies, as is known from FPGAs, for example, which depend on the particular function of the PAEs and in particular on the lengths of bus connections. The slowest configuration then ensures that the proper operation of this configuration is also ensured, and simultaneously reduces the power demand of all other configurations which is advantageous in particular when different portions of the data processing such as through the other configurations, which would possibly run at higher clock frequencies, are not needed prior to the slowest configuration. Also in cases where it must be absolutely ensured that proper operation takes place, the possibly only negligible performance loss occurring by clocking-down other configurations, which could run faster per se, is often acceptable.

In an optimized embodiment, the frequency is adapted only to the configurations which are currently carried out on a VPU, in other words, the global frequency may be reset/reconfigured with each configuration.

In an enhanced embodiment, the clock may then be configured globally, as well as, as described above, individually for each configurable element.

It should be noted that different variants are possible, individually or in combination. In order to show a detailed example, it is assumed in the following, without this necessarily being the case, that the clock may be controlled individually in each PAE. This offers the following possibilities, for example:

a) Controlled Enabling and Disabling of the Clock

It is preferred that the processing clock of PAEs is disabled, i.e., the PAEs operate only in case of need; clock enabling, i.e., activating the PAE, may take place, for example, under at least one of the following conditions, namely

when valid data is present; when the result of the previous computation is approved; due to one or more trigger signals; due to an expected or valid timing mark, compare DE 101 10 530.4 (PACT18).

In order to cause clock enabling, each individual condition may be used either individually or in combination with other conditions, clock enabling being computed based on the logical combination of conditions. It should be noted that it is possible to put the PAEs into a power-saving operating mode while a clock is disabled, for example, through additionally partly switched-off or reduced power supply, or, should it be necessary because of other reasons, through extremely reduced sleeping clocks.

b) Different Frequencies Per PAE

Technologies for controlling sequences in VPUs are known from PCT/DE 97/02949 (PACT02/PCT), PCT/DE 97/02998 (PACT04/PCT), and PCT/DE 00/01869 (PACT13/PCT). Special sequencers (SWTs) which control a large number of PAEs and which are responsible for their (re)configu-

6

ration are configured in PCT/DE 97/02998 (PACT04/PCT). The (re)configuration is controlled by using status signals which are generated by the PAEs (triggers) and passed on to the SWTs, namely in that the SWT responds to the triggers, making the particular continuation of a sequence dependent on the triggers.

A small memory for their configuration is assigned to each individual PAE in PCT/DE 97/02949 (PACT02/PCT). A sequencer passes through the memory and addresses the individual configurations. The sequencer is controlled by triggers and/or by the status of its PAE (into which it may be integrated, for example).

During data processing, it is now possible that different sequencers in different PAEs have to carry out a different number of operations per transmitted data packet (compare DE 101 39 170.6 (PACT11), DE 101 42 903.7 (PACT11a), DE 101 44 732.9 (PACT11b), DE 101 45 792.8 (PACT11c), DE 101 54 260.7 (PACT11d), DE 102 07 225.6 (PACT11e), PCT/DE 00/01869 (PACT13/PCT)). This is described using a configuration as an example in which 3 sequencers are involved in processing a data packet, requiring a different number of operations for data packet processing. Example:

Sequencer 1 (Seq1) requires 10 operations for processing a data packet,

Sequencer 2 (Seq2) requires 5 operations for processing a data packet,

Sequencer 3 (Seq3) requires 20 operations for processing a data packet.

In order to obtain an optimum operation/power consumption ratio, the individual sequencers would have to be clocked as follows:

$$F_{\max} = F_{Seq2}/4 = F_{Seq1}/2 = F_{Seq3}$$

or at a maximum operating frequency of, for example, 100 MHz:  $F_{Seq1}=50$  MHz,  $F_{Seq2}=25$  MHz,  $F_{Seq3}=100$  MHz.

It is suggested in particular to use different clock sources for each PAE and/or group of PAEs. For example, different techniques may be used for this purpose, either individually or jointly:

1) Clock dividers, individually programmable per PAE, which enable an individually configurable divider ratio based on one or more mutual base clocks.

2) Clock multipliers (PLLs), individually programmable per PAE, which enable an individually configurable divider ratio based on one or more mutual base clocks.

3) Deriving the particular PAE clock from the data stream of the particular data to be processed, e.g., by oversampling.

An exemplary embodiment having different algorithms is illustrated in FIG. 1.

c) Configuration Clock

Optimization of the power consumption is also favored in that the circuit components, necessary for executing a configuration, are clocked selectively, i.e., it is suggested to clock each PAE addressed and/or to completely disable the clock of those circuit components necessary for executing a configuration or a reconfiguration when no configuration or reconfiguration is being executed and/or to use static registers.

In particular example embodiments, the operating frequency of the PAEs or groups of PAEs may be made dependent on different and/or additional factors. The following is listed below as an example:

1. Temperature Measurement

If the operating temperature reaches certain threshold values, the operating clock is reduced correspondingly. The



reduction may take place selectively by initially operating those PAEs on a lower clock which represent the most irrelevant performance loss.

In a particularly preferred embodiment, multiple temperature measurements may be performed in different regions and clocking may be adapted locally.

## 2. Buffer Filling Levels

IO-FIFOs (input-output-first-in-first-out-circuits) which decouple peripheral data transmissions from data processing within a VPU are described in DE 102 06 653.1 (PACT15), DE 102 07 224.8 (PACT15a), (PACT15b). One buffer for input data (input buffer) and/or one buffer for output data (output buffer) may be implemented, for example. A particularly efficient variable for determining the clock frequency may, for example, be determined from the filling level of the particular data buffers. The following effects and measures may occur, for example:

- a) An input buffer is largely full and/or the filling level rises abruptly: Clocking increase to accelerate processing.
- b) An input buffer is largely empty and/or the filling level drops abruptly: Clocking decrease to decelerate processing.
- c) An output buffer is largely full and/or the filling level rises abruptly: Clocking decrease to decelerate processing.
- d) An output buffer is largely empty and/or the filling level drops abruptly: Clocking increase to accelerate processing.

Depending on the application and the system, suitable combinations may be implemented accordingly.

It should be pointed out that such a clock frequency determination is implementable if a filling level determination means for a buffer, in particular an input and/or output buffer, alternatively also an intermediate buffer within a VPU array, is provided and if this filling level determination means is connected to a clock preselecting means for preselecting logic cell clocking so that this clock preselecting means is able to change the logic cell clocking in response to the buffer filling level.

## 3. Battery Charge State

It is imperative to be careful with the power supply, e.g., a battery, for mobile units. Depending on the power reserve, which may be determined based on the existing methods according to the related art, the frequency of PAEs and/or groups of PAEs is determined and is reduced in particular when the power reserve is low.

Besides or in addition to optimizing data processing clocking it is also possible to accomplish an optimization of the data transmission with respect to the relationship between data transmission and data processing.

In a particular embodiment, the clock controls of PAEs described may be enhanced in such a way that, by using a sequencer-like activation and a suitable register set, for example, multiple, preferably different, configuration words may be executed successively in multiple clocks. A sequencer, sequentially processing a number of configuration inputs, may be additionally assigned to the configuration registers and/or to a configuration memory which is possibly also decoupled and implemented separately (compare DE 102 06 653.1 (PACT15), DE 102 07 224.8 (PACT15a, PACT15b)). The sequencer may be designed as a microcontroller. In particular, the sequencer may be programmable/configurable in its function such as Altera's module EPS448 (ALTERA Data Book 1993). Possible embodiments of such PAEs are described, for example, in the following patent applications which are included in their entirety for the purpose of disclosure: PCT/DE 97/02949 (PACT02/PCT), PCT/

DE 97/02998 (PACT04/PCT), PCT/DE 00/01869 (PACT13/PCT), DE 101 10 530.4 (PACT18), DE 102 06 653.1 (PACT15), DE 102 07 224.8 (PACT15a, PACT15b).

For the following, it is initially assumed that multiple configuration words are combined into one configuration (PACKEDCONF) and are configured on a PAE. The PACKEDCONF is processed in such a way that the individual configuration words are executed in chronological succession. The data exchange and/or status exchange between the individual timed configurations takes place via a suitable data feedback in the PAEs; for example by using a suitable register set and/or another data exchange and/or status exchange means such as suitable memories and the like.

This method allows a different timing for PAEs and bus systems. While PAEs process data at very high clock rates, for example, operands and/or results are transmitted via a bus at only a fraction of the clock rate of the PAEs. The transmission time via a bus may be correspondingly longer.

It is preferred if not only the PAEs or other logic units in a configurable and/or reconfigurable module are clockable at a different rate, but also if different clocking is provided for parts of a bus system. It is possible here to provide multiple buses in parallel whose speed is clocked differently, i.e., a bus which is clocked particularly high for providing a high-performance connection, parallel to a bus which is clocked lower for providing a power-saving connection. The connection clocked high may be used when longer signal paths have to be compensated, or when PAEs, positioned close together, operate at a high frequency and therefore also have to exchange data at a high frequency in order to provide a good transmission here over short distances in which the latency plays a minor role at best. Therefore, it is suggested in a possible embodiment that a number of PAEs, positioned together locally and combined in a group, operate at a high frequency and possibly also sequentially and that local and correspondingly short bus systems are clocked high corresponding to the data processing rate of the group, while the bus systems, inputting the operands and outputting the results, have slower clock and data transmission rates. For the purpose of optimizing the power consumption, it would be alternatively possible to implement slow clocking and to supply data at a high speed, e.g., when a large quantity of inflowing data may be processed with only a minor operational effort, thus at low clock rates.

In addition to the possibility of providing bus systems which are clocked using different frequencies it is also possible to provide multiple bus systems which are operable independently from one another and to then apply the PAEs in a multiplex-like manner as required. This alone makes it possible to operate reconfigurable modules particularly efficiently in resource multiplexing, independently from the still existing possibility of differently clocking different bus systems or different bus system parts. It is possible here to assign different configurations to different resources according to different multiplexing methods.

According to PCT/DE 00/01869 (PACT13/PCT), a group of PAEs may be designed as a processor in particular.

In the following embodiments, for example, different configurations are assigned to data-processing PAEs using time-division multiplexing, while bus systems are assigned to the different configurations using space-division multiplexing.

In the assignment of resources, i.e., the assignment of tasks to PAEs or a group of PAEs to be carried out by the compiler or a similar unit, the given field may then be considered as a field of the n-fold variable and code sections may be transferred to this field of resources, which is virtually scaled up by the factor n, without the occurrence of problems, particularly

when code sections are transferred in such a way that no interdependent code sections have to be configured into a PAE which is used in a multiplex-like manner.

In the previous approach, a PACKEDCONF was composed of at least one configuration word or a bundle of configuration words for PAEs which belong to one single application. In other words, only configuration words which belong together were combined in the PACKEDCONF.

In an enhanced embodiment, at least one or more configuration words per each different configuration are entered into a PACKEDCONF in such a way that the configuration word or words which belong together in a configuration are combined in a configuration group and the configuration groups thus created are combined in the PACKEDCONF.

The individual configuration groups may be executed in chronological succession, thus in time-division multiplexing by a timeslice-like assignment. This results in time division multiplexing of different configuration groups on one PAE. As described above, the configuration word or the configuration words within a configuration group may also be executed in chronological succession.

Multiplexers which select one of the configuration groups are assigned to the configuration registers and/or to a configuration memory, which is possibly also decoupled and implemented separately (compare DE 102 06 653.1 (PACT15), DE 102 07 224.8 (PACT15a, PACT15b)). In an enhanced embodiment, a sequencer (as described above) may be additionally assigned which makes the sequential processing of configuration words within configuration groups possible.

Using the multiplexers and the optional sequencer, a resource (PAE) may be assigned to multiple different configurations in a time-division multiplex method.

Among one another, different resources may synchronize the particular configuration group to be applied, for example by transmitting a configuration group number or a pointer.

The execution of the configuration groups may take place linearly in succession and/or cyclically, with a priority being observed. It should be noted here in particular that different sequences may be processed in a single processor element and that different bus systems may be provided at the same time so that no time is wasted in establishing a bus connection which may take some time due to the long transmission paths. If a PAE assigns its first configuration to a first bus system and, on execution of the first configuration, couples the same to the bus system, then it may, in a second configuration, couple a different or partially different bus system to the former if spacial multiplexing for the bus system is possible.

The execution of a configuration group, each configuration group being composed of one or more configuration words, may be made dependent on the reception of an execution release via data and/or triggers and/or an execution release condition.

If the execute release (condition) for a configuration group is not given, the execute release (condition) may either be awaited, or the execution of a subsequent configuration group may be continued. The PAEs preferably go into a power-saving operating mode during the wait for an execute release (condition), for example with a disabled clock (gated clock) and/or partially disabled or reduced power supply. If a configuration group cannot be activated, then, as mentioned above, the PAEs preferably also go into a power-saving mode.

The storage of the PACKEDCONF may take place by using a ring-type memory or other memory or register means, the use of a ring-type memory resulting in the fact that after the execution of the last input, the execution of the first input may be started again (compare PCT/DE 97/02998 (PACT04/PCT)). It should be noted that it is also possible to skip to a

particular execution directly and/or indirectly and/or conditionally within the PACKEDCONF and/or a configuration group.

In a preferred method, PAEs may be designed for processing of configurations in a corresponding time-division multiplexing method. The number of bus systems between the PAEs is increased such that sufficient resources are available for a sufficient number of configuration groups. In other words, the data-processing PAEs operate in a time-division multiplex method, while the data-transmitting and/or data-storing resources are adequately available.

This represents a type of space division multiplexing, a first bus system being assigned to a first temporarily processed configuration, and a second bus system being assigned to an additional configuration; the second bus system runs or is routed spacially separated from the first bus system.

It is possible at the same time and/or alternatively that the bus systems are also entirely or partially operated in time-division multiplexing and that multiple configuration groups share one bus system. It may be provided here that each configuration group transmits its data as a data packet, for example, a configuration group ID being assigned to the data packet (compare APID in DE 102 06 653.1 (PACT15), DE 102 07 224.8 (PACT15a, PACT15b)). Subsequently it may be provided to store and sort the particular data packets transmitted based on their assigned identification data, namely between different buses if required and for coordinating the IDs.

In an enhanced method, memory sources may also be run in a time-division multiplex, e.g., by implementing multiple segments and/or, at a change of the configuration group, by writing the particular memory/memories as described in PCT/DE 97/02998 (PACT04/PCT) and/or PCT/DE 00/01869 (PACT13/PCT) into a different or even external memory or by loading from the same. In particular the methods as described in DE 102 06 653.1 (PACT15), DE 102 07 224.8 (PACT15a, PACT15b) may be used (e.g., MMU paging and/or APID).

The adaptation of the operating voltage to the clock should be noted as a further possibility for conserving resources.

Semiconductor processes typically allow higher clock frequencies when they are operated at higher operating voltages. However, this causes substantially higher power consumption and may also reduce the service life of a semiconductor.

An optimum compromise may be achieved in that the voltage supply is made dependent on the clock frequency. Low clock frequencies may be operated at a low supply voltage, for example. With increasing clock frequencies, the supply voltage is also increased (preferably up to a defined maximum).

#### BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows a reconfigurable data processing unit.

FIG. 2 shows a configuration unit feeding configuring data via a configuration line into a respective cell.

FIG. 3a shows a 3x3 field of configurable cells.

FIG. 3b shows a table indicating numbers of clock cycles and clock rates for the cells of FIG. 3a.

FIG. 4 shows the data processing unit (VPU) according to FIG. 1.

FIG. 5 shows an example embodiment of the operation of a PAE according to FIG. 2.

FIG. 6 shows an example embodiment of the operation of a PAE, including activation of a sequencer.

FIG. 6a shows an example of the transmission of data on a data bus.

FIG. 7 shows an example embodiment of the operation of a PAE, in which multiple configuration groups share the PAE.

FIG. 7a shows example bus transmission in accordance with FIG. 7.

#### DETAILED DESCRIPTION OF EXAMPLE EMBODIMENTS

The present invention, as an example, is explained in greater detail below with reference to the Figures. It should be noted that this exemplary description is not limiting and that in isolated cases and in different figures identical or similar units may be denoted using different reference numbers.

As an example, FIG. 1 shows a reconfigurable data processing unit (VPU) (0101). A configuration unit (CT, 0103) for the control and execution of the configuration and reconfiguration is superordinated to an array of PAEs (0102) which are configurable and reconfigurable independently from one another. In this connection, particular reference is made to the various applications of the applicant and the disclosure content of the patents and technologies described above. In addition, a central clock generator (0104) is assigned to the data processing unit. In a possible embodiment, the clock rate of the central clock generator may be preselected by configuration unit 0103. In a possible embodiment, the clock rate of each PAE and/or groups of PAEs and their bus connections may also be preselected by configuration unit 0103.

According to FIG. 2, configuration unit 0103 feeds configuring data via a configuration line 0103a into respective cells 0102 of which only one is illustrated as an example. Furthermore, the clock signal of central clock generator 0104 is fed to cell 0102 via a clock line 0104a. Via a data bus input 0205a and a data bus output 0205b, reconfigurable cell 0102 communicates with other cells and additionally has a data processing unit, e.g., an arithmetic logic unit ALU 0206, and preferably an internal data memory 0207 and a configuration memory 0208 into which configuring instructions from configuration unit 0103 are fed via a configuration instruction extractor 0209 in order to configure the data processing unit, e.g., ALU 0206, as a response. In addition, configuration (instruction) extractor 0209 is connected to a frequency divider/multiplier factor preselecting input 0210a of a frequency divider/frequency multiplier 0210 which is designed to divide or multiply the clock signal of central clock generator 0104 on clock line 0104a according to a clock ratio preselected via input 0210a and to feed the clock signal to the data processing unit, e.g., arithmetic logic unit ALU 0206, and possibly other units of reconfigurable cell 0102 via a line 0211. Using an optional data bus monitoring circuit 0212, 0210 may be activated in such a way that the frequency is controlled depending on the data reception or the data transmission.

Furthermore, a multiplexer 0213 for selecting different configurations and/or configuration groups may optionally be integrated dependent on 0212. Furthermore, the multiplexer may optionally be activated by a sequencer 0214 in order to make sequential data processing possible. In particular, intermediate results may be managed in data memory 0207.

While the general configuration of the cell was described in part in the applicant's applications described above, the presently described clock dividing system, the associated circuit, and the optimization of its operation are at least novel and it should be pointed out that these facts may and shall be associated with the required hardware changes.

The entire system and in particular configuration unit 0103 is designed in such a way that, together with a configuring signal with which a configuration word is fed via configura-

tion line 0103a via configuration word extractor 0209 to data processing unit 0206 or upstream and/or downstream and/or associated memory 0208, a clock dividing/multiplying signal may also be transmitted which is extracted by configuration word extractor 0209 and transmitted to frequency divider/multiplier 0210, so that, as a response, 0210 may clock data processing unit 0206 and possibly also other units. It should be pointed out that, as a response to an input signal to the cell, there are also other possibilities instead of unit 0209 to vary clocking of an individual data processing unit 0206 with reference to a central clock unit 0104, via data bus monitoring circuit 0212, for example.

Described only as an example with reference to FIGS. 3 and 4, an entire field of all reconfigurable logic units 0102 may be operated using the above-described embodiment, but possibly also by implementing the units in a different way.

For example, a 3x3 field of reconfigurable cells is configured in such a way, according to FIG. 3a, that a first cell 0102a is used for analyzing an input/output signal. Cells 0102b, 0102c are presently not needed and are therefore denoted as not configured (n. c.). Cells 0102d through 0102i together form a group which executes a complex arithmetic operation; an addition takes place in cell 0102d, a subtraction takes place in cell 0102e, a multiplication takes place in cell 0102f, a loop is run in cell 0102g, a multiple addition being executed within the loop, a division takes place in cell 0102h, and an addition in turn takes place in cell 0102i. Cells 0102d through 0102i are connected to one another in group 0301, indicated by dot and dash lines, in such a way that data is sequentially and pipeline-like processed by the cells. As is indicated in the second row of the table in FIG. 3b, the operations within cells 0102d and 0102e are executed in a different number of clock cycles. The number of clock cycles is denoted there and it is clear that an addition or a subtraction may be executed in one clock cycle; the division, however, requires 32 clock cycles. The third line of the table in FIG. 3b denotes which value is assigned to the frequency divider of each cell in order to achieve optimum power usage at a constant data throughput through the cell. Only the cell in which the division takes place is operated at the highest clock; the clock ratio here is 1. This cell requires the longest time for the operation assigned to it. Since a new result has to be delivered only every 32 clock pulses to cell 0102h executing the division, cells 0102d and 0102e are clocked slower by the appropriate factor of 32; the frequency divider ratio for these cells is therefore 32, as can be seen in FIG. 3b. The multiplication running in two clock cycles has a frequency divider ratio of 16, and the more complex loop of cell 0102g running in 16 clock cycles is assigned a frequency divider ratio of only 2. These clock ratios are initially known at the configuration, in which the individual cells are compiled in groups and are assigned to each cell within the group since they were determined by the compiler at program compilation and may therefore be input into the cell at its configuration. It is denoted in the fourth row from the top which clock rate results from a central clock of 256 MHz.

If the processor unit having the separately clockable reconfigurable logic cells is operated in an application where the voltage may drop, e.g., due to exhausting voltage supply capacities, it may be provided that, at a drop in the supply voltage, the entire frequency is reduced to a critical value U1; all cells are subsequently clocked slower by one half so that division cell 0102h too runs only at 128 MHz, while cell 0102d is clocked at 4 MHz. Cell 0102a, executing a query of the mouse pointer having a lower priority, is no longer clocked at 8 MHz as previously but rather at 2 MHz, i.e., depending on the prioritization, different slowdowns accord-

ing to the importance of the task are assigned to the respective groups at a voltage drop or under other circumstances.

If, for other reasons, the temperature still rises, the heat generation in the logic cell field may be further reduced by an additional clock rate reduction for the logic cells, as is indicated in the last row of FIG. 3b. It is understood that, for example, a particular individual sensor for determining the condition such as the supply voltage and/or the temperature may be provided whose sensor signal is fed to the cells in a conditioned manner; a corresponding sensor system may be assigned to each cell and/or the central clock is possibly modifiable.

This makes it possible to optimally operate a processor field energy-efficiently; the cooling capacity required is reduced and it is clear that, since as a rule not all cells may and/or must be permanently operated at the highest clock frequency, heat sinks and the like may be dimensioned appropriately smaller which in turn offers additional cost advantages.

It should be noted that in addition to the query regarding a supply voltage, a temperature, the prioritization of computations, and the like, other conditions may determine the clock. For example, a hardware switch or a software switch may be provided with which the user indicates that only low clocking or higher clocking is desired. This makes an even more economical and targeted handling of the available power possible. It may be provided in particular that, at the user's request or at an external request, the central clock rate in total may be reduced; the clock divider ratios within the cell array, however, are not changed in order to avoid the requirement of reconfiguring all cells, e.g., at an extreme temperature rise. Moreover, it should be pointed out that a hysteresis characteristic may be provided in determining the clock rates, when a temperature-sensitive change of the clock frequencies is to be performed, for example.

FIG. 4 shows the data processing unit (VPU) according to FIG. 1. Different groups within the VPU are operated using different frequencies  $f$  which are derived from a frequency normal  $n$  generated by 0104. It should be expressly noted that multiple frequency normals ( $n_1 \dots n_m$ ) may be generated by multiple 0104 and may be used within one VPU.

FIG. 5 shows a simple exemplary embodiment for the operation of a PAE according to FIG. 2. A data bus (0205a) delivers operands ia1 and ia2 to an ALU (0206) which in turn delivers the result of the computation oa to 0205b. The PAE is only activated, i.e., clocked and/or supplied with current, when data bus monitoring circuit 0212 recognizes the acceptance of the previous result oa by the receiver and the arrival of operands ia1 and ia2 necessary for the operation. In other words, the PAE is only activated when all working conditions and requirements are met. The clock release is carried out by 0210, the clock source is 0104a.

FIG. 6 corresponds to FIG. 5 with the exception that a sequencer (0214) is additionally activated which controls a multiclock configuration (e.g., a complex computation such as a matrix multiplication or the like). The sequencer extracts the operations from the configuration memory or from a section of the configuration memory. In the example shown, operations op1, op2, op3, op4, op5 are carried out sequentially. Result oa is conveyed after completion and the PAE has to be activated again.

The data transmission occurring on data bus 0205a/b is illustrated in FIG. 6a. It should be pointed out that the data routing via the bus may take place in a conventional manner, i.e., collision and deadlock situations may be prevented for one configuration at a time in a conventional manner.

In order to execute op1, operands ia must be available via 0205a (0601); the data transmissions for the remaining cycles may be undefined in principle.

Thereafter, 0205a may preferably transmit the subsequent operands (0602) for which the execution time of op2, op3, op4, op5 is available, thus creating a temporal decoupling, allowing the use of slower and/or, in particular, longer bus systems.

During the execution of op2, op3, op4, op5, data of other configurations may alternatively (0603) be transmitted via the same bus system 0205a using a time-division multiplex method.

Following op5, result oa is applied to bus 0205b (0601); the data transmissions for the remaining cycles may be undefined in principle.

The time prior to op5, i.e., during the execution of op1, op2, op3, op4, may be used for transmitting the previous result (0602). This again creates a temporal decoupling, allowing the use of slower and/or, in particular, longer bus systems.

During the execution of op1, op2, op3, op4, data of other configurations may alternatively (0603) be transmitted via the same bus system 0205b using a time-division multiplex method. For clock multiplication, 0210 may use a PLL. A PLL may be used in particular in such a way that the operating clock of the PAE for executing op1, op2, op3, op4, op5 is five times that of the bus clock. In this case, the PAE may act as a PAE without a sequencer having only one (uniclock) configuration and the same clock as the bus clock.

FIG. 7 corresponds to FIG. 6 plus the addition that multiple configuration groups (ga, gb, gc) share the PAE in a time-division multiplexed manner and each group has connections to a separate (space-division multiplexed) bus system (ia/oa, ib/ob, ic/oc). A multiplexer in 0214 cyclically selects the groups ga, gb, gc. Provided the data monitoring circuit 0212 generates a valid execution release (condition) for a configuration group, the particular configuration group is executed; otherwise the execution release (condition) may be awaited or, preferably, a different subsequent configuration group may be selected. The configuration groups may be run through cyclically.

One configuration group may contain multiple configuration words ( $ga=\{ka1, ka2\}$ ,  $gb=\{kb1\}$ ,  $gc=\{kc1, kc2, kc3\}$ ). The configuration words may be executed sequentially in 0214 using a sequencer.

FIG. 7a shows the bus transmissions according to the example in FIG. 7. 0701 corresponds to 0601, 0702 corresponds to 0602, 0703 corresponds to 0603; a separate bus system is used thereby for each group ga, gb, gc.

In addition, a possible bus transmission using a time-division multiplex for the bus systems is illustrated in 0704. The input data of all groups is transmitted via an input bus system and the output data of all groups is transmitted via an output bus system. The undefined intermediate cycles are either unused or are free for other data transmissions.

What is claimed is:

1. A method of data processing by operating a matrix of clocked functional units which are configurable with respect to clock rate, interconnection, and functionality, comprising:
  - setting a configuration for execution of a complex operation on data packets, wherein the setting of the configuration comprises:
    - selecting a group of the functional units for the configuration, the group including at least a first functional unit and a second functional unit; and
    - for each functional unit of the group setting a respective functionality of the functional unit; and
  - determining a first clock rate for the first functional unit for execution of its set functionality and a second clock rate for the second functional unit for execution of its set functionality, the second clock rate differing from the first clock rate;

15

wherein:

each of the first and second clock rates is generated from a global master clock; and

the first clock rate is determined, at least in part, based on the functionality set for the second functional unit.

2. The method as recited in claim 1, further comprising: preselecting a clock rate for at least one of the functional units, the clock rate being preselected one of together with its configuration or determined by its configuration.

3. The method as recited claim 2, wherein the group of the functional units is jointly configured for executing algebraic or other operations which require a different number of clock cycles and wherein at least one of the functional units, executing an operation which requires fewer clock cycles than an operation requiring the most clock cycles within the group, is clocked slower than at least one other functional unit.

4. The method as recited in claim 3, further comprising: configuring at least one group of functional units for sequential data processing.

5. The method as recited in claim 3, further comprising: configuring at least two groups of the matrix for executing at least two different tasks which are assigned different priorities, a first one of the groups, appointed for executing a task having a lower priority than that of a second one of the groups, being clocked using a lower clock frequency than that of the second one of the groups.

6. The method as recited in claim 3, further comprising: determining at least one of a condition of a voltage supply source and a temperature, wherein at least one of the first clock rate and the second clock rate is determined as a function of the at least one of the voltage and temperature.

16

7. The method as recited in claim 4, further comprising: determining at least one of a condition of a voltage supply source and a temperature, wherein at least one of the first clock rate and the second clock rate is determined as a function of the at least one of the voltage and temperature.

8. The method as recited in claim 4, further comprising: configuring at least two groups of the matrix for executing at least two different tasks which are assigned different priorities, a first one of the groups, appointed for executing a task having a lower priority than that of a second one of the groups, being clocked using a lower clock frequency than that of the second one of the groups.

9. The method as recited in claim 6, further comprising: configuring at least two groups of the matrix for executing at least two different tasks which are assigned different priorities, a first one of the groups, appointed for executing a task having a lower priority than that of a second one of the groups, being clocked using a lower clock frequency than that of the second one of the groups.

10. The method as recited in claim 7, further comprising: configuring at least two groups of the matrix for executing at least two different tasks which are assigned different priorities, a first one of the groups, appointed for executing a task having a lower priority than that of a second one of the groups, being clocked using a lower clock frequency than that of the second one of the groups.

\* \* \* \* \*