

Exhibit C-1

Invalidity of U.S. Patent No. 7,594,052 in view of Weber

U.S. Patent No. 7,165,094/U.S. Patent Application Publication No. 2002/0129173 (“Weber”), which was filed on March 9, 2001 and published no later than September 12, 2002, qualifies as prior art under at least 35 U.S.C. §§ 102 and 103 as to the asserted claims of U.S. Patent No. 7,594,052 (“’052 patent”), as discussed in greater detail in the chart that follows.¹

To the extent Plaintiff alleges that Weber does not disclose any particular limitation of claims 1–7 (the “Asserted Claims”) of the ’052 patent, either expressly or inherently, it would have been obvious to a person of ordinary skill in the art as of the priority date of the ’052 patent to modify Weber and/or to combine the teachings of Weber with other prior art references, including but not limited to the prior art references presently found in Defendants’ Invalidity Contentions and any supplements thereto and the relevant section of charts for other prior art for the ’052 patent in a manner that would have rendered the Asserted Claims invalid as obvious. The inclusion of the particular limitation into the methods and systems described in Weber would have been obvious for at least the following reasons: It would have been obvious to modify the methods and/or systems disclosed in Weber, alone or in combination with the references identified in Section V.C of Defendants’ Invalidity Contentions, to meet this limitation based on: (a) the teachings in Weber itself; (b) a person of ordinary skill in the art implementing a predictable variation of the methods and/or systems disclosed in Weber or in the other identified references; (c) the logic, judgment, intelligence, creativity, innovation and common sense a person of ordinary skill in the art would employ; (d) the inferences and creative and routine steps that a person of ordinary skill in the art would employ; (e) market demand for methods and/or systems to include this limitation; and/or (f) design needs or market pressures to solve a problem with a finite number of solutions that could have produced the desired outcome.

Defendants reserve the right to amend or supplement this claim chart at a later date pursuant to the Docket Control Order issued by the Court on June 16, 2023 (Dkt. No. 59).

¹ The citations are to the issued patent version of Weber, and they also disclose the corresponding citations to the patent application publication version of Weber, which are omitted for brevity. Nothing in these claim charts should be construed as an admission regarding infringement, either literally or under the doctrine of equivalents, or as an admission regarding Defendants’ understanding of the proper scope of the Asserted Claims. Given Plaintiff’s vague and ambiguous infringement contentions, the exemplary citations herein necessarily account for a variety of possible infringement arguments and claim constructions. Defendants reserve the right to rely on additional citations or sources of evidence that also may be applicable, or that may become applicable in light of claim construction, changes in Plaintiff’s infringement contentions, and/or information obtained during discovery as the case progresses. Various positions put forth in this document are predicated on Plaintiff’s overly broad interpretation of its claims as evidenced by the infringement allegations received to date. Those positions are not intended to and do not necessarily reflect Defendants’ interpretation of the true and proper scope of Plaintiff’s claims, and Defendants reserve the right to adopt claim construction positions that differ from or even conflict with various positions put forth in this document.

A. Claim 1

'052 Patent Claim Limitation	Weber
[1pre] Integrated circuit, comprising:	To the extent the preamble is limiting, Weber discloses and/or renders obvious the preamble based on at least the following citations. See, e.g.: • Weber at 1:12–32 Electronic computing and communications systems continue to include greater numbers of features and to increase in complexity. At the same time, electronic computing and communications systems decrease in physical size and cost per function. Rapid advances in semiconductor technology such as four-layer deep-sub-micron complimentary metal-oxide semiconductor (CMOS) technology, have enabled true “system-on-a-chip” designs. These complex designs may incorporate, for example, one or more processor cores, a digital signal processing (DSP) core, several communications interfaces, and graphics support in application-specific logic. In some systems, one or several of these extremely complex chips must communicate with each other and with other system components. Significant new challenges arise in the integration, verification and testing of such systems because efficient communication must take place between sub-systems on a single complex chip as well as between chips on a system board. One benefit to having an efficient and flexible method for communication between sub-systems and chips is that system components can be reused in other systems with a minimum of redesign. • Weber at 4:15–26 The present invention is a communications system and method for allowing multiple functional blocks or sub-systems of a complex electronics system to communicate with each other through a shared communications resource, such as a shared communications bus. In one embodiment, a communications protocol allows multiple functional block on a single semiconductor device to communicate to each other. In another embodiment, the communications protocol may be used to allow multiple functional blocks on different semiconductor devices to communicate to each other through a shared off-chip communications resource, such as a bus. • Weber at Fig. 1

'052 Patent Claim Limitation	Weber
	FIG. 1 • Weber at 4:43–5:24 FIG. 1 is a block diagram of a complex electronics system 100. Shared communications bus 112 connects sub-systems 102, 104, 106, 108, and 110. Sub-systems are typically functional blocks including a interface module for interfacing to a shared bus. Sub-systems may themselves include one or more functional blocks and may or may not include an integrated or physically separate interface module. In one embodiment, the sub-systems connected by communications bus 112 are separate integrated circuit chips. Sub-system 104 is an application specific integrated circuit (ASIC) which, as is known, is an integrated circuit designed to perform a particular function. Sub-system 106 is a dynamic random access memory (DRAM). Sub-system 108 is an erasable, programmable, read only memory (EPROM). Sub-system 110 is a field programmable gate array (FPGA). Sub-system 102 is a fully custom integrated circuit designed specifically to operate in system 100. Other

'052 Patent Claim Limitation	Weber
	embodiments may contain additional sub-systems of the same types as shown, or other types not shown. Other embodiments may also include fewer sub-systems than the sub-systems shown in system 100. Integrated circuit 102 includes sub-systems 102A, 102B, 102C, 102D and 102E. ASIC 104 includes functional blocks 101A, 104B and 104C. FPGA 110 includes functional blocks 110A and 110B. A functional block may be a particular block of logic that performs a particular function. A functional block may also be a memory component on an integrated circuit. • Weber at 5:3–8 System 100 is an example of a system that may consist of one or more integrated circuits or chips. A functional block may be a logic block on an integrated circuit such as, for example, functional block 102E, or a functional block may also be an integrated circuit such as fully custom integrated circuit 102 that implements a single logic function. See also, e.g., Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21. To the extent Weber does not expressly disclose the preamble, a person of ordinary skill in the art would have determined that the preamble is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant’s Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the ’052 Patent in a manner that would result in the subject matter of the preamble given, at the very least, the person’s understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[1a] a plurality of processing modules (M, S),	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: • Abstract A communication system. One embodiment includes at least two functional blocks, wherein an first functional block communicates with a second functional block by establishing a connection. A connection is a logical state in which data may pass between the first functional block and the second

'052 Patent Claim Limitation	Weber
	functional block. One embodiment includes a bus coupled to each of the functional blocks and configured to carry a plurality of signals. The plurality of signals includes a thread identifier that indicates a transaction stream that the data transfer is part of and a busy signal identified by the thread identifier. The busy signal is issued by the target functional block when resources will be unavailable to perform a transfer. • Weber at 1:12–32 Electronic computing and communications systems continue to include greater numbers of features and to increase in complexity. At the same time, electronic computing and communications systems decrease in physical size and cost per function. Rapid advances in semiconductor technology such as four-layer deep-sub-micron complimentary metal-oxide semiconductor (CMOS) technology, have enabled true “system-on-a-chip” designs. These complex designs may incorporate, for example, one or more processor cores, a digital signal processing (DSP) core, several communications interfaces, and graphics support in application-specific logic. In some systems, one or several of these extremely complex chips must communicate with each other and with other system components. Significant new challenges arise in the integration, verification and testing of such systems because efficient communication must take place between sub-systems on a single complex chip as well as between chips on a system board. One benefit to having an efficient and flexible method for communication between sub-systems and chips is that system components can be reused in other systems with a minimum of redesign. • Weber at 1:32–48 One challenge in the integration, verification and testing of modern electronic systems stems from the fact that modern electronic systems in many application areas have functionality, cost and form-factor requirements that mandate the sharing of resources, such as memory, among multiple functional blocks, where functional blocks can be any entity that interfaces to a communication system. In such systems, the functional blocks typically possess different performance characteristics and requirements, and the communications system and shared resources must simultaneously satisfy the total requirements. Key requirements of typical functional blocks are bandwidth and latency constraints that can vary over several orders of magnitude between functional blocks. In order to

'052 Patent Claim Limitation	Weber
	simultaneously satisfy constraints that vary so widely, communications systems must provide high degrees of predictability. • Weber at 2:63–3:9 One embodiment of the invention includes several functional blocks, including at least one initiator functional block and one target functional block. Some initiator functional blocks may also function as target functional blocks. In one embodiment, the initiator functional block is coupled to an initiator interface module and the target functional block is coupled to a target interface module. The initiator functional block and the target functional block communicate to their respective interface modules and the interface modules communicate with each other. The initiator functional block communicates with the target functional block by establishing a connection, wherein a connection is a logical state in which data may pass between the initiator functional block and the target functional block. • Weber at 4:15–26 The present invention is a communications system and method for allowing multiple functional blocks or sub-systems of a complex electronics system to communicate with each other through a shared communications resource, such as a shared communications bus. In one embodiment, a communications protocol allows multiple functional block on a single semiconductor device to communicate to each other. In another embodiment, the communications protocol may be used to allow multiple functional blocks on different semiconductor devices to communicate to each other through a shared off-chip communications resource, such as a bus. • Weber at Fig. 1

**'052 Patent Claim
Limitation**

Weber

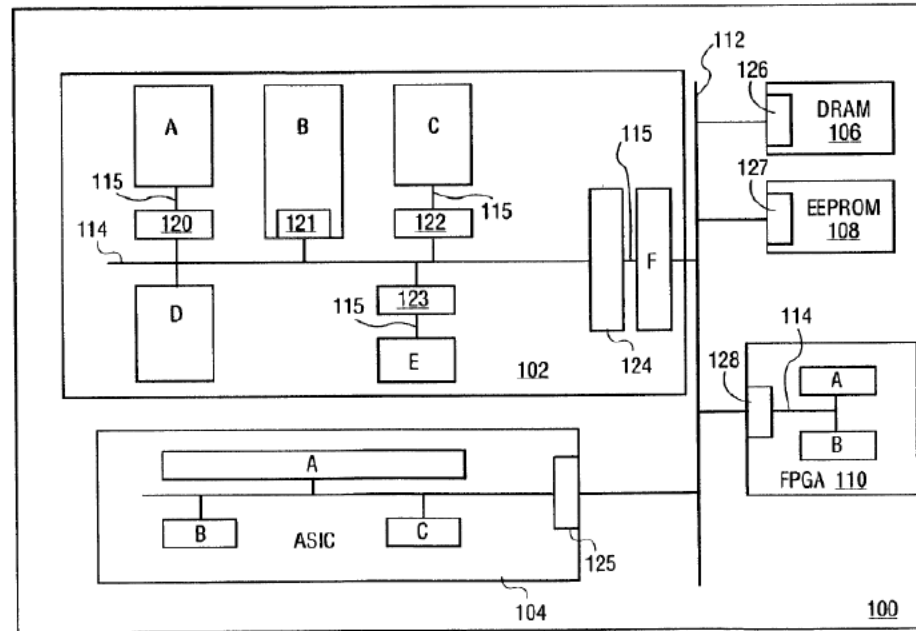


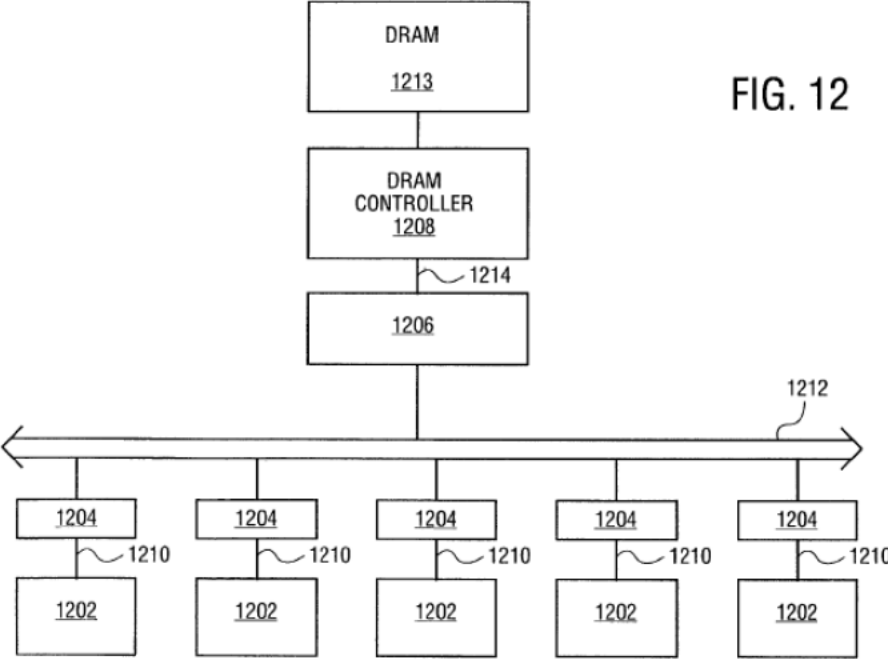
FIG. 1

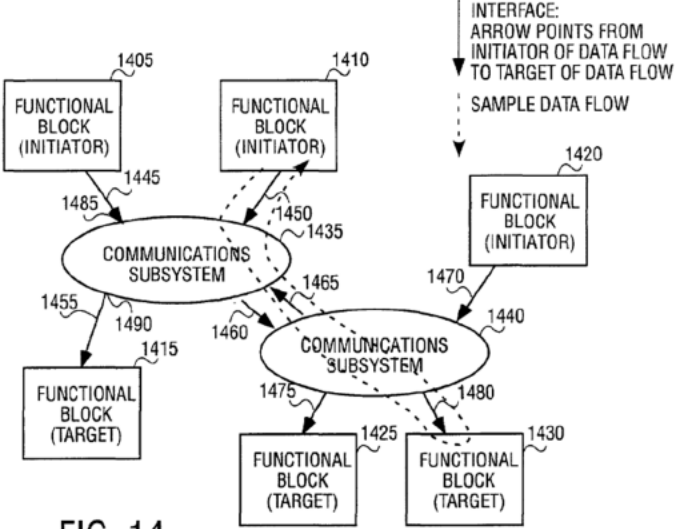
- Weber at 4:43–5:24

FIG. 1 is a block diagram of a complex electronics system 100. Shared communications bus 112 connects sub-systems 102, 104, 106, 108, and 110. Sub-systems are typically functional blocks including a interface module for interfacing to a shared bus. Sub-systems may themselves include one or more functional blocks and may or may not include an integrated or physically separate interface module. In one embodiment, the sub-systems connected by communications bus 112 are separate integrated circuit chips. Sub-system 104 is an application specific integrated circuit (ASIC) which, as is known, is an integrated circuit designed to perform a particular function. Sub-system 106 is a dynamic random access memory (DRAM). Sub-system 108 is an erasable, programmable, read only memory (EPROM). Sub-system 110 is a field programmable gate array (FPGA). Sub-system 102 is a fully custom integrated circuit designed specifically to operate in system 100. Other

'052 Patent Claim Limitation	Weber
	embodiments may contain additional sub-systems of the same types as shown, or other types not shown. Other embodiments may also include fewer sub-systems than the sub-systems shown in system 100. Integrated circuit 102 includes sub-systems 102A, 102B, 102C, 102D and 102E. ASIC 104 includes functional blocks 101A, 104B and 104C. FPGA 110 includes functional blocks 110A and 110B. A functional block may be a particular block of logic that performs a particular function. A functional block may also be a memory component on an integrated circuit. • Weber at 5:3–8 System 100 is an example of a system that may consist of one or more integrated circuits or chips. A functional block may be a logic block on an integrated circuit such as, for example, functional block 102E, or a functional block may also be an integrated circuit such as fully custom integrated circuit 102 that implements a single logic function. • Weber at 5:29–41 In one embodiment there are four fundamental types of functional blocks. The four fundamental types are initiator, target, bridge, and snooping blocks. A typical target is a memory device, a typical initiator is a central processing unit (CPU). Functional blocks all communicate with one another via shared communications bus 112 or shared communications bus 114 and the protocol of one embodiment. Initiator and target functional blocks may communicate a shared communications bus through interface modules. An initiator functional block may communicate with a shared communications bus through an initiator interface module and a target functional block may communicate with a shared communications bus through a target interface module. • Weber at Fig. 4

'052 Patent Claim Limitation	Weber
	FIG. 4 Weber at 6:53–7:8 FIG. 4 is a block diagram of a part of a computer system 1000 according to one embodiment. FIG. 4 is useful in illustrating multilevel connection identification. System 1000 includes initiator functional block 1002, which is connected to initiator interface module 1004 by interconnect 1010. Initiator interface module 1004 is connected to target interface module 1006 by shared communications bus 1012. Target interface module 1006 is connected to target functional block 1008 by an interconnect 1010. Typically, shared communications bus 1012 is analogous to shared communications bus 112 of FIG. 1 or to shared communications bus 114 of FIG. 1. Interconnects 1010 are typically analogous to interconnect 115 of FIG. 1 in that they connect functional blocks to interface modules and are point-to-point, rather than shared, interconnects. Interconnects 1010 are typically physically shorter than shared communications bus 1012 because of their local nature. As will be explained more fully below, system 1000 uses two different levels of connection identification depending upon the

'052 Patent Claim Limitation	Weber
	requirements of a particular functional block. “Global” connection identification information is sent on shared communications bus 1012, while “local” connection information, or thread identification information, is sent in interconnects 1010. Weber at Fig. 12  FIG. 12 Weber at 20:59–21:3 Referring to FIG. 12, an example of a use of Thread ID, consider a series of identical direct memory access (DMA) engines in a system. In FIG. 12, elements 1202 are identical DMA engines, each connected to an initiator interface module 1204. Initiator interface modules 1204 are connected to shared communications bus 1212. Target interface module 1206 is also connected to shared communications bus 1212 and transmits data from bus 1212 to DRAM controller 1208, which is a

'052 Patent Claim Limitation	Weber
	target functional block. Target interface module 1206 is connected to DRAM controller 1208 by interconnect 1214. DRAM controller 1208 controls access to DRAM 1213. • Weber at Fig. 14  FIG. 14 • Weber at 15:38–49 An exemplary system is shown in FIG. 14. The system consists of one or more functional blocks 1405, 1410, 1415, 1420, 1425, 1430 and one or more communication subsystems 1435, 1440 that are all tied together using a set of interfaces 1445, 1450, 1455, 1460, 1465, 1470, 1475, 1480. The functional blocks 145, 1410, 1415, 1420, 1425, 1430 perform the actual work done by the system (such as computation). In this document, the focus is on communication between functional blocks. Functional blocks are broken down into initiators 145, 1410, 1420 and targets 1415, 1425, 1430, depending on whether a functional block initiates or is the target of a data flow.

'052 Patent Claim Limitation	Weber
	See also, e.g., Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant’s Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the ’052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person’s understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[1b] wherein at least one first of said processing modules (M) requests at least one communication service to at least one second processing module (S) based on specific communication properties and at least one communication service identification,	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: • Weber at Abstract A communication system. One embodiment includes at least two functional blocks, wherein an first functional block communicates with a second functional block by establishing a connection. A connection is a logical state in which data may pass between the first functional block and the second functional block. One embodiment includes a bus coupled to each of the functional blocks and configured to carry a plurality of signals. The plurality of signals includes a thread identifier that indicates a transaction stream that the data transfer is part of and a busy signal identified by the thread identifier. The busy signal is issued by the target functional block when resources will be unavailable to perform a transfer. • Weber at 1:32–48 One challenge in the integration, verification and testing of modern electronic systems stems from the fact that modern electronic systems in many application areas have functionality, cost and form-factor requirements that mandate the sharing of resources, such as memory, among multiple functional blocks, where functional blocks can be any entity that interfaces to a communication system. In such systems, the functional blocks typically possess different performance characteristics and requirements, and the communications system and shared resources must simultaneously satisfy

'052 Patent Claim Limitation	Weber
	the total requirements. Key requirements of typical functional blocks are bandwidth and latency constraints that can vary over several orders of magnitude between functional blocks. In order to simultaneously satisfy constraints that vary so widely, communications systems must provide high degrees of predictability. • Weber at 2:63–3:9 One embodiment of the invention includes several functional blocks, including at least one initiator functional block and one target functional block. Some initiator functional blocks may also function as target functional blocks. In one embodiment, the initiator functional block is coupled to an initiator interface module and the target functional block is coupled to a target interface module. The initiator functional block and the target functional block communicate to their respective interface modules and the interface modules communicate with each other. The initiator functional block communicates with the target functional block by establishing a connection, wherein a connection is a logical state in which data may pass between the initiator functional block and the target functional block. • Weber at 8:29–56 Connection identifier (CONNID) lines 226 carry a multi-bit signal driven by an initiator bus owner to indicate which connection the current transfer is part of. A connection is a logical state, established by an initiator, in which data may pass between the initiator and an associated target. The CONNID typically transmits information including the identity of the functional block initiating the transfer and ordering information regarding an order in which the transfer must be processed. In one embodiment, the information conveyed by the CONNID includes information regarding the priority of the transfer with respect to other transfers. In one embodiment the CONNID is a eight-bit code. An initiator interface module sends a unique CONNID along with an initial address transfer of a connection. Later transfers associated with this connection (for example, data transfers) also provide the CONNID value so both sender and receiver (as well as any device monitoring transfers on shared communications bus 1012) can unambiguously identify transfers associated with the connection. One advantage of using a CONNID is that transfers belonging to different transactions can be interleaved arbitrarily between multiple devices on a per cycle basis. In one embodiment, shared communications bus 1012 implements a fully pipelined protocol that requires strict control over transaction ordering in order to guarantee proper system operation. Without the use of a CONNID,

'052 Patent Claim Limitation	Weber
	ordering constraints within a particular transaction may be violated because transfers associated with a particular connection are not identified. Weber at 9:1–17 The present invention uses a CONNID signal, in part, to allow overlapping of dependent commands. Therefore, use of a CONNID improves system performance and efficiency. Another benefit of the CONNID of the present invention is that communication system predictability is enhanced because it allows a shared functional block to respond to requests based upon quality of service guarantees that may vary between connections. For example, data requested to operate a computer display cannot tolerate unpredictable delay because delay causes the display to flicker. Therefore, the CONNID may be used to prioritize data requests from a display controller so that requests from the display controller to a common resource are serviced before other requests. The present invention also allows for flexible reconfiguration of the CONNID to retune system performance. Weber at 12:1–20 Second, in one embodiment the connection identifier allows the target to choose which requests it rejects. The target may associate meanings such as transfer priority to the CONNID values, and therefore decide which requests to act upon based upon a combination of the CONNID value and the internal state of the target. For instance, a target might have separate queues for storing transfer requests of different priorities. Referring to FIG. 9, the target might have a queue for low priority requests (which present with an odd CONNID) and a queue for high priority requests (which present with an even CONNID). Thus, the CONNID 1 WRITE ADDRE0 request of cycle 1 would be rejected if the low-priority queue were full, whereas the CONNID 2 READ ADDR0 transfer could be completed successfully based upon available high-priority queue resources. Such differences in transfer priorities are very common in highly-integrated electronic systems, and the ability for the target to deliver higher quality of service to higher priority transfer requests adds significantly to the overall predictability of the system. Weber at 12:53–67 Finally, the CONNID values that are associated with particular initiator transactions should typically be chosen to provide useful information such as transfer priorities but also to minimize

'052 Patent Claim Limitation	Weber
	implementation cost. It is useful to choose the specific CONNID values at system design time, so the values can be guaranteed to be unique and can be ordered to simplify comparison and other operations. Furthermore, it is frequently useful to be able to change the CONNID values during operation of the communications system so as to alter the performance and predictability aspects of the system. Preferred implementations of the present invention enable flexible system configuration by storing the CONNID values in ROM or RAM resources of the functional blocks, so they may be readily reconfigured at either system build time or system run time. • Weber at Fig. 11

'052 Patent Claim Limitation	Weber
	FIG. 11 • Weber at 13:14–38 Signals shown in FIG. 11 are labeled with signal names. In addition, some signal names are followed by a notation or notations in parentheses or brackets. The notations are as follows: (I) The signal is optional and is independently configurable

'052 Patent Claim Limitation	Weber
	(A) The signal must be configured together with signals having similar notations (AI) The signal is independently configurable if (A) interface modules exist [#] Maximum signal width The clock signal is the clock of a connected functional block. The command (Cmd) signal indicates the type of transfer on the bus. Commands can be issued independent of data. The address (Addr) signal is typically an indication of a particular resource that an initiator functional block wishes to access. Request Accept (ReqAccept) is a handshake signal whereby slave 1104 allows master 1102 to release Cmd, Addr and DataOut from one transfer and reuse them for another transfer. If slave 1104 is busy and cannot participate in a requested transfer, master 1102 must continue to present Cmd, Addr and DataOut. DataOut is data sent from a master to a slave, typically in a write transfer. DataIn typically carries read data. • Weber at 14:1–13 Request Thread Identifier (ReqThreadID), in one embodiment, is a four-bit signal that provides the thread number associated with a current transaction intended for slave 1104. All commands executed with a particular thread ID must execute in order with respect to one another, but they may execute out of order with respect to commands from other threads. Response Thread Identifier (RespThreadID) provides a thread number associated with a current response. Because responses in a thread may return out of order with respect to other threads, RespThreadID is necessary to identify which thread's command is being responded to. In one embodiment, ReqThreadID and RespThreadID are optional signals, but if one is used, both should be used. • Weber at 14:14–22 The Request Connection Identifier (ReqConnID) provides the CONNID associated with the current transaction intended for slave or target. CONNIDs provide a mechanism by which a system entity may associate particular transactions with the system entity. One use of the CONNID is in establishing request priority among various initiators. Another use is in associating actions or data transfers with initiator identity rather than the address presented with the transaction request. • Weber at 15:30–37

'052 Patent Claim Limitation	Weber
	Provision of per-thread flow control permits end-to-end performance guarantees to be determined across the interconnect. End-to-end data channels are linked using the threads of the interconnect, and per-thread flow control allows for end-to-end flow control between a particular initiator and target. One embodiment of end-to-end performance guarantees is described with reference to FIGS. 14, 15 a, 15 b, 16, 17, 18, 19, 20 and 21. - Weber at Fig. 14 FIG. 14 - Weber at 15:38–49 An exemplary system is shown in FIG. 14. The system consists of one or more functional blocks 1405, 1410, 1415, 1420, 1425, 1430 and one or more communication subsystems 1435, 1440 that are all tied together using a set of interfaces 1445, 1450, 1455, 1460, 1465, 1470, 1475, 1480. The functional blocks 145, 1410, 1415, 1420, 1425, 1430 perform the actual work done by the system (such as computation). In this document, the focus is on communication between functional blocks.

'052 Patent Claim Limitation	Weber
	Functional blocks are broken down into initiators 145, 1410, 1420 and targets 1415, 1425, 1430, depending on whether a functional block initiates or is the target of a data flow. Weber at 15:50–64 For purpose of discussion herein, a data flow is a set of commands, data items or events being communicated over time between two functional blocks. Any given request from an initiator to a target is a request to perform some action. For example, read and write requests cause a target to retrieve and store the data respectively. A channel is a portion of a set of physical resources in a component that is dedicated to service a particular data flow. For example, a target may have different data channels to service different incoming data flows. A communication subsystem transmits data items from one of its input ports to an output port, using a dedicated data channel for each data flow. Each of the ports leads from/to another communications subsystem or a functional block via an interface. Interfaces also have dedicated channels for each data flow. Weber at 15:65–16:9 Thus, for example, the end-to-end progress of a data flow may be as follows. The data flow originates at an initiator. It travels to an input port of a first communication subsystem via a channel of an interface. Once inside the first communication subsystem, it travels to an output port, via a channel. It then may cross another interface to other communication subsystems until it finally reaches a target. It is serviced by a dedicated channel inside the target and completes if no response is needed. If a response is needed, the data flow then reverses its direction through the communications subsystem(s), and completes when it reaches the originating initiator. Weber at 16:10–21 In one embodiment, the interfaces tying the different components together have the ability to tag each request and response transmitted with the identity of the channel it belongs to, and the ability to transmit flow control on a per-channel basis. These features of the interfaces are desirable to achieve independent data channels that allow the quality of service guarantees given by an individual component to be propagated to another component that it is connected to via an interface. Thus, for example with reference to FIG. 4, the quality of service guarantees include the signal propagation time from initiator functional block 1002 to target functional block 1008.

'052 Patent Claim Limitation	Weber
	• Weber 16:22–45 FIG. 15 a shows an example of a functional block's quality of service guarantees. In this case, a memory system is shown that has differential quality of service guarantees for three channels. The quality of service guarantees given are minimum bandwidth and maximum latency while servicing an 8-byte read or write request. In this example, each channel can sustain 10 M 8-byte requests per second. Channel 1 has been optimized for lower latency and guarantees a maximum service latency of 100 ns, while channels 2 and 3 only guarantee a maximum service latency of 200 ns. It is common for the performance guarantees given by a particular functional block to be much more complex than shown here and to depend on a lot of different factors such as request size, inter-arrival interval, etc. The metrics used here have been chosen to keep the example simple. While maximum latency and minimum bandwidth are two common types of performance parameters, quality of service guarantees using other parameters such as, maximum outstanding requests, maximum variance in service latency, etc. are equally possible. In addition, if a particular component is too complex to easily achieve differential quality of service guarantees, the methodology can be applied recursively to that component by breaking it down into smaller, more manageable pieces. • Weber 16:46–61 FIG. 15 b shows an example of a communication subsystem's quality of service guarantees. This particular communication system has two ports for connecting initiators X 210 and Y 215 and one port for connecting a target Z 220. Port X 210 in turn supports two different incoming channels, port Y one incoming channel and port Z, three outgoing channels. This yields three data channels (A, B, and C) within the communications subsystem. In the present example, no quality of service guarantees have been given for channel A, again to simplify the discussion herein and also to note that sometimes certain data flows do not have guarantees. Channel B is serviced with a minimum bandwidth of 4 bytes every 4 cycles and a maximum latency of 3 cycles for each direction. Channel C is serviced with a minimum bandwidth of 4 bytes every 2 cycles and a maximum latency of 5 cycles for each direction. • Weber at Figs. 17–18

'052 Patent Claim Limitation	Weber
	<pre> graph TD START1([START]) --> 1710[FOR INITIATOR DATA FLOWS IN THE SYSTEM] 1710 --> 1715[MAP DATA FLOW THROUGH THE SYSTEM] 1715 --> 1720[ALIGN/CONVERT PERFORMANCE GUARANTEES] 1720 --> 1730[AGGREGATE PERFORMANCE GUARANTEES] 1730 --> D1{MORE DATA FLOWS?} D1 -- YES --> 1710 D1 -- NO --> END1([END]) </pre> FIG. 17 <pre> graph TD START2([START]) --> 1810[FOR COMPONENTS IN PATH FROM INITIATOR TO TARGET] 1810 --> 1815[PICK AN AVAILABLE CHANNEL IN THIS COMPONENT] 1815 --> 1820[MARK CHANNEL AS USED] 1820 --> D2{MORE COMPONENTS?} D2 -- YES --> 1810 D2 -- NO --> END2([END]) </pre> FIG. 18 • Weber at 17:26–32 One embodiment of a process for calculating end-to-end system performance guarantees for different data flows is shown in FIG. 17. For each initiator data flow in the system 1710, a mapping of the data flow through the channels of the system is first determined 1715. An exemplary mapping process is further illustrated in FIG. 18 and will be discussed below. • Weber at 17:59–18:6 FIG. 18 focuses in on one embodiment of the data flow mapping process that is part of the global mechanism shown in FIG. 17. For each component in a path from initiator to target, e.g., through

'052 Patent Claim Limitation	Weber
	several interfaces and one or more communication subsystems step 1710, an available channel is chosen to carry the initiator channel step 1715. Picking a particular data flow out of the set of available channels for a given component can be a complex task. Many possible ways of accomplishing the task and optimizing the overall system performance exist. For example, one could choose the channel that most closely matches the performance requirements of the initiator's data flow. It is also possible to split a data flow among multiple paths from initiator to target. In the present example, it is simply assumed that a valid mapping is chosen. • Weber at Fig. 20 The diagram illustrates a network architecture. At the top, there are three initiators: INITIATOR A (2005), INITIATOR B (2020), and INITIATOR C (2045). INITIATOR A has two output channels labeled 1 and 2, with a reference numeral 710. INITIATOR B has one output channel labeled 1, with a reference numeral 2025. INITIATOR C has three output channels labeled 1, 3, and 2, with a reference numeral 2065. These initiators are connected to a central COMMUNICATIONS SUBSYSTEM (2030) via three interfaces: INTERFACE 1 (2060), INTERFACE 2 (2040), and INTERFACE 3 (2070). INTERFACE 1 connects to PORT X (2055), INTERFACE 2 connects to PORT Y (2040), and INTERFACE 3 connects to PORT Z (2037). The ports are connected to the communications subsystem via channels labeled A, B, and C. The communications subsystem (2030) is represented by a central block with internal components 2035 and 2037. The target (2045) is connected to the communications subsystem via a reference numeral 2075. FIG. 20 • Weber at 19:27–34

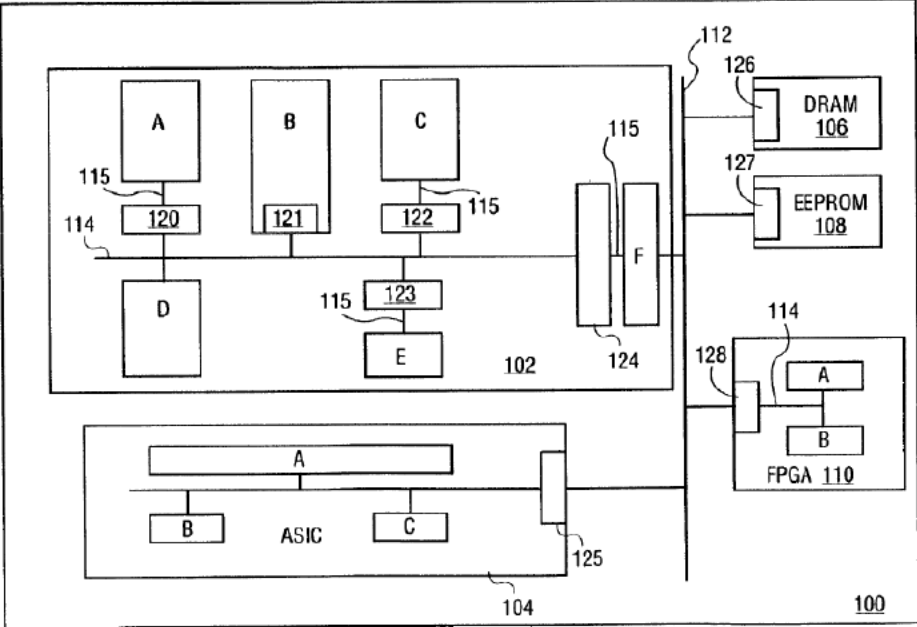
'052 Patent Claim Limitation	Weber
	An example application of the quality of service aggregation mechanism using a simple system is shown in FIG. 20. An initiator A 2005 sources two data flows shown as 1 2010 and 2 2015. A second initiator B sources only one data flow 2025. The communications subsystem 2030 receives the initiator data flows on ports X 2035 and Y 2040, respectively, and sends all of these flows to a target C 2045 via port Z 2050. • Weber at 19:35–56 The first table in FIG. 21 shows one embodiment of the mapping of each initiator data flow through the system. So, for example, initiator A data flow 1 2010 is sent to communications subsystem port X 2035 via channel A 2060 of interface 1. It traverses the communication subsystem 2030 via channel B 2037, exits via port Z 2065, and is sent to target C 2045 via interface 3 channel A 2070. At target C 2045 channel 1 2075 is chosen to handle this data flow. The second table of FIG. 21 shows an example of quality of service guarantees after unit alignment for the three different initiator data flows. In this example, it is assumed that initiator A flow 1 is using 4-byte requests, initiator A flow 2 is using 16-byte requests, and initiator B flow 1 is using 8-byte requests. The unit conversion and alignment is accomplished using the process described above. The last table of FIG. 21 shows the exemplary system end-to-end quality of service guarantees for the three initiator data flows. In order to calculate the minimum bandwidth in the system for each data flow, the minimum bandwidth of each mapped data flow through each component is found. Maximum latencies are found by adding each of the component latencies along a data flow from initiator to target. • Weber at 19:57–20:6 The Request Connection Identifier (ReqConnID) provides the CONNID associated with the current transaction intended for slave or target. CONNIDs provide a mechanism by which a system entity may associate particular transactions with the system entity. One use of the CONNID is in establishing request priority among various initiators. Another use is in associating actions or data transfers with initiator identity rather than the address presented with the transaction request.

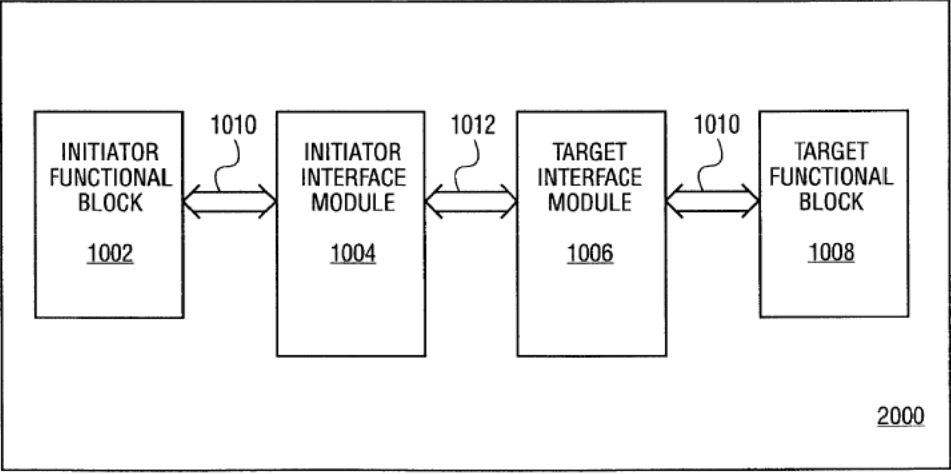
'052 Patent Claim Limitation	Weber
	The embodiment of FIG. 11 provides end-to-end connection identification with CONNID as well as point-to-point, or more local identification with Thread ID. A Thread ID is an identifier of local scope that simply identifies transfers between the interface module and its connected functional block. In contrast, the CONNID is an identifier of global scope that identifies transfers between two interface modules (and, if required, their connected functional blocks). • Weber at 20:25–58 Referring to FIG. 4, a CONNID should be unique when transferred between interface modules 1004 and 1006 on shared communications bus 1012. The CONNID may be sent over a local interconnect, such as interconnect 1010. In many cases, however, it is much more efficient to use only Thread ID between a functional block and its interface module. For example initiator functional block 1002 may not require all the information provided by the CONNID. Also, in some systems, multiple identical initiator functional blocks 1002 may exist with the same CONNID so that a particular target functional block 1008 receiving a transfer will not know which connection it is actually part of unless logic in initiator interface module 1004 translates the “local” CONNID to a unique “global” CONNID. The design and implementation of such a translation functionality in an interface module is complicated and expensive. In such cases, the CONNID may be sent between interface modules over shared communications bus 1012 while the Thread ID is sent between a functional block and an interface module. In the case of an initiator functional block, a one-to-one static correspondence may exist between Thread ID and CONNID. For example if the Thread ID is “1”, a single CONNID is mapped for a particular interface module, solving the problem of multiple, identical functional blocks. In the case of a target functional block, there is a one-to-one dynamic correspondence between a Thread ID and a CONNID. If a target functional block supports two simultaneous threads, the target interface module acquires the CONNID of an open connection and associates it with a thread as needed. For example, a target interface module receives a CONNID of “7”, and then maps CONNID 7 to thread “0”. Thereafter, all transfers with CONNID 7 are associated with thread 0 until connection 7 is closed. • Weber at 21:4–34

'052 Patent Claim Limitation	Weber
	A DMA engine is an example of an initiator functional block that also functions as a target functional block. When the DMA engine is programmed by software, it acts as a target. Thereafter, the DMA engine is an initiator. Because a DMA engine performs both read and write operations, two connections can be associated with a single DMA engine. If some buffering is available in the DMA engine, read and write operations may be decoupled so that both types of operations can be performed concurrently. A read may occur from a long latency storage device which requires the read data to be buffered on the DMA engine before a write operation writes the data. In one embodiment, each of DMA engines 1202 uses a Thread ID to identify the read stream and a different Thread ID to identify the write stream. The DMA engine does not require more information, such as what other functional block participates in a transaction. Therefore, a CONNID is not required to be sent from the DMA engine 1202 to a connected interface module 1204. Mapping of a Thread ID to a CONNID occurs in the interface module 1204. In one embodiment, each initiator interface module 1204 maps a unique CONNID to each of two Thread IDs from a connected DMA engine 1202. Each of DMA engines 1202 use a single bit, for example, Thread ID of FIG. 11, to distinguish between its two threads. For each transfer over shared communications bus a unique CONNID is sent to target interface module 1206. The CONNID may include priority information, for example, assigning high priority to requests for graphics data. The high priority graphics data request is immediately serviced by DRAM controller 1208 while lower priority request may be required to wait. <i>See also, e.g.,</i> Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

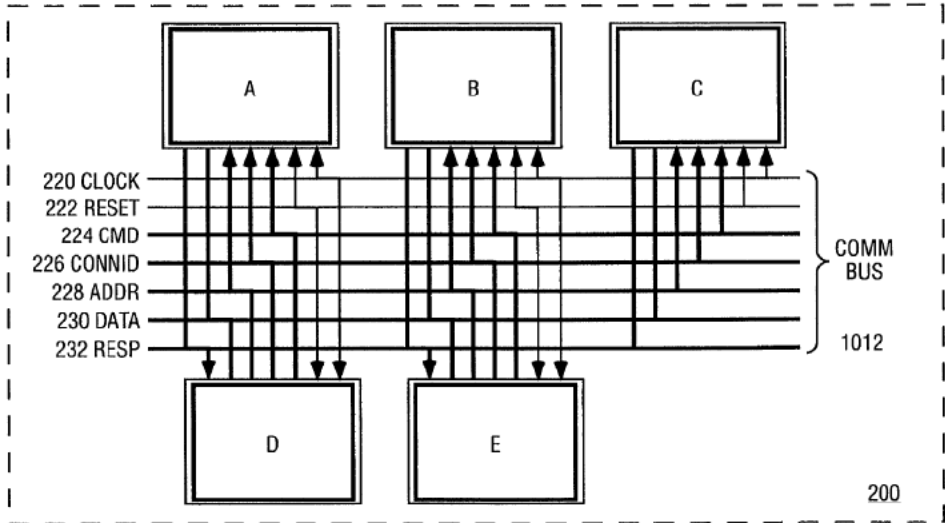
'052 Patent Claim Limitation	Weber
[1c] an interconnect means (N) for coupling said plurality of processing modules (M, S) and for enabling a connection based communication having a set of connection properties,	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: • Weber at Abstract A communication system. One embodiment includes at least two functional blocks, wherein an first functional block communicates with a second functional block by establishing a connection. A connection is a logical state in which data may pass between the first functional block and the second functional block. One embodiment includes a bus coupled to each of the functional blocks and configured to carry a plurality of signals. The plurality of signals includes a thread identifier that indicates a transaction stream that the data transfer is part of and a busy signal identified by the thread identifier. The busy signal is issued by the target functional block when resources will be unavailable to perform a transfer. • Weber at 1:32–48 One challenge in the integration, verification and testing of modern electronic systems stems from the fact that modern electronic systems in many application areas have functionality, cost and form-factor requirements that mandate the sharing of resources, such as memory, among multiple functional blocks, where functional blocks can be any entity that interfaces to a communication system. In such systems, the functional blocks typically possess different performance characteristics and requirements, and the communications system and shared resources must simultaneously satisfy the total requirements. Key requirements of typical functional blocks are bandwidth and latency constraints that can vary over several orders of magnitude between functional blocks. In order to simultaneously satisfy constraints that vary so widely, communications systems must provide high degrees of predictability. • Weber at 2:63–3:9 One embodiment of the invention includes several functional blocks, including at least one initiator functional block and one target functional block. Some initiator functional blocks may also function as target functional blocks. In one embodiment, the initiator functional block is coupled to an initiator interface module and the target functional block is coupled to a target interface module. The initiator functional block and the target functional block communicate to their respective interface modules

'052 Patent Claim Limitation	Weber
	and the interface modules communicate with each other. The initiator functional block communicates with the target functional block by establishing a connection, wherein a connection is a logical state in which data may pass between the initiator functional block and the target functional block. • Weber at 3:10–31 One embodiment also includes a bus configured to carry multiple signals, wherein the signals include a connection identifier signal that indicates a particular connection that a data transfer between an initiator functional block and a target functional block is part of. The connection identifier includes information about the connection, such as which functional block is the source of a transmission, a priority of a transfer request, and transfer ordering information. One embodiment also includes a thread identifier, which provides a subset of the information provided by the connection identifier. In one embodiment, the thread identifier is an identifier of local scope that identifies transfers between an interface module and a connected functional block, where in some embodiments, an interface module connects a functional block to a shared communications bus. The connection identifier is a an identifier of global scope that transfers information between interface modules or between functional blocks through their interface modules. Some functional blocks may require all the information provided by the connection identifier, while other functional blocks may require only the subset of information provided by the thread identifier. • Weber at 4:15–26 The present invention is a communications system and method for allowing multiple functional blocks or sub-systems of a complex electronics system to communicate with each other through a shared communications resource, such as a shared communications bus. In one embodiment, a communications protocol allows multiple functional block on a single semiconductor device to communicate to each other. In another embodiment, the communications protocol may be used to allow multiple functional blocks on different semiconductor devices to communicate to each other through a shared off-chip communications resource, such as a bus. • Weber at Fig. 1

'052 Patent Claim Limitation	Weber
	 FIG. 1 • Weber at 5:9–18 Shared communications bus 112 provides a shared communications bus between sub-systems of system 100. Shared communication bus 114 provides a shared communications bus between sub-systems or functional blocks on a single integrated circuit. Some of the functional blocks shown are connected to interface modules through which they send and receive signals to and from shared communications bus 112 or shared communications bus 114. Interconnect 115 is a local point-to-point interconnect for connecting interface modules to functional blocks. • Weber at 5:29–41

'052 Patent Claim Limitation	Weber
	In one embodiment there are four fundamental types of functional blocks. The four fundamental types are initiator, target, bridge, and snooping blocks. A typical target is a memory device, a typical initiator is a central processing unit (CPU). Functional blocks all communicate with one another via shared communications bus 112 or shared communications bus 114 and the protocol of one embodiment. Initiator and target functional blocks may communicate a shared communications bus through interface modules. An initiator functional block may communicate with a shared communications bus through an initiator interface module and a target functional block may communicate with a shared communications bus through a target interface module. • Weber at Fig. 4  <pre> graph LR 1002[INITIATOR FUNCTIONAL BLOCK 1002] <--> 1010 1004[INITIATOR INTERFACE MODULE 1004] 1004 <--> 1012 1006[TARGET INTERFACE MODULE 1006] 1006 <--> 1010 1008[TARGET FUNCTIONAL BLOCK 1008] subgraph 2000 1002 1004 1006 1008 end </pre> FIG. 4 • Weber at 6:53–7:8

'052 Patent Claim Limitation	Weber
	FIG. 4 is a block diagram of a part of a computer system 1000 according to one embodiment. FIG. 4 is useful in illustrating multilevel connection identification. System 1000 includes initiator functional block 1002, which is connected to initiator interface module 1004 by interconnect 1010. Initiator interface module 1004 is connected to target interface module 1006 by shared communications bus 1012. Target interface module 1006 is connected to target functional block 1008 by an interconnect 1010. Typically, shared communications bus 1012 is analogous to shared communications bus 112 of FIG. 1 or to shared communications bus 114 of FIG. 1. Interconnects 1010 are typically analogous to interconnect 115 of FIG. 1 in that they connect functional blocks to interface modules and are point-to-point, rather than shared, interconnects. Interconnects 1010 are typically physically shorter than shared communications bus 1012 because of their local nature. As will be explained more fully below, system 1000 uses two different levels of connection identification depending upon the requirements of a particular functional block. “Global” connection identification information is sent on shared communications bus 1012, while “local” connection information, or thread identification information, is sent in interconnects 1010. • Weber at Fig. 5

'052 Patent Claim Limitation	Weber
	 FIG. 5 • Weber at 7:9–42 FIG. 5 is a block diagram of one embodiment of a shared communications bus 1012. Shared communications bus 1012 is shown connected to entities A, B, C, D and E, which may be interface modules or functional blocks. Shared communications bus 1012 is composed of a set of wires. Data wires 230 provide direct, high efficiency transport of data traffic between functional blocks on shared communications bus 1012. In one embodiment, shared communications bus 1012 supports a bus protocol that is a framed, time division multiplexed, fully pipelined, fixed latency communication protocol using separate address, data and connection identification wires. The bus protocol supports fine grained interleaving of transfers to enable high operation concurrency, and uses retry transactions to efficiently implement read transactions from target devices with long or variable latency. Details of the arbitration method used to access shared communications bus 1012 are not required to understand the present invention. The delay from when an initiator functional block

'052 Patent Claim Limitation	Weber
	drives the command and address until the target functional block drives the response is known as the latency of shared communications bus 1012. The bus protocol supports arbitration among many initiator functional blocks and target functional blocks for access to the bus. In the embodiment shown, arbitration for access to shared communications bus 1012 is performed by an initiator interface module, such as module 1004 of FIG. 4. In other embodiments, arbitration is performed by functional blocks directly, or by a combination of interface modules and functional blocks. In one embodiment, a bus grant lasts for one pipelined bus cycle. The protocol does not forbid a single functional block from becoming a bus owner for consecutive bus cycles, but does require that the functional block successfully win arbitration on consecutive cycles to earn the right. • Weber at 8:29–56 Connection identifier (CONNID) lines 226 carry a multi-bit signal driven by an initiator bus owner to indicate which connection the current transfer is part of. A connection is a logical state, established by an initiator, in which data may pass between the initiator and an associated target. The CONNID typically transmits information including the identity of the functional block initiating the transfer and ordering information regarding an order in which the transfer must be processed. In one embodiment, the information conveyed by the CONNID includes information regarding the priority of the transfer with respect to other transfers. In one embodiment the CONNID is a eight-bit code. An initiator interface module sends a unique CONNID along with an initial address transfer of a connection. Later transfers associated with this connection (for example, data transfers) also provide the CONNID value so both sender and receiver (as well as any device monitoring transfers on shared communications bus 1012) can unambiguously identify transfers associated with the connection. One advantage of using a CONNID is that transfers belonging to different transactions can be interleaved arbitrarily between multiple devices on a per cycle basis. In one embodiment, shared communications bus 1012 implements a fully pipelined protocol that requires strict control over transaction ordering in order to guarantee proper system operation. Without the use of a CONNID, ordering constraints within a particular transaction may be violated because transfers associated with a particular connection are not identified. • Weber at 12:1–20

'052 Patent Claim Limitation	Weber
	Second, in one embodiment the connection identifier allows the target to choose which requests it rejects. The target may associate meanings such as transfer priority to the CONNID values, and therefore decide which requests to act upon based upon a combination of the CONNID value and the internal state of the target. For instance, a target might have separate queues for storing transfer requests of different priorities. Referring to FIG. 9, the target might have a queue for low priority requests (which present with an odd CONNID) and a queue for high priority requests (which present with an even CONNID). Thus, the CONNID 1 WRITE ADDRE0 request of cycle 1 would be rejected if the low-priority queue were full, whereas the CONNID 2 READ ADDR0 transfer could be completed successfully based upon available high-priority queue resources. Such differences in transfer priorities are very common in highly-integrated electronic systems, and the ability for the target to deliver higher quality of service to higher priority transfer requests adds significantly to the overall predictability of the system. • Weber at 12:53–67 Finally, the CONNID values that are associated with particular initiator transactions should typically be chosen to provide useful information such as transfer priorities but also to minimize implementation cost. It is useful to choose the specific CONNID values at system design time, so the values can be guaranteed to be unique and can be ordered to simplify comparison and other operations. Furthermore, it is frequently useful to be able to change the CONNID values during operation of the communications system so as to alter the performance and predictability aspects of the system. Preferred implementations of the present invention enable flexible system configuration by storing the CONNID values in ROM or RAM resources of the functional blocks, so they may be readily reconfigured at either system build time or system run time. • Weber at 15:30–37 Provision of per-thread flow control permits end-to-end performance guarantees to be determined across the interconnect. End-to-end data channels are linked using the threads of the interconnect, and per-thread flow control allows for end-to-end flow control between a particular initiator and target. One embodiment of end-to-end performance guarantees is described with reference to FIGS. 14, 15 a, 15 b, 16, 17, 18, 19, 20 and 21.

'052 Patent Claim Limitation	Weber
	- Weber at Fig. 14 FIG. 14 - Weber at 15:38–49 An exemplary system is shown in FIG. 14. The system consists of one or more functional blocks 1405, 1410, 1415, 1420, 1425, 1430 and one or more communication subsystems 1435, 1440 that are all tied together using a set of interfaces 1445, 1450, 1455, 1460, 1465, 1470, 1475, 1480. The functional blocks 145, 1410, 1415, 1420, 1425, 1430 perform the actual work done by the system (such as computation). In this document, the focus is on communication between functional blocks. Functional blocks are broken down into initiators 145, 1410, 1420 and targets 1415, 1425, 1430, depending on whether a functional block initiates or is the target of a data flow. - Weber at 15:50–64 For purpose of discussion herein, a data flow is a set of commands, data items or events being communicated over time between two functional blocks. Any given request from an initiator to a target is a request to perform some action. For example, read and write requests cause a target to

'052 Patent Claim Limitation	Weber
	retrieve and store the data respectively. A channel is a portion of a set of physical resources in a component that is dedicated to service a particular data flow. For example, a target may have different data channels to service different incoming data flows. A communication subsystem transmits data items from one of its input ports to an output port, using a dedicated data channel for each data flow. Each of the ports leads from/to another communications subsystem or a functional block via an interface. Interfaces also have dedicated channels for each data flow. • Weber at 15:65–16:9 Thus, for example, the end-to-end progress of a data flow may be as follows. The data flow originates at an initiator. It travels to an input port of a first communication subsystem via a channel of an interface. Once inside the first communication subsystem, it travels to an output port, via a channel. It then may cross another interface to other communication subsystems until it finally reaches a target. It is serviced by a dedicated channel inside the target and completes if no response is needed. If a response is needed, the data flow then reverses its direction through the communications subsystem(s), and completes when it reaches the originating initiator. • Weber at 16:10–21 In one embodiment, the interfaces tying the different components together have the ability to tag each request and response transmitted with the identity of the channel it belongs to, and the ability to transmit flow control on a per-channel basis. These features of the interfaces are desirable to achieve independent data channels that allow the quality of service guarantees given by an individual component to be propagated to another component that it is connected to via an interface. Thus, for example with reference to FIG. 4, the quality of service guarantees include the signal propagation time from initiator functional block 1002 to target functional block 1008. • Weber 16:22–45 FIG. 15 a shows an example of a functional block's quality of service guarantees. In this case, a memory system is shown that has differential quality of service guarantees for three channels. The quality of service guarantees given are minimum bandwidth and maximum latency while servicing an 8-byte read or write request. In this example, each channel can sustain 10 M 8-byte requests per second. Channel 1 has been optimized for lower latency and guarantees a maximum service latency

'052 Patent Claim Limitation	Weber
	of 100 ns, while channels 2 and 3 only guarantee a maximum service latency of 200 ns. It is common for the performance guarantees given by a particular functional block to be much more complex than shown here and to depend on a lot of different factors such as request size, inter-arrival interval, etc. The metrics used here have been chosen to keep the example simple. While maximum latency and minimum bandwidth are two common types of performance parameters, quality of service guarantees using other parameters such as, maximum outstanding requests, maximum variance in service latency, etc. are equally possible. In addition, if a particular component is too complex to easily achieve differential quality of service guarantees, the methodology can be applied recursively to that component by breaking it down into smaller, more manageable pieces. • Weber 16:46–61 FIG. 15 b shows an example of a communication subsystem's quality of service guarantees. This particular communication system has two ports for connecting initiators X 210 and Y 215 and one port for connecting a target Z 220. Port X 210 in turn supports two different incoming channels, port Y one incoming channel and port Z, three outgoing channels. This yields three data channels (A, B, and C) within the communications subsystem. In the present example, no quality of service guarantees have been given for channel A, again to simplify the discussion herein and also to note that sometimes certain data flows do not have guarantees. Channel B is serviced with a minimum bandwidth of 4 bytes every 4 cycles and a maximum latency of 3 cycles for each direction. Channel C is serviced with a minimum bandwidth of 4 bytes every 2 cycles and a maximum latency of 5 cycles for each direction. • Weber at Figs. 17–18

'052 Patent Claim Limitation	Weber
	<pre> graph TD START1([START]) --> 1710[FOR INITIATOR DATA FLOWS IN THE SYSTEM] 1710 --> 1715[MAP DATA FLOW THROUGH THE SYSTEM] 1715 --> 1720[ALIGN/CONVERT PERFORMANCE GUARANTEES] 1720 --> 1730[AGGREGATE PERFORMANCE GUARANTEES] 1730 --> D1{MORE DATA FLOWS?} D1 -- YES --> 1710 D1 -- NO --> END1([END]) </pre> FIG. 17 <pre> graph TD START2([START]) --> 1810[FOR COMPONENTS IN PATH FROM INITIATOR TO TARGET] 1810 --> 1815[PICK AN AVAILABLE CHANNEL IN THIS COMPONENT] 1815 --> 1820[MARK CHANNEL AS USED] 1820 --> D2{MORE COMPONENTS?} D2 -- YES --> 1810 D2 -- NO --> END2([END]) </pre> FIG. 18 - Weber at 17:26–32 One embodiment of a process for calculating end-to-end system performance guarantees for different data flows is shown in FIG. 17. For each initiator data flow in the system 1710, a mapping of the data flow through the channels of the system is first determined 1715. An exemplary mapping process is further illustrated in FIG. 18 and will be discussed below. - Weber at 17:59–18:6 FIG. 18 focuses in on one embodiment of the data flow mapping process that is part of the global mechanism shown in FIG. 17. For each component in a path from initiator to target, e.g., through

'052 Patent Claim Limitation	Weber
	several interfaces and one or more communication subsystems step 1710, an available channel is chosen to carry the initiator channel step 1715. Picking a particular data flow out of the set of available channels for a given component can be a complex task. Many possible ways of accomplishing the task and optimizing the overall system performance exist. For example, one could choose the channel that most closely matches the performance requirements of the initiator's data flow. It is also possible to split a data flow among multiple paths from initiator to target. In the present example, it is simply assumed that a valid mapping is chosen. • Weber at Fig. 20 The diagram, labeled FIG. 20, illustrates a network architecture. At the top, there are three initiators: INITIATOR A (2005) with components 710, 1, and 2; INITIATOR B (2020) with component 1; and INITIATOR C (2045) with components 1, 3, and 2. INITIATOR A is connected to INTERFACE 1 (2060) via paths A and B. INITIATOR B is connected to INTERFACE 2 (2040) via path A. INITIATOR C is connected to INTERFACE 3 (2065) via paths A, B, and C. These three interfaces connect to a central COMMUNICATIONS SUBSYSTEM (2030). The subsystem has three ports: PORT X (2055) connected to INTERFACE 1, PORT Y (2040) connected to INTERFACE 2, and PORT Z (2037) connected to INTERFACE 3. The subsystem contains internal components 2050, 2035, and 2030. Arrows indicate data flow from the initiators through the interfaces and ports into the communications subsystem, and then from the subsystem to the target (C). • Weber at 19:27–34

'052 Patent Claim Limitation	Weber
	An example application of the quality of service aggregation mechanism using a simple system is shown in FIG. 20. An initiator A 2005 sources two data flows shown as 1 2010 and 2 2015. A second initiator B sources only one data flow 2025. The communications subsystem 2030 receives the initiator data flows on ports X 2035 and Y 2040, respectively, and sends all of these flows to a target C 2045 via port Z 2050. • Weber at 19:35–56 The first table in FIG. 21 shows one embodiment of the mapping of each initiator data flow through the system. So, for example, initiator A data flow 1 2010 is sent to communications subsystem port X 2035 via channel A 2060 of interface 1. It traverses the communication subsystem 2030 via channel B 2037, exits via port Z 2065, and is sent to target C 2045 via interface 3 channel A 2070. At target C 2045 channel 1 2075 is chosen to handle this data flow. The second table of FIG. 21 shows an example of quality of service guarantees after unit alignment for the three different initiator data flows. In this example, it is assumed that initiator A flow 1 is using 4-byte requests, initiator A flow 2 is using 16-byte requests, and initiator B flow 1 is using 8-byte requests. The unit conversion and alignment is accomplished using the process described above. The last table of FIG. 21 shows the exemplary system end-to-end quality of service guarantees for the three initiator data flows. In order to calculate the minimum bandwidth in the system for each data flow, the minimum bandwidth of each mapped data flow through each component is found. Maximum latencies are found by adding each of the component latencies along a data flow from initiator to target. • Weber at Fig. 12

'052 Patent Claim Limitation	Weber
	FIG. 12 Weber at 20:59–21:3 Referring to FIG. 12, an example of a use of Thread ID, consider a series of identical direct memory access (DMA) engines in a system. In FIG. 12, elements 1202 are identical DMA engines, each connected to an initiator interface module 1204. Initiator interface modules 1204 are connected to shared communications bus 1212. Target interface module 1206 is also connected to shared communications bus 1212 and transmits data from bus 1212 to DRAM controller 1208, which is a target functional block. Target interface module 1206 is connected to DRAM controller 1208 by interconnect 1214. DRAM controller 1208 controls access to DRAM 1213. See also, e.g., Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21.

'052 Patent Claim Limitation	Weber
	To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[1d] at least one network interface (NI) associated to said at least one first of said processing modules for controlling the communication between said at least one first of said plurality of processing modules (M) and said interconnect means (N), and	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: • Weber at 2:63–3:9 One embodiment of the invention includes several functional blocks, including at least one initiator functional block and one target functional block. Some initiator functional blocks may also function as target functional blocks. In one embodiment, the initiator functional block is coupled to an initiator interface module and the target functional block is coupled to a target interface module. The initiator functional block and the target functional block communicate to their respective interface modules and the interface modules communicate with each other. The initiator functional block communicates with the target functional block by establishing a connection, wherein a connection is a logical state in which data may pass between the initiator functional block and the target functional block. • Weber at Fig. 1

**'052 Patent Claim
Limitation**

Weber

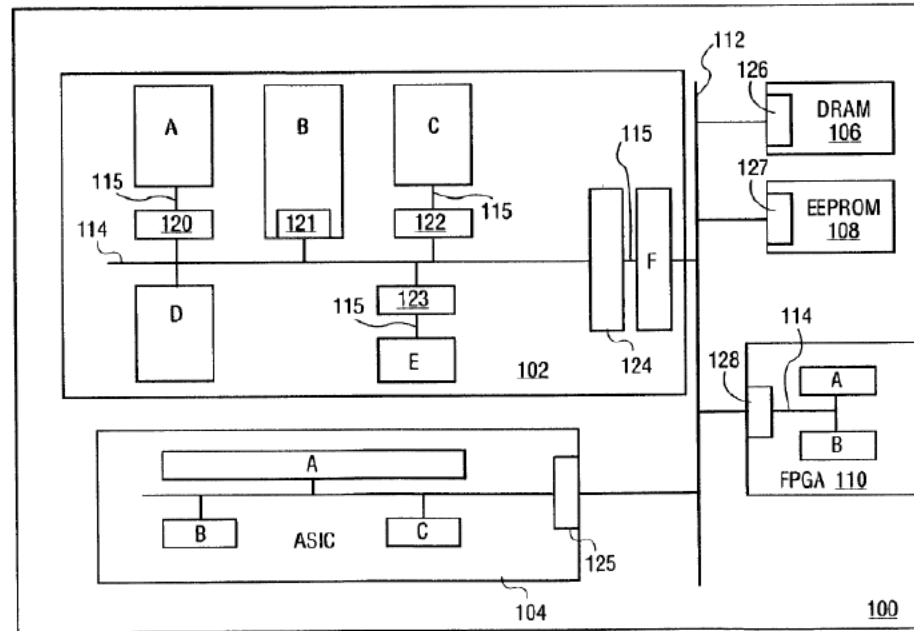
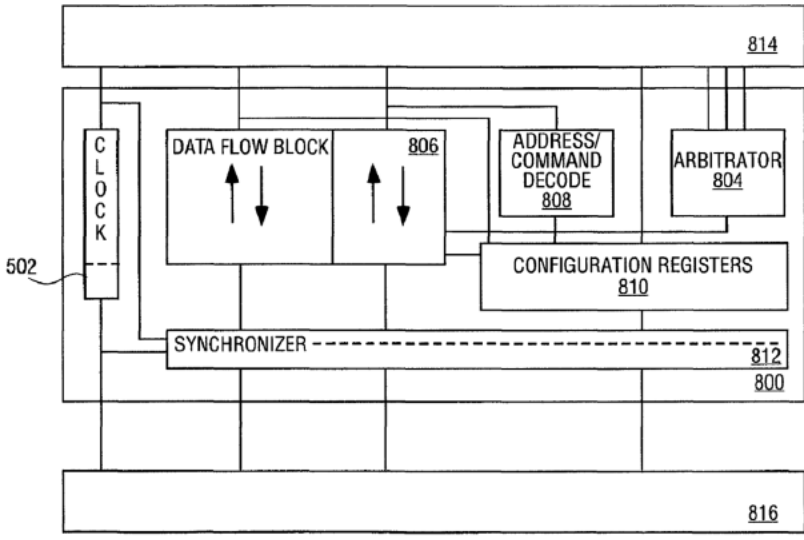


FIG. 1

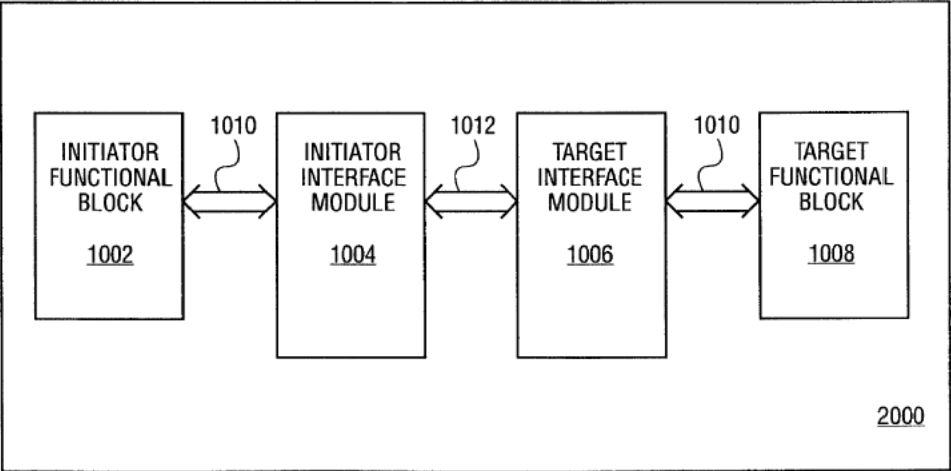
- Weber at 5:9–28

Shared communications bus 112 provides a shared communications bus between sub-systems of system 100. Shared communication bus 114 provides a shared communications bus between sub-systems or functional blocks on a single integrated circuit. Some of the functional blocks shown are connected to interface modules through which they send and receive signals to and from shared communications bus 112 or shared communications bus 114. Interconnect 115 is a local point-to-point interconnect for connecting interface modules to functional blocks. Interface modules 120–127 are connected to various functional blocks as shown. In this embodiment, interface modules 120, 122, 123 and 124 are physically separated from their connected functional block (A, B, C, E and 102, respectively). Interface modules 121, and 125–128 are essentially part of

'052 Patent Claim Limitation	Weber
	their respective functional blocks or sub-systems. Some functional blocks, such as 102D, do not require a dedicated interface module. The arrangement of sub-systems, functional blocks and interface modules is flexible and is determined by the system designer. • Weber at 5:29–47 In one embodiment there are four fundamental types of functional blocks. The four fundamental types are initiator, target, bridge, and snooping blocks. A typical target is a memory device, a typical initiator is a central processing unit (CPU). Functional blocks all communicate with one another via shared communications bus 112 or shared communications bus 114 and the protocol of one embodiment. Initiator and target functional blocks may communicate a shared communications bus through interface modules. An initiator functional block may communicate with a shared communications bus through an initiator interface module and a target functional block may communicate with a shared communications bus through a target interface module. An initiator interface module issues and receives read and write requests to and from functional blocks other than the one with which it is associated. In one embodiment, an initiator interface module is typically connected to a CPU, a digital signal processing (DSP) core, or a direct memory access (DMA) engine. • Weber at Fig. 2

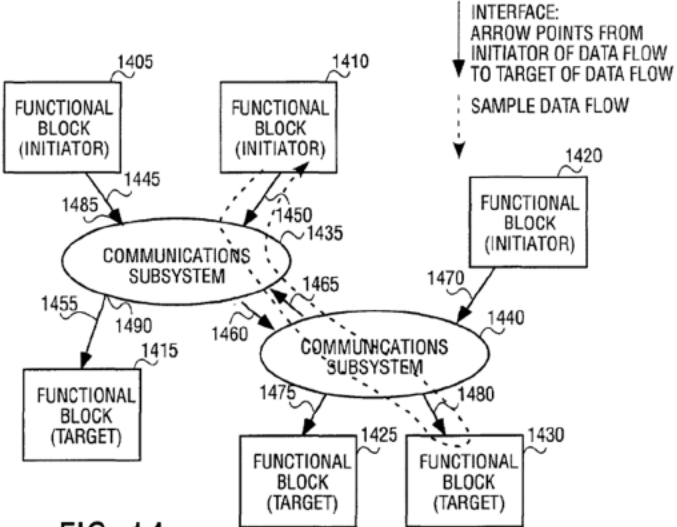
'052 Patent Claim Limitation	Weber
	 FIG. 2 - Weber at 5:48–57 FIG. 2 is a block diagram of an embodiment of an initiator interface module 800. Initiator interface module 800 includes clock generator 802, data flow block 806, arbitrator block 804, address/command decode block 808, configuration registers 810, and synchronizer 812. Initiator interface module 800 is connected to a shared communications bus 814 and to an initiator functional block 816. In one embodiment, shared communications bus 814 is a shared communications bus that connects sub-systems, as bus 112 does in FIG. 1. - Weber at Fig. 3

'052 Patent Claim Limitation	Weber
	<pre> graph TD 914[914] --- 906[906] 914 --- 908[908] 914 --- 916[916] 914 --- 918[918] 902[902] --- 906 902 --- 912[912] 906 --- 908 906 --- 916 908 --- 916 912 --- 916 916 --- 918 </pre> FIG. 3 Weber at 6:34–46 FIG. 3 is a block diagram of an embodiment of a target interface module 900. Target interface module 900 is connected to shared communications bus 914 and to target functional block 918. Target interface module 900 includes clock generator 902, data flow block 906, address/command decode block 908, synchronizer 912, and state registers in state control block 916. Blocks of target interface module 900 that are named similarly to blocks of initiator module 800 function in substantially the same way as explained with respect to initiator block 800. State registers and state control block 916 include registers that store, for example, client address base and an identifier for target functional block 918.

'052 Patent Claim Limitation	Weber
	• Weber at Fig. 4  FIG. 4 • Weber at 6:53–7:8 FIG. 4 is a block diagram of a part of a computer system 1000 according to one embodiment. FIG. 4 is useful in illustrating multilevel connection identification. System 1000 includes initiator functional block 1002, which is connected to initiator interface module 1004 by interconnect 1010. Initiator interface module 1004 is connected to target interface module 1006 by shared communications bus 1012. Target interface module 1006 is connected to target functional block 1008 by an interconnect 1010. Typically, shared communications bus 1012 is analogous to shared communications bus 112 of FIG. 1 or to shared communications bus 114 of FIG. 1. Interconnects 1010 are typically analogous to interconnect 115 of FIG. 1 in that they connect functional blocks to interface modules and are point-to-point, rather than shared, interconnects. Interconnects 1010 are typically physically shorter than shared communications bus 1012 because of their local nature. As will be explained more fully

'052 Patent Claim Limitation	Weber
	below, system 1000 uses two different levels of connection identification depending upon the requirements of a particular functional block. “Global” connection identification information is sent on shared communications bus 1012, while “local” connection information, or thread identification information, is sent in interconnects 1010. • Weber at Fig. 11

'052 Patent Claim Limitation	Weber
	The diagram shows a point-to-point interconnect 1010 between a MASTER 1102 and a SLAVE 1104. The signals are as follows: - Clock (bidirectional) - Cmd [3] (MASTER to SLAVE) - Addr [N] (MASTER to SLAVE) - DataOut [N] (MASTER to SLAVE) - ReqAccept (SLAVE to MASTER) - Resp [3] (SLAVE to MASTER) - DataIn [N] (SLAVE to MASTER) - RespAccept (MASTER to SLAVE) - Width [3] (I) (MASTER to SLAVE) - Burst [4] (I) (MASTER to SLAVE) - InterruptOut (I) (MASTER to SLAVE) - ErrorOut (I) (MASTER to SLAVE) - InterruptIn (I) (SLAVE to MASTER) - ErrorIn (I) (SLAVE to MASTER) - Reset (I) (SLAVE to MASTER) - ReqThreadID [4] (A) (MASTER to SLAVE) - RespThreadID [4] (A) (SLAVE to MASTER) - ReqThreadBusy [16] (AI) (SLAVE to MASTER) - RespThreadBusy [16] (AI) (MASTER to SLAVE) - ReqConnID [8] (AI) (MASTER to SLAVE) FIG. 11 • Weber at 13:1–13 FIG. 11 shows an interconnect 1010, which is a point-to-point interconnect as shown in FIG. 4. Interconnect 1010 includes additional signals as compared to the protocol described with reference to FIG. 5. As will be explained below, some of the additional signals are particularly useful as signals

'052 Patent Claim Limitation	Weber
	sent over point-to-point interconnects such as interconnects 1010. The protocol of interconnect 1010 controls point-to-point transfers between a master entity 1102 and a slave entity 1104 over a dedicated (non-shared) interconnect. Referring to FIG. 5, a master entity may be, for example, initiator functional block 1002 or target interface module 1006. A slave entity may be, for example, initiator interface module 1004 or target functional block 1008. • Weber at Fig. 14  FIG. 14 • Weber at 15:38–49 An exemplary system is shown in FIG. 14. The system consists of one or more functional blocks 1405, 1410, 1415, 1420, 1425, 1430 and one or more communication subsystems 1435, 1440 that are all tied together using a set of interfaces 1445, 1450, 1455, 1460, 1465, 1470, 1475, 1480. The functional blocks 145, 1410, 1415, 1420, 1425, 1430 perform the actual work done by the system (such as computation). In this document, the focus is on communication between functional blocks.

'052 Patent Claim Limitation	Weber
	Functional blocks are broken down into initiators 145, 1410, 1420 and targets 1415, 1425, 1430, depending on whether a functional block initiates or is the target of a data flow. • Weber at 15:65–16:9 Thus, for example, the end-to-end progress of a data flow may be as follows. The data flow originates at an initiator. It travels to an input port of a first communication subsystem via a channel of an interface. Once inside the first communication subsystem, it travels to an output port, via a channel. It then may cross another interface to other communication subsystems until it finally reaches a target. It is serviced by a dedicated channel inside the target and completes if no response is needed. If a response is needed, the data flow then reverses its direction through the communications subsystem(s), and completes when it reaches the originating initiator. • Weber at 16:10–21 In one embodiment, the interfaces tying the different components together have the ability to tag each request and response transmitted with the identity of the channel it belongs to, and the ability to transmit flow control on a per-channel basis. These features of the interfaces are desirable to achieve independent data channels that allow the quality of service guarantees given by an individual component to be propagated to another component that it is connected to via an interface. Thus, for example with reference to FIG. 4, the quality of service guarantees include the signal propagation time from initiator functional block 1002 to target functional block 1008. • Weber at Fig. 12

'052 Patent Claim Limitation	Weber
	FIG. 12 Weber at 20:59–21:3 Referring to FIG. 12, an example of a use of Thread ID, consider a series of identical direct memory access (DMA) engines in a system. In FIG. 12, elements 1202 are identical DMA engines, each connected to an initiator interface module 1204. Initiator interface modules 1204 are connected to shared communications bus 1212. Target interface module 1206 is also connected to shared communications bus 1212 and transmits data from bus 1212 to DRAM controller 1208, which is a target functional block. Target interface module 1206 is connected to DRAM controller 1208 by interconnect 1214. DRAM controller 1208 controls access to DRAM 1213. See also, e.g., Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21.

'052 Patent Claim Limitation	Weber
	To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[1e] a mapping means (A) for mapping the requested at least one communication service based on said specific communication properties to a connection based on a set of connection properties according to said at least one communication service identification,	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: • Weber at Abstract A communication system. One embodiment includes at least two functional blocks, wherein an first functional block communicates with a second functional block by establishing a connection. A connection is a logical state in which data may pass between the first functional block and the second functional block. One embodiment includes a bus coupled to each of the functional blocks and configured to carry a plurality of signals. The plurality of signals includes a thread identifier that indicates a transaction stream that the data transfer is part of and a busy signal identified by the thread identifier. The busy signal is issued by the target functional block when resources will be unavailable to perform a transfer. • Weber at 2:63–3:9 One embodiment of the invention includes several functional blocks, including at least one initiator functional block and one target functional block. Some initiator functional blocks may also function as target functional blocks. In one embodiment, the initiator functional block is coupled to an initiator interface module and the target functional block is coupled to a target interface module. The initiator functional block and the target functional block communicate to their respective interface modules and the interface modules communicate with each other. The initiator functional block communicates with the target functional block by establishing a connection, wherein a connection is a logical state in which data may pass between the initiator functional block and the target functional block.

'052 Patent Claim Limitation	Weber
	• Weber at 3:10–31 One embodiment also includes a bus configured to carry multiple signals, wherein the signals include a connection identifier signal that indicates a particular connection that a data transfer between an initiator functional block and a target functional block is part of. The connection identifier includes information about the connection, such as which functional block is the source of a transmission, a priority of a transfer request, and transfer ordering information. One embodiment also includes a thread identifier, which provides a subset of the information provided by the connection identifier. In one embodiment, the thread identifier is an identifier of local scope that identifies transfers between an interface module and a connected functional block, where in some embodiments, an interface module connects a functional block to a shared communications bus. The connection identifier is a an identifier of global scope that transfers information between interface modules or between functional blocks through their interface modules. Some functional blocks may require all the information provided by the connection identifier, while other functional blocks may require only the subset of information provided by the thread identifier. • Weber at 8:29–56 Connection identifier (CONNID) lines 226 carry a multi-bit signal driven by an initiator bus owner to indicate which connection the current transfer is part of. A connection is a logical state, established by an initiator, in which data may pass between the initiator and an associated target. The CONNID typically transmits information including the identity of the functional block initiating the transfer and ordering information regarding an order in which the transfer must be processed. In one embodiment, the information conveyed by the CONNID includes information regarding the priority of the transfer with respect to other transfers. In one embodiment the CONNID is a eight-bit code. An initiator interface module sends a unique CONNID along with an initial address transfer of a connection. Later transfers associated with this connection (for example, data transfers) also provide the CONNID value so both sender and receiver (as well as any device monitoring transfers on shared communications bus 1012) can unambiguously identify transfers associated with the connection. One advantage of using a CONNID is that transfers belonging to different transactions can be interleaved arbitrarily between multiple devices on a per cycle basis. In one embodiment, shared communications bus 1012 implements a fully pipelined protocol that requires strict control over transaction ordering in order to guarantee proper system operation. Without the use of a CONNID,

'052 Patent Claim Limitation	Weber
	ordering constraints within a particular transaction may be violated because transfers associated with a particular connection are not identified. Weber at 9:1–17 The present invention uses a CONNID signal, in part, to allow overlapping of dependent commands. Therefore, use of a CONNID improves system performance and efficiency. Another benefit of the CONNID of the present invention is that communication system predictability is enhanced because it allows a shared functional block to respond to requests based upon quality of service guarantees that may vary between connections. For example, data requested to operate a computer display cannot tolerate unpredictable delay because delay causes the display to flicker. Therefore, the CONNID may be used to prioritize data requests from a display controller so that requests from the display controller to a common resource are serviced before other requests. The present invention also allows for flexible reconfiguration of the CONNID to retune system performance. Weber at 12:1–20 Second, in one embodiment the connection identifier allows the target to choose which requests it rejects. The target may associate meanings such as transfer priority to the CONNID values, and therefore decide which requests to act upon based upon a combination of the CONNID value and the internal state of the target. For instance, a target might have separate queues for storing transfer requests of different priorities. Referring to FIG. 9, the target might have a queue for low priority requests (which present with an odd CONNID) and a queue for high priority requests (which present with an even CONNID). Thus, the CONNID 1 WRITE ADDRE0 request of cycle 1 would be rejected if the low-priority queue were full, whereas the CONNID 2 READ ADDR0 transfer could be completed successfully based upon available high-priority queue resources. Such differences in transfer priorities are very common in highly-integrated electronic systems, and the ability for the target to deliver higher quality of service to higher priority transfer requests adds significantly to the overall predictability of the system. Weber at 12:53–67 Finally, the CONNID values that are associated with particular initiator transactions should typically be chosen to provide useful information such as transfer priorities but also to minimize

'052 Patent Claim Limitation	Weber
	implementation cost. It is useful to choose the specific CONNID values at system design time, so the values can be guaranteed to be unique and can be ordered to simplify comparison and other operations. Furthermore, it is frequently useful to be able to change the CONNID values during operation of the communications system so as to alter the performance and predictability aspects of the system. Preferred implementations of the present invention enable flexible system configuration by storing the CONNID values in ROM or RAM resources of the functional blocks, so they may be readily reconfigured at either system build time or system run time. • Weber at 14:1–13 Request Thread Identifier (ReqThreadID), in one embodiment, is a four-bit signal that provides the thread number associated with a current transaction intended for slave 1104. All commands executed with a particular thread ID must execute in order with respect to one another, but they may execute out of order with respect to commands from other threads. Response Thread Identifier (RespThreadID) provides a thread number associated with a current response. Because responses in a thread may return out of order with respect to other threads, RespThreadID is necessary to identify which thread's command is being responded to. In one embodiment, ReqThreadID and RespThreadID are optional signals, but if one is used, both should be used. • Weber at 14:14–22 The Request Connection Identifier (ReqConnID) provides the CONNID associated with the current transaction intended for slave or target. CONNIDs provide a mechanism by which a system entity may associate particular transactions with the system entity. One use of the CONNID is in establishing request priority among various initiators. Another use is in associating actions or data transfers with initiator identity rather than the address presented with the transaction request. • Weber at 15:30–37 Provision of per-thread flow control permits end-to-end performance guarantees to be determined across the interconnect. End-to-end data channels are linked using the threads of the interconnect, and per-thread flow control allows for end-to-end flow control between a particular initiator and target. One embodiment of end-to-end performance guarantees is described with reference to FIGS. 14, 15 a, 15 b, 16, 17, 18, 19, 20 and 21.

'052 Patent Claim Limitation	Weber
	- Weber at Fig. 14 FIG. 14 - Weber at 15:38–49 An exemplary system is shown in FIG. 14. The system consists of one or more functional blocks 1405, 1410, 1415, 1420, 1425, 1430 and one or more communication subsystems 1435, 1440 that are all tied together using a set of interfaces 1445, 1450, 1455, 1460, 1465, 1470, 1475, 1480. The functional blocks 145, 1410, 1415, 1420, 1425, 1430 perform the actual work done by the system (such as computation). In this document, the focus is on communication between functional blocks. Functional blocks are broken down into initiators 145, 1410, 1420 and targets 1415, 1425, 1430, depending on whether a functional block initiates or is the target of a data flow. - Weber at 15:50–64 For purpose of discussion herein, a data flow is a set of commands, data items or events being communicated over time between two functional blocks. Any given request from an initiator to a

'052 Patent Claim Limitation	Weber
	target is a request to perform some action. For example, read and write requests cause a target to retrieve and store the data respectively. A channel is a portion of a set of physical resources in a component that is dedicated to service a particular data flow. For example, a target may have different data channels to service different incoming data flows. A communication subsystem transmits data items from one of its input ports to an output port, using a dedicated data channel for each data flow. Each of the ports leads from/to another communications subsystem or a functional block via an interface. Interfaces also have dedicated channels for each data flow. • Weber at 15:65–16:9 Thus, for example, the end-to-end progress of a data flow may be as follows. The data flow originates at an initiator. It travels to an input port of a first communication subsystem via a channel of an interface. Once inside the first communication subsystem, it travels to an output port, via a channel. It then may cross another interface to other communication subsystems until it finally reaches a target. It is serviced by a dedicated channel inside the target and completes if no response is needed. If a response is needed, the data flow then reverses its direction through the communications subsystem(s), and completes when it reaches the originating initiator. • Weber at 16:10–21 In one embodiment, the interfaces tying the different components together have the ability to tag each request and response transmitted with the identity of the channel it belongs to, and the ability to transmit flow control on a per-channel basis. These features of the interfaces are desirable to achieve independent data channels that allow the quality of service guarantees given by an individual component to be propagated to another component that it is connected to via an interface. Thus, for example with reference to FIG. 4, the quality of service guarantees include the signal propagation time from initiator functional block 1002 to target functional block 1008. • Weber 16:22–45 FIG. 15 a shows an example of a functional block's quality of service guarantees. In this case, a memory system is shown that has differential quality of service guarantees for three channels. The quality of service guarantees given are minimum bandwidth and maximum latency while servicing an 8-byte read or write request. In this example, each channel can sustain 10 M 8-byte requests per

'052 Patent Claim Limitation	Weber
	second. Channel 1 has been optimized for lower latency and guarantees a maximum service latency of 100 ns, while channels 2 and 3 only guarantee a maximum service latency of 200 ns. It is common for the performance guarantees given by a particular functional block to be much more complex than shown here and to depend on a lot of different factors such as request size, inter-arrival interval, etc. The metrics used here have been chosen to keep the example simple. While maximum latency and minimum bandwidth are two common types of performance parameters, quality of service guarantees using other parameters such as, maximum outstanding requests, maximum variance in service latency, etc. are equally possible. In addition, if a particular component is too complex to easily achieve differential quality of service guarantees, the methodology can be applied recursively to that component by breaking it down into smaller, more manageable pieces. • Weber 16:46–61 FIG. 15 b shows an example of a communication subsystem's quality of service guarantees. This particular communication system has two ports for connecting initiators X 210 and Y 215 and one port for connecting a target Z 220. Port X 210 in turn supports two different incoming channels, port Y one incoming channel and port Z, three outgoing channels. This yields three data channels (A, B, and C) within the communications subsystem. In the present example, no quality of service guarantees have been given for channel A, again to simplify the discussion herein and also to note that sometimes certain data flows do not have guarantees. Channel B is serviced with a minimum bandwidth of 4 bytes every 4 cycles and a maximum latency of 3 cycles for each direction. Channel C is serviced with a minimum bandwidth of 4 bytes every 2 cycles and a maximum latency of 5 cycles for each direction. • Weber at Figs. 17–18

'052 Patent Claim Limitation	Weber
	<pre> graph TD subgraph FIG_17 [FIG. 17] S1([START]) --> L1[FOR INITIATOR DATA FLOWS IN THE SYSTEM 1710] L1 --> M1[MAP DATA FLOW THROUGH THE SYSTEM 1715] M1 --> A1[ALIGN/CONVERT PERFORMANCE GUARANTEES 1720] A1 --> AG1[AGGREGATE PERFORMANCE GUARANTEES 1730] AG1 --> D1{MORE DATA FLOWS?} D1 -- YES --> L1 D1 -- NO --> E1([END]) end subgraph FIG_18 [FIG. 18] S2([START]) --> L2[FOR COMPONENTS IN PATH FROM INITIATOR TO TARGET 1810] L2 --> P1[PICK AN AVAILABLE CHANNEL IN THIS COMPONENT 1815] P1 --> M2[MARK CHANNEL AS USED 1820] M2 --> D2{MORE COMPONENTS?} D2 -- YES --> L2 D2 -- NO --> E2([END]) end </pre> FIG. 17 FIG. 18 FIG. 17 FIG. 18 Weber at 17:26–32 One embodiment of a process for calculating end-to-end system performance guarantees for different data flows is shown in FIG. 17. For each initiator data flow in the system 1710, a mapping of the data flow through the channels of the system is first determined 1715. An exemplary mapping process is further illustrated in FIG. 18 and will be discussed below. Weber at 17:59–18:6 FIG. 18 focuses in on one embodiment of the data flow mapping process that is part of the global mechanism shown in FIG. 17. For each component in a path from initiator to target, e.g., through

'052 Patent Claim Limitation	Weber
	several interfaces and one or more communication subsystems step 1710, an available channel is chosen to carry the initiator channel step 1715. Picking a particular data flow out of the set of available channels for a given component can be a complex task. Many possible ways of accomplishing the task and optimizing the overall system performance exist. For example, one could choose the channel that most closely matches the performance requirements of the initiator's data flow. It is also possible to split a data flow among multiple paths from initiator to target. In the present example, it is simply assumed that a valid mapping is chosen. • Weber at Fig. 20 The diagram illustrates a network architecture. At the top, there are three initiators: INITIATOR A (2005), INITIATOR B (2020), and INITIATOR C (2045). INITIATOR A has two output channels labeled 1 and 2, with a reference numeral 710. INITIATOR B has one output channel labeled 1, with a reference numeral 2025. INITIATOR C has three output channels labeled 1, 3, and 2, with a reference numeral 2065. These initiators are connected to a central COMMUNICATIONS SUBSYSTEM (2030) via three interfaces: INTERFACE 1 (2060), INTERFACE 2 (2040), and INTERFACE 3 (2070). The COMMUNICATIONS SUBSYSTEM (2030) contains three ports: PORT X (2055), PORT Y (2040), and PORT Z (2037). The connections are as follows: INITIATOR A's channel 1 connects to PORT X via path A; INITIATOR A's channel 2 connects to PORT X via path B; INITIATOR B's channel 1 connects to PORT Y via path A; INITIATOR C's channel 1 connects to PORT Z via path A; INITIATOR C's channel 3 connects to PORT Z via path B; and INITIATOR C's channel 2 connects to PORT Z via path C. The COMMUNICATIONS SUBSYSTEM (2030) is represented by a central block with internal components 2050 and 2035. The entire system is labeled FIG. 20. • Weber at 19:27–34

'052 Patent Claim Limitation	Weber
	An example application of the quality of service aggregation mechanism using a simple system is shown in FIG. 20. An initiator A 2005 sources two data flows shown as 1 2010 and 2 2015. A second initiator B sources only one data flow 2025. The communications subsystem 2030 receives the initiator data flows on ports X 2035 and Y 2040, respectively, and sends all of these flows to a target C 2045 via port Z 2050. • Weber at 19:35–56 The first table in FIG. 21 shows one embodiment of the mapping of each initiator data flow through the system. So, for example, initiator A data flow 1 2010 is sent to communications subsystem port X 2035 via channel A 2060 of interface 1. It traverses the communication subsystem 2030 via channel B 2037, exits via port Z 2065, and is sent to target C 2045 via interface 3 channel A 2070. At target C 2045 channel 1 2075 is chosen to handle this data flow. The second table of FIG. 21 shows an example of quality of service guarantees after unit alignment for the three different initiator data flows. In this example, it is assumed that initiator A flow 1 is using 4-byte requests, initiator A flow 2 is using 16-byte requests, and initiator B flow 1 is using 8-byte requests. The unit conversion and alignment is accomplished using the process described above. The last table of FIG. 21 shows the exemplary system end-to-end quality of service guarantees for the three initiator data flows. In order to calculate the minimum bandwidth in the system for each data flow, the minimum bandwidth of each mapped data flow through each component is found. Maximum latencies are found by adding each of the component latencies along a data flow from initiator to target. • Weber at 19:57–20:6 The Request Connection Identifier (ReqConnID) provides the CONNID associated with the current transaction intended for slave or target. CONNIDs provide a mechanism by which a system entity may associate particular transactions with the system entity. One use of the CONNID is in establishing request priority among various initiators. Another use is in associating actions or data transfers with initiator identity rather than the address presented with the transaction request. The embodiment of FIG. 11 provides end-to-end connection identification with CONNID as well as point-to-point, or more local identification with Thread ID. A Thread ID is an identifier of local

'052 Patent Claim Limitation	Weber
	scope that simply identifies transfers between the interface module and its connected functional block. In contrast, the CONNID is an identifier of global scope that identifies transfers between two interface modules (and, if required, their connected functional blocks). • Weber at 20:25–58 Referring to FIG. 4, a CONNID should be unique when transferred between interface modules 1004 and 1006 on shared communications bus 1012. The CONNID may be sent over a local interconnect, such as interconnect 1010. In many cases, however, it is much more efficient to use only Thread ID between a functional block and its interface module. For example initiator functional block 1002 may not require all the information provided by the CONNID. Also, in some systems, multiple identical initiator functional blocks 1002 may exist with the same CONNID so that a particular target functional block 1008 receiving a transfer will not know which connection it is actually part of unless logic in initiator interface module 1004 translates the “local” CONNID to a unique “global” CONNID. The design and implementation of such a translation functionality in an interface module is complicated and expensive. In such cases, the CONNID may be sent between interface modules over shared communications bus 1012 while the Thread ID is sent between a functional block and an interface module. In the case of an initiator functional block, a one-to-one static correspondence may exist between Thread ID and CONNID. For example if the Thread ID is “1”, a single CONNID is mapped for a particular interface module, solving the problem of multiple, identical functional blocks. In the case of a target functional block, there is a one-to-one dynamic correspondence between a Thread ID and a CONNID. If a target functional block supports two simultaneous threads, the target interface module acquires the CONNID of an open connection and associates it with a thread as needed. For example, a target interface module receives a CONNID of “7”, and then maps CONNID 7 to thread “0”. Thereafter, all transfers with CONNID 7 are associated with thread 0 until connection 7 is closed. • Weber at 21:24–34 In one embodiment, each initiator interface module 1204 maps a unique CONNID to each of two Thread Ids from a connected DMA engine 1202. Each of DMA engines 1202 use a single bit, for example, Thread ID of FIG. 11, to distinguish between its two threads. For each transfer over shared

'052 Patent Claim Limitation	Weber
	communications bus a unique CONNID is sent to target interface module 1206. The CONNID may include priority information, for example, assigning high priority to requests for graphics data. The high priority graphics data request is immediately serviced by DRAM controller 1208 while lower priority request may be required to wait. See also, e.g., Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant’s Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the ’052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person’s understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[1f] wherein said at least one communication service identification comprises at least one communication thread or at least one address range, said address range for identifying one or more second processing modules (S) or a memory region within said one or more second processing modules (S).	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: • Weber at Abstract A communication system. One embodiment includes at least two functional blocks, wherein an first functional block communicates with a second functional block by establishing a connection. A connection is a logical state in which data may pass between the first functional block and the second functional block. One embodiment includes a bus coupled to each of the functional blocks and configured to carry a plurality of signals. The plurality of signals includes a thread identifier that indicates a transaction stream that the data transfer is part of and a busy signal identified by the thread identifier. The busy signal is issued by the target functional block when resources will be unavailable to perform a transfer. • Weber at 2:63–3:9

'052 Patent Claim Limitation	Weber
	One embodiment of the invention includes several functional blocks, including at least one initiator functional block and one target functional block. Some initiator functional blocks may also function as target functional blocks. In one embodiment, the initiator functional block is coupled to an initiator interface module and the target functional block is coupled to a target interface module. The initiator functional block and the target functional block communicate to their respective interface modules and the interface modules communicate with each other. The initiator functional block communicates with the target functional block by establishing a connection, wherein a connection is a logical state in which data may pass between the initiator functional block and the target functional block. • Weber at 3:10–31 One embodiment also includes a bus configured to carry multiple signals, wherein the signals include a connection identifier signal that indicates a particular connection that a data transfer between an initiator functional block and a target functional block is part of. The connection identifier includes information about the connection, such as which functional block is the source of a transmission, a priority of a transfer request, and transfer ordering information. One embodiment also includes a thread identifier, which provides a subset of the information provided by the connection identifier. In one embodiment, the thread identifier is an identifier of local scope that identifies transfers between an interface module and a connected functional block, where in some embodiments, an interface module connects a functional block to a shared communications bus. The connection identifier is a an identifier of global scope that transfers information between interface modules or between functional blocks through their interface modules. Some functional blocks may require all the information provided by the connection identifier, while other functional blocks may require only the subset of information provided by the thread identifier. • Weber at Fig. 11

'052 Patent Claim Limitation	Weber
	FIG. 11 • Weber at 13:14–38 Signals shown in FIG. 11 are labeled with signal names. In addition, some signal names are followed by a notation or notations in parentheses or brackets. The notations are as follows: (I) The signal is optional and is independently configurable

'052 Patent Claim Limitation	Weber
	(A) The signal must be configured together with signals having similar notations (AI) The signal is independently configurable if (A) interface modules exist [#] Maximum signal width The clock signal is the clock of a connected functional block. The command (Cmd) signal indicates the type of transfer on the bus. Commands can be issued independent of data. The address (Addr) signal is typically an indication of a particular resource that an initiator functional block wishes to access. Request Accept (ReqAccept) is a handshake signal whereby slave 1104 allows master 1102 to release Cmd, Addr and DataOut from one transfer and reuse them for another transfer. If slave 1104 is busy and cannot participate in a requested transfer, master 1102 must continue to present Cmd, Addr and DataOut. DataOut is data sent from a master to a slave, typically in a write transfer. DataIn typically carries read data. • Weber at 14:1–13 Request Thread Identifier (ReqThreadID), in one embodiment, is a four-bit signal that provides the thread number associated with a current transaction intended for slave 1104. All commands executed with a particular thread ID must execute in order with respect to one another, but they may execute out of order with respect to commands from other threads. Response Thread Identifier (RespThreadID) provides a thread number associated with a current response. Because responses in a thread may return out of order with respect to other threads, RespThreadID is necessary to identify which thread's command is being responded to. In one embodiment, ReqThreadID and RespThreadID are optional signals, but if one is used, both should be used. • Weber at 14:14–22 The Request Connection Identifier (ReqConnID) provides the CONNID associated with the current transaction intended for slave or target. CONNIDs provide a mechanism by which a system entity may associate particular transactions with the system entity. One use of the CONNID is in establishing request priority among various initiators. Another use is in associating actions or data transfers with initiator identity rather than the address presented with the transaction request. • Weber at 15:30–37

'052 Patent Claim Limitation	Weber
	Provision of per-thread flow control permits end-to-end performance guarantees to be determined across the interconnect. End-to-end data channels are linked using the threads of the interconnect, and per-thread flow control allows for end-to-end flow control between a particular initiator and target. One embodiment of end-to-end performance guarantees is described with reference to FIGS. 14, 15 a, 15 b, 16, 17, 18, 19, 20 and 21. • Weber at 19:57–20:6 The Request Connection Identifier (ReqConnID) provides the CONNID associated with the current transaction intended for slave or target. CONNIDs provide a mechanism by which a system entity may associate particular transactions with the system entity. One use of the CONNID is in establishing request priority among various initiators. Another use is in associating actions or data transfers with initiator identity rather than the address presented with the transaction request. The embodiment of FIG. 11 provides end-to-end connection identification with CONNID as well as point-to-point, or more local identification with Thread ID. A Thread ID is an identifier of local scope that simply identifies transfers between the interface module and its connected functional block. In contrast, the CONNID is an identifier of global scope that identifies transfers between two interface modules (and, if required, their connected functional blocks). • Weber at 20:25–58 Referring to FIG. 4, a CONNID should be unique when transferred between interface modules 1004 and 1006 on shared communications bus 1012. The CONNID may be sent over a local interconnect, such as interconnect 1010. In many cases, however, it is much more efficient to use only Thread ID between a functional block and its interface module. For example initiator functional block 1002 may not require all the information provided by the CONNID. Also, in some systems, multiple identical initiator functional blocks 1002 may exist with the same CONNID so that a particular target functional block 1008 receiving a transfer will not know which connection it is actually part of unless logic in initiator interface module 1004 translates the “local” CONNID to a unique “global” CONNID. The design and implementation of such a translation functionality in an interface module is complicated and expensive. In such cases, the CONNID may be sent between interface modules over shared communications bus 1012 while the Thread ID is sent between a functional block and an interface module.

'052 Patent Claim Limitation	Weber
	In the case of an initiator functional block, a one-to-one static correspondence may exist between Thread ID and CONNID. For example if the Thread ID is “1”, a single CONNID is mapped for a particular interface module, solving the problem of multiple, identical functional blocks. In the case of a target functional block, there is a one-to-one dynamic correspondence between a Thread ID and a CONNID. If a target functional block supports two simultaneous threads, the target interface module acquires the CONNID of an open connection and associates it with a thread as needed. For example, a target interface module receives a CONNID of “7”, and then maps CONNID 7 to thread “0”. Thereafter, all transfers with CONNID 7 are associated with thread 0 until connection 7 is closed. • Weber at 21:4–34 A DMA engine is an example of an initiator functional block that also functions as a target functional block. When the DMA engine is programmed by software, it acts as a target. Thereafter, the DMA engine is an initiator. Because a DMA engine performs both read and write operations, two connections can be associated with a single DMA engine. If some buffering is available in the DMA engine, read and write operations may be decoupled so that both types of operations can be performed concurrently. A read may occur from a long latency storage device which requires the read data to be buffered on the DMA engine before a write operation writes the data. In one embodiment, each of DMA engines 1202 uses a Thread ID to identify the read stream and a different Thread ID to identify the write stream. The DMA engine does not require more information, such as what other functional block participates in a transaction. Therefore, a CONNID is not required to be sent from the DMA engine 1202 to a connected interface module 1204. Mapping of a Thread ID to a CONNID occurs in the interface module 1204. In one embodiment, each initiator interface module 1204 maps a unique CONNID to each of two Thread Ids from a connected DMA engine 1202. Each of DMA engines 1202 use a single bit, for example, Thread ID of FIG. 11, to distinguish between its two threads. For each transfer over shared communications bus a unique CONNID is sent to target interface module 1206. The CONNID may include priority information, for example, assigning high priority to requests for graphics data. The high priority graphics data request is immediately serviced by DRAM controller 1208 while lower priority request may be required to wait.

'052 Patent Claim Limitation	Weber
	<i>See also, e.g.,</i> Weber at 1:7–2:52, 2:56–3:31, 4:15–26, 4:43–7:42, 8:29–9:17, 9:18–12:67, 14:1–15:12, 15:38–18:30, 19:1–21:34, Figs. 1–21. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant’s Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the ’052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person’s understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

B. Claim 2

'052 Patent Claim Limitation	Weber
[2] Integrated circuit according to claim 1, wherein said mapping means (A) is arranged in said at least one network interface (NI).	Weber discloses and/or renders obvious this claim based on at least the following citations. See, e.g.: <i>See supra</i> Claim 1; <i>see also supra</i> Claim [1d]–[1e]. To the extent Weber does not expressly disclose this claim, a person of ordinary skill in the art would have determined that this claim is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant’s Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the ’052 Patent in a manner that would result in the subject matter of this claim given, at the very least, the person’s understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

C. Claim 3

'052 Patent Claim Limitation	Weber
[3] Integrated circuit according to claim 1, wherein said at least one communication thread is mapped to at least one connection based on a set of connection properties.	Weber discloses and/or renders obvious this claim based on at least the following citations. <i>See, e.g.,</i> <i>See supra</i> Claim 1; <i>see also supra</i> Claim [1e]–[1f]. To the extent Weber does not expressly disclose this claim, a person of ordinary skill in the art would have determined that this claim is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this claim given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

D. Claim 4

'052 Patent Claim Limitation	Weber
[4] Integrated circuit according to claim 3, wherein said communication service identification further comprises at least one address range in said at least one second processing module (S), wherein said at least one address range is mapped to at least one connection based on a set of connection properties.	Weber discloses and/or renders obvious this claim based on at least the following citations, at least in view of Plaintiff's apparent interpretation and/or application of this claim. <i>See, e.g.,</i> <i>See supra</i> Claim 3; <i>see also supra</i> Claim [1e]–[1f]. To the extent Weber does not expressly disclose this claim, a person of ordinary skill in the art would have determined that this claim is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this claim

'052 Patent Claim Limitation	Weber
	given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

E. Claim 5

'052 Patent Claim Limitation	Weber
[5] Integrated circuit according to claim 1, wherein said at least one address range is mapped to at least one connection based on a set of connection properties.	Weber discloses and/or renders obvious this claim based on at least the following citations, at least in view of Plaintiff's apparent interpretation and/or application of this claim. <i>See, e.g.,</i> <i>See supra</i> Claim 1; <i>see also supra</i> Claim [1e]–[1f]. To the extent Weber does not expressly disclose this claim, a person of ordinary skill in the art would have determined that this claim is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this claim given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

F. Claim 6

'052 Patent Claim Limitation	Weber
[6pre-a] Method of communication service mapping in an integrated	To the extent the preamble is limiting, Weber discloses and/or renders obvious this part of the preamble based on at least the following citations. <i>See, e.g.:</i> <i>See supra</i> claim [1pre], [1a].

'052 Patent Claim Limitation	Weber
circuit, having a plurality of processing modules (M, S),	To the extent Weber does not expressly disclose this part of the preamble, a person of ordinary skill in the art would have determined that this part of the preamble is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this part of the preamble given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[6pre-b] wherein at least one first of said processing modules (M) requests at least one communication service to at least one second processing module (S) based on specific communication properties and at least one communication service identification,	To the extent the preamble is limiting, Weber discloses and/or renders obvious this part of the preamble based on at least the following citations. See, e.g.: <i>See supra</i> claim [1b]. To the extent Weber does not expressly disclose this part of the preamble, a person of ordinary skill in the art would have determined that this part of the preamble is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this part of the preamble given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[6pre-c] wherein said at least one communication service identification comprises at least one communication thread or at least one address range, said address range for identifying one or more second processing modules (S) or a memory region	To the extent the preamble is limiting, Weber discloses and/or renders obvious this part of the preamble based on at least the following citations. See, e.g.: <i>See supra</i> claim [1f]. To the extent Weber does not expressly disclose this part of the preamble, a person of ordinary skill in the art would have determined that this part of the preamble is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art

'052 Patent Claim Limitation	Weber
within said one or more second processing modules (S), comprising the steps of:	references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this part of the preamble given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[6a] coupling said plurality of processing modules (M, S) by an interconnect means (N) and	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1c]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[6b] enabling a connection based communication having a set of connection properties,	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1c]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

'052 Patent Claim Limitation	Weber
[6c] controlling the communication between said at least one first of said plurality of processing modules (M) and said interconnect means (N) by at least one network interface (NI) associated to said at least one first of said processing modules,	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1d]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[6d] mapping the requested at least one communication service based on said specific communication properties to a connection based on a set of connection properties according to said at least one communication service identification.	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1e]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.

G. Claim 7

'052 Patent Claim Limitation	Weber
[7pre] Data processing system, comprising	To the extent the preamble is limiting, Weber discloses and/or renders obvious the preamble based on at least the following citations. See, e.g.: <i>See supra</i> claim [1pre]. To the extent Weber does not expressly disclose the preamble, a person of ordinary skill in the art would have determined that the preamble is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of the preamble given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[7a] a plurality of processing modules (M, S),	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1a]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[7b] wherein at least one first of said processing modules (M) requests at	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.:

'052 Patent Claim Limitation	Weber
least one communication service to at least one second processing module (S) based on specific communication properties and at least one communication service identification,	<i>See supra</i> claim [1b]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[7c] wherein said at least one communication service identification comprises at least one communication thread or at least one address range, said address range for identifying one or more second processing modules (S) or a memory region within said one or more second processing modules (S),	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1f]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[7d] an interconnect means (N) for coupling said plurality of processing modules (M, S) and for enabling a connection based communication having a set of connection properties,	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1c]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been

'052 Patent Claim Limitation	Weber
	motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[7e] at least one network interface (NI) associated to said at least one first of said processing modules for controlling the communication between said at least one first of said plurality of processing modules (M) and said interconnect means (N), and	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1d]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.
[7f] a mapping means (A) for mapping the requested at least one communication service based on said specific communication properties to a connection based on a set of connection properties according to said at least one communication service identification.	Weber discloses and/or renders obvious this limitation based on at least the following citations. See, e.g.: <i>See supra</i> claim [1e]. To the extent Weber does not expressly disclose this limitation, a person of ordinary skill in the art would have determined that this limitation is either inherent and/or obvious to one of ordinary skill in the art in view of the teachings of Weber. Further, one of ordinary skill in the art would have been motivated to modify Weber or combine it with any of the present prior art references found in Defendant's Invalidity Contentions and any supplements thereto and the relevant section charts for other prior art for the '052 Patent in a manner that would result in the subject matter of this limitation given, at the very least, the person's understanding of the state of the art, the problems addressed and solved in the prior art, and the teachings of Weber.