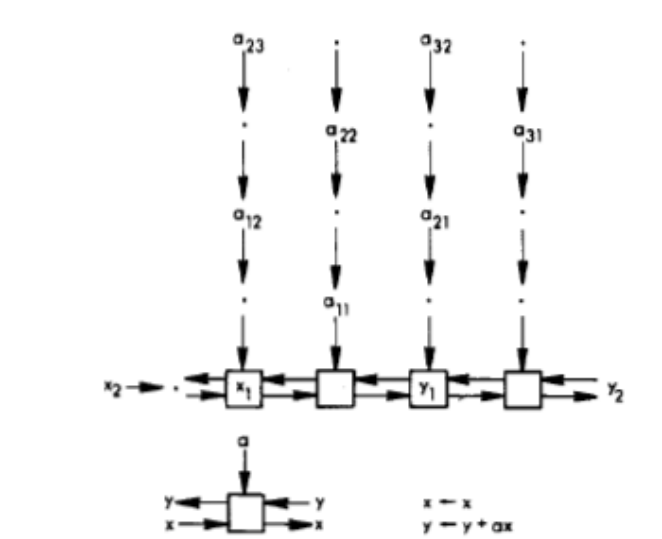
Asserted Claim of '800 Patent	Exemplary Disclosure of Gaudiot
[1A] A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:	At least under Plaintiff's apparent theories of infringement and interpretations of th alleging that any of Defendant's accused products satisfy this claim limitation, Gau combination with one or more references, discloses: Gaudiot at Abstract ("New technologies of integration allow the design of powerfu may include several thousands of elementary processors. These multiprocessors m range of applications in signal and data processing. However, assuring the proper i large number of processors and the ultimate safe execution of the user programs pr scheduling problem. The scheduling of operations upon the availability of their op termed the data-driven mode of execution and offers an elegant solution to the issu is described in this paper and several architectures which have been proposed or in (systolic arrays, data-flow machines, etc.) are examined in detail. The problems ass data-driven execution are also studied. A multi-level approach to high-speed digita processing is then evaluated.") Gaudiot at 1224-25 ("A. Systolic Arrays [32] The primary goal of a systolic array is to make use of the large amount of processi available in VLSl technology by using repetitive circuitry to perform signal proces matrix operations, image processing, etc. In summary, a systolic array is simply a c interconnected Processing Elements (PES). In order to incorporate as many proces

	the structure of the PES themselves is kept to a maximum simplicity and usually inc operation units. For design simplification, there are few types of PES in the same sy same token, interconnections are kept to a nearest neighbor topology in order to min communication delays as well as power distribution issues. Note that topologies incl neighbors (linear arrays), four neighbors (square arrays), or six neighbors (hexagona required by the problem to solve. This is notably due to the fact that scheduling mec based upon local criteria such as data availability. However, it should be noted that t clock in all the computation cells. Linear systolic arrays can tolerate clock skews at multidimensional designs require slower clocks in order to compensate. In order to s mechanisms, the design of a systolic array emphasizes an efficient mapping of the p architecture. An example of a band matrix-vector multiplication is shown in Fig. 6. the synchronization of the processors and of the input data rate has been mapped to requirements of the problem. Note that each processor is designed to operate upon th arguments. In summary, it should be noted that systolic arrays are very efficient at c intensive problems which involve many repetitive low-level calculations. Also, the v their design renders their function fixed at design time.") Gaudiot at Figure 6:

	 Fig. 6. A linear systolic array. To the extent Plaintiff asserts this limitation is not expressly or inherently disclosed apparent claim construction, or any other claim construction, the claimed subject m been obvious to a person of ordinary skill in the art considering this reference in co the knowledge of one of ordinary skill in the art at the time of the alleged invention disclosures in one or more of the references identified in Section I.B.2 of the cover
[1B] transforming an algorithm into a data driven calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor;	At least under Plaintiff's apparent theories of infringement and interpretations of th alleging that any of Defendant's accused products satisfy this claim limitation, Gau combination with one or more references, discloses: Gaudiot at Abstract ("New technologies of integration allow the design of powerfu may include several thousands of elementary processors. These multiprocessors ma range of applications in signal and data processing. However, assuring the proper i large number of processors and the ultimate safe execution of the user programs pr scheduling problem. The scheduling of operations upon the availability of their ope

	termed the data-driven mode of execution and offers an elegant solution to the issue is described in this paper and several architectures which have been proposed or imp (systolic arrays, data-flow machines, etc.) are examined in detail. The problems asso data-driven execution are also studied. A multi-level approach to high-speed digital processing is then evaluated.") Gaudiot at 1220-21 ("The overall objective of this paper is to demonstrate the applic driven principles of execution to the design of high-performance signal and data pro architectures. Several approaches will be demonstrated and their particular domain o will be contrasted. The description of low-level processing systems is beyond the sc and the interested readerr is referred to an excellent survey by Allen [3]. Instead, we here on the issues related to building high-performance multiprocessors for signal pr applications. In Section II, we show the type of problems considered in signal proce flow principles of execution as they relate to digital signal processing problems are detail in Section Ill while several existing data-driven architectures are described in Section V, we analyze a multi-level datadriven architecture and examine its program environment. Conclusions are drawn in Section VI.") Gaudiot at 1222 ("C. Structure Handling This is a crucial issue in signal processing for this kind of application requires that n elements which belong to the same structure be processed in a parallel or pipelined f the basic premises of data-flow principles states that an output is a function of its inp regardless of the state of the machine at the time of execution.") Gaudiot at 1224 ("IV. DATA-DRIVEN ARCHITECTURES We now describe in detail several systems which operate at runtime, compile-time, under data-driven execution. Although it is generally considered that data-flow princ execution are in effect at runtime, we extend their domain of application to design o and refer to them as data-driven systems. We thus initially examine multiprocessor s data dependencies have been frozen at design time (systolic arrays). We then consid programmable systolic arrays (the Wavefront Array Processor) and multiprocessors compile time by the use of data-flow program graphs (the ESL polycyclic processor

	Gaudiot at 1224-25 (“A. Systolic Arrays [32] The primary goal of a systolic array is to make use of the large amount of processing available in VLSl technology by using repetitive circuitry to perform signal processi matrix operations, image processing, etc. In summary, a systolic array is simply a co interconnected Processing Elements (PES). In order to incorporate as many processo the structure of the PES themselves is kept to a maximum simplicity and usually inc operation units. For design simplification, there are few types of PES in the same sy same token, interconnections are kept to a nearest neighbor topology in order to min communication delays as well as power distribution issues. Note that topologies incl neighbors (linear arrays), four neighbors (square arrays), or six neighbors (hexagona required by the problem to solve. This is notably due to the fact that scheduling mec based upon local criteria such as data availability. However, it should be noted that t clock in all the computation cells. Linear systolic arrays can tolerate clock skews at multidimensional designs require slower clocks in order to compensate. In order to s mechanisms, the design of a systolic array emphasizes an efficient mapping of the p architecture. An example of a band matrix-vector multiplication is shown in Fig. 6. the synchronization of the processors and of the input data rate has been mapped to requirements of the problem. Note that each processor is designed to operate upon th arguments. In summary, it should be noted that systolic arrays are very efficient at c intensive problems which involve many repetitive low-level calculations. Also, the v their design renders their function fixed at design time.”) Gaudiot at Figure 6:

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.