



(51) International Patent Classification:

G06F 13/42 (2006.01) G06F 11/20 (2006.01)
G06F 11/07 (2006.01) G06F 15/173 (2006.01)

(21) International Application Number:

PCT/US2022/011757

(22) International Filing Date:

10 January 2022 (10.01.2022)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

63/140,434 22 January 2021 (22.01.2021) US

(71) Applicant: **THE REGENTS OF THE UNIVERSITY OF CALIFORNIA** [US/US]; 1111 Franklin Street, 12th Floor, Oakland, California 94607-5200 (US).

(72) Inventors: **DWYER, Daniel**; 631 Adams St., Albany, California 94706 (US). **GRACE, Carl**; 138 Arbor St., San Francisco, California 94134 (US).

(74) Agent: **OVANEZIAN, Daniel E.** et al.; WOMBLE BOND DICKINSON (US) LLP, Po Box 7037, Atlanta, Georgia 30357-0037 (US).

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

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: AD-HOC NETWORK OF READOUT APPLICATION-SPECIFIC INTEGRATED CIRCUITS FOR RELIABLE DETECTOR INSTRUMENTATION

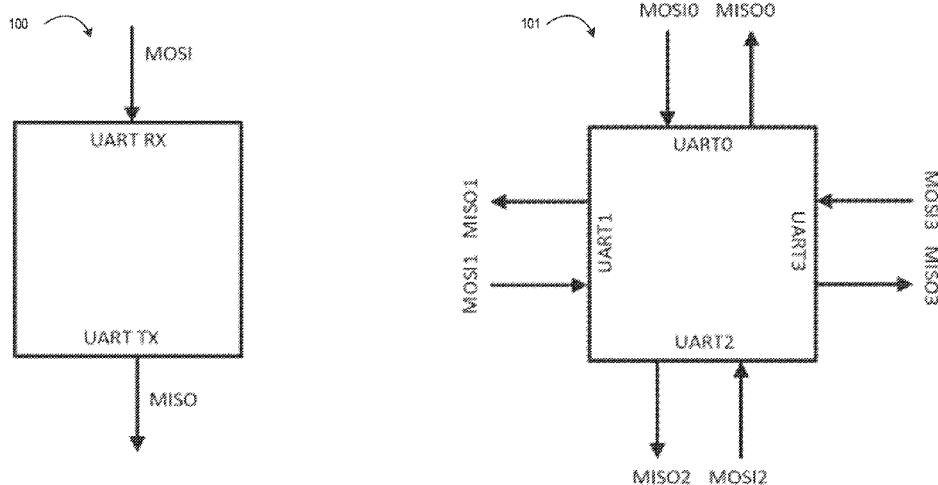


FIGURE 1

(57) Abstract: An application-specific integrated circuit (ASIC) of a computer network comprises three or more universal asynchronous receiver transmitters (UARTs). The ASIC further comprises three or more upstream communication channels, one of the three or more upstream communication channels per UART, respectively. The ASIC further comprises three or more downstream communication channels, one of the three or more downstream communication channels per UART, respectively.



TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

- *with international search report (Art. 21(3))*

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit under 35 U.S.C. § 119(e) of U.S. Provisional Application No. 63/140,434, filed on 22 January 2021, the entire contents of which is hereby incorporated by reference herein

STATEMENT OF GOVERNMENT SUPPORT

[0002] This invention was made with government support under Contract No. DE-AC02-05CH11231 awarded by the U.S. Department of Energy. The government has certain rights in this invention.

FIELD

[0003] The present invention relates to application-specific integrated circuits (ASICs), and more particularly, relates to ad-hoc networks of readout ASICs for reliable detector instrumentation.

BACKGROUND

[0004] An ad-hoc network is a computer network that is spontaneously formed when computing devices connect and communicate with each other. The devices communicate with each other directly instead of relying on a base station or access points, as in wireless local area networks (LANs), for data transfer coordination. Each device participates in routing activity, by determining the route using the routing algorithm and forwarding data to other devices via this route.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] Figure 1 illustrates examples of ASICs embodying the Daisy Chain approach and the Hydra ad-hoc network approach, according to one embodiment.

[0006] Figures 2A-C illustrate an example hydra network, according to one embodiment.

[0007] Figure 3 illustrates an example method of using a hydra network, according to one embodiment.

[0008] Figure 4 is a block diagram of an example computing device that may perform one or more of the operations described herein, in accordance with some embodiments of the present disclosure.

DETAILED DESCRIPTION

[0009] Various international physics collaborations use noble liquid detectors. Cryogenic electronics vendors and equipment manufacturers want to instrument large areas with detectors reliably. Any application where there is a desire to address a large number of individual components robustly and have the capability to work around damage when repair or replacement is problematic could benefit from this invention.

[0010] Many emerging noble liquid particle detectors require large numbers of component chips that need to be addressed individually. Because these components are placed into a sealed cryostat, there is a strong need to communicate with them using a minimum number of wires.

[0011] Historically, one solution was to organize the chips into a daisy chain. However, repair or replacement of components in these noble liquid detectors is problematic, and failure of a single component can cause the entire daisy chain to fail unless complex countermeasures are taken. The embodiments described herein, dubbed Hydra I/O, provide for a software-defined ad-hoc network of readout chips that can be reconfigured on the fly to route around failed components.

Advantageously, Hydra I/O is robust, scalable, and highly reliable and could find application beyond noble liquid detectors to any system that needs a robust, reliable network of large numbers of components that can be communicated with individually using a reduced number of wires.

[0012] The embodiments described herein significantly simplify the creation of large, reliable arrays of detector readout ASICs. The embodiments move much of the complexity from the custom integrated circuit domain, which may be costly and time consuming, to the software domain, which can be addressed by more people. The solution provided herein is more reliable, lower power, and requires simpler hardware than a comparable bus architecture. Hydra I/O offers a superior solution to the problem of reliably addressing large numbers of components through a reduced number of communication channels.

[0013] In summary, the embodiments described herein provide for a new technique to solve the difficult problem of addressing a large number of ASICs through a reduced number of communication channels by creating a robust and scalable ad-hoc network that offers long-term reliability even in the face of failures of individual nodes of the network.

[0014] Figure 1 illustrates examples of ASICs embodying the Daisy Chain approach and the Hydra ad-hoc network approach (100 and 101), according to one embodiment. One technical problem with existing solutions is how to implement pixelated readout of large detectors (such as the DUNE Near Detector, for example) in a reliable way. Because of the large number of chips used, it is important to limit the physical lines of communications (both to reduce the number of cryostat penetrations and to simplify PCB design). In addition, communication with the ASICs must be extremely reliable even in the face of individual ASIC failures. This is important because the cryostats may be sealed for up to 20 years or more, and repair or replacement of failed ASICs may be impossible or at least very difficult. Therefore, it is desirable to communicate with the ASICs in such a way that the network is robust against individual failures. Because the detectors operate at cryogenic temperatures, it is also important that any such solution minimize the number of cryostat penetrations (e.g., to reduce cooling cost).

[0015] The first illustrated ASIC (LArPix_v1, 100) uses a daisy chaining approach to satisfy these requirements. Each ASIC in a network is connected to its two nearest neighbors in one dimension and all the ASICs may be organized in a series of columns. All network data may be forwarded to the next upstream ASIC until it eventually exits the cryostat (or other system).

[0016] As is, this approach is straightforward and easy to implement but may not be not scalable, since a single ASIC failure may make it impossible to communicate with all ASICs upstream of the failed ASIC on the daisy chain. This may be particularly problematic because of the challenge of not being able to repair or replace broken modules in an enclosed system or network.

[0017] One solution to this issue is a bus architecture. A bus is defined here as a subsystem that allows components to communicate with each other using a set of shared wires. The use of a bus may be problematic in the context described herein for a variety of reasons. First, in a bus architecture a single bad actor could bring down the entire bus without complex countermeasures. Second, the bus may need a complex arbitration scheme as various components could try to control the bus at random times.

[0018] One goal of the embodiments described herein is to provide for a more robust, scalable organizational scheme for the ASICs that retains the benefits of the simple daisy chain, while addressing its drawbacks and also avoiding some of the disadvantages of a bus architecture.

[0019] The solution provided herein is referred to as Hydra I/O (or, simply “Hydra”). Advantageously, Hydra allows an ad-hoc, software-defined network of ASICs that can be reconfigured on the fly (e.g., dynamically, in real time, etc.) to route around failed ASICs. In short, each ASIC is connected to its nearest neighbors on all four sides (such physical layout and geometry is only provided as an example, and not required), and which side is the dominant direction to shift data toward the outside is configurable. Therefore, the organization is not constrained to be daisy chained and networks of arbitrary complexity can be constructed in an ad hoc manner. This means that single failed ASICs can be easily routed around (which may be problematic from an implementation standpoint in the daisy chain approach).

[0020] The implementation of Hydra I/O is as simple as possible from the ASIC perspective, and puts as much of the functionality required to define a network of ASICs in software as possible. In short, the ASICs themselves know nothing about Hydra I/O and will only be able to power up and down Universal Asynchronous Receiver Transmitter units (UARTs) and respond to commands that match its Chip identifier (ID) (e.g., each ASIC has a unique chip ID). It should be noted that although UARTs are referred to herein for convenience and brevity, any number of alternative digital communications modalities may be used.

[0021] The daisy chain approach and the ad-hoc Hydra approach are contrasted in Figure 1 (see 100 in contrast to 101). The daisy chain version of the ASIC 100 has a single UART per chip and

uses a daisy chaining approach to communicate with multiple ASICs. In the daisy chain, each ASIC (e.g., 100) is connected to its two nearest neighbors in one dimension and the ASICs are organized in a series of columns. The network data is forwarded to the next upstream ASIC until it finally exits the system (e.g., a cryostat). This approach is simple and easy to implement but may not be scalable since a single ASIC failure may make it impossible to communicate with all ASICs upstream of the failed ASIC on the daisy chain.

[0022] To circumvent the reliability problems of a daisy chain approach to ASIC 100 interconnection, a new architecture called Hydra is employed. As seen at 101 of Figure 1, an example ASIC 101 employing Hydra may have four on-board UARTs and each ASIC is connected to its nearest neighbors on all four sides. In one embodiment, which side is the dominant direction to shift data toward the outside is configurable. Therefore, the organization is not constrained to be daisy chained and two-dimensional networks of arbitrary complexity can be constructed in an ad hoc way. This means that a single failed ASIC (e.g., 101) can be routed around (which may be problematic from an implementation standpoint in the daisy chain approach). It should be noted that four UARTs, channels, etc. is used herein for convenience, any number of three or more respective components is contemplated herein.

[0023] In the embodiments described herein, there are downstream and upstream communication channels (e.g., “channels,” for short). Downstream channels may be used to send configuration and other commands through the network. Upstream channels may be used to return requested configuration data and any other suitable data to the controller. In one embodiment, downstream and upstream channels may use the same master output, slave input (MOSI) ports and master input, slave output (MISO) ports, so the difference may be logical rather than physical. In other embodiments, physical ports may be used. Additionally, downstream and/or upstream packets may be marked as such (e.g., via a flag), respectively, to differentiate them from upstream packets.

[0024] In a variety of embodiments, ASIC 101 may include additional optional features. For example, in one implementation of the embodiments described herein, all ASICs in a Hydra network may be synchronously clocked using a single clock source. Problematically, such a configuration represents an additional single point of failure and requires significant power dissipation to distribute the clock to all the individual ASICs in a Hydra network. An alternative is for each Hydra to be self-clocked using an internal oscillator. Advantageously, such a configuration would have the benefit of significantly reduced power dissipation and reduced risk of an entire Hydra network being brought down due to a fault on the clock network.

[0025] In one embodiment, such a configuration may bring the additional challenge that with internal oscillators the various ASICs in a Hydra network may not be mutually synchronized. This could be addressed in principle using the Hydra network itself. The network could be configured

without concern of slightly different clocking between ASICs (this is because the communication channel is asynchronous). Once the network is established, the network may be traversed with each ASIC synchronizing itself to the ASICs to which it is connected. Once the entire network was traversed, all ASICs would be mutually synchronized.

[0026] In another embodiment, of Hydra-IO, faults due to a specific ASIC in the network that interrupt the flow of data or configuration commands can be avoided through re-routing the communication pathways around the faulty chips. However, there may be classes of faults for which this would not be sufficient. For example, if a chip went into a mode causing it to pollute the network with spurious data, it may in some situations be difficult or impossible to disable it through its configuration interface.

[0027] In another example, if a chip has a fault causing it to draw too much current (e.g., due to damaged devices or short circuit) it may be difficult to address that ASIC through configuration commands. To solve these issues and others, Hydra may be extended to give each network node the capability to execute a hard shutdown of any ASIC to which it is connected (either directly or indirectly). The traditional Hydra functionality may then be exercised to reroute configuration commands and data around the newly shut down ASIC. Advantageously, such a configuration would expand the class of faults to which Hydra could respond and still remain resilient.

[0028] Figures 2A-C illustrate an example hydra network, according to one embodiment. In one embodiment, the upstream Hydra network in Figure 2A is used to configure the various ASICs in the network. Each chip is given a unique Chip ID and MOSI ports are enabled to make an efficient topology to configure a large number of ASICs. The corresponding downstream network is shown in Figure 2B. In the case of an ASIC failure, the Hydra network can be reconfigured to route around the damage. This is shown in Figure 2C. The red ASIC has failed and the red, dotted connections are no longer operable. Without Hydra's ability to reconfigure, not only would the red chip be lost, but all upstream ASICs would be out of communication as well. Using Hydra, however, the green connection can be configured by enabling a different MOSI port on an ASIC downstream of the failed ASIC, and all of the chips in the network are once again reachable.

[0029] Figure 3 illustrates an example method 300 of using a hydra network, according to one embodiment. The method 300 may be performed by processing logic that comprises hardware (e.g., circuitry, dedicated logic, programmable logic, microcode, etc.), software (e.g., instructions run on a processing device to perform hardware simulation), or a combination thereof. In one embodiment, components of any of Figures 1 and 2A-C may perform one or more of the following operations. In another embodiment, any other suitable processing device may perform the described operations.

[0030] Referring to Figure 3, at 302, processing logic may receive, from a computer network comprising a plurality of application-specific integrated circuits (ASICs), an indication that a first

ASIC of the plurality of ASICs is expressing an abnormality. In one embodiment, such an abnormality may be a complete or partial failure of any aspect of the affective ASIC. In response to receiving the indication, processing logic may reprogram (e.g., by a processing device) a second ASIC of the plurality of ASICs to bypass the first ASIC, wherein the second ASIC is downstream from the first ASIC.

[0031] In one embodiment, to reprogram the second ASIC, processing logic may deactivate a first logical master output, slave input (MOSI) port and logical master input, slave output (MISO) port pair of the second ASIC and activate a second logical MOSI port and logical MISO port pair of the second ASIC.

[0032] Figure 4 is a block diagram of an example computing device that may perform one or more of the operations described herein, in accordance with some embodiments of the present disclosure.

[0033] FIG. 4 is a block diagram of an example computing device 400 that may perform one or more of the operations described herein, in accordance with some embodiments of the present disclosure. Computing device 400 may be connected to other computing devices in a LAN, an intranet, an extranet, and/or the Internet. The computing device may operate in the capacity of a server machine in client-server network environment or in the capacity of a client in a peer-to-peer network environment. The computing device may be provided by a personal computer (PC), a set-top box (STB), a server, a network router, switch or bridge, or any machine capable of executing a set of instructions (sequential or otherwise) that specify actions to be taken by that machine. Further, while only a single computing device is illustrated, the term “computing device” shall also be taken to include any collection of computing devices that individually or jointly execute a set (or multiple sets) of instructions to perform the methods discussed herein.

[0034] The example computing device 400 may include a processing device (e.g., a general purpose processor, a PLD, etc.) 402, a main memory 404 (e.g., synchronous dynamic random access memory (DRAM), read-only memory (ROM)), a static memory 406 (e.g., flash memory and a data storage device 418), which may communicate with each other via a bus 430.

[0035] Processing device 402 may be provided by one or more general-purpose processing devices such as a microprocessor, central processing unit, or the like. In an illustrative example, processing device 402 may comprise a complex instruction set computing (CISC) microprocessor, reduced instruction set computing (RISC) microprocessor, very long instruction word (VLIW) microprocessor, or a processor implementing other instruction sets or processors implementing a combination of instruction sets. Processing device 402 may also comprise one or more special-purpose processing devices such as an application specific integrated circuit (ASIC), a field programmable gate array (FPGA), a digital signal processor (DSP), network processor, or the like.

The processing device 402 may be configured to execute the operations described herein, in accordance with one or more aspects of the present disclosure, for performing the operations and steps discussed herein. In one embodiment, processing device 402 represents processing device 120 of FIG. 1A. In another embodiment, processing device 402 represents a processing device of a client device (e.g., client device 150 of FIG. 1A).

[0036] Computing device 400 may further include a network interface device 408 which may communicate with a network 420. The computing device 400 also may include a video display unit 410 (e.g., a liquid crystal display (LCD) or a cathode ray tube (CRT)), an alphanumeric input device 412 (e.g., a keyboard), a cursor control device 414 (e.g., a mouse) and an acoustic signal generation device 416 (e.g., a speaker). In one embodiment, video display unit 410, alphanumeric input device 412, and cursor control device 414 may be combined into a single component or device (e.g., an LCD touch screen).

[0037] Data storage device 418 may include a computer-readable storage medium 428 on which may be stored one or more sets of instructions, e.g., instructions for carrying out the operations described herein, in accordance with one or more aspects of the present disclosure. Instructions implementing ASIC service 427 may also reside, completely or at least partially, within main memory 404 and/or within processing device 402 during execution thereof by computing device 400, main memory 404 and processing device 402 also constituting computer-readable media. The instructions may further be transmitted or received over a network 420 via network interface device 408.

[0038] While computer-readable storage medium 428 is shown in an illustrative example to be a single medium, the term “computer-readable storage medium” should be taken to include a single medium or multiple media (e.g., a centralized or distributed database and/or associated caches and servers) that store the one or more sets of instructions. The term “computer-readable storage medium” shall also be taken to include any medium that is capable of storing, encoding or carrying a set of instructions for execution by the machine and that cause the machine to perform the methods described herein. The term “computer-readable storage medium” shall accordingly be taken to include, but not be limited to, solid-state memories, optical media and magnetic media.

CLAIMS

What is claimed is:

1. An application-specific integrated circuit (ASIC) of a computer network, comprising:
three or more universal asynchronous receiver transmitters (UARTs);
three or more upstream communication channels, one of the three or more upstream communication channels per UART, respectively;
three or more downstream communication channels, one of the three or more downstream communication channels per UART, respectively.
2. The ASIC of claim 1, wherein the three or more upstream channels are to send configuration commands through the computer network.
3. The ASIC of claim 1, wherein the three or more downstream channels are to return requested configuration data to a network controller of the computer network.
4. The ASIC of claim 1, further comprising three or more logical master output, slave input (MOSI) ports and three or more logical master input, slave output (MISO) ports, wherein each pair of the three or more downstream channels and the three or more upstream channels use a different logical MOSI port and MISO port, respectively.
5. The ASIC of claim 1, wherein each of the three or more MOSI ports and the three or more MISO ports is configurable to be enabled or disabled.
6. The ASIC of claim 1, wherein downstream packets sent over the three or more downstream communication channels are flagged as such to differentiate them from upstream packets sent over the three or more upstream communication channels.
7. The ASIC of claim 1, wherein the ASIC is synchronously clocked using a single clock source.
8. The ASIC of claim 1, wherein the ASIC is self-clocked using an internal oscillator.
9. The ASIC of claim 1, wherein the ASIC is capable to execute a hard shutdown of any other ASIC to which it is connected.
10. A computer network, comprising:
a plurality of application-specific integrated circuits (ASICs), wherein each of the plurality of ASICs comprises:
four universal asynchronous receiver transmitters (UARTs);
four upstream communication channels, one of the four upstream communication channels per UART, respectively;
four downstream communication channels, one of the four downstream communication channels per UART, respectively.
11. The computer network of claim 10, wherein the four upstream channels are to send

configuration commands through the computer network.

12. The computer network of claim 10, wherein the four downstream channels are to return requested configuration data to a network controller of the computer network.

13. The computer network of claim 10, further comprising four logical master output, slave input (MOSI) ports and four logical master input, slave output (MISO) ports, wherein each pair of the four downstream channels and the four upstream channels use a different logical MOSI port and MISO port, respectively.

14. The computer network of claim 10, wherein each of the four MOSI ports and the four MISO ports is configurable to be enabled or disabled.

15. The computer network of claim 10, wherein downstream packets sent over the four downstream communication channels are flagged as such to differentiate them from upstream packets sent over the four upstream communication channels.

16. The computer network of claim 10, wherein the plurality of ASICs is synchronously clocked using a single clock source.

17. The computer network of claim 10, wherein each of the plurality of ASICs is self-clocked using an internal oscillator.

18. The computer network of claim 10, wherein each of the plurality of ASICs is capable to execute a hard shutdown of any other of the plurality of ASICs to which it is connected.

19. A method, comprising:

receiving, from a computer network comprising a plurality of application-specific integrated circuits (ASICs), an indication that a first ASIC of the plurality of ASICs is expressing an abnormality; and

in response to receiving the indication, reprogramming, by a processing device, a second ASIC of the plurality of ASICs to bypass the first ASIC, wherein the second ASIC is downstream from the first ASIC.

20. The method of claim 19, wherein to reprogram the second ASIC the method comprises: deactivating a first logical master output, slave input (MOSI) port and logical master input, slave output (MISO) port pair of the second ASIC and activating a second logical MOSI port and logical MISO port pair of the second ASIC.

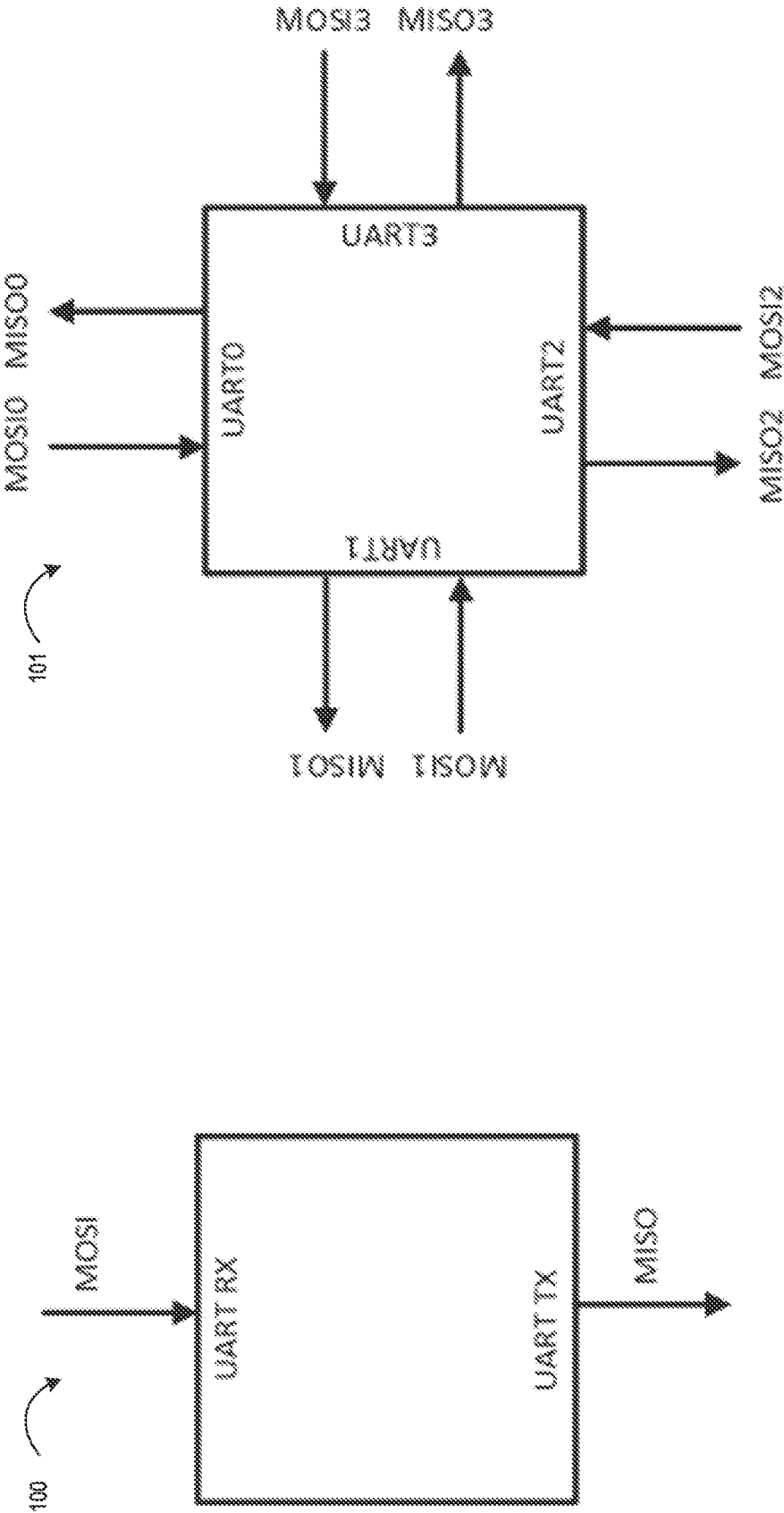
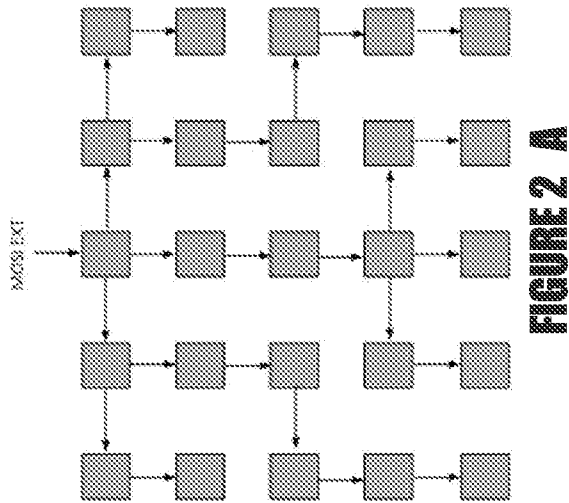
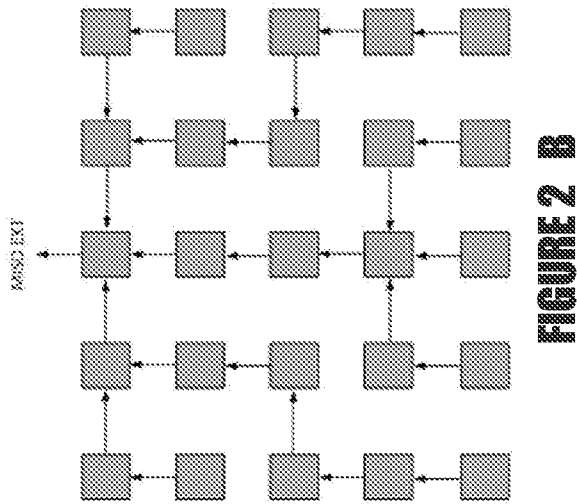
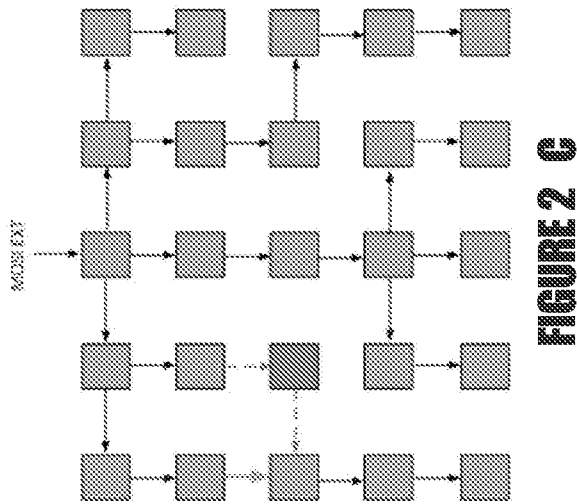
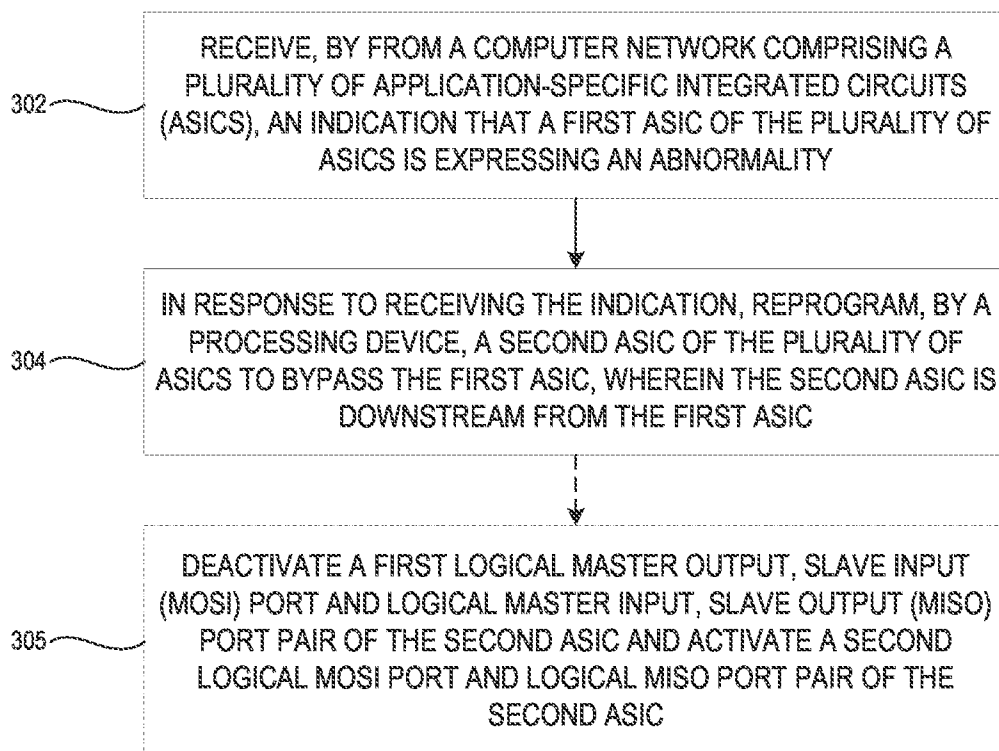
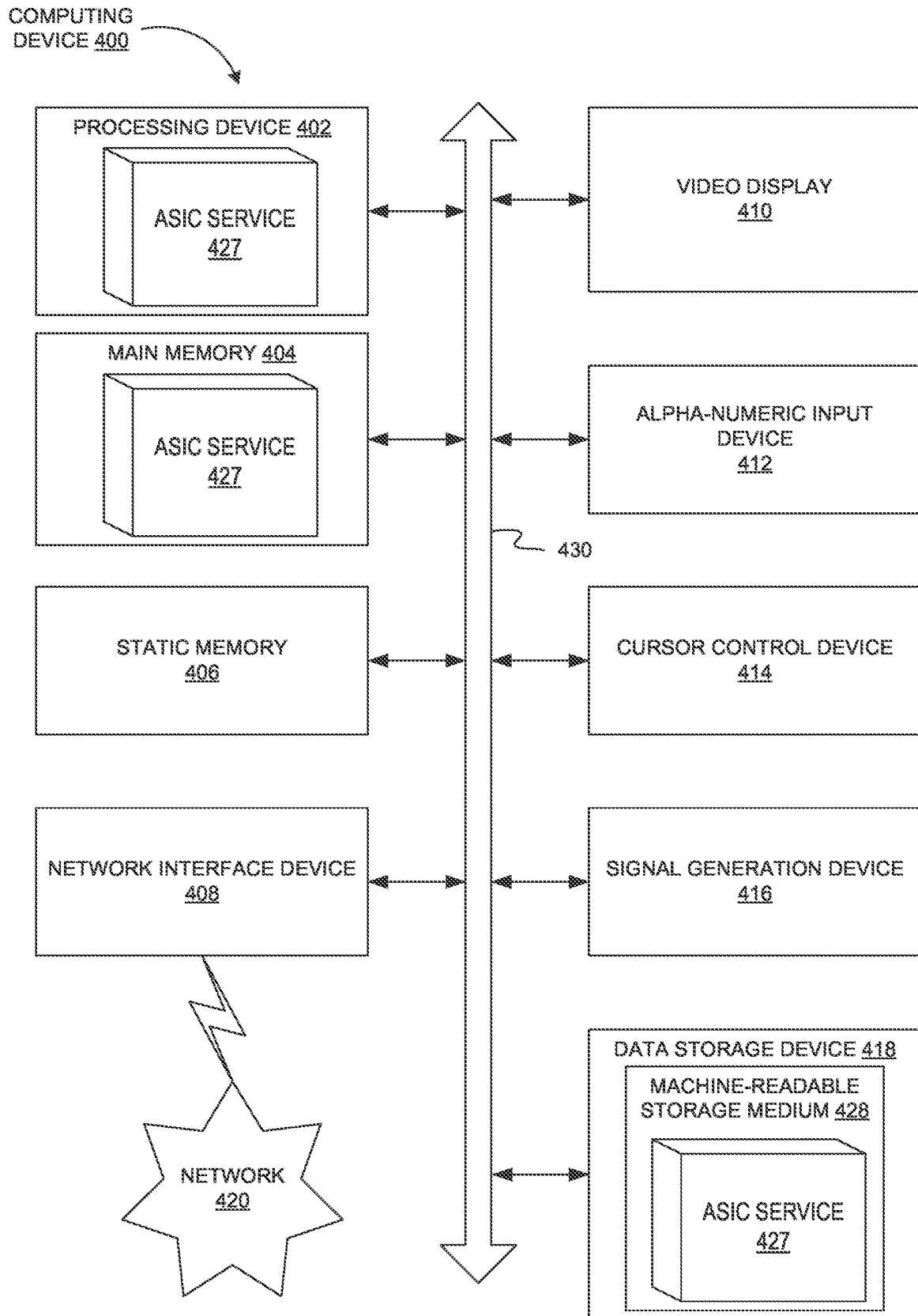


FIGURE 1



300

**FIGURE 3**

**FIGURE 4**

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2022/011757

A. CLASSIFICATION OF SUBJECT MATTER INV. G06F13/42 G06F11/07 G06F11/20 G06F15/173 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Dwyer Dan: "LArPix: Pixelated Charge Readout for Large LArTPCs", , 13 November 2019 (2019-11-13), pages 1-21, XP055906900, Retrieved from the Internet: URL:https://indico.fnal.gov/event/21535/contributions/63304/attachments/39680/48018/DDwyer_LArPix_MOOD_13Nov2019.pdf [retrieved on 2022-03-30] pages 17-19 ----- -/--	1-20
☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.		
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance;; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance;; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search		Date of mailing of the international search report
30 March 2022		12/04/2022
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Albert, Jozsef

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2022/011757

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Russell Brooke: "LArPix Analysis Tutorial on behalf of LArPix @ LBNL ND-LAr Analysis Meeting", , 22 October 2020 (2020-10-22), XP055906867, Retrieved from the Internet: URL:https://indico.fnal.gov/event/46138/contributions/200624/attachments/136498/169759/LArPix_Analysis_Tutorial__Oct21st2020.pdf [retrieved on 2022-03-30] page 9 -----	1-20
A	Dwyer Dan: "Dan Dwyer (LBNL) ArgonCube Collaboration Meeting LArPix-v2: Pixelated Charge Readout for the ArgonCube 2x2 Demonstrator", , 9 December 2019 (2019-12-09), XP055906875, Retrieved from the Internet: URL:https://indico.fnal.gov/event/20432/contributions/57611/attachments/36079/43952/DDwyer_LArPix_ACCollab_09Dec2019.pdf [retrieved on 2022-03-30] pages 6-8 -----	1-20