



- (51) **International Patent Classification:**
G06F 11/14 (2006.01) *G06F 11/20* (2006.01)
- (21) **International Application Number:**
PCT/IB2021/056772
- (22) **International Filing Date:**
26 July 2021 (26.07.2021)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant:** MOLEX, LLC [US/US]; 2222 Wellington Court, Lisle, Illinois 60532 (US).
- (72) **Inventor:** SCOTT, Eric; c/o Molex, LLC, 2222 Wellington Court, Lisle, Illinois 60532 (US).

- (74) **Agent:** SHELDON,, Stephen L.; c/o Molex, LLC, 2222 Wellington Court, Lisle, Illinois 60532 (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,

(54) **Title:** FAULT-TOLERANT DISTRIBUTED COMPUTING FOR VEHICULAR SYSTEMS

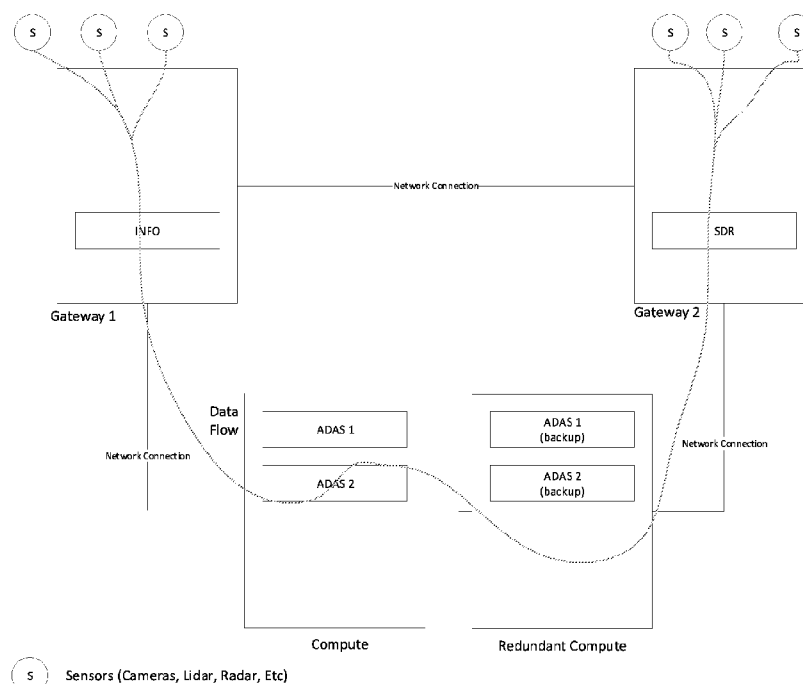


FIG. 17

(57) **Abstract:** Exemplary embodiments are disclosed of systems and methods of providing fault-tolerant distributed computing for vehicular systems. In an exemplary embodiment, a system for providing fault-tolerant distributed computing for a vehicular system includes a first processing unit and a second processing unit communicatively coupled to the first processing unit. The first processing unit is configured to support a first set of processes. The second processing unit is configured to support a second set of processes, to monitor the state of the first set of processes, and to support at least one additional process of the first set of processes when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit. The second processing unit is configured to support the additional process substantially instantaneously.

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

Published:

- *with international search report (Art. 21(3))*
- *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

FAULT-TOLERANT DISTRIBUTED COMPUTING FOR VEHICULAR SYSTEMS

TECHNICAL FIELD

[0001] This disclosure relates to fault-tolerant distributed computing for vehicular systems.

DESCRIPTION OF RELATED ART

[0002] Fault tolerance is a known system requirement for certain types of computing systems. In distributed networks, it is possible to have multiple instances running in parallel and to add additional instances as needed, both to address failures and to address the need for additional computing resources.

[0003] Systems within autonomous or semi-autonomous vehicles need to continue to operate, at least to some degree, despite failures or situations where the vehicle suffers damage. The mobile network which interconnects the various systems within the vehicle is one of these key systems. Thus, there is a need to provide a substantial degree of fault-tolerance for the applications running on the nodes in vehicular distributed computing systems.

SUMMARY

[0004] This section provides a general summary of the disclosure, and is not a comprehensive disclosure of its full scope or all of its features.

[0005] Exemplary embodiments are disclosed of systems and methods of providing fault-tolerant distributed computing for vehicular systems. In an exemplary embodiment, a system for providing fault-tolerant distributed computing for a vehicular system includes a first processing unit and a second processing unit communicatively coupled to the first processing unit. The first processing unit is configured to support a first set of processes. The second processing unit is configured to support a second set of processes, to monitor the state of the first set of processes, and to support at least one additional process of the first set of processes when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit. The second processing unit is configured to support the additional process substantially instantaneously.

[0006] Further areas of applicability will become apparent from the description provided herein. The description and specific examples in this summary are intended for purposes of illustration only and are not intended to limit the scope of the present disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The present application is illustrated by way of example and not limited in the accompanying figures in which like reference numerals may indicate similar elements and in which:

[0008] FIG. 1 illustrates an exemplary system including a central computer (gateway, switch, power distribution), zonal controls (gateway, switch, power distribution), and an autonomous computer for which may be implemented an exemplary embodiment of system/method for providing fault-tolerant networking and/or enable failover operation in a vehicle.

[0009] FIG. 2 is a block diagram of a distributed computing system including four gateways GW1, GW2, GW3, and GW4 according to an exemplary embodiment of the present disclosure.

[0010] FIG. 3 is another block diagram of the distributed computing system shown in FIG. 2 after the fourth gateway GW4 has failed in its entirety as indicated by the X through the fourth gateway GW4.

[0011] FIG. 4 is a block diagram of a distributed computing system including three gateways that are connected to each other via high speed digital links and that include a redundant connection to a controller area network (CAN) device according to an exemplary embodiment of the present disclosure.

[0012] FIG. 5 is another block diagram of the distributed computing system shown in FIG. 4 after a task has failed on one of the gateways and another gateway has taken action in response to the other gateway's failed task according to exemplary embodiments of the present disclosure.

[0013] FIG. 6 is a table that includes example actions (*e.g.*, Start/Stop a software task or HW process, load or unload programmable logic, *etc.*) that may be taken by a Safety Supervisor Manager of a gateway in the event of a failure of a process or an entire gateway according to exemplary embodiments of the present disclosure.

[0014] FIG. 7 is a table that includes examples of descriptions of different states shown in FIG. 6.

[0015] FIG. 8 illustrates an example of low latency failover according to exemplary embodiments of the present disclosure.

[0016] FIG. 9 is a flow chart illustrating an example operation of a fault-tolerant system according to exemplary embodiments of the present disclosure.

[0017] FIG. 10 is a flow chart illustrating an example method of dynamically reconfiguring Action tables according to exemplary embodiments of the present disclosure.

[0018] FIG. 11 is a block diagram of an exemplary distributed computing system including two zonal gateway nodes (*e.g.*, electronic control units (ECUs), *etc.*) connected to each other and to a computer or data server node via network connections (*e.g.*, high speed digital links, *etc.*) according to an exemplary embodiment of the present disclosure.

[0019] FIG. 12 is another block diagram of the distributed computing system shown in FIG. 11 after the data server node has failed and the software tasks ADAS 1 and ADAS 2 have been moved from on the data server node to the two zonal gateway modes Gateway 1 and Gateway 2, respectively.

[0020] FIG. 13 is a block diagram of an exemplary distributed computing system including a computer or data server node connected via network connections (*e.g.*, high speed digital links, *etc.*) to two zonal gateway nodes (*e.g.*, electronic control units (ECUs), *etc.*) and to an active remote antenna (*e.g.*, 5G mmWave antenna, *etc.*) according to an exemplary embodiment of the present disclosure.

[0021] FIG. 14 is another block diagram of the distributed computing system shown in FIG. 13 after the second zonal gateway node has failed and the remote antenna hardware has taken over the function of the mmWave radar of the failed second zonal gateway node.

[0022] FIG. 15 is a block diagram of an exemplary distributed computing system including two controllers or zonal gateway nodes that are connected to each other and to a computer or data server node via network connections (*e.g.*, high speed digital links, *etc.*) and that include redundant connections (*e.g.*, CAN links, *etc.*) to power steering according to an exemplary embodiment of the present disclosure.

[0023] FIG. 16 is another block diagram of the distributed computing system shown in FIG. 15 after the steering control task on the first controller has failed and the control over the power steering unit's software control has been failed over to the second controller.

[0024] FIG. 17 is a block diagram of a distributed computing system including a redundant ADAS compute or data server node.

DETAILED DESCRIPTION

[0025] The detailed description that follows describes exemplary embodiments and the features disclosed are not intended to be limited to the expressly disclosed combination(s). Therefore, unless otherwise noted, features disclosed herein may be combined to form additional combinations that were not otherwise shown for purposes of brevity.

[0026] Systems within autonomous or semi-autonomous vehicles need to continue to operate, at least to some degree, despite failures or situations where the vehicle suffers damage. After recognizing the need to provide a substantial degree of fault-tolerance for the applications running on the nodes in vehicular distributed computing systems, exemplary embodiments of systems and methods were developed and/or are disclosed herein that provide fault-tolerant distributed computing and failover operation for vehicular systems, *e.g.*, when a vehicular system component or subcomponent fails, *etc.*

[0027] In exemplary embodiments, a system for providing fault-tolerant distributed computing for a vehicular system includes a plurality of distributed processing units (*e.g.*, gateways, electronic control units (ECUs), electronic control modules (ECMs), telematics control units (TCUs), *etc.*) that are configured to pass relevant information to each other, such as sharing their states with each other. The processing units are configured such that actions of a failed processing unit may be taken over by one or more of the other remaining processing units. For example, tasks, processes, code, *etc.* may be distributed across multiple processing units such that when there is a failure in and/or loss of communication with one of the processing units, the remaining processing unit(s) are operable for assuming and/or taking over critical tasks of that failed processing unit.

[0028] FIG. 1 illustrates an exemplary system including a central computer (gateway, switch, power distribution), zonal controls (gateway, switch, power distribution), and an autonomous computer for which may be implemented an exemplary embodiment of a system/method for providing fault-tolerant distributed computing and enable failover operation for a vehicular system. Other components can also be considered but have been left out for clarify and simplification. Nodes disclosed herein may be considered and/or include gateways, electronic control units (ECUs), electronic control modules (ECMs), telematics control units (TCUs), *etc.*

[0029] FIG. 2 illustrates an exemplary distributed computing system including four gateways GW1, GW2, GW3, and GW4 (*e.g.*, electronic control units (ECUs), electronic control modules (ECMs), telematics control units (TCUs), *etc.*). Sensors and actuators are connected with the gateways. During normal operation in a none failed operational state as shown in FIG. 2, the field programmable gate array (FPGA) function x in the first gateway GW1 is disabled. Accordingly, the FPGA Function x in the first gateway GW1 is not used during normal operation, while all other tasks in the system are running normally. Safety Supervisor passes relative task state information between gateways.

[0030] As shown in FIG. 2, each Safety Supervisor is able to share status information with each other Safety Supervisor as indicated by the arrows. Accordingly, FIG. 2 generally shows a broadcast situation in which each node obtains the status information for each other node.

[0031] FIG. 3 illustrates the distributed computing system shown in FIG. 2 after the fourth gateway GW4 has failed in its entirety as indicated by the X through the fourth gateway GW4. As indicated by the arrows, the first, second, and third gateways GW1, GW2, and GW3 have taken over the operations of the fourth gateway GW4 according to exemplary embodiments of the present disclosure. In FIG. 3, non-functioning components are denoted by the Xs through the sensors, and disabled tasks are denoted by Xs through the Infotasks in the first, second, and third gateways GW1, GW2, and GW3 and the X through the FPGA Function y in the first gateway GW1.

[0032] As shown in FIG. 3, the FPGA Function y is moved from the fourth gateway GW4 to the first gateway GW1 to the FPGA function x in the first gateway GW1, which was previously disabled during normal operation when none of the gateways had failed as shown in FIG. 2. With continued reference to FIG. 3, the ADAS (advanced driver-assistance system) Functions x are moved from the fourth gateway GW4 to the second and third gateways GW2 and GW3. FPGA Function y is disabled (as indicated by the X) in the first gateway GW1 to accommodate and/or make room for FPGA Function x from the fourth gateway GW4. Infotainment tasks in the first, second, and third gateways GW1, GW2, and GW3, are disabled (as indicated by the Xs) or shut down due to resource loading. These actions can be done via a static configuration table and/or a dynamic algorithm deciding what tasks to move based on current load.

[0033] Accordingly, the tasks that are critical have been redistributed from the failed fourth gateway GW4. And, tasks that are not critical have been disabled or shutdown in the first, second,

and third gateways GW1, GW2, and GW3 to provide, make available, or free up resources for handling the critical tasks redistributed from the fourth gateway GW4.

[0034] In FIG. 3, the four boxes labeled Safety Supervisor represent supervisor tasks that monitor in real time the health and status of the run-time tasks in the other gateways. Each Safety Supervisor is configured to be operable to determine whether or not the run-time tasks are operating properly, whether or not the run-time tasks have malfunctioned, and/or whether or not a malfunction has occurred with the central processing unit (CPU) on which the run-time tasks should be or were running.

[0035] In exemplary embodiments, each critical task is treated independently. Each critical task has a calculated or includes a target failover gateway for actions to be taken.

[0036] History/context of the tasks may or may not be required. For example, sensor data parsing/calculations may not be needed going forward in time, but the state of an output may still be shared through Safety Supervisor or other method (*e.g.*, shared data model(s), *etc.*) with the failover task.

[0037] FIG. 4 illustrates a distributed computing system in which three gateways (*e.g.*, electronic control units (ECUs), electronic control modules (ECMs), telematics control units (TCUs), *etc.*) are connected via 10G Ethernet links (broadly, high speed digital links) according to an exemplary embodiment of the present disclosure. The three gateways include a redundant connection to a controller area network (CAN) device. The CAN device is operable as an actuator and/or a sensor. Each Safety Supervisor Manager is configured to transfer a Heartbeat (*e.g.*, a packet of data sent on a regular basis, *etc.*) plus information on the state of the current tasks. In turn, the receiving gateway monitors this information to decide what actions to take in the event a task(s) fails or a heartbeat fails indicating a loss of communication.

[0038] FIG. 5 illustrates the distributed computing system shown in FIG. 4 after a task has failed on one of the gateways (as indicated by the X through the ADAS Function x) and another gateway has taken action in response to the other gateway's failed task according to exemplary embodiments of the present disclosure. As shown in FIG. 5, a secondary function (ADAS Function x) is started up (as indicated by the arrow) and another function (Info Task) is disabled (as indicated by the X) to free up resources.

[0039] In the exemplary embodiment shown in FIGS. 4 and 5, the links between the gateways are illustrated as 10G Ethernet links. The 10G Ethernet links are relatively high bandwidth digital

links that allows for a low latency changeover (*e.g.*, substantially instantaneously or within a response time of a vehicle safety system, *etc.*) in the event a task(s) fails or a heartbeat fails indicating a loss of communication. In other embodiments, the links between the gateways may comprise other suitable high speed digital links that allow for low latency changeover in the event a task(s) fails or a heartbeat fails indicating a loss of communication.

[0040] In exemplary embodiments, a Safety Supervisor Manager within each gateway is operable for monitoring in real time and controlling the state of the gateways within the gateway cluster. More specifically, each Safety Supervisor Manager independently monitors in real time the state of its peers/other gateways in the cluster and takes action based on this information. Each Safety Supervisor Manager in the cluster keeps track of all deemed relevant processes within its own respective CPU or Hardware (HW) processing unit and reports this to the other Safety Supervisor Managers within the cluster. Each Safety Supervisor Manager in the cluster also monitors in real time the state of all deemed relevant processes within the CPUs or HW processing units of its peers/other gateways in the cluster.

[0041] In the event of a failure of a process or a failure of an entire gateway, a Safety Supervisor Manager may perform actions quickly (*e.g.*, substantially instantaneously or within a response time of a vehicle safety system, *etc.*) according to a predefined state or based on an algorithm for optimizing resource usage. Examples of actions may include Start/Stop a software task or HW process, load or unload programmable logic (*e.g.*, FPGA logic, *etc.*), *etc.* Failovers start and stop may be based on system time (*e.g.*, microsecond resolution, millisecond resolution, *etc.*). For example, system time accuracy / resolution may be based on 10 ns, with a period or cycle time in millisecond resolution.

[0042] FIG. 6 is an Actions table that includes example actions (*e.g.*, Start/Stop a software task or HW process, load or unload programmable logic, *etc.*) that may be taken by a Safety Supervisor Manager of a gateway in the event of a failure of a process or a failure of an entire gateway. The actions may be performed according to a predefined state and/or based on an algorithm for optimizing resource usage in accordance with exemplary embodiments of the present disclosure. FIG. 6 is an example only as the table may be modified to accommodate system needs.

[0043] Every process in the table (*e.g.*, KeyManager, Adaptive Cruise, ADAS Function 1, ADAS Function 2, ADAS Function 2, Media Player) may be treated independently such that if a process fails in a gateway then all processes in that gateway may be considered failed. Entry 5

(ADAS Function 3) in the table may be considered a special case where the process fails but is recoverable, and the system allows for recovery after performing self-checks and the secondary is shut down.

[0044] FIG. 7 is a table that includes examples of descriptions of different states shown in FIG. 6. But the different state descriptions provided in FIG. 7 are example only, and the present disclosure is not limited to these descriptions.

[0045] In exemplary embodiments, the communication scheme for the Safety Supervisor Managers is independent of failover operations. Redundancy and safety requirements are taken into account when selecting communications methods.

[0046] In exemplary embodiments, messages may be sent via several communication protocols and topologies, such as Broadcast, LLDP (Link Layer Discover Protocol) Peer to Peer (data passed on from node to node), Multicast (all Supervisors listen to each other's multicast traffic), Internet Protocol, Shared memory based Inter Process Communication schemes, *etc.* If TSN Qbv is supported in an exemplary embodiment, then messages may utilize this mechanism via control frame QOS tag.

[0047] In exemplary embodiments, message content may be the same for each type, *e.g.*, a protocol data unit (PDU) containing the local Gateway's state. Each PDU may be encrypted (*e.g.*, ECC HMAC (elliptical curve cryptography hash-based message authentication code), *etc.*), for example, to provide assurance the data has not been tampered with and is correct according to a safety requirement, prevent replay attacks, check with integrated circuit base unit (IC BU) on safely transferring data frames schemes, *etc.* Each PDU may have a system Time Stamp with the data. Packet format (sent via Multicast UDP message, on port X) may be:

```
{ "PROCESS_STATE": [
  { "DEVICE_NAME": "GW_FR_LEFT", "Entity Name": "OBD_Server_x", "STATE":
    "Running", "ERROR", "0"},
  { "DEVICE_NAME": "GW_FR_LEFT", "Entity Name": "Router_x", "STATE": "Failed",
    "ERROR", "-10"},
]}
```

[0048] Safety action should not take more than the time between Action start and prep info next. If so, then the state should be clearly identified in the state table and an in-progress value should be sent at the next cycle.

[0049] In exemplary embodiments, each gateway (*e.g.*, electronic control units (ECUs), electronic control modules (ECMs), telematics control units (TCUs), *etc.*) is configured with accurate timing such that actions are taken in sync with each other. For example, Ethernet 802.1AS and Time aware shapers may be used in exemplary embodiments. But alternative embodiments may include another mechanism of synchronizing as the need for synchronization may depend on system requirements. For example, other exemplary embodiments may include precision time protocol (PTP) and time-sensitive networking (TSN) to help keep the time more aligned between gateways. System cycle time is determined by the applications. Information from a process on a given gateway is prepared before the end of the system cycle so that it can be transferred to each other gateway during the control phase. Actions in the processes in the affected gateways start after the control phase and either must finish before the next control phase or report pending in the update. The system is synchronized to act on the status of the system at the same time.

[0050] As used herein, task may refer to and/or include a software component or process that is running on a processor. Hardware (HW) process may refer to and/or include an IP core within a Field programmable device such that the hardware can be reconfigured at runtime. Gateway may refer to and/or include a device capable of running a task or HW processes. By way of example only, a gateway may comprise a domain controller, an automotive domain specific gateway (*e.g.*, Body Control Module, electronic control unit (ECU), electronic control module (ECM), telematics control unit (TCU), *etc.*), *etc.* Computer server may refer to and/or include a processing unit used for running applications, such as path planning algorithms and multimedia applications, *etc.* Telecommunication control unit (TCU) may refer to and/or include a device(s) used to communicated via wireless technologies outside a vehicle. Data server may refer to and/or include a processing unit dedicated to storing and retrieving data. Network may refer to and/or include a form of communications to transfer data from one device to another, such as Ethernet, controller area network (CAN), local interconnect network (LIN), *etc.* Safety critical task may refer to and/or include a task or HW process that is deemed safe by the overall system by following the ISO26262, IEC61508 standards, or other related safety standards.

[0051] In exemplary embodiments, redundancy enables the failover to work as traffic needs to flow between the Safety Supervisor processes. The type of redundancy may vary but the redundancy should have a failover occur in less time than the cycle time (response time of the system).

[0052] In exemplary embodiments, a critical process is considered a process that implements at least the following features. Interprocess communication (IPC) interface allows for the Safety Supervisor to acquire its state, *e.g.*, application programming interface (API) for current state (running, recovered, pending, *etc.*), and watchdog counter and heartbeat. Non-critical tasks do not need to include this API but may be beneficial for diagnostics. Safety Supervisor may compare the stats from the operating system (OS) (central processing unit (CPU) utilization, memory usage, *etc.*) against expected values.

[0053] In an exemplary embodiment, a system may include two gateways with at least one controller area network (CAN) device attached to both gateways in a redundant fashion. If one gateway fails or goes down, then the signal comes from the other gateway to actuate, for example, a vehicle's brakes, lane change steering, *etc.* In this exemplary embodiment, the key manager is moved from one gateway to the next gateway. Fusion of data is still functioning but limited due to the loss of sensors from the gateway failure.

[0054] In exemplary embodiments, tasks may be moved based on predefined information, *e.g.*, gateway operational states, in a static table (*e.g.*, FIG. 6, *etc.*). In alternative embodiments, tasks may be moved dynamically whereby the current workload of all gateways are shared, and the safety supervisors calculate and share the best location to transfer the task in a failed condition. This calculation may be performed on each cycle so that all gateways in the cluster execute on the same known state of the cluster.

[0055] FIG. 8 illustrates an example of low latency failover according to exemplary embodiments of the present disclosure that may be configured for providing fault-tolerant distributed computing for a vehicular system with low latency changeover (*e.g.*, substantially instantaneously or within a response time of a vehicle safety system, *etc.*). In FIG. 8, "Recv Status" represents status coming from other nodes in the system. "Send Status" represents status information sent from a local node to all other nodes in the system (*e.g.*, via broadcast or similar mechanism). "Prep Status" represents the preparation of local node status information before the sync time period has expired. "Seq" represents the current sequence number of the frames being sent, which is used to line up actions across all nodes. All nodes work on the same Sequence number in the same time cycle. "Cycle" represents a cyclic time period.

[0056] In the example shown in FIG. 8, the required reaction/response time or synchronization time period is about 7 milliseconds (ms) in order to be substantially instantaneously. This 7 ms

response time corresponds with a vehicle crossing a lane (1 meter change in direction) at 250 kilometers per hour (km/hr) or 69.4 meters per second (m/s). In this example, the response time is 14 ms/2 for time to react. In other exemplary embodiments, the response time may be different or as critical.

[0057] The time is synchronized between nodes (Node N, Node N+1, *etc.*) as each safety supervisor (broadly, node) needs to be in sync to make sure the failover is synchronized. The accuracy may not necessarily be critical for the safety supervisor if all status information is updated within the same cycle for all nodes so as to ensure they are all working of the same data. In some exemplary embodiments, some jitter in the failover system can be supported as long as all nodes react within the cycle time window.

[0058] Exemplary embodiments disclosed herein may be configured for providing fault-tolerant distributed computing for a vehicular system while reducing the need for CPU power and duplication. Due to the possibility to start and stop application/task on failures, the overall system requirements may be reduced by avoiding the need to have duplicate hardware to run critical tasks in the event of a failure. By way of example, this may save between 30% to 50% of required CPU support in a vehicular system, *e.g.*, computer power for less critical applications (*e.g.*, radio, user video switching, *etc.*) may be repurposed for critical applications (*e.g.*, safety) if there is a failure somewhere. Exemplary embodiments may allow consolidation of a safety application into one computer box if there was another ready to take over. Exemplary embodiments may allow for monitoring at a much lower rate and running of noncritical task(s) with the freed resources.

[0059] See, for example, the distributed computer systems shown in FIG. 11 and FIG. 17, respectively. As shown in FIG. 17, the distributed computing system includes a redundant ADAS server node. In this example, the gateway nodes GW1 and GW2 each have an arbitrary compute capacity of 150 compute units, and the Compute and Redundant Compute each have a capacity of 500 compute units, such that the system total is 1300 compute units. For the failover example shown in FIGS. 11 and 12 in which there is no Redundant Compute (or its 500 compute units), the system would have a total of 800 compute units, thus providing a 40% reduction in compute units. The Failover scenario may have to scale back due to limited resources even with dropping noncritical tasks in the gateway nodes GWs. But as this is for an Emergency situation, limiting the performance can be expected, *e.g.*, limited to 50% of rated speed because the reduced sensor input processing isn't fast enough, *etc.*

[0060] The reduction of CPU needs and redundancy allows for a reduction in wiring thereby decreasing the weight and cost of the overall system. The reduced power consumption allows for longer battery life, use of lower cost silicon as operating temperatures may be lower with less power consumption, and/or use of less expensive/complex heat mitigation solutions.

[0061] In exemplary embodiments, each safety supervisor in the system is configured to act independently (*e.g.*, via static and/or dynamic algorithms, *etc.*) based on the state/status of its peers/other gateways in the system. Each Safety supervisor broadcasts its status to all other safety supervisors. Each subsystem or “information” may have its own update rate based on the criticality of the application. For example, steering control may have a required response time of 14 ms for lane departure. But loss of a sensor may not need to respond as fast, *e.g.*, because of the availability of data from the last scan and/or because the map of the environment will not change that fast.

[0062] Actions taken during the response time out requirements (*e.g.*, 1 cycle, 100 cycles, or 1000 cycles as shown in the Action table illustrated in FIG. 6, *etc.*) may be based on static actions tables (predefined steps in case of a failover event), which supports fast and predictable reaction times. The Actions tables (*e.g.*, FIG. 6, *etc.*) may be dynamically configured as a secondary task and be updated in the status report phase. As show in FIG. 10, the dynamic reconfiguration of the Action tables may be done via Artificial Intelligence or Machine learning, *etc.* All changes to the Actions tables are coordinated with acknowledgment from the Safety supervisors and only put in into action after the coordination with acknowledgement as shown in FIG. 10.

[0063] Status updates may include sending all status information from an Action table or only sending the changes. A heartbeat of indication of activity from each node is sent so that the corresponding action event does not time out in other nodes. If changes are sent, the Action table is updated in each node with that node’s changes.

[0064] After each cycle, each node is synchronized with the same information such that each node is therefore able to act on any failure events in a synchronized way. Time between cycles does not have to be exact, but each node must have the same data within the x period. Each cycle keeps track of its cycle number. Time synchronization in the system for data visualization may be done with time stamps from each sensor which a sensor fusion mechanism uses. Actions do not have to complete before the next cycle, but action status is updated indicating when an action is in progress before the cycle in finished.

[0065] The following examples of hardware failover (reconfiguration) are representative, and not exhaustive, of possible use cases for exemplary embodiments disclosed herein. In a first hardware failover example, an advanced driver-assistance system (ADAS) computer or data server node fails or goes down. The hardware failover includes moving object detection algorithm(s) to a zonal gateway node, *e.g.*, with limited functionality (*e.g.*, reduced maximum speed, *etc.*). The object detection may be handled by a software-defined radio (SDR) system instead of high network throughput.

[0066] Continuing with this first hardware failover example, FIG. 11 illustrates an exemplary distributed computing system including two zonal gateway nodes (*e.g.*, electronic control units (ECUs), *etc.*) connected to each other and to a computer or data server node via network connections (*e.g.*, high speed digital links, *etc.*). FIG. 11 also illustrates the data flow between sensors 5 (*e.g.*, cameras, Lidar, Radar, *etc.*), Gateway 1, Gateway 2, and the data server node (*e.g.*, ADAS 1, ADAS 2) during normal operation in a none failed operational state.

[0067] As shown in FIG. 12, the data server node has failed. In response, the hardware failover includes moving ADAS 1 and ADAS 2 from the data server node to Gateway 1 and Gateway 2, respectively. FIG. 12 also illustrates the failover data flow between ADAS 1 on Gateway 1 and ADAS 2 on Gateway 2.

[0068] In a second hardware failover example, a software-defined radio (SDR) may be reconfigured from a 5G mmWave radio to radar. If a front left radar sensor (or gateway node associated therewith) fails or goes down, then the front left SDR mmWave radio (5G remote antenna) may be reconfigured to act as a radar.

[0069] Continuing with this second hardware failover example, FIG. 13 illustrates an exemplary distributed computing system including a computer or data server node connected via network connections (*e.g.*, high speed digital links, *etc.*) to two zonal gateway nodes (*e.g.*, electronic control units (ECUs), *etc.*) and to an active remote antenna (*e.g.*, 5G mmWave antenna, *etc.*). The two zonal gateway nodes are also connected to each other via a network connection (*e.g.*, high speed digital links, *etc.*). FIG. 13 also illustrates the data flow between sensors 5 (*e.g.*, cameras, Lidar, Radar, 5mmWave Radar, *etc.*), first gateway node Gateway 1, second gateway node Gateway 2, the active remote antenna, and the data server node (*e.g.*, ADAS 1, ADAS 2, software-defined radio (SDR)) during normal operation in a none failed operational state.

[0070] As shown in FIG. 14, the second gateway node (Gateway 2) has failed and is unable to report data back to the ADAS applications on the data server node. In response, the hardware failover includes the remote antenna hardware taking over the function of the mmWave radar of the failed second gateway node from a hardware perspective, *e.g.*, the SDR mmWave radio (5G remote antenna) is reconfigured to act as a mmWave radar. A software task (mmWave radar failover app) is started on the data server node. The software-defined radio (SDR) on the data server node is disabled as denoted by the X through the SDR. The active antenna can therefore be used to keep sensor coverage in a corner of the vehicle at which the second zonal gateway mode went down by bringing down non-critical portion(s) of the vehicular communications system, *e.g.*, until the vehicle is able to stop at a safe stop location.

[0071] In a third hardware failover example, power steering is controlled via a first controller or zonal gateway node during normal operation in a none failed operational state. In response to a failure of the steering control task of the first controller, the hardware failover includes switching the power steering control from the first controller over to a second controller to handle this critical input/out (I/O). The hardware failover may include switching the primary network connection to the PS (Power Steering) module or other safety critical module (*e.g.*, CAN). This third hardware failover example may also be implemented with and used for vehicle brake controller hardware failover.

[0072] Continuing with this third hardware failover example, FIG. 15 illustrates an exemplary distributed computing system including two controllers or zonal gateway nodes (*e.g.*, electronic control units (ECUs), *etc.*) connected to each other and to a computer or data server node via network connections (*e.g.*, high speed digital links, *etc.*). The distributed computing system includes a failover mechanism (*e.g.*, redundant CAN links, *etc.*) so that control over the power steering unit's software control can be failed over to another controller. This can also be used for brakes or other critical vehicular systems.

[0073] During normal operation in a none failed operational state, the CAN link from the first controller (Gateway 1) is active, while the CAN link from the second controller (Gateway 2) is for monitoring only. Accordingly, power steering is controllable via the steering control task on the first controller during normal operation in a none failed operational state. FIG. 15 also illustrates the data flow between the two controllers during normal operation in a none failed operational state.

[0074] FIG. 16 illustrates the distributed computing system shown in FIG. 15 after the steering control task on the first controller has failed as denoted by the X. In response, control over the power steering unit's software control has been failed over to the second controller to handle this critical input/out (I/O) associated with power steering. Non critical task(s) (*e.g.*, air-conditioning (AC) control) on the second controller has been disabled. The steering control task on the second controller is active and no longer for monitoring only. The CAN link from the first controller is inactive. The CAN link from the second controller is active.

[0075] Exemplary embodiments are disclosed of fault-tolerant systems and methods of for providing fault-tolerant distributed computing for vehicular systems. In an exemplary embodiment, a system for providing fault-tolerant distributed computing for a vehicular system includes a first processing unit and a second processing unit communicatively coupled to the first processing unit. The first processing unit is configured to support a first set of processes. The second processing unit is configured to support a second set of processes, to monitor the state of the first set of processes, and to support at least one additional process of the first set of processes when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit. The second processing unit is configured to support the additional process substantially instantaneously.

[0076] In an exemplary embodiment, the first set of processes comprises a vehicle device control process configured to control a vehicle device. The at least one additional process assumes control of the vehicle device when the vehicle device control process has failed. The vehicle device control process may comprise one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection.

[0077] In an exemplary embodiment, the system is configured such that: during normal operation in a none failed operational state, at least one of the second set of processes are disabled in the second processing unit; and when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the at least one additional process of the first set of processes is supported by the previously disabled at least one of the second set of processes and at least one other of the second set of processes is disabled to provide resources in the second processing unit for supporting the at least one additional process of the first set of processes redistributed from the first processing unit to the second processing unit.

[0078] In an exemplary embodiment, the first processing unit is configured to monitor the state of the second set of processes, and to support at least one additional process of the second set of processes when the monitored state of the second set of processes indicates a failure in and/or loss of communication with the second processing unit.

[0079] In an exemplary embodiment, the system is configured such that when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the system determines, according to predefined states, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

[0080] In an exemplary embodiment, the system is configured such that when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the system dynamically determines, based on a current workload of the first and second processing units, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

[0081] In an exemplary embodiment, the system includes multiple electronic control units configured to be operable for communicating information to each other including sharing the state of their processes, such that in response to a failure in and/or loss of communication with an electronic control unit, one or more processes of said electronic control unit are supported by at least one of the remaining electronic control units. The system may be configured such that tasks, processes, and/or code is distributed across the multiple electronic control units and such that in response to the failure in and/or loss of communication with said electronic control unit, the at least one of the remaining electronic control units are operable for supporting one or more critical tasks of said electronic control unit. For example, the system may be configured such that the at least one of the remaining electronic control units are operable for supporting said one or more critical tasks of said electronic control unit including one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection. Or, for example, the system may be configured such that a software-defined radio (SDR) of the at least one of the remaining electronic control units is reconfigured from a 5G mmWave radio to radar,

and a remote antenna of the at least one of the remaining electronic control units takes over a function of the mmWave radar of said electronic control unit.

[0082] In an exemplary embodiment, the system includes multiple electronic control units including first and second processing units. The system is configured such that in the event of a failure in and/or loss of communication with an electronic control unit: one or more critical processes are redistributed from said electronic control unit to at least one of the remaining electronic control units; and one or more non-critical processes are disabled or shutdown in said at least one of the remaining electronic control units to provide resources for supporting the one or more critical processes redistributed from said electronic control unit to said at least one of the remaining electronic control units. The one or more critical processes may include or may be related to one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection. During normal operation in a none failed operational state, at least one process is disabled in said at least one of the remaining electronic control units. In the event of the failure in and/or loss of communication with the said electronic control unit, the one or more critical processes are redistributed from said electronic control unit to be supported by the previously disabled at least one process of said at least one of the remaining electronic control units.

[0083] In an exemplary embodiment, the first processing unit comprises a first electronic control unit, and the second processing unit comprises a second electronic control unit. the system is configured such that in the event of a failure in and/or loss of communication with the first or second electronic control unit: one or more critical processes are redistributed from said first or second electronic control unit to the other one of said first or second electronic control unit; and one or more non-critical processes are disabled or shutdown in the other one of the first or second electronic control unit to provide resources for supporting the one or more critical processes redistributed from said first or second electronic control unit to the other one of said first or second electronic control unit.

[0084] In an exemplary embodiment, the first processing unit comprises a first electronic control unit, and the second processing unit comprises a second electronic control unit. The system further comprises third and fourth electronic control units communicatively coupled to each other and to the first and second electronic control units. The system is configured such that in the event of a failure in and/or loss of communication with the fourth electronic control unit: one or more

critical processes are redistributed from the fourth electronic control unit to at least one of the first, second, and third electronic control units; and one or more non-critical processes are disabled or shutdown in the at least one of the first, second, and third electronic control units to provide resources for supporting the one or more critical processes redistributed from the fourth electronic control unit.

[0085] In an exemplary embodiment, the system the system includes multiple electronic control units including the first and second processing units. The system further comprises a controller area network device. The multiple electronic control units include a redundant connection to the controller area network device. The system is configured such that in the event of a failure in and/or loss of communication with an electronic control unit, one or more signals to actuate one or more vehicle control devices are sent by one or more of the remaining electronic control units.

[0086] In an exemplary embodiment, an autonomous or semi-autonomous vehicle comprises one or more vehicular systems within the vehicle configured to be operable for controlling operation of the vehicle. The one or more vehicular systems are interconnected by a system as disclosed, which provides fault-tolerant distributed computing for the one or more vehicular systems with low latency changeover in the one or more vehicular systems.

[0087] Also disclosed are exemplary methods of for providing fault-tolerant distributed computing for vehicular system.. In an exemplary embodiment, a computer-implemented method of providing fault-tolerant distributed computing for a vehicular comprises monitoring the state of a first set of processes of a first processing unit via a second processing unit, the second processing unit configured to support a second set of processes. When the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit, the method comprises substantially instantaneously: redistributing at least one of the first set of processes from the first processing unit to the second processing unit; and disabling at least one of the second set of processes to provide resources in the second processing unit for supporting the at least one of the first set of processes redistributed from the first processing unit to the second processing unit.

[0088] In an exemplary embodiment, the first set of processes comprises a vehicle device control process configured to control a vehicle device. The method comprises the second processing unit assuming control of the vehicle device when the vehicle device control process has

failed. The vehicle device control process may comprise one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection.

[0089] In an exemplary embodiment, the method includes disabling at least one of the second set of processes in the second processing unit during normal operation in a none failed operational state. When the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit, the method further includes supporting the at least one process of the first set of processes by the previously disabled at least one of the second set of processes.

[0090] In an exemplary embodiment, the method includes: monitoring the state of the second set of processes of the second processing unit via the first processing unit. When the state of the second set of processes as monitored by the first processing unit indicates a failure in and/or loss of communication with the second processing unit, the method further includes: redistributing at least one of the second set of processes from the second processing unit to first processing unit; and disabling at least one of the first set of processes to provide resources in the first processing unit for supporting the at least one of the second set of processes redistributed from the second processing unit to the first processing unit.

[0091] In an exemplary embodiment, when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the method includes: determining, according to predefined states, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes; and/or dynamically determining, based on a current workload of the first and second processing units, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

[0092] In an exemplary embodiment, the first and second processing units include a redundant connection to a controller area network device. In the event of a failure in and/or loss of communication with one of the first or second processing unit, the method includes sending one

or more signals to actuate one or more vehicle control devices from the other one of said first or second processing unit.

[0093] Exemplary embodiments of non-transitory computer-readable storage media including executable instructions for providing fault-tolerant distributed computing for vehicular systems are also disclosed. In an exemplary embodiment, a non-transitory computer-readable storage media includes executable instructions for providing fault-tolerant distributed computing for a vehicular system, such that when executed by at least one processor: the state of a first set of processes of a first processing unit is monitored by a second processing unit, the second processing unit configured to support a second set of processes; and when the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit: at least one of the first set of processes is redistributed substantially instantaneously from the first processing unit to the second processing unit; and at least one of the second set of processes is disabled substantially instantaneously to provide resources in the second processing unit for supporting the at least one of the first set of processes redistributed from the first processing unit to the second processing unit.

[0094] In an exemplary embodiment, the first set of processes comprises a vehicle device control process configured to control a vehicle device. The non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor, the second processing unit assumes control of the vehicle device when the vehicle device control process has failed. The vehicle device control process may comprise one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection.

[0095] In an exemplary embodiment, the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor: at least one of the second set of processes is disabled in the second processing unit during normal operation in a none failed operational state; and when the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit, at least one process of the first set of processes is supported by the previously disabled at least one of the second set of processes.

[0096] In an exemplary embodiment, the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor: the state of the

second set of processes of the second processing unit is monitored by the first processing unit; and when the state of the second set of processes as monitored by the first processing unit indicates a failure in and/or loss of communication with the second processing unit: at least one of the second set of processes is redistributed from the second processing unit to first processing unit; and at least one of the first set of processes is disabled to provide resources in the first processing unit for supporting the at least one of the second set of processes redistributed from the second processing unit to the first processing unit.

[0097] In an exemplary embodiment, the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor and when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit: a determination is made according to predefined states as to which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes; and/or a determination is made dynamically based on a current workload of the first and second processing units as to which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

[0098] In an exemplary embodiment, the first and second processing units include a redundant connection to a controller area network device. The non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor: in the event of a failure in and/or loss of communication with one of the first or second processing unit, one or more signals to actuate one or more vehicle control devices are sent from the other one of said first or second processing unit.

[0099] Exemplary embodiments disclosed herein may provide one or more (but not necessarily any or all) of the following features and/or advantages. For example, exemplary embodiments disclosed herein may be configured for providing fault-tolerant distributed computing for a vehicular system with low latency changeover (*e.g.*, substantially instantaneously, within a response time of a vehicle safety system, *etc.*) in the event of a failure in and/or loss of communication with a processing unit (*e.g.*, gateway, *etc.*). Exemplary embodiments disclosed herein may be configured (*e.g.*, include high speed digital links, *etc.*) that allow for substantially

instantaneous changeover (*e.g.*, within a response time of a vehicle safety system, *etc.*) in the event of a failure in and/or loss of communication with a processing unit (*e.g.*, gateway, *etc.*). Exemplary embodiments disclosed herein may be configured for substantially instantaneous remapping and repurposing programmable hardware (*e.g.*, box to box, *etc.*) for new hardware operations, *e.g.*, not just changing the software. Exemplary embodiments disclosed herein may be configured for providing fault-tolerant distributed computing for a vehicular system with a reduced or minimal number of data links, with less required computational power, with less wiring, and with less data paths.

[00100] As will be appreciated based on the foregoing specification, the above-described embodiments of the disclosure may be implemented using computer programming or engineering techniques including computer software, firmware, hardware, or any combination or subset thereof. Exemplary embodiments may include one or more processors and memory coupled to (and in communication with) the one or more processors. A processor may include one or more processing units (*e.g.*, in a multi-core configuration, *etc.*) such as, and without limitation, a central processing unit (CPU), a microcontroller, a reduced instruction set computer (RISC) processor, an application specific integrated circuit (ASIC), a programmable logic device (PLD), a gate array, and/or any other circuit or processor capable of the functions described herein.

[00101] It should be appreciated that the functions described herein, in some embodiments, may be described in computer executable instructions stored on a computer readable media, and executable by at least one processor. The computer readable media is a non-transitory computer readable storage medium. By way of example, and not limitation, such computer-readable media can include dynamic random access memory (DRAM), static random access memory (SRAM), read only memory (ROM), erasable programmable read only memory (EPROM), solid state devices, flash drives, CD-ROMs, thumb drives, floppy disks, tapes, hard disks, other optical disk storage, magnetic disk storage or other magnetic storage devices, any other type of volatile or nonvolatile physical or tangible computer-readable media, or other medium that can be used to carry or store desired program code in the form of instructions or data structures and that can be accessed by a computer. Combinations of the above should also be included within the scope of computer-readable media.

[00102] Computer-executable instructions may be stored in the memory for execution by a processor to particularly cause the processor to perform one or more of the functions described

herein, such that the memory is a physical, tangible, and non-transitory computer readable storage media. Such instructions often improve the efficiencies and/or performance of the processor that is performing one or more of the various operations herein. It should be appreciated that the memory may include a variety of different memories, each implemented in one or more of the functions or processes described herein.

[00103] It should also be appreciated that one or more aspects of the present disclosure transform a general-purpose computing device into a special-purpose computing device when configured to perform the functions, methods, and/or processes described herein.

[00104] The disclosure provided herein describes features in terms of preferred and exemplary embodiments thereof. Numerous other embodiments, modifications and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure.

CLAIMS

We claim:

1. A system for providing fault-tolerant distributed computing for a vehicular system, the system comprising:

a first processing unit configured to support a first set of processes; and

a second processing unit communicatively coupled to the first processing unit, the second processing unit configured to support a second set of processes, to monitor the state of the first set of processes, and to support an additional process of the first set of processes when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, wherein the second processing unit is configured to support the additional process substantially instantaneously.

2. The system of claim 1, wherein:

the first set of processes comprises a vehicle device control process configured to control a vehicle device; and

the at least one additional process assumes control of the vehicle device when the vehicle device control process has failed.

3. The system of claim 2, wherein the vehicle device control process comprises one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection.

4. The system of claim 1, wherein the system is configured such that:

during normal operation in a none failed operational state, at least one of the second set of processes is disabled in the second processing unit; and

when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the at least one additional process of the first set of processes is supported by the previously disabled at least one of the second set of processes and at least one other of the second set of processes is disabled to provide resources in the second

processing unit for supporting the at least one additional process of the first set of processes redistributed from the first processing unit to the second processing unit.

5. The system of any one of claims 1 to 4, wherein the first processing unit is configured to monitor the state of the second set of processes, and to support at least one additional process of the second set of processes when the monitored state of the second set of processes indicates a failure in and/or loss of communication with the second processing unit.

6. The system of any one of claims 1 to 4, wherein the system is configured such that when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the system determines, according to predefined states, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

7. The system of any one of claims 1 to 4, wherein the system is configured such that when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the system dynamically determines, based on a current workload of the first and second processing units, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

8. The system of any one of claims 1 to 4, wherein the system includes multiple electronic control units including the first and second processing units, the multiple electronic control units configured to be operable for communicating information to each other including sharing the state of their processes, such that in response to a failure in and/or loss of communication with an electronic control unit, one or more processes of said electronic control unit are supported by at least one of the remaining electronic control units.

9. The system of claim 8, wherein the system is configured such that tasks, processes, and/or code is distributed across the multiple electronic control units and such that in response to the failure in and/or loss of communication with said electronic control unit, the at least one of the remaining electronic control units are operable for supporting one or more critical tasks of said electronic control unit.

10. The system of claim 9, wherein the system is configured such that:
the at least one of the remaining electronic control units are operable for supporting said one or more critical tasks of said electronic control unit including one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection; and/or

a software-defined radio (SDR) of the at least one of the remaining electronic control units is reconfigured from a 5G mmWave radio to radar, and a remote antenna of the at least one of the remaining electronic control units takes over a function of the mmWave radar of said electronic control unit.

11. The system of any one of claims 1 to 4, wherein:
the system includes multiple electronic control units including the first and second processing units;

the system is configured such that in the event of a failure in and/or loss of communication with an electronic control unit:

one or more critical processes are redistributed from said electronic control unit to at least one of the remaining electronic control units, the one or more critical processes including or related to one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection; and

one or more non-critical processes are disabled or shutdown in said at least one of the remaining electronic control units to provide resources for supporting the one or more critical processes redistributed from said electronic control unit to said at least one of the remaining electronic control units.

12. The system of claim 11, wherein:

during normal operation in a none failed operational state, at least one process is disabled in said at least one of the remaining electronic control units; and

in the event of the failure in and/or loss of communication with the said electronic control unit, the one or more critical processes are redistributed from said electronic control unit to be supported by the previously disabled at least one process of said at least one of the remaining electronic control units.

13. The system of any one of claims 1 to 4, wherein:

the first processing unit comprises a first electronic control unit;

the second processing unit comprises a second electronic control unit;

the system is configured such that in the event of a failure in and/or loss of communication with the first or second electronic control unit:

one or more critical processes are redistributed from said first or second electronic control unit to the other one of said first or second electronic control unit; and

one or more non-critical processes are disabled or shutdown in the other one of the first or second electronic control unit to provide resources for supporting the one or more critical processes redistributed from said first or second electronic control unit to the other one of said first or second electronic control unit.

14. The system of any one of claims 1 to 4, wherein:

the first processing unit comprises a first electronic control unit;

the second processing unit comprises a second electronic control unit;

the system further comprises third and fourth electronic control units communicatively coupled to each other and to the first and second electronic control units;

the system is configured such that in the event of a failure in and/or loss of communication with the fourth electronic control unit:

one or more critical processes are redistributed from the fourth electronic control unit to at least one of the first, second, and third electronic control units; and

one or more non-critical processes are disabled or shutdown in the at least one of the first, second, and third electronic control units to provide resources for supporting the one or more critical processes redistributed from the fourth electronic control unit.

15. The system of any one of claims 1 to 4, wherein:

the system includes multiple electronic control units including the first and second processing units;

the system further comprises a controller area network device;

the multiple electronic control units include a redundant connection to the controller area network device; and

the system is configured such that in the event of a failure in and/or loss of communication with an electronic control unit, one or more signals to actuate one or more vehicle control devices are sent by one or more of the remaining electronic control units.

16. An autonomous or semi-autonomous vehicle comprising one or more vehicular systems within the vehicle configured to be operable for controlling operation of the vehicle, the one or more vehicular systems interconnected by the system of any one of claims 1 to 4, thereby providing fault-tolerant distributed computing for the one or more vehicular systems with low latency changeover in the one or more vehicular systems.

17. A computer-implemented method of providing fault-tolerant distributed computing for a vehicular system, the method comprising:

monitoring the state of a first set of processes of a first processing unit via a second processing unit, the second processing unit configured to support a second set of processes; and

when the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit, the method comprises substantially instantaneously:

redistributing at least one of the first set of processes from the first processing unit to the second processing unit; and

disabling at least one of the second set of processes to provide resources in the second processing unit for supporting the at least one of the first set of processes redistributed from the first processing unit to the second processing unit.

18. The method of claim 17, wherein:

the first set of processes comprises a vehicle device control process configured to control a vehicle device; and

the method comprises the second processing unit assuming control of the vehicle device when the vehicle device control process has failed.

19. The method of claim 18, wherein the vehicle device control process comprises one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection.

20. The method of any one of claims 17 to 19, wherein the method includes:

disabling at least one of the second set of processes in the second processing unit during normal operation in a none failed operational state; and

when the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit, supporting the at least one process of the first set of processes by the previously disabled at least one of the second set of processes.

21. The method of any one of claims 17 to 19, wherein the method includes:

monitoring the state of the second set of processes of the second processing unit via the first processing unit; and

when the state of the second set of processes as monitored by the first processing unit indicates a failure in and/or loss of communication with the second processing unit:

redistributing at least one of the second set of processes from the second processing unit to first processing unit; and

disabling at least one of the first set of processes to provide resources in the first processing unit for supporting the at least one of the second set of processes redistributed from the second processing unit to the first processing unit.

22. The method of any one of claims 17 to 19, wherein when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit, the method includes:

determining, according to predefined states, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes; and/or

dynamically determining, based on a current workload of the first and second processing units, which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

23. The method of any one of claims 17 to 19, wherein:

the first and second processing units include a redundant connection to a controller area network device; and

in the event of a failure in and/or loss of communication with one of the first or second processing unit, the method includes sending one or more signals to actuate one or more vehicle control devices from the other one of said first or second processing unit.

24. A non-transitory computer-readable storage media including executable instructions for providing fault-tolerant distributed computing for a vehicular system, such that when executed by at least one processor:

the state of a first set of processes of a first processing unit is monitored by a second processing unit, the second processing unit configured to support a second set of processes; and

when the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit:

at least one of the first set of processes is redistributed substantially instantaneously from the first processing unit to the second processing unit; and

at least one of the second set of processes is disabled substantially instantaneously to provide resources in the second processing unit for supporting the at least one of the first set of processes redistributed from the first processing unit to the second processing unit.

25. The non-transitory computer-readable storage media of claim 24, wherein:

the first set of processes comprises a vehicle device control process configured to control a vehicle device; and

the executable instructions include executable instructions that when executed by the at least one processor, the second processing unit assumes control of the vehicle device when the vehicle device control process has failed.

26. The non-transitory computer-readable storage media of claim 25, wherein the vehicle device control process comprises one or more of vehicular steering, vehicular braking, advanced driver-assistance system (ADAS) functioning, and/or object detection.

27. The non-transitory computer-readable storage media of any one of claims 24 to 26, wherein the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor:

at least one of the second set of processes is disabled in the second processing unit during normal operation in a none failed operational state; and

when the state of the first set of processes as monitored by the second processing unit indicates a failure in and/or loss of communication with the first processing unit, at least one process of the first set of processes is supported by the previously disabled at least one of the second set of processes.

28. The non-transitory computer-readable storage media of any one of claims 24 to 26, wherein the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor:

the state of the second set of processes of the second processing unit is monitored by the first processing unit; and

when the state of the second set of processes as monitored by the first processing unit indicates a failure in and/or loss of communication with the second processing unit:

at least one of the second set of processes is redistributed from the second processing unit to first processing unit; and

at least one of the first set of processes is disabled to provide resources in the first processing unit for supporting the at least one of the second set of processes redistributed from the second processing unit to the first processing unit.

29. The non-transitory computer-readable storage media of any one of claims 24 to 26, wherein the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor and when the monitored state of the first set of processes indicates a failure in and/or loss of communication with the first processing unit:

a determination is made according to predefined states as to which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes; and/or

a determination is made dynamically based on a current workload of the first and second processing units as to which one or more critical processes are redistributed from the first processing unit to the second processing unit and which one or more non-critical processes are disabled or shutdown in the second processing unit for supporting the one or more critical processes.

30. The non-transitory computer-readable storage media of any one of claims 24 to 26, wherein:

the first and second processing units include a redundant connection to a controller area network device; and

the non-transitory computer-readable storage media includes executable instructions that when executed by the at least one processor: in the event of a failure in and/or loss of

communication with one of the first or second processing unit, one or more signals to actuate one or more vehicle control devices are sent from the other one of said first or second processing unit.

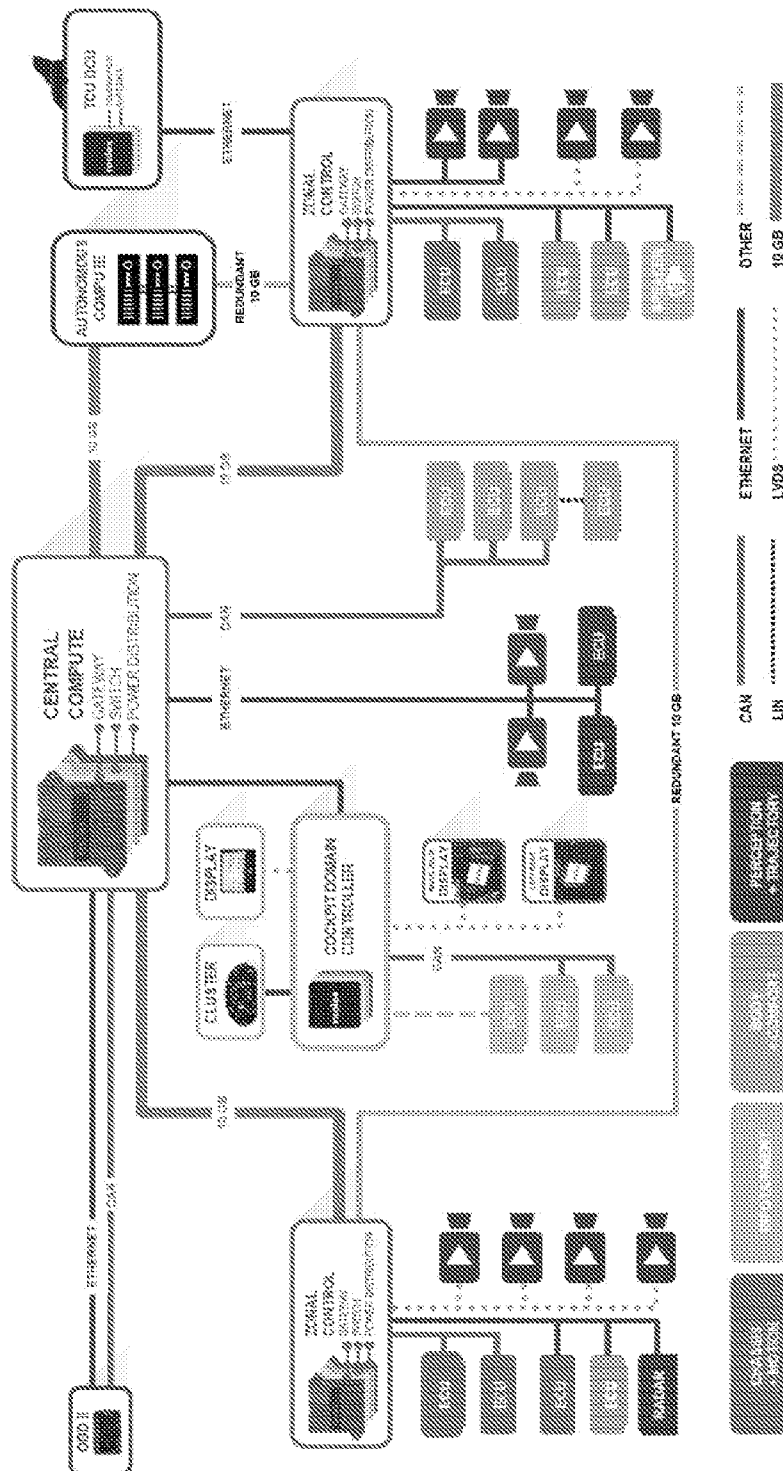


FIG. 1

Load Balanced Distributed Control/Fail Safe

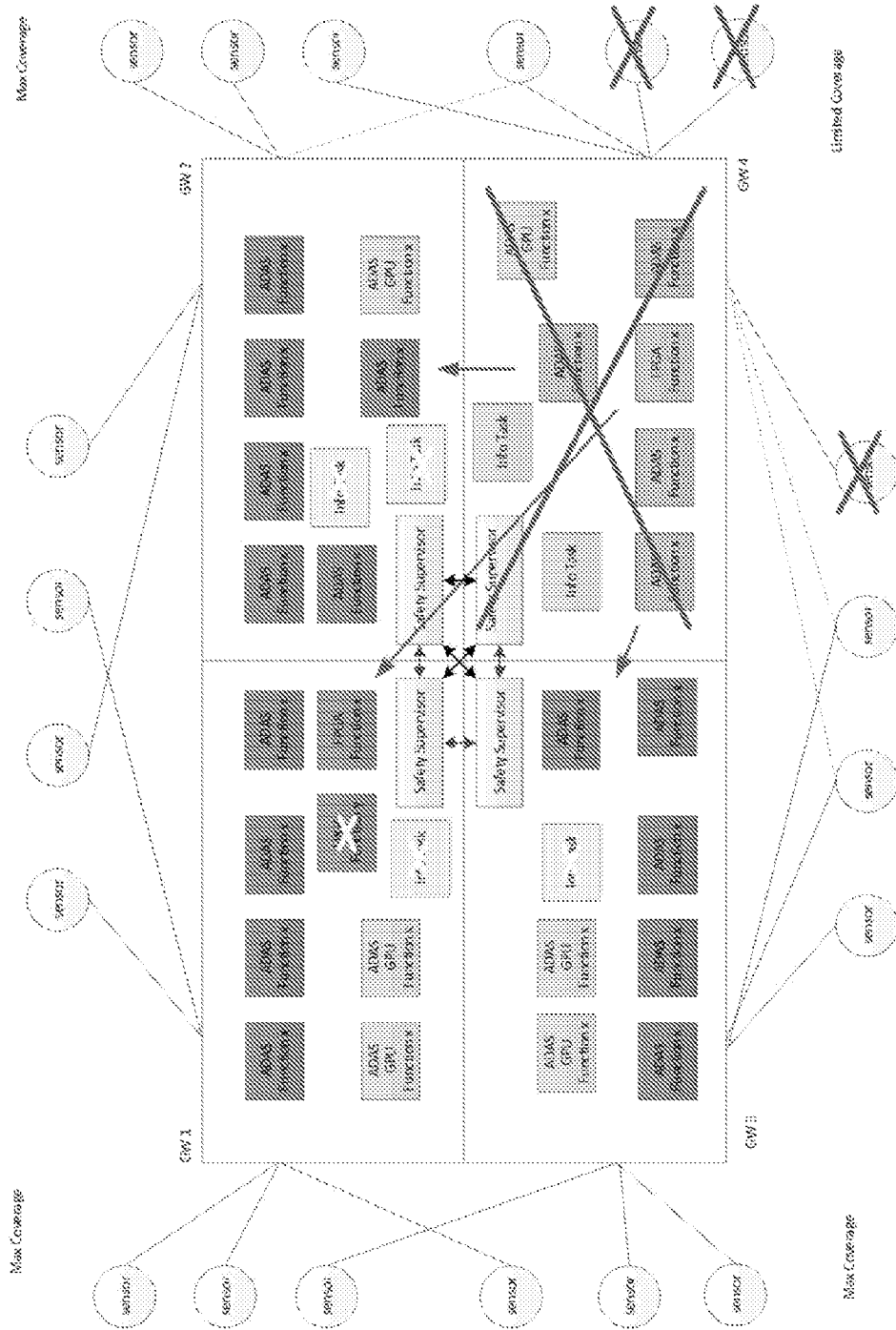


FIG. 3

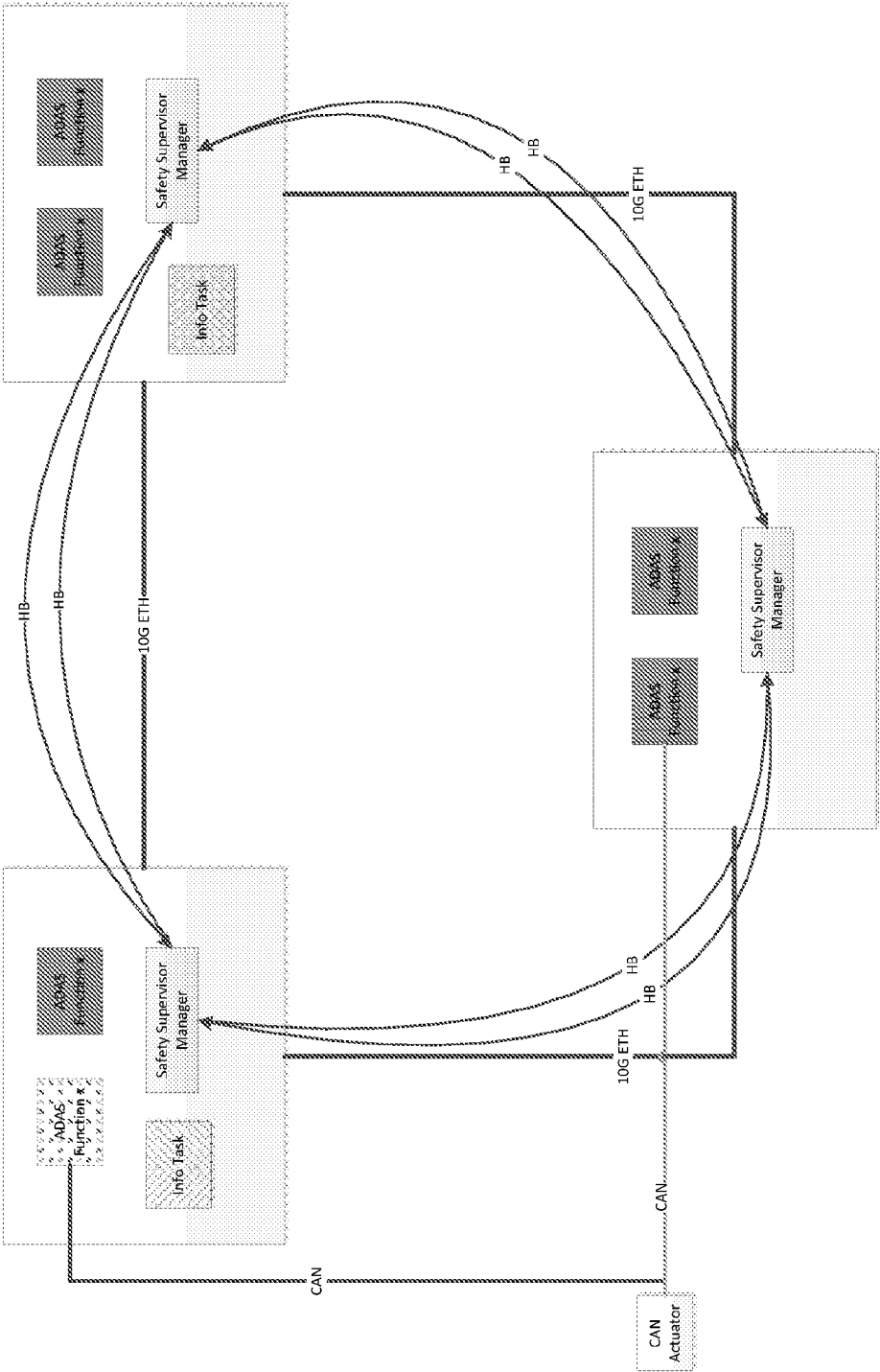


FIG. 4

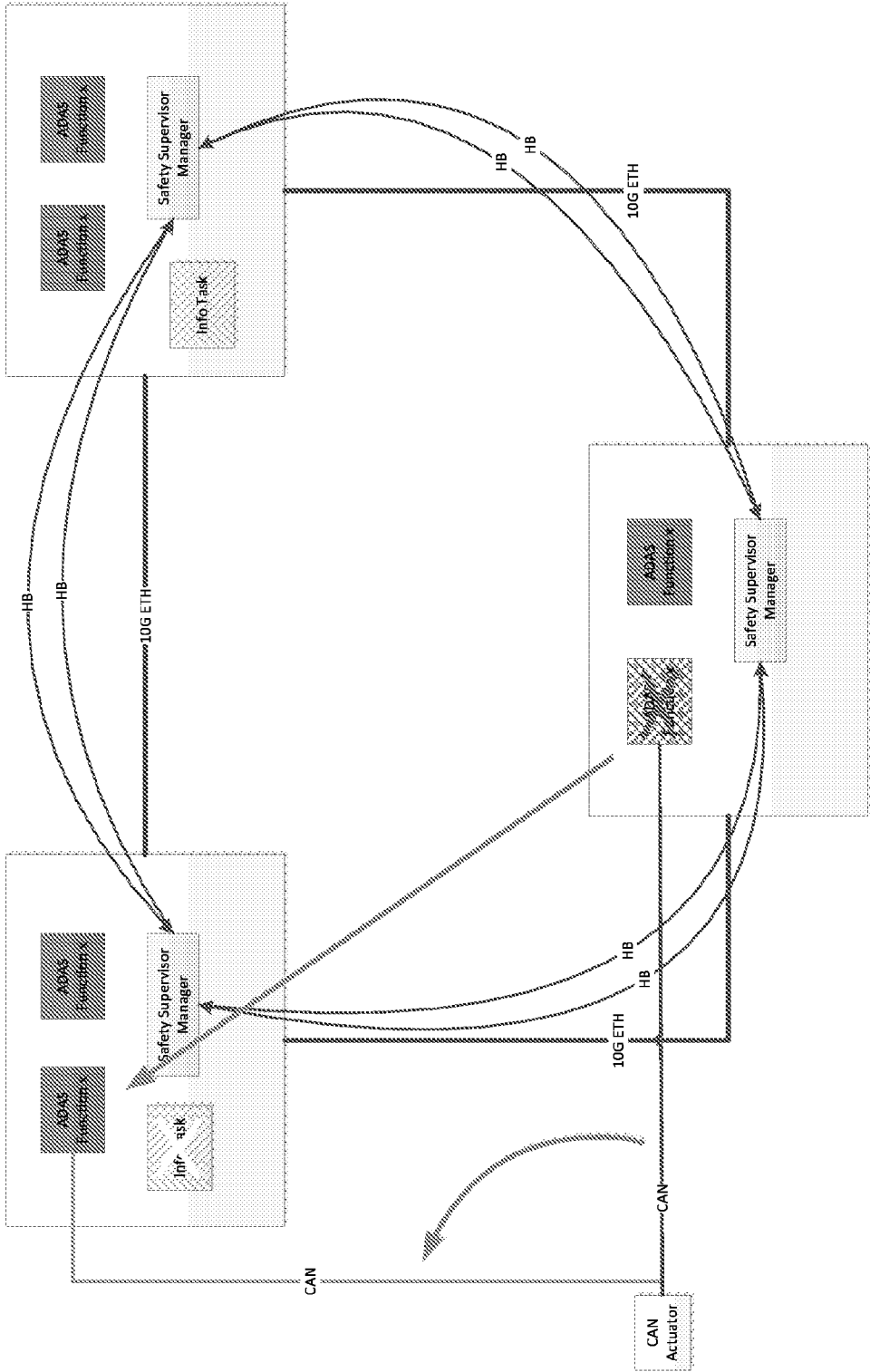


FIG. 5

Entry	Process	Type	Gateway (Primary)	State	Time out	Action on Primary *	Action on Secondary	Process	Gateway Secondary	Recoverable
1	KeyManager	Critical	Front Right	Failed	200 cycles	Stop	Start Stop	KeyManager Media Player	Front Left Front Left	Stay Stopped
2	Adaptive Cruise	Critical	Front Right	Failed	1 Cycle	Stop	Start Stop	Adaptive Cruise Media Player	Front Left Front Left	Stay Stopped
3	ADAS Function 1	Critical	Rear	Failed	1 cycle	Stop	Start Stop	ADAS Function 1 Media Player	Front Right Front Right	Stay Stopped
4	ADAS Function 2	Critical	Front Left	Out of Spec	1 cycle	Stop	Start	ADAS Function 2	Front Right	Restart
5	ADAS Function 2	Critical	Front Left	Recovered	1 cycle	Start	Stop	ADAS Function 2	Front Right	N/A
6	Media Player	Non Critical	Front Left	Failed	2000 cycles	None	N/A	N/A	N/A	N/A

* Only if available to do so,

FIG. 6

State	Description
Failed	Process is not available or not responding to queries
Out of Spec	The Process has used more resources than expected
Recovered	Process is recovered from error state
Running	Process is running normally
Disabled	Process has been disabled (on purpose)
Pending Start	Process is Starting up
Stopped	Process has been Stopped

FIG. 7

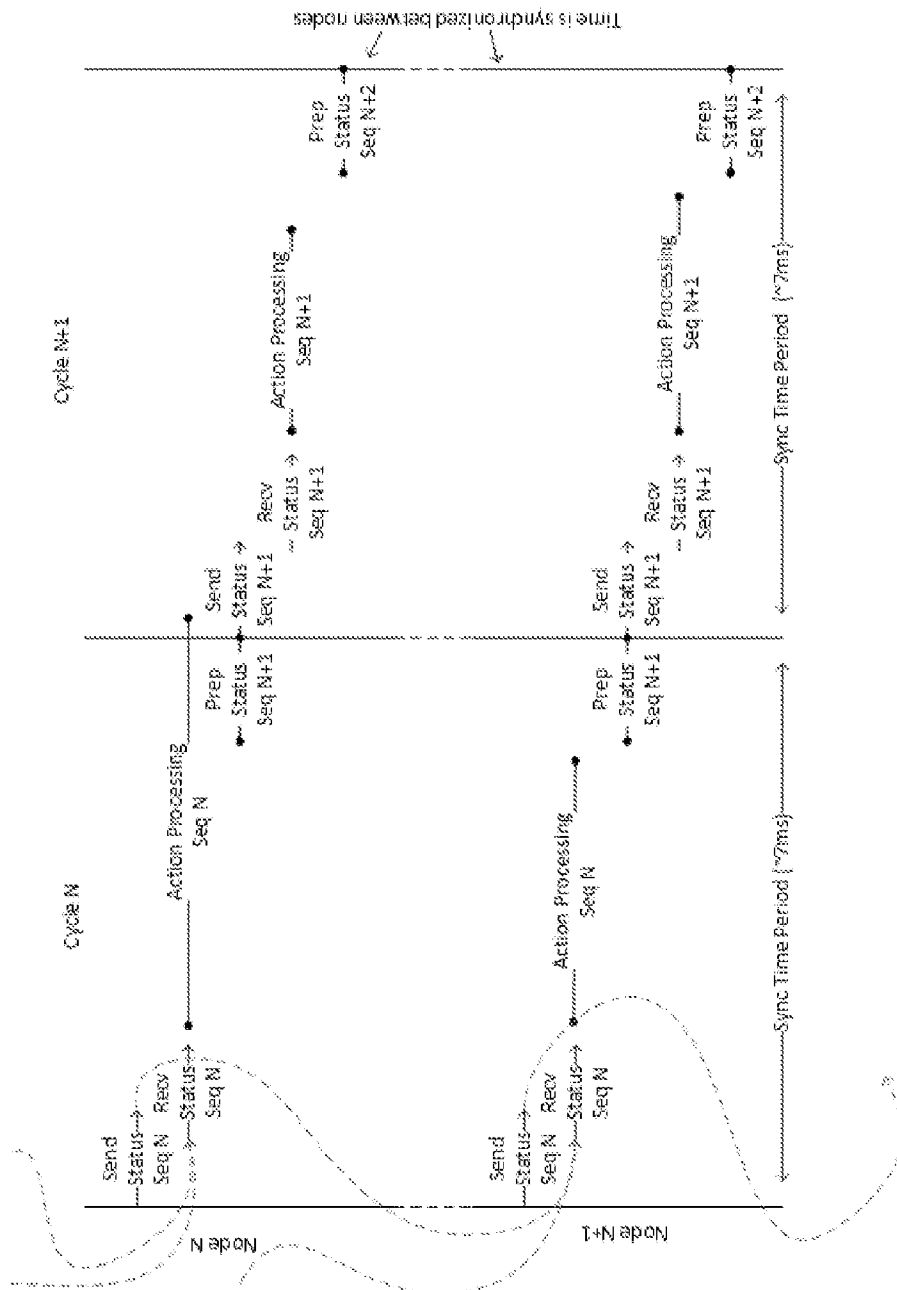


FIG. 8

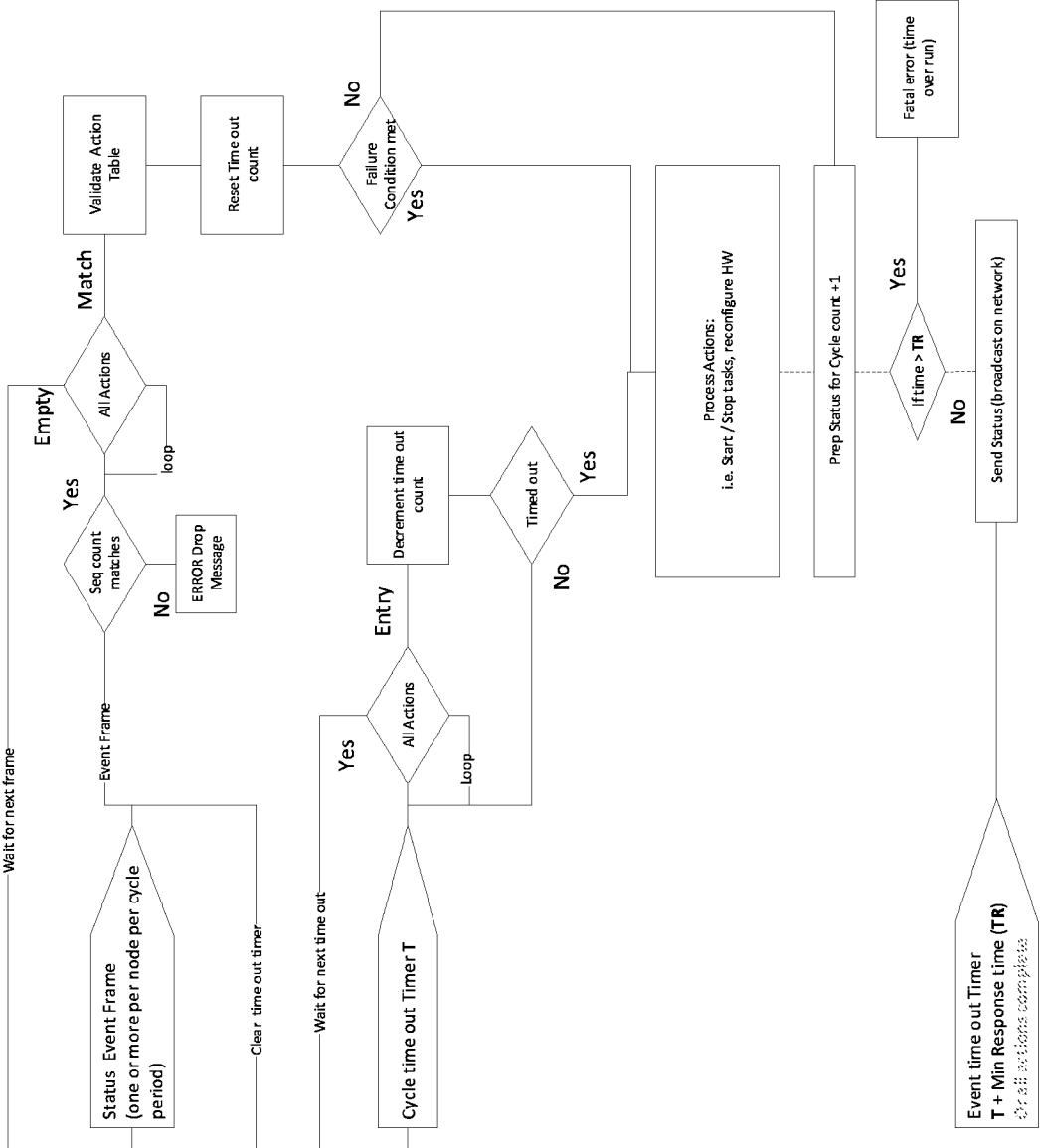


FIG. 9

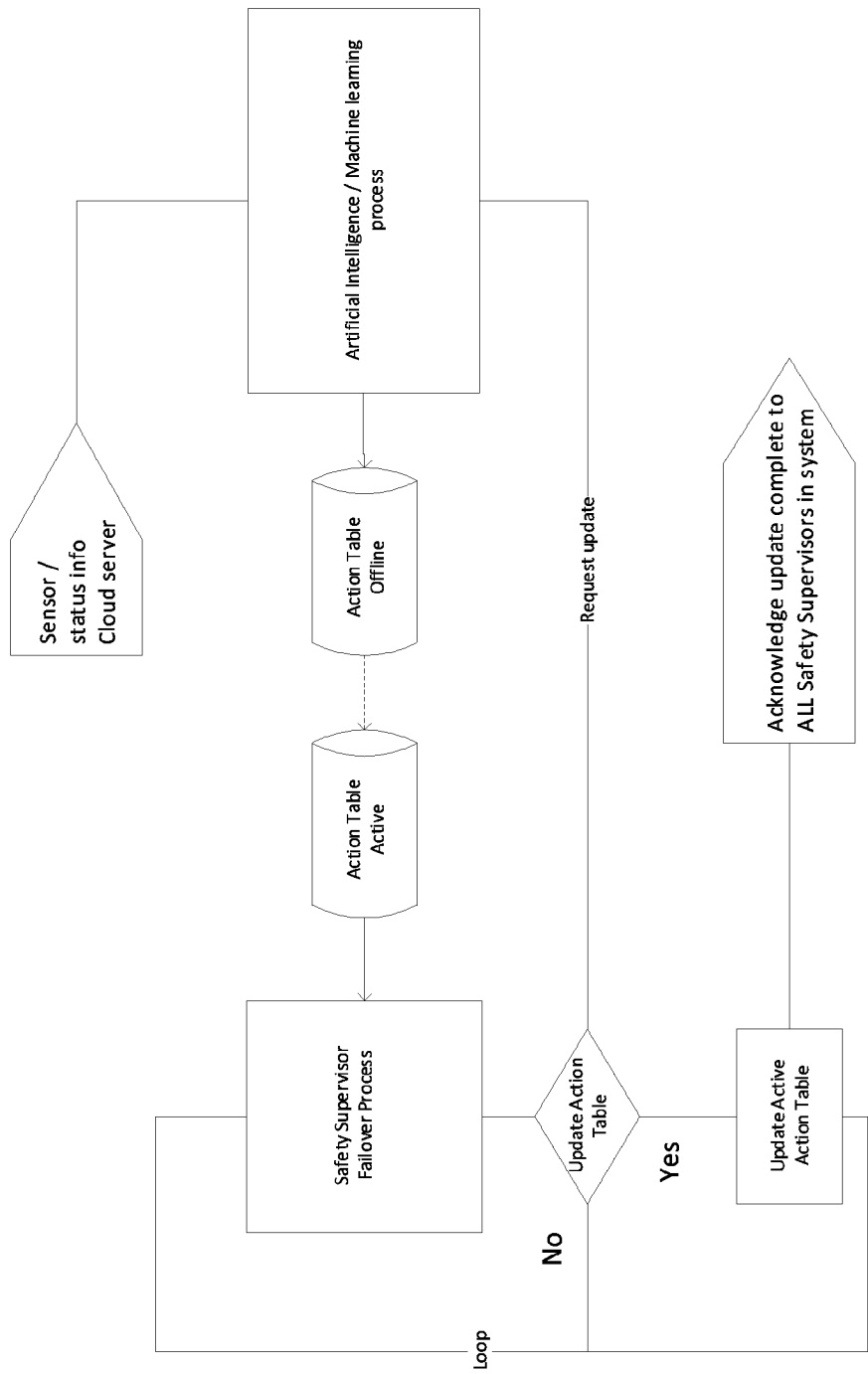
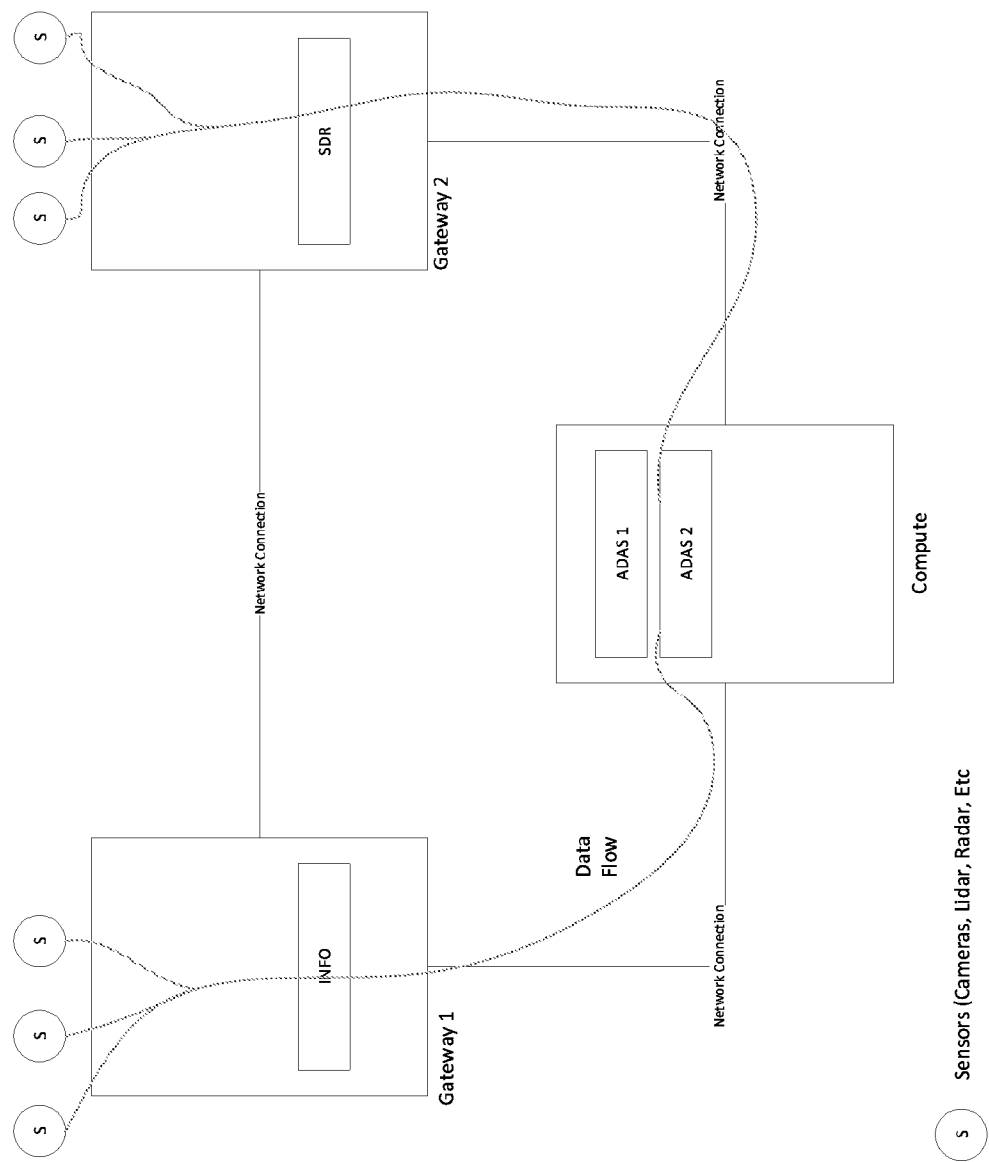


FIG. 10



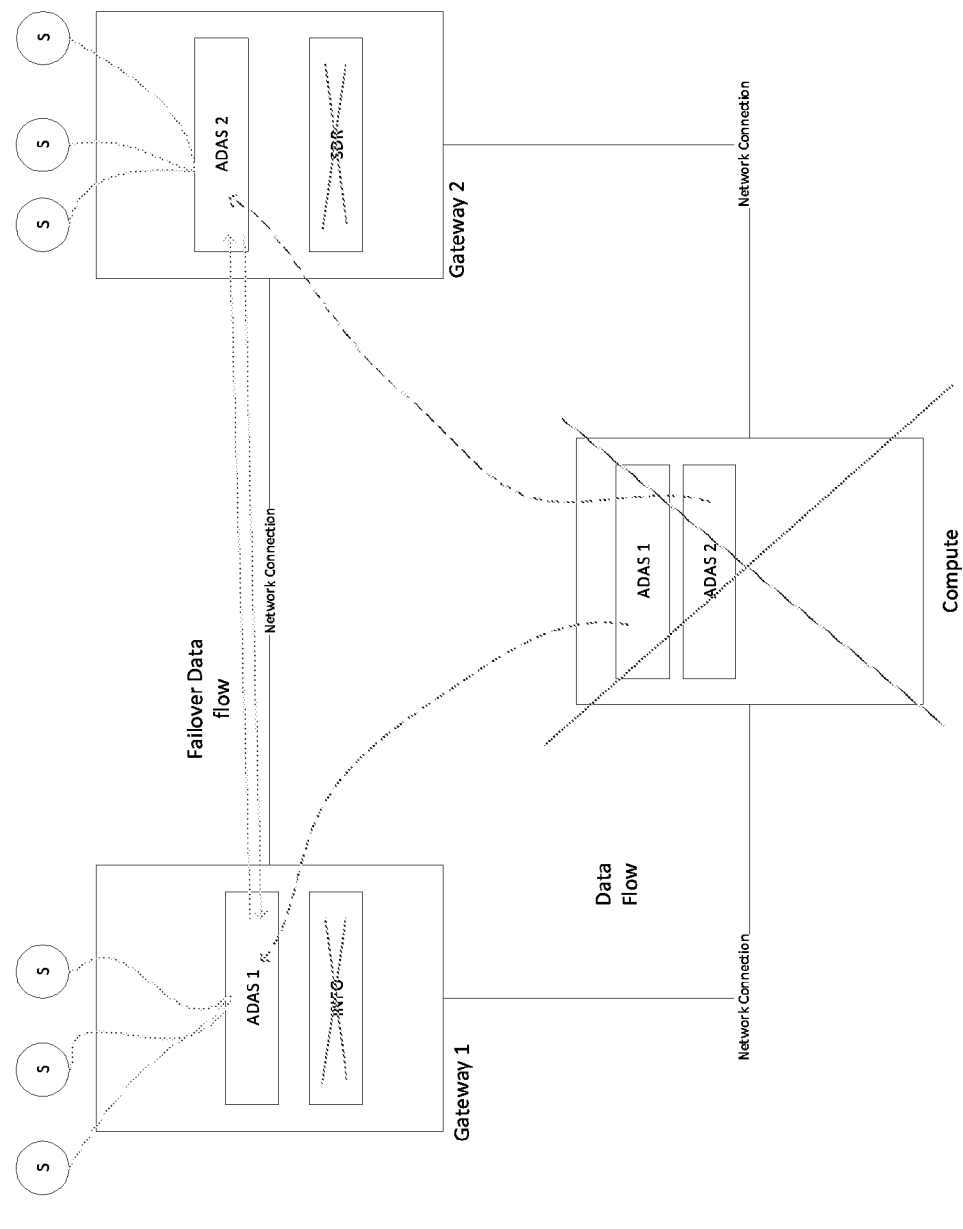


FIG. 12

Sensors (Cameras, Lidar, Radar, Etc

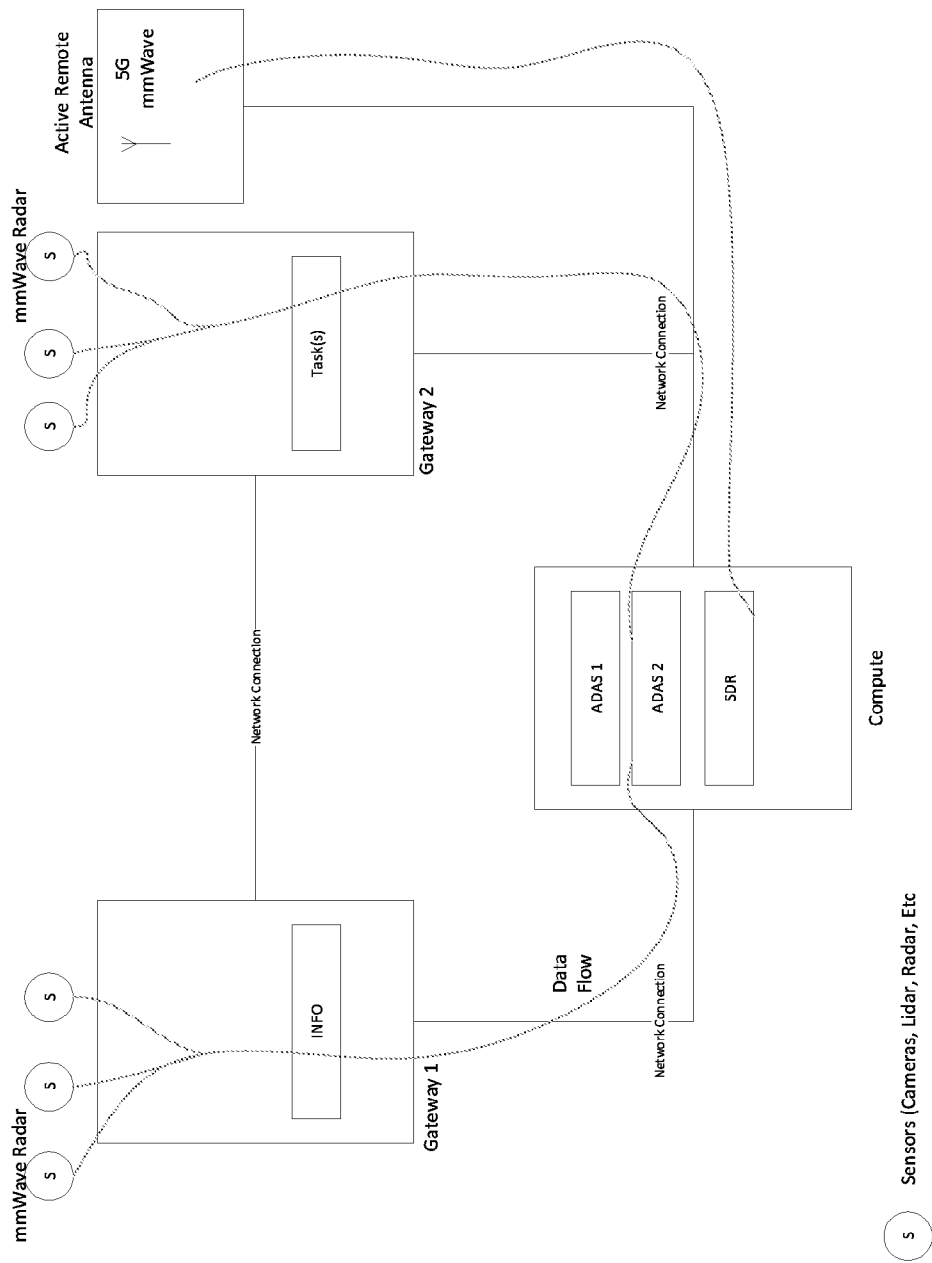


FIG. 13

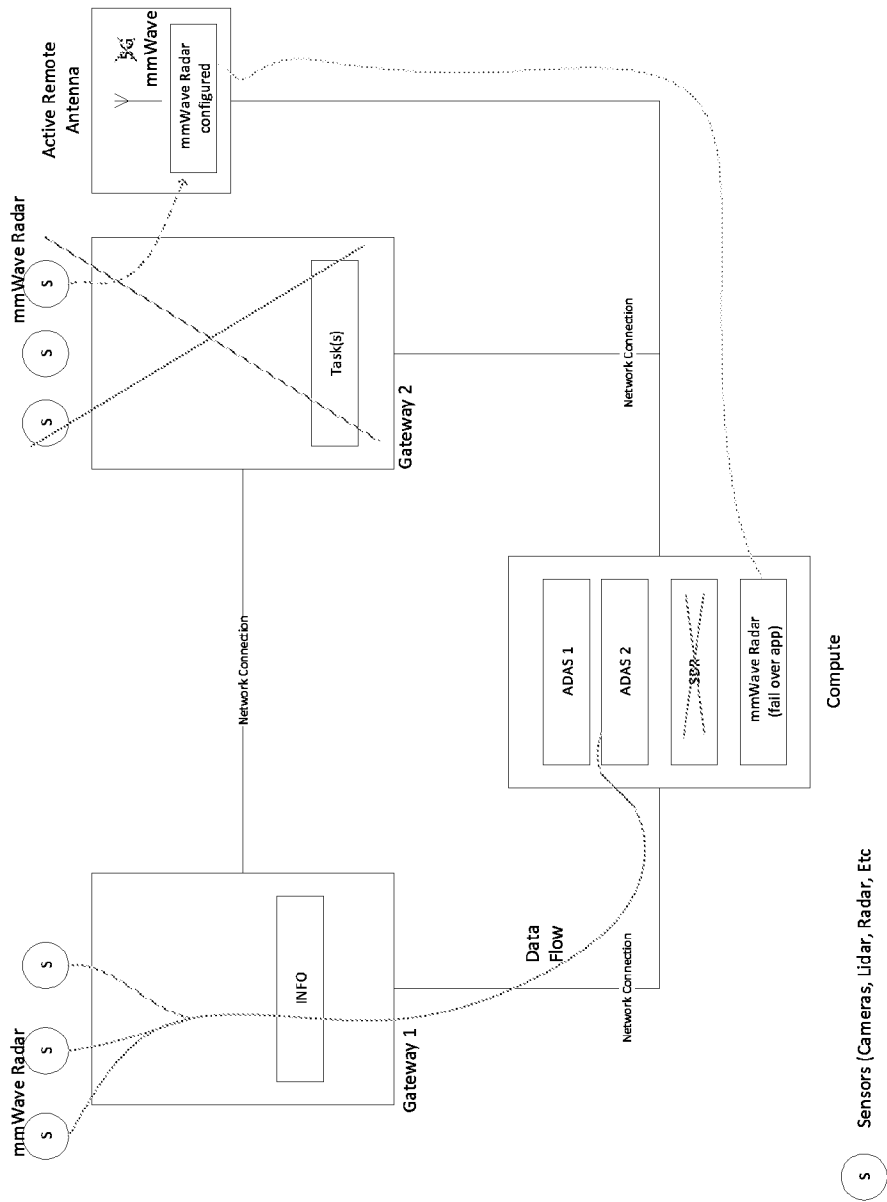


FIG. 14

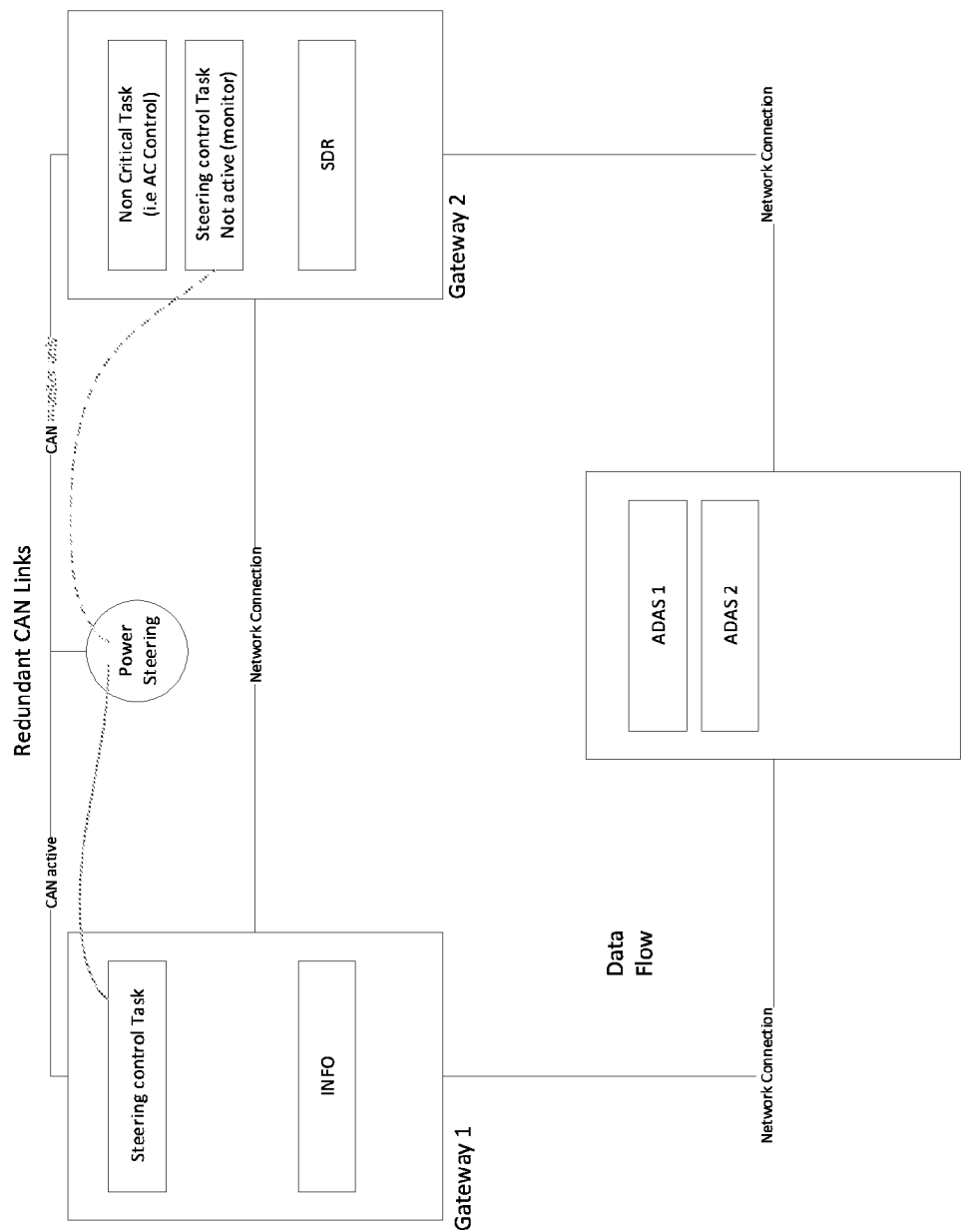


FIG. 15

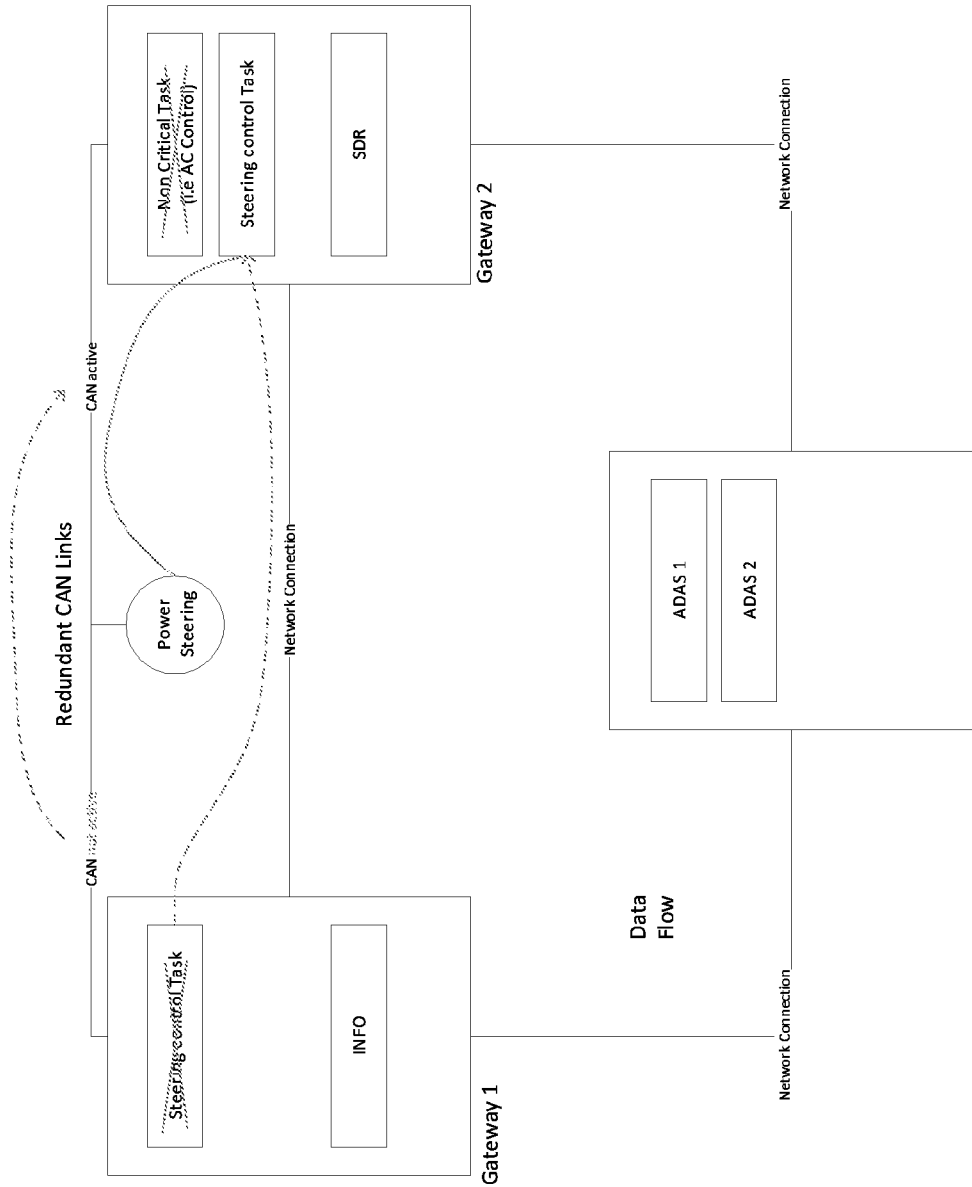


FIG. 16

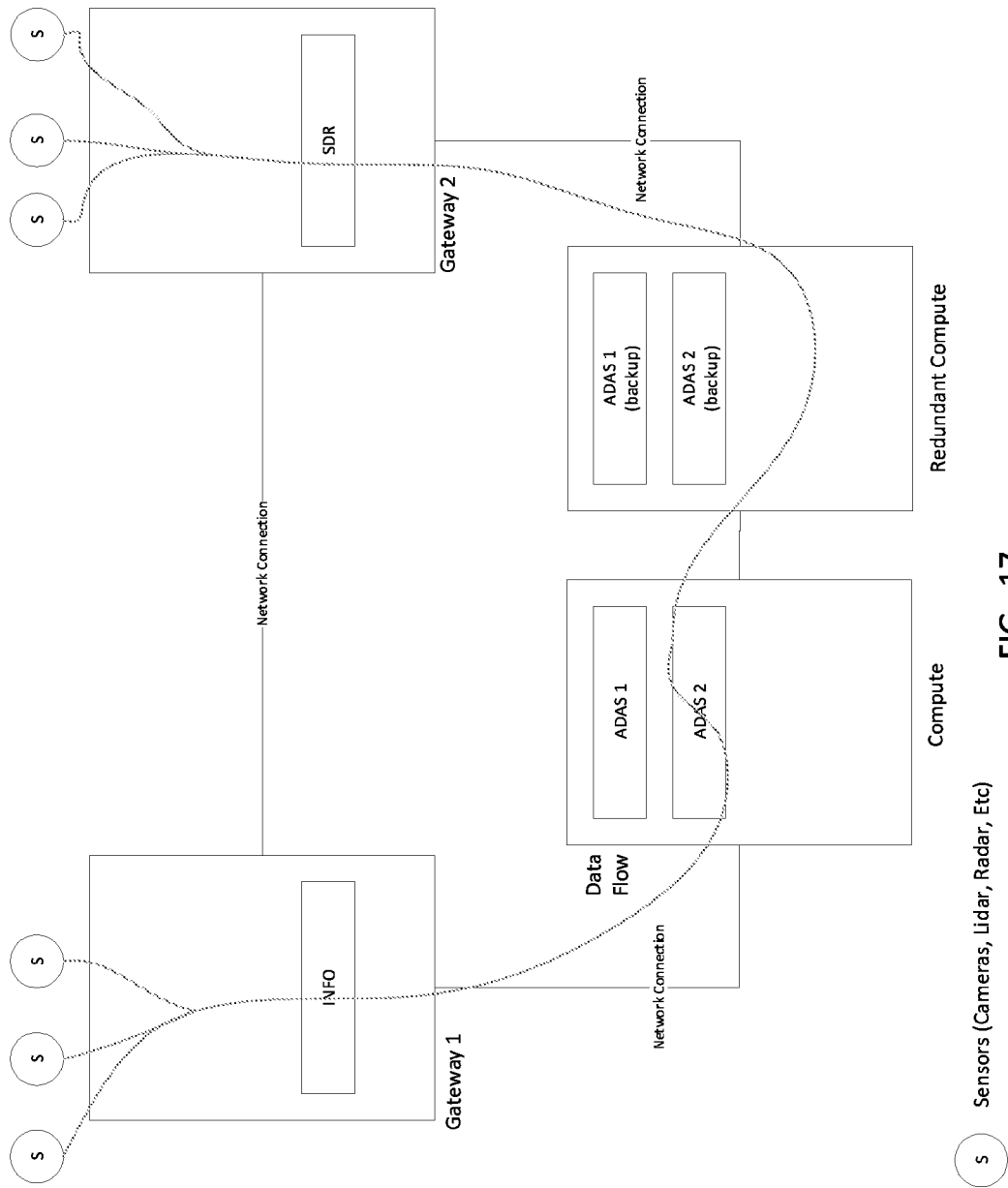


FIG. 17

INTERNATIONAL SEARCH REPORT

International application No

PCT/IB2021/056772

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06F11/14 G06F11/20

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	US 7 117 390 B1 (KLARER PAUL R [US] ET AL) 3 October 2006 (2006-10-03)	1-3, 5, 8-10, 12, 15-19, 23-26, 30
Y	abstract; figures 1, 2, 4 column 1, line 21 - column 4, line 27 ----- -/--	4, 6, 7, 11, 13, 14, 20-22, 27-29



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

28 March 2022

Date of mailing of the international search report

05/04/2022

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

Bozas, Ioannis

INTERNATIONAL SEARCH REPORT

International application No

PCT/IB2021/056772

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 2017/223532 A1 (SCHNEIDER ELECTRIC SYSTEMS USA INC [US]) 28 December 2017 (2017-12-28) abstract paragraph [0030] - paragraph [0034] paragraph [0063] - paragraph [0094] paragraph [0125] the whole document -----	4, 6, 7, 11, 13, 14, 20-22, 27-29
A	WO 2006/075332 A2 (QLUSTERS SOFTWARE ISRAEL LTD [IL]; MARKOVICH OFFER [IL] ET AL.) 20 July 2006 (2006-07-20) abstract; figures 1-3 page 1 - page 8 the whole document -----	1-30
A	WO 2019/094843 A1 (NVIDIA CORP [US]) 16 May 2019 (2019-05-16) abstract; figure 15 paragraph [0230] - paragraph [0283] the whole document -----	1-30

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/IB2021/056772

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 7117390	B1	03-10-2006	NONE
WO 2017223532	A1	28-12-2017	AU 2017280347 A1 07-02-2019
			AU 2017280350 A1 07-02-2019
			AU 2017280352 A1 07-02-2019
			CA 3028993 A1 28-12-2017
			CA 3028995 A1 28-12-2017
			CA 3029024 A1 28-12-2017
			CN 109643090 A 16-04-2019
			CN 109643264 A 16-04-2019
			CN 109937406 A 25-06-2019
			EP 3475773 A1 01-05-2019
			EP 3475826 A1 01-05-2019
			EP 3475827 A1 01-05-2019
			RU 2019101794 A 24-07-2020
			RU 2019101798 A 24-07-2020
			RU 2019101807 A 27-07-2020
			US 2019205180 A1 04-07-2019
			US 2019347171 A1 14-11-2019
			US 2019361753 A1 28-11-2019
			US 2021263783 A1 26-08-2021
			US 2021373969 A1 02-12-2021
			WO 2017223532 A1 28-12-2017
			WO 2017223535 A1 28-12-2017
			WO 2017223537 A1 28-12-2017
WO 2006075332	A2	20-07-2006	NONE
WO 2019094843	A1	16-05-2019	CN 111587407 A 25-08-2020
			EP 3707572 A1 16-09-2020
			JP 2021508863 A 11-03-2021
			US 2019258251 A1 22-08-2019
			WO 2019094843 A1 16-05-2019