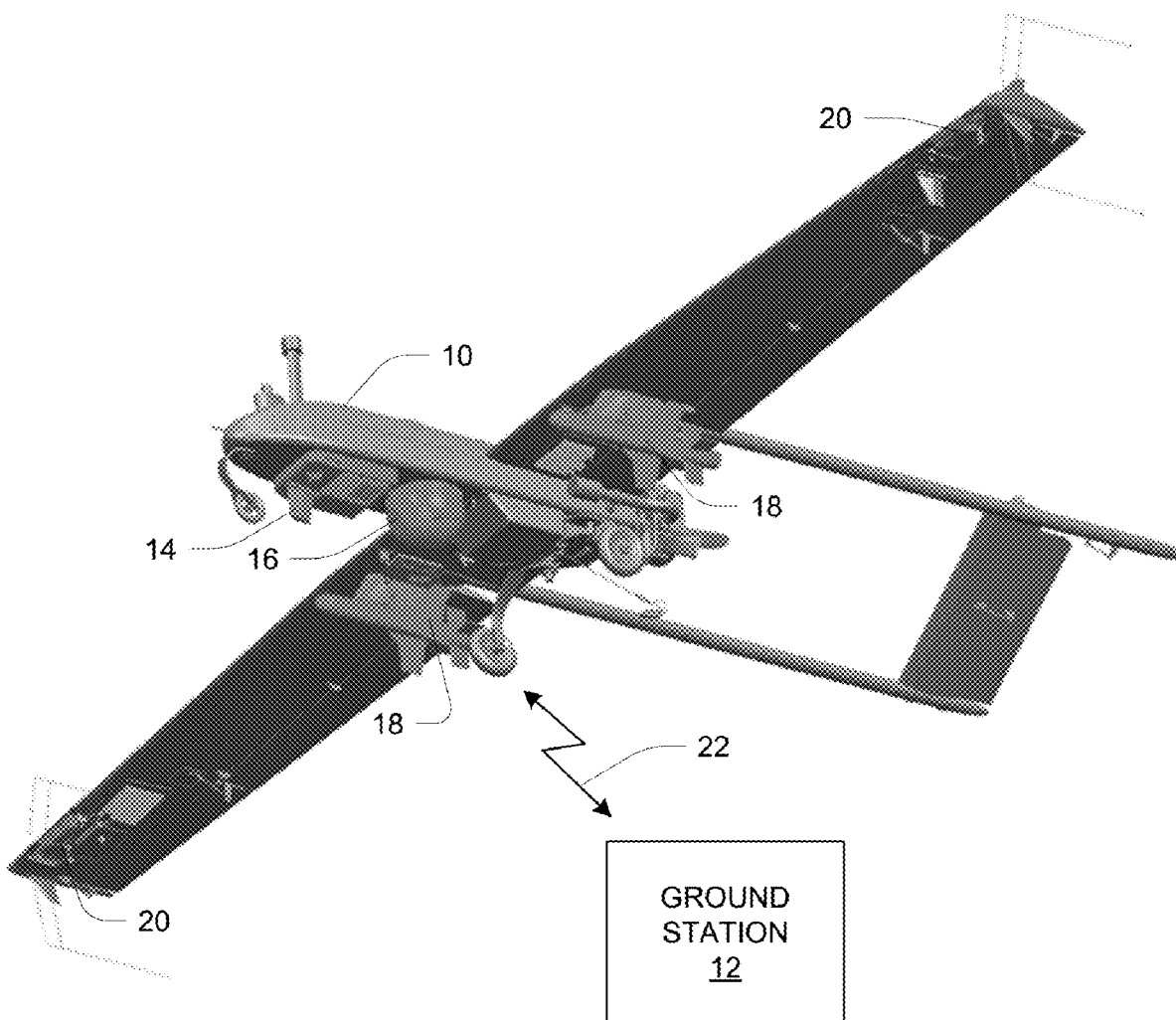




US 20230244232A1

(19) **United States**(12) **Patent Application Publication** (10) **Pub. No.: US 2023/0244232 A1**
Maust et al. (43) **Pub. Date: Aug. 3, 2023**(54) **PAYLOAD INTERFACE MODULE FOR
REMOTELY PILOTED VEHICLES**(52) **U.S. Cl.**
CPC **G05D 1/0094** (2013.01); **G05D 1/0022**
(2013.01)(71) Applicant: **Textron Systems Corporation**, Hunt
Valley, MD (US)(72) Inventors: **Urbane Dwane Maust**, Bel Air, MD
(US); **Mark Andrew Popowych**,
Kingsville, MD (US); **Christopher
David Bevard**, Cockeysville, MD (US)(21) Appl. No.: **18/103,000**(22) Filed: **Jan. 30, 2023****Related U.S. Application Data**(60) Provisional application No. 63/304,226, filed on Jan.
28, 2022.**Publication Classification**(51) **Int. Cl.**
G05D 1/00 (2006.01)(57) **ABSTRACT**

In a remotely piloted vehicle system, a ground station has a payload controller (PC), ground data terminal (GDT), and first payload interface module (PIM) having a first interface of a payload-specific type to the PC and a first network interface to the GDT, and the first PIM converts between signals of the first interface and corresponding messages of the first network interface. Aa remotely piloted vehicle has a payload (PL), a vehicle data terminal (VDT), and a second PIM having a second interface of the payload-specific type to the PL and a second network interface to the VDT. The VDT is communicatively coupled to the GDT for transfer of messages, and the second PIM converts between messages of the second network interface and corresponding signals of the second interface. The PIMs form respective endpoints of a ground-to-vehicle channel between the PC and PL.



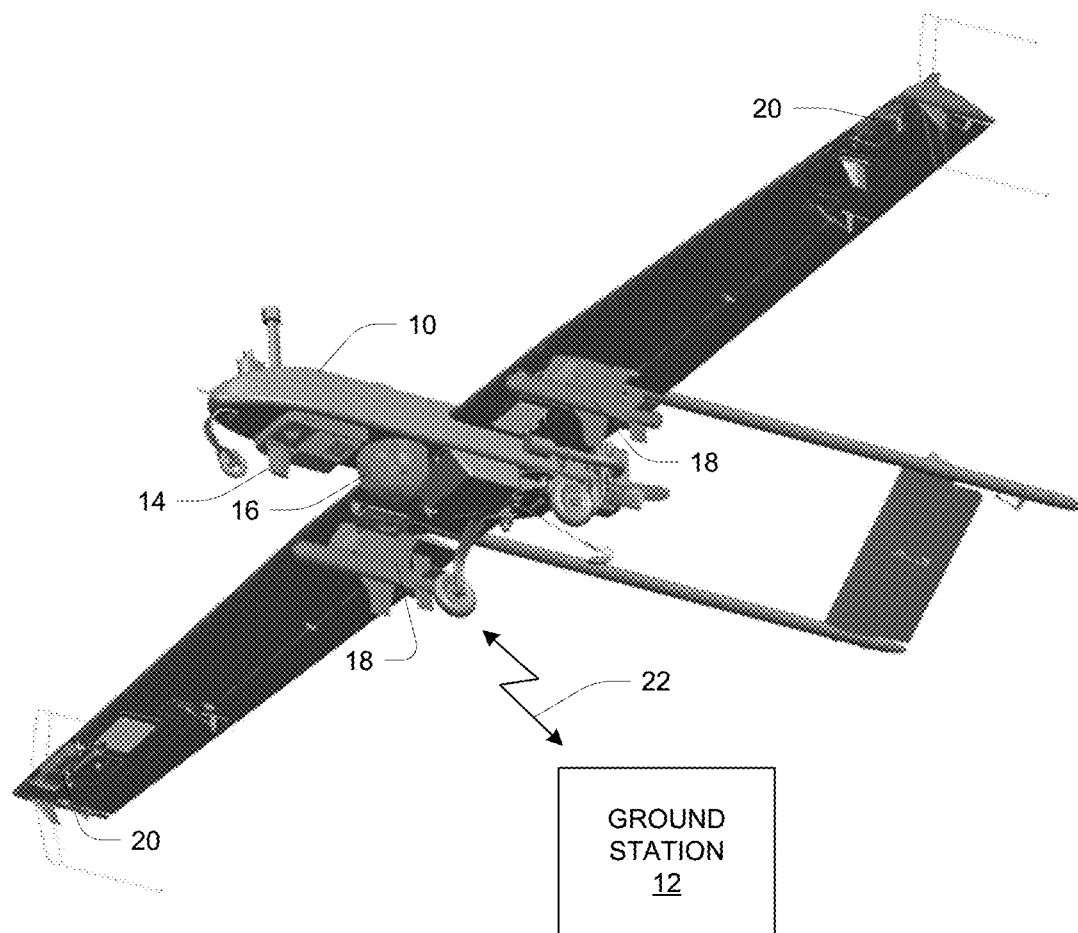


Fig. 1

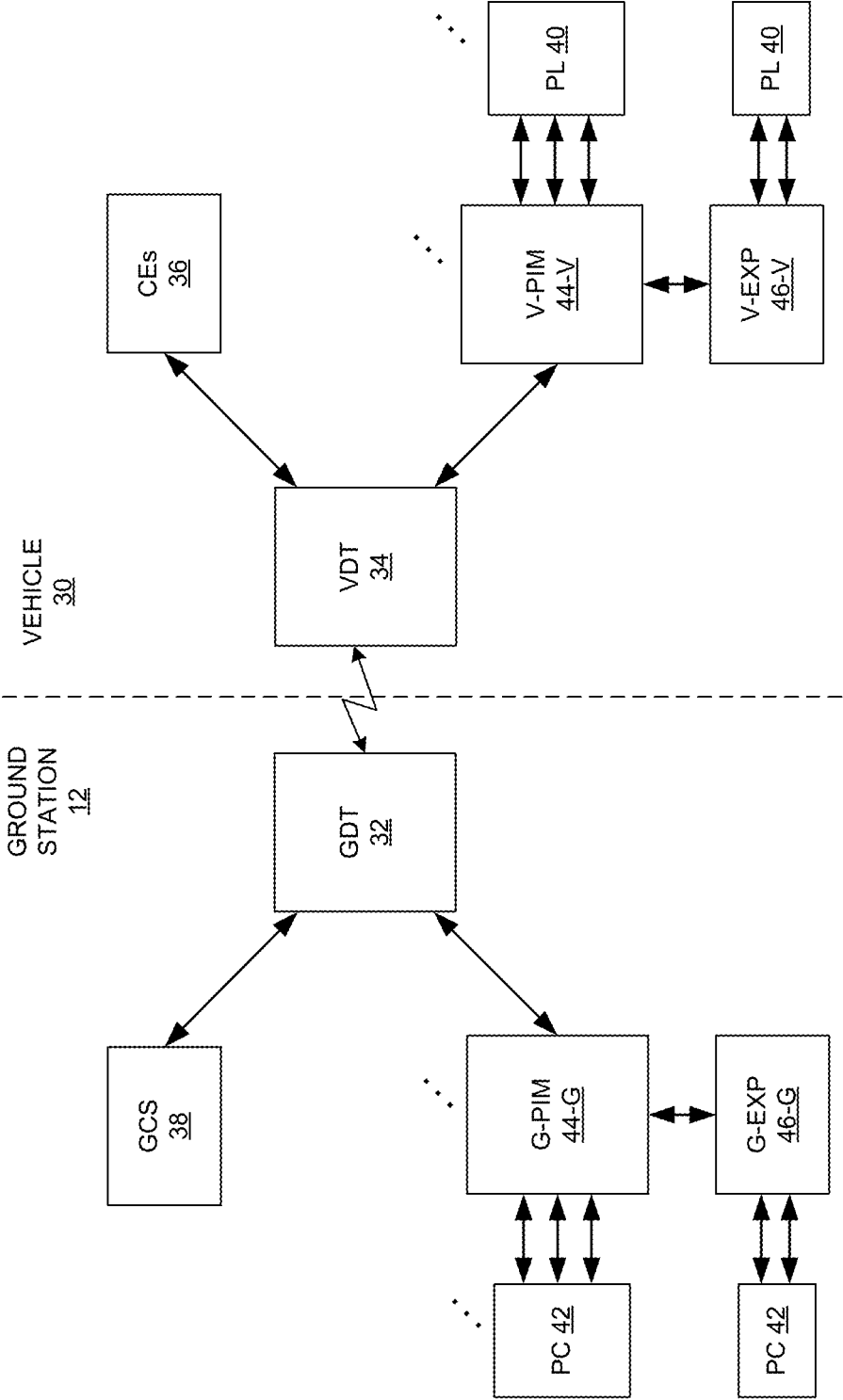


Fig. 2

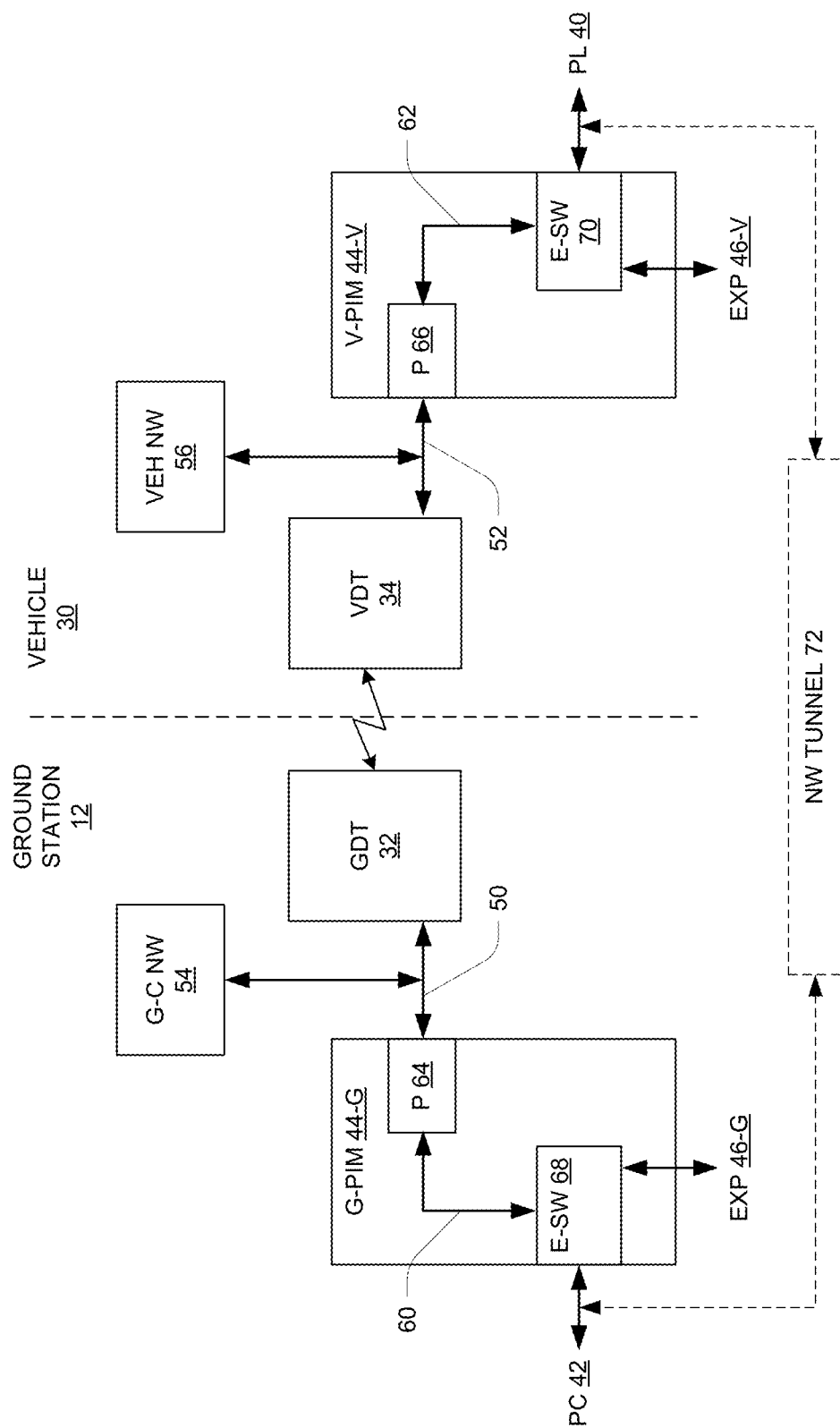


Fig. 3

Fig. 4

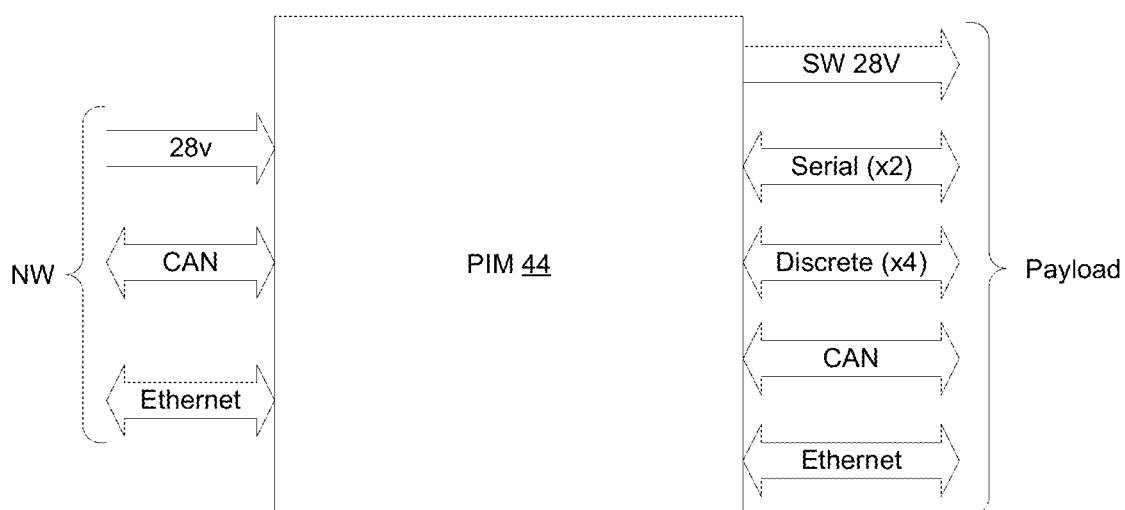
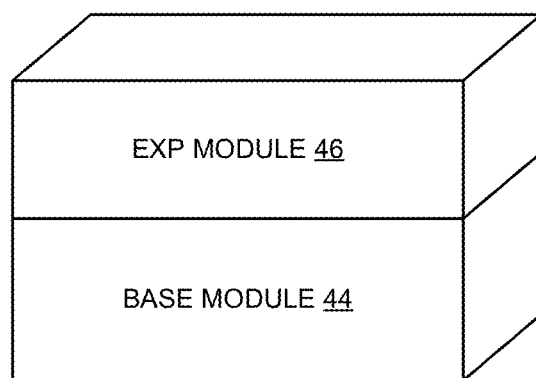


Fig. 5

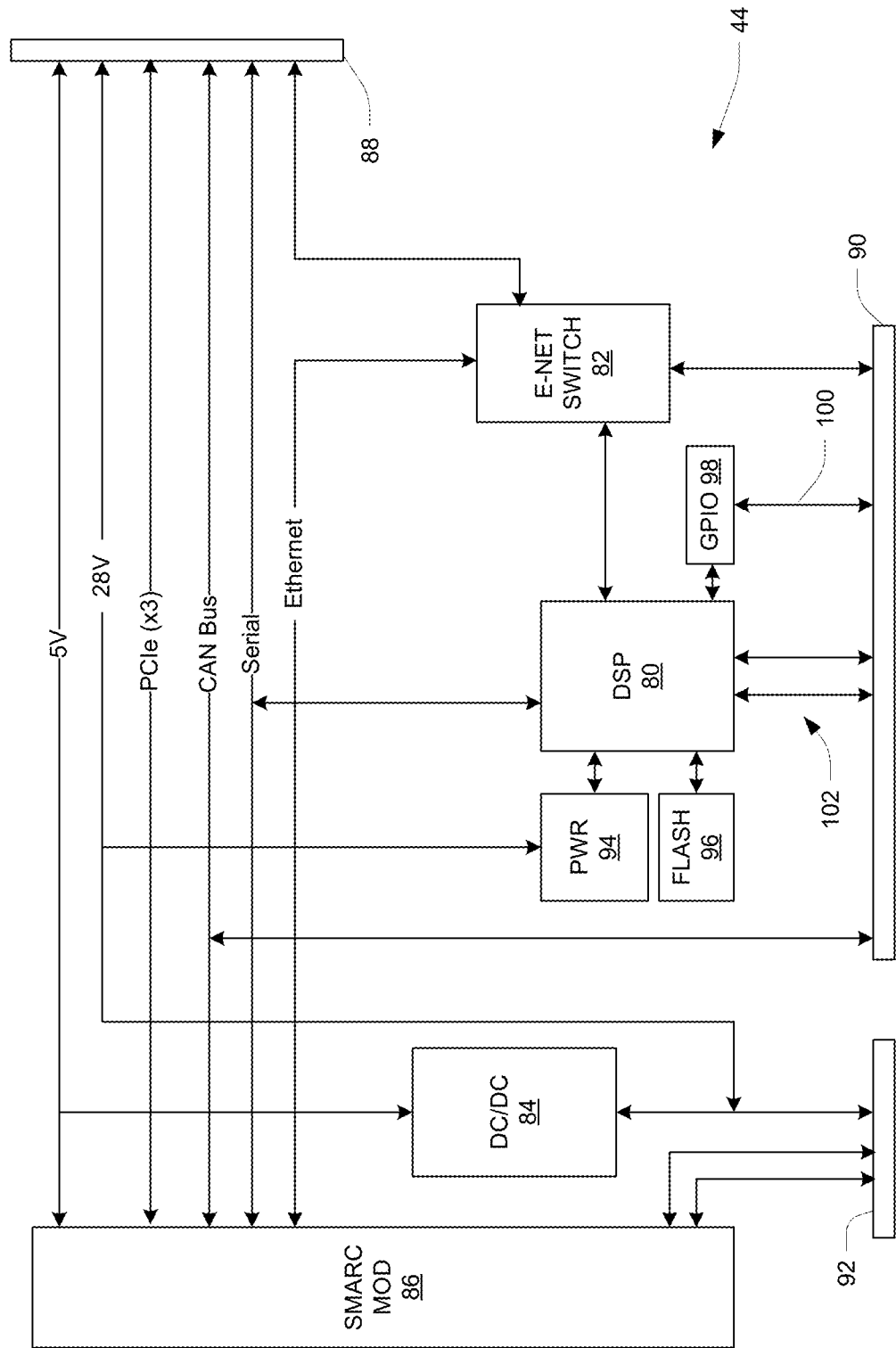


Fig. 6

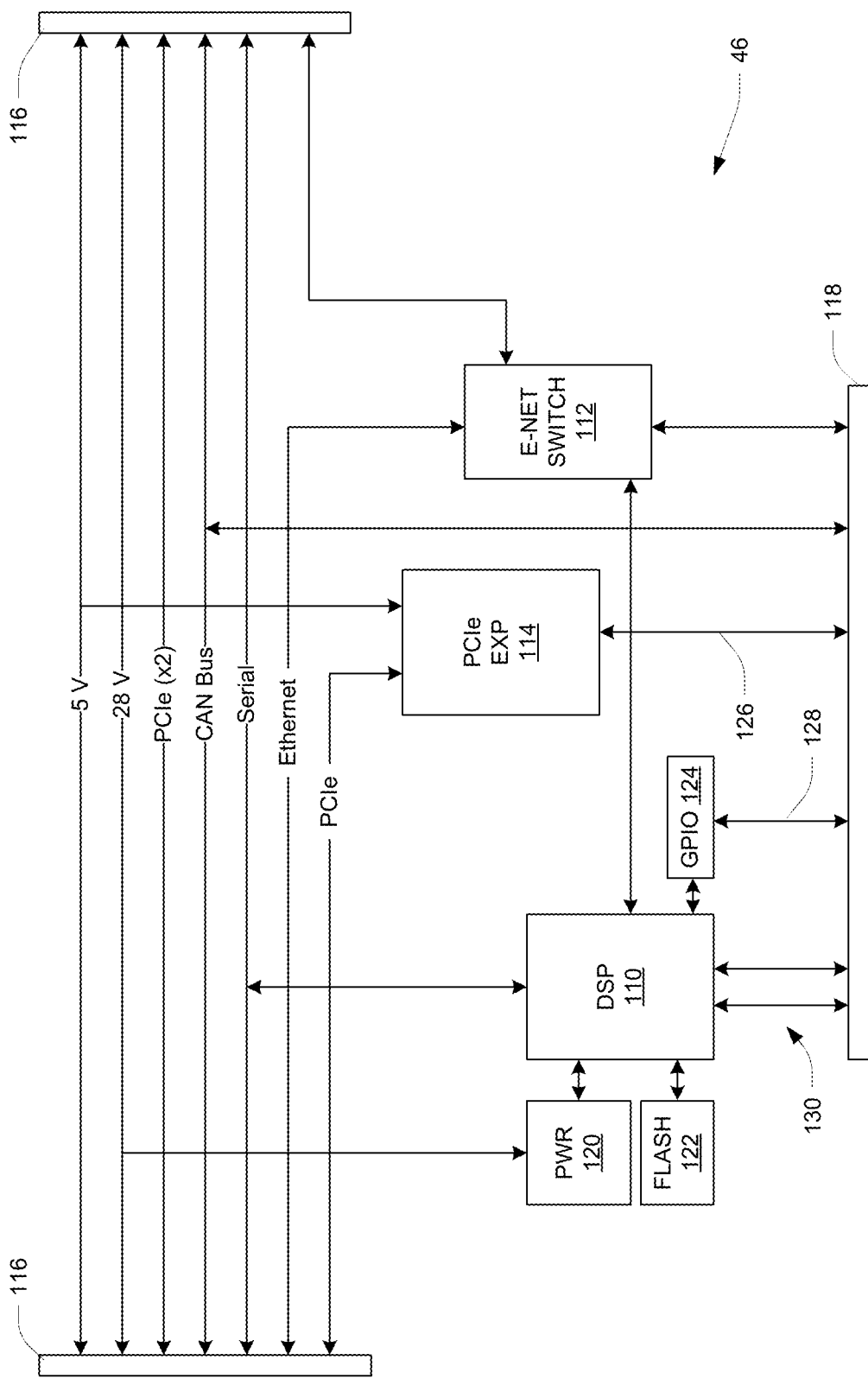


Fig. 7

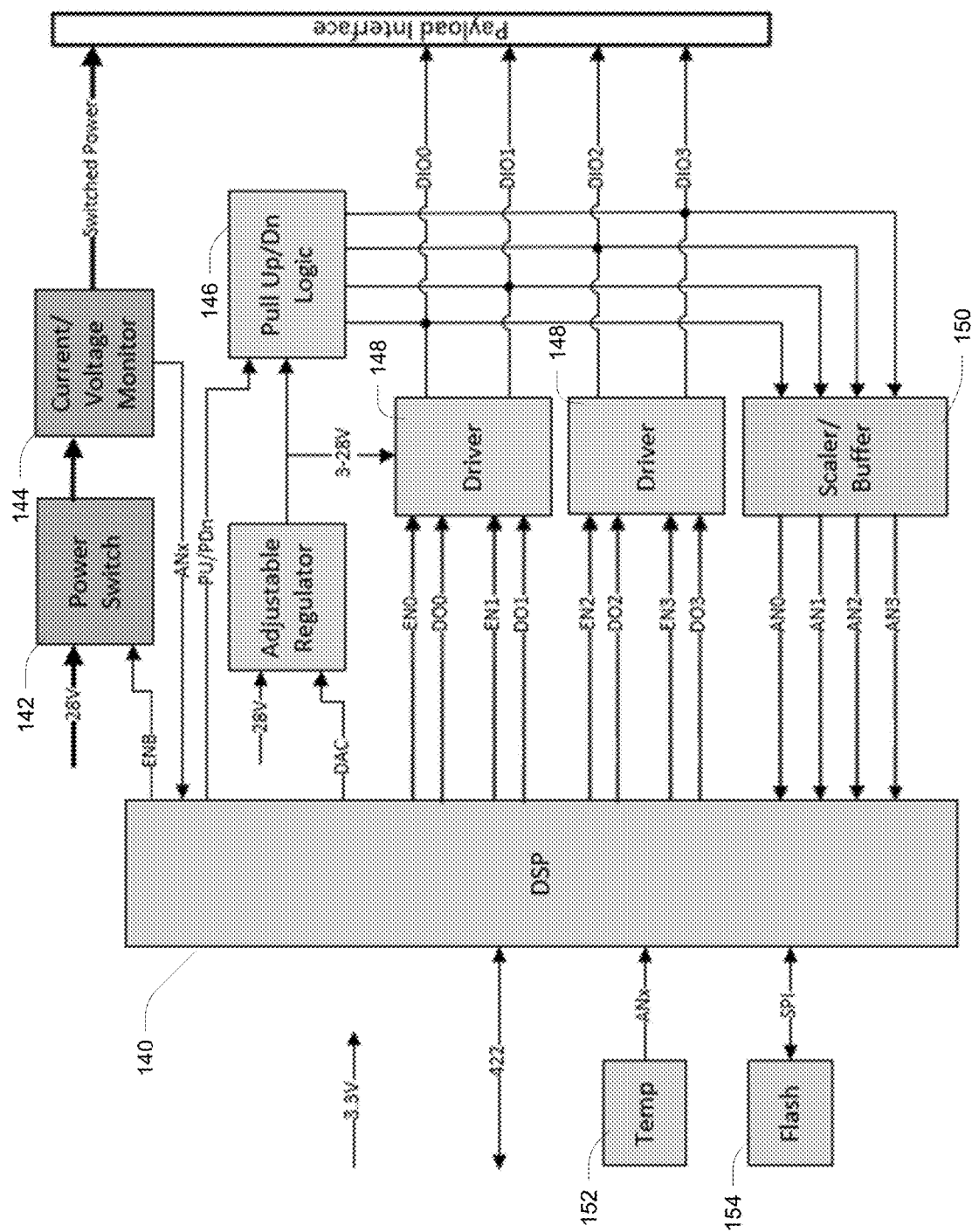
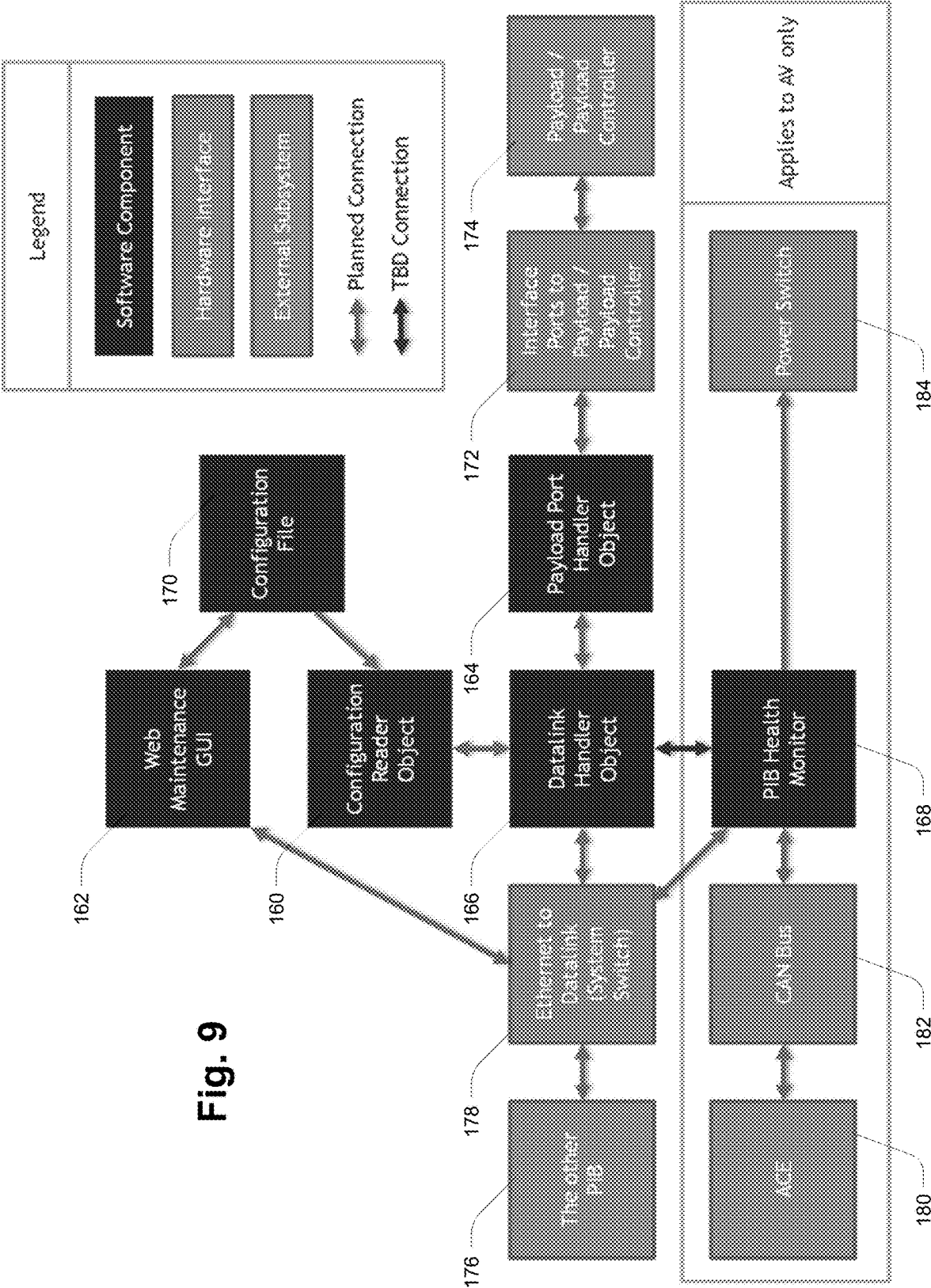


Fig. 8



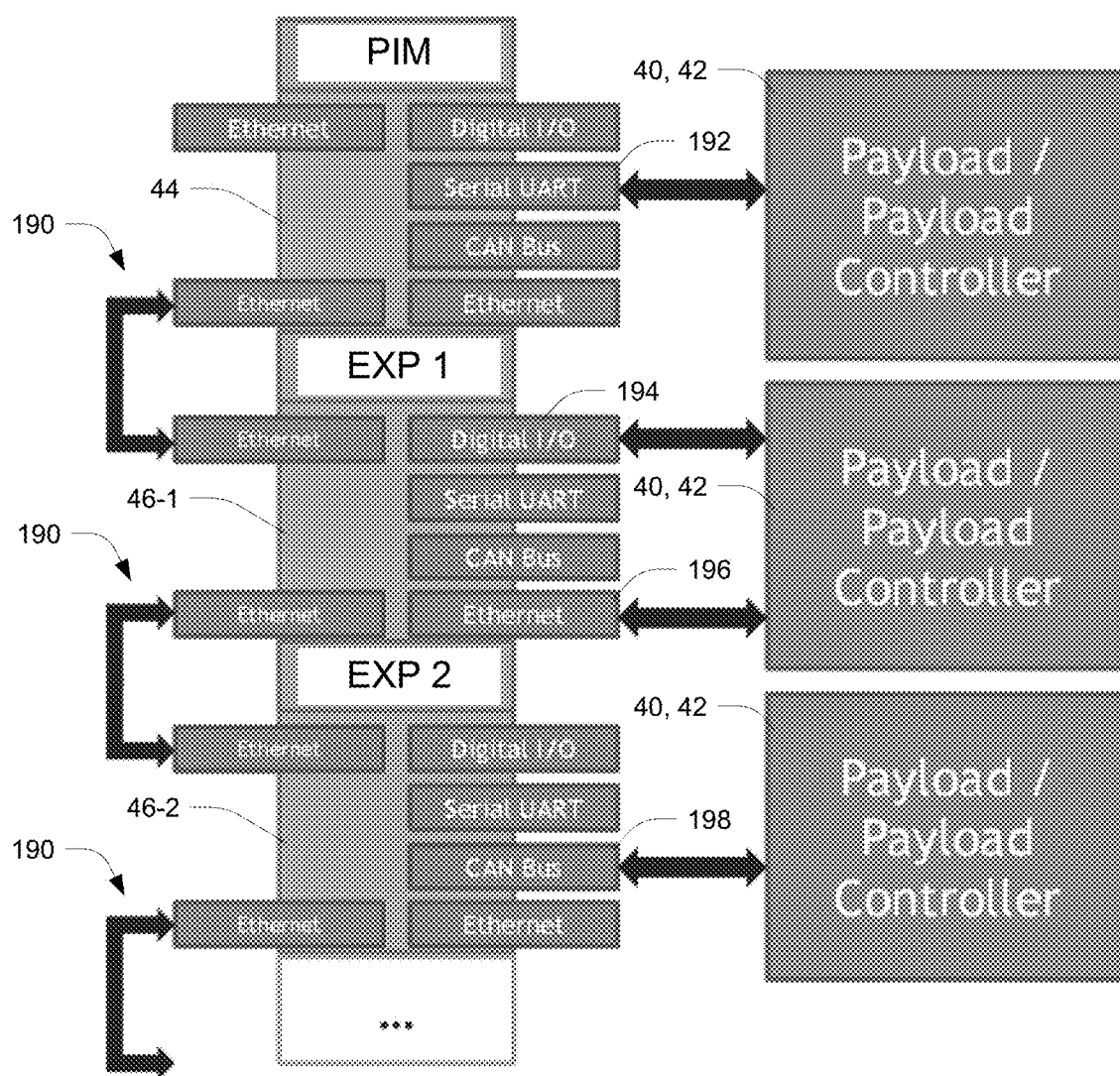


Fig. 10

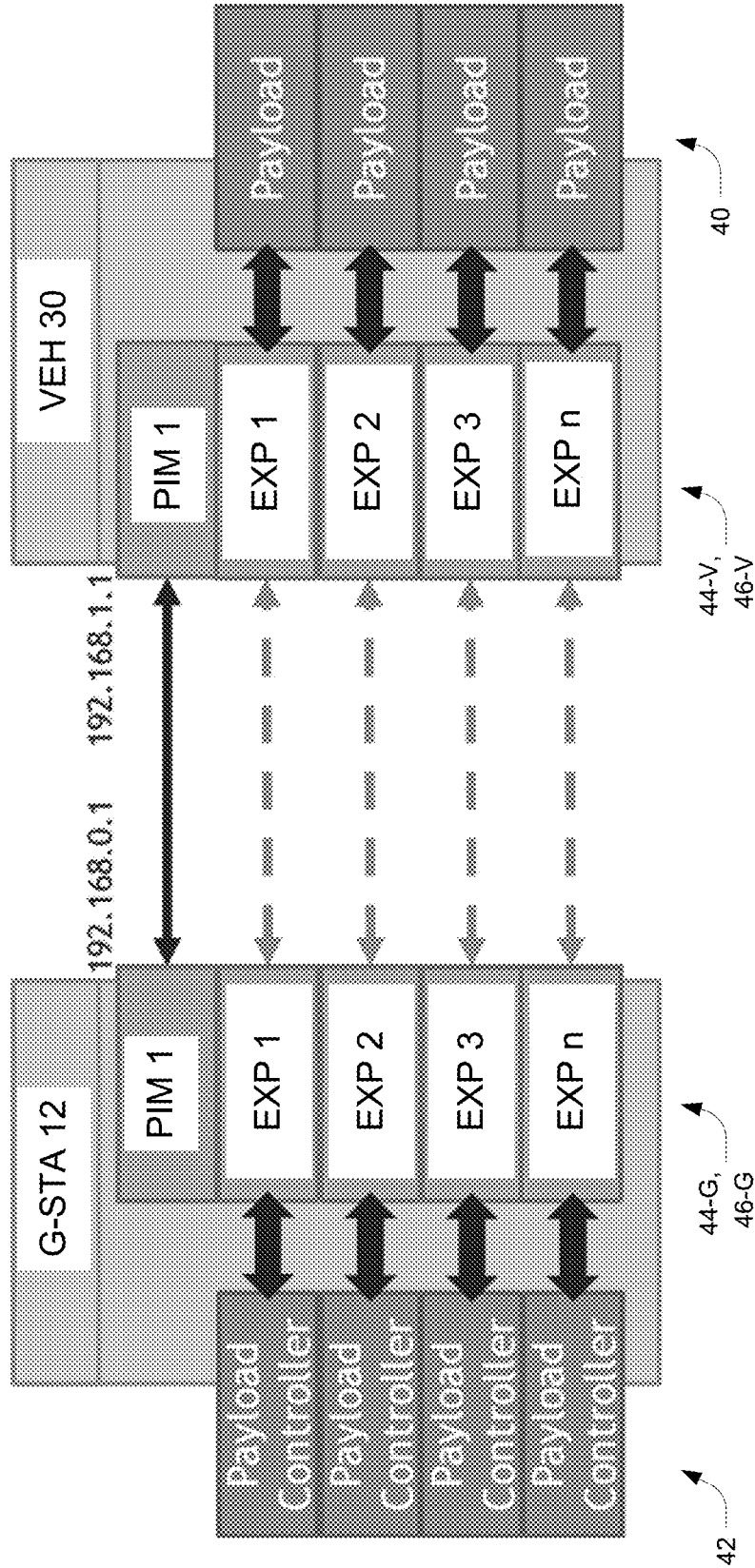


Fig. 11

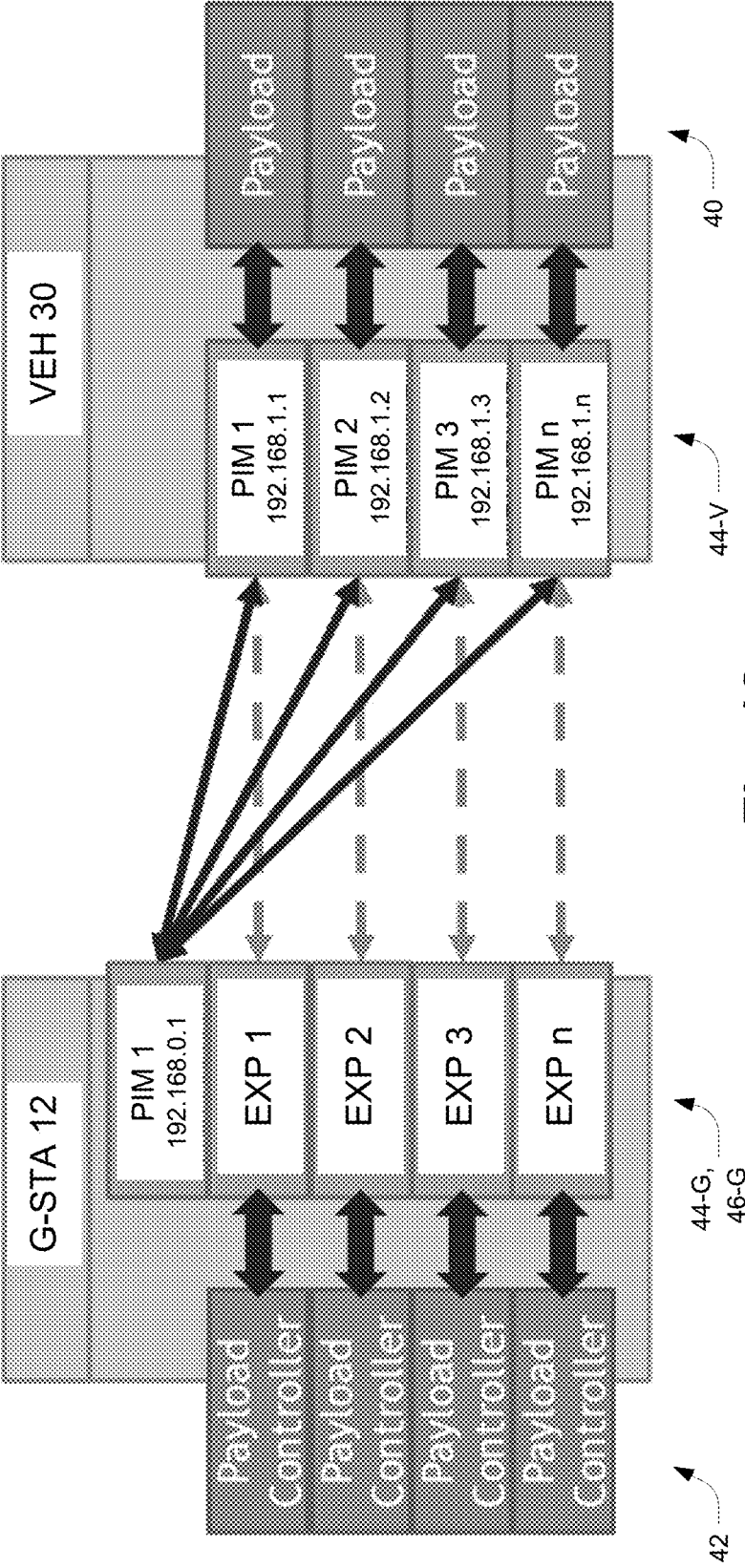


Fig. 12

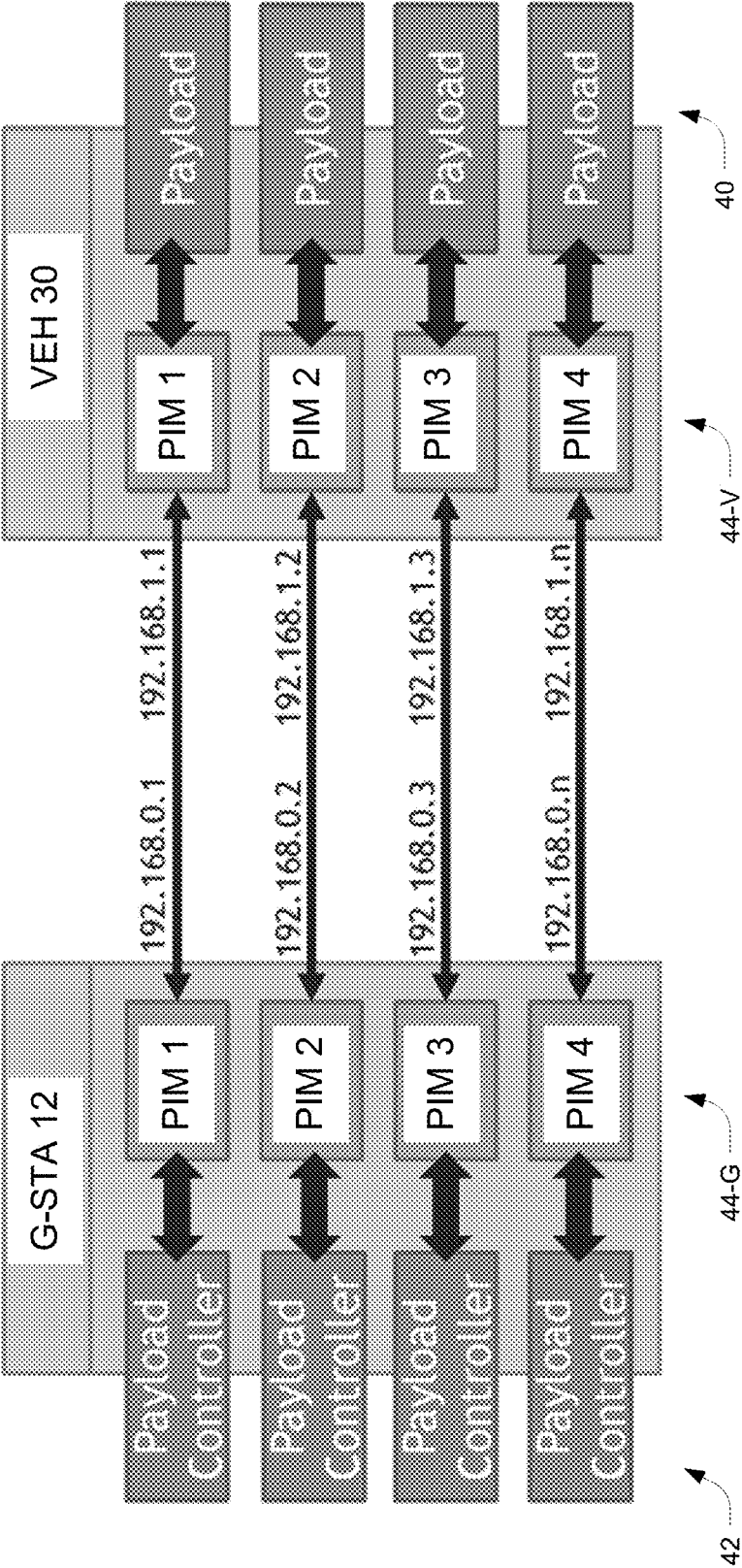


Fig. 13

PAYLOAD INTERFACE MODULE FOR REMOTELY PILOTED VEHICLES

BACKGROUND

[0001] This invention is generally in the field of remotely piloted vehicles (RPVs) such as unmanned aerial systems (UAS), and in particular it relates to remote control of and communications with payloads carried by an RPV.

SUMMARY

[0002] A Payload Interface Module (PEW) is a part of a Universal Payload Architecture (UPA). It provides standard interfaces for certain payloads to be carried by a UAS, such as electro-optical and infrared (EO/IR) sensor payloads for example. In one embodiment, PIM interfaces can include:

[0003] 1. Power for the payload which is controlled and monitored by the UAS

[0004] 2. Ethernet

[0005] 3. Serial ports (e.g., one RS-232, one RS-422)

[0006] 4. Discrete I/O pins—Each pin is generally monitored as an input and can be individually assigned as an output. Drive voltage for outputs may be programmable.

[0007] In the past, payloads such as EO/IR payloads carried by a UAS have been tightly integrated with the UAS, requiring long development times and purpose-built interfaces for each payload. According to UPA, a standard set of interfaces are provided that satisfy the needs of various payload types that are identified as potential candidates to be carried by the UAS.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The foregoing and other objects, features and advantages will be apparent from the following description of particular embodiments of the invention, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views.

[0009] FIG. 1 is an isometric underside view of an unmanned aircraft system (UAS);

[0010] FIG. 2 is a functional block diagram of a ground-to-vehicle communications and control system employing payload interface modules (PIMs);

[0011] FIG. 3 is a functional block diagram of the system of FIG. 2 highlighting network-related structure and functionality;

[0012] FIG. 4 is an isometric view of example housings for a PIM and associated expansion module;

[0013] FIG. 5 is a schematic illustration of PIM interfaces;

[0014] FIG. 6 is a detailed schematic diagram of a PIM according to one embodiment;

[0015] FIG. 7 is a detailed schematic diagram of an expansion module according to one embodiment;

[0016] FIG. 8 is a block diagram of general-purpose input/output (GPIO) aspects of a PIM according to one embodiment;

[0017] FIG. 9 is a block diagram illustrating a software organization of a PIM according to one embodiment.

[0018] FIG. 10 is a schematic illustration of scalability using a PIM and associated expansion modules;

[0019] FIGS. 11-13 are schematic illustrations of different communication models that may be supported in various embodiments.

DETAILED DESCRIPTION

Overview

[0020] A payload interface module (PIM, also referred to as “payload interface box” or PIB) as disclosed herein can provide an interface to variety of payloads, including for example an EO/IR payload, communications radios, radar (e.g., synthetic aperture radar for mapping), electronic warfare (EW) gear, and weapons. A PIM onboard a UAS is paired with a corresponding PIM on the ground, enabling customized remoting of control functionality.

[0021] In one embodiment, the PIM is architected with a DSP microcontroller (DSP) providing low level processing to manage direct communication with hardware elements that provide interfaces between the UAS system and a corresponding payload and its controller. A power switch (e.g., high-side switch) controls power to the payload, with the DSP making measurements of voltage and current. Power can be commanded off by the DSP if an overcurrent condition is detected. The DSP also controls and virtualizes communications with the payload through the serial ports and discrete I/O pins. These virtualized communications occur between a vehicle-borne PIM and a paired PIM in a ground station, making them available to the native payload controller for that payload. The UAS system may utilize a routed ethernet interface between the air and ground segments. The paired air and ground PIMs create an infrastructure that simplifies the implementation of payloads using ethernet, by making an ethernet port at the payload appear as if it is on the same subnet as an ethernet port at the payload controller on the ground, which is done using an ethernet internet protocol (IP) tunnel through that routed network. Bandwidth usage of the ethernet connection between the payload and its controller may be monitored and can be throttled if necessary.

[0022] The PIM architecture has also been designed to be expandable. Power and data interfaces needed to add more, or different types of payload interface channels are designed in allowing for additional flexible for future capabilities.

[0023] Prior UAS systems have sought to solve the problem of long integration times for new payloads. In one example, a set of interfaces have been predefined that would generically be available for new payload types. A PIM implementation within UPA as disclosed herein places a microprocessor running an operating system on top of a hardware architecture having generic interfaces. This allows the PIM to handle more complex network environments by leveraging interface support available in the operating system. The PIM architecture also supports the concept of providing not only an interface to the payload but also a designed in endpoint with the complement of that interface on the ground for a connection to the payload's native controller through the pairing of air and ground PIMs.

[0024] Thus, the following objectives are addressed by the disclosed improvements, generally directed to a distributed universal payload architecture in a remotely piloted vehicle such as an autonomous aircraft:

[0025] 1. Enables expansion of aircraft capabilities, beyond specialized reconnaissance & communication relay, creating a universal utility vehicle.

[0026] 2. Enables rapid integration of new payloads in response to real-time emerging threats.

[0027] 3. Enables rapid interchange of payloads in the field in response to varying mission sets.

Embodiments

[0028] FIG. 1 shows a remotely piloted vehicle, namely an unmanned aircraft or “drone” 10, carrying various payloads that are used during operation, along with a ground station 12 that is in communication with the vehicle 10 to provide command-and-control (C2) functionality, both for general flight control as well as remotely controlled use of the payloads. In this example, payload areas of the vehicle 10 include a chin pod payload bay 14, a main payload bay 16, two center-wing bays 18, and two outer wing bays 20. As described more below, at each location there are one or more functional devices, generally referred to as “payloads”, such as sensors (e.g., electro-optical (EO) or infrared (IR)), communications radios, radar (e.g., synthetic aperture radar for mapping), electronic warfare (EW) gear, weapons, etc. Both the vehicle 10 and ground station 12 employ specialized modules called “payload interface modules” (PIMs) to provide multi-layer interfacing of payload-specific components (controller, payload itself) to a ground-to-vehicle link 22 and distributed network structure for remote C2 functionality accordingly, as described in greater detail herein.

[0029] The following are general features of the payload C2 system employing distributed PIMs as described herein:

[0030] Electrical Power: Discretely Controllable Electrical Power To Various Payload Bays

[0031] Communications—Payload C2 & Data Transmission Within Air Vehicle And Between Air & Ground

[0032] Ground Control—Payload C2 & Data Transmission On The Ground

[0033] In this description, the more general term “vehicle” is used to refer to a remotely piloted vehicle, such as a UAS 10, employing a PIM-based payload architecture as described herein. The disclosed techniques have more general applicability to other types of remotely piloted vehicles including for example water-borne or land-borne vehicles.

[0034] FIG. 2 is a functional block diagram of the overall system, showing components of the ground station 12 as well as the remotely piloted vehicle 30 (e.g., UAS 10). Ground-to-vehicle communication (e.g., radio communication) is provided by a ground data terminal (GDT) 32 and vehicle data terminal (VDT) 34. The vehicle 30 generally includes various controlled elements (CEs) 36 such as flight control elements, position and environment sensors, etc., for which a ground control subsystem (GCS) 38 provides C2 functionality. The payload subsystem includes vehicle-borne payloads (PLs) 40 and associated ground-based payload controllers (PCs) 42 (plurality indicated by ellipses). These are interfaced to the associated data terminal 34, 32 by respective PIMs shown as vehicle PIMs (V-PIM) 44-V and ground PIMs (G-PIM) 44-G, which as described below may have a common overall design and implementation and be customized to the respective ground/vehicle environment by configuration. In some embodiments the system may employ expansion modules V-EXP 46 (46-V, 46-G) to provide for local expansion to additional payloads 40 and controllers 42.

[0035] In this description, a local PL 40 or PC 42 connected to a PIM 44 or expansion module 46 is also referred to more generally as a “payload element”, reflecting the use of a common-design PIM at either the ground or vehicle end in a system. It will be understood that in this use, “payload element” refers to either a PL or a PC to which the PIM is connected.

[0036] FIG. 3 illustrates certain network- and communication-related aspects of the system. The ground system 12 employs a ground system network 50 while the vehicle 30 employs a vehicle network 52, both of which may be realized for example as TCP/IP networks on underlying connection hardware such as Ethernet® etc. The ground network 50 is connected to one or more PIMs 44-G and functionally separate ground-control network (G-C NW) elements 54, and similarly the vehicle network 52 is connected to one or more PIMs 44-V and functionally separate vehicle network (VEH NW) elements 56. Each PIM 44 includes a respective internal network 60, 62, a respective port (P) 64, 66 connected to the respective network 50, 52, and a respective Ethernet switch (E-SW) 68, 70 providing connections to an associated PL 40 or PC 42 as well as to any local expansion module 46 if present. As described herein, the PIM-related structure and functionality provide a virtualized end-to-end connection between payload 40 and controller 42, indicated in FIG. 3 as a network tunnel 72.

[0037] The following are general features that may be realized by or using a PIM 44 as disclosed herein:

[0038] 1. The PIM design creates an end-to-end duplication of each of the interfaces present at a payload 40 on a paired PIM 44 at a corresponding payload controller 42, allowing for a non-integrated payload to be operated by its payload controller without additional configuration or software within the unmanned system. Previous solutions have required specific and expensive system integration to make those interfaces available to a payload controller.

[0039] 2. The PIM design includes computing hardware capable of supporting an Operating System (e.g., Linux, Windows).

[0040] Leverages software support available natively from the Operating System

[0041] Enables the support of additional types of interfaces that are not suitable for remote use via signal virtualization (e.g., USB)

[0042] 3. The PIM can be designed with an expansion port, allowing additional (e.g., up to 2 or 4) payload interfaces to be added without the need to duplicate the processor portion of the PIM. Additional types of interfaces can be implemented in each expansion payload interface. Payload power is provided for each additional interface. The PIM software is designed to handle the complexity of pairing payload and payload controller PIMs when the use of expansion interfaces is introduced.

[0043] 4. The PIM ethernet network is built using a processor containing two ports to allow separation between the payload network and remaining vehicle network components 56. This means that network routing within the remote vehicle network 52 only needs to be established when the PIMs are implemented into the system. When payloads are introduced the payload network is independent from the vehicle and ground control networks 52, 50. The payload networks can be realized to appear as if they are on the same subnet using an IP tunnel established between the PIMs through the vehicle and ground control networks.

[0044] FIG. 4 is a simplified view of an overall mechanical implementation having both the PIM (base module) 44 and expansion module 46 in stackable box-shaped housings. In one embodiment, the housings have a length/width footprint of 4" by 3", and heights in the range of 1 to 2 inches.

[0045] FIG. 5 illustrates external interfaces of a PIM 44, shown as network-side (NW), for connection to the ground

network **50** or vehicle network **52** location, and payload-side, for connection to a payload **40** or payload controller **42**. In the illustrated embodiment, the NW-side connections include a power connection to a 28-volt bus (28V), a CANBUS connection (CAN, controller area network), and an Ethernet connection. The payload-side connections include a switch power connection (Switched 28V), 2 serial buses, 4 discrete I/O connections, a CANBUS connection and an Ethernet connection.

[0046] As described in more detail below, in one embodiment these may be the overall capabilities of a base module **44** and expansion module **46**:

[0047] Base Module

[0048] Processor and interface to AV

[0049] Interface to single Payload

[0050] Interface addressing and virtualization.

[0051] Activity monitoring and logging

[0052] Expansion Module(s)

[0053] Interface to 2nd Payload in AV

[0054] Interface to Control Stations on ground.

[0055] Expandable to additional Payloads

[0056] Future addition of interface types

[0057] The following are additional hardware/electrical aspects in various embodiments:

[0058] I. Network interface module for control of a payload or payloads

[0059] Scalable with one processor able to control levels of I/O

[0060] Designed to handle distributed or concentric networking

[0061] Low size, weight, power

[0062] Supports quick turn development

[0063] II. Example hardware specifics:

[0064] Interfaces to include a base set: 10/100 Ethernet

[0065] Serial: (1) CAN Bus 2.0, (1) RS-232, (1) RS-422

[0066] One controllable power channel with programmable trip and UV monitoring

[0067] Processing to allow an OS

[0068] Mil-STD-704 power (28 VDC)

[0069] Mil-STD-461E (CE101/2, RE101/2, RS101/3)

[0070] Mil-STD-810F (assorted)

[0071] D Connectors for initial application (other options also being explored)

[0072] FIG. 6 is a schematic diagram of a base PIM **44** according to one embodiment. It includes a digital signal processor (DSP) **80**, Ethernet (E-NET) switch **82**, DC-DC converter **84** and a daughter module shown as a "Smart Mobility Card" (SMARC) module **86**. It also includes a set of connectors for external connections including an expansion connector **88**, a payload connector **90**, and an avionics connector **92**. Associated with the DSP **80** are power circuitry (PWR) **94**, Flash memory **96**, and general-purpose I/O (GPIO) circuitry **98**, which is described in some detail below with reference to FIG. 8. The PIM **44** carries and terminates a variety of signals including a 5-volt bus 5V, a 28-volt bus 28V, three PCI-Express (PCIe) buses, a CAN bus, one or more serial buses, and four Ethernet buses all terminated at the Ethernet switch **82**. DSP-associated signals further include Data signals **100** and additional serial bus signals **102**.

[0073] FIG. 7 is a schematic diagram of an expansion module **46** according to one embodiment. It includes a digital signal processor (DSP) **110**, Ethernet (E-NET) switch **112**, and a PCI expansion (EXP) module **114**. It also includes

a set of connectors for external connections including two expansion connectors **116** and a payload connector **118**. Associated with the DSP **110** are power circuitry (PWR) **120**, Flash memory **122**, and general-purpose I/O (GPIO) circuitry **124**, which is described in some detail below with reference to FIG. 8. The expansion module **46** carries and terminates a variety of signals including a 5-volt bus 5V, a 28-volt bus 28V, three PCI-Express (PCIe) buses, a CAN bus, one or more serial buses, and four Ethernet buses all terminated at the Ethernet switch **112**. Additional signals include function signals **126**, Data signals **128**, and additional serial bus signals **130**.

[0074] Below are circuitry/features that may be included in an expansion module **46** in various embodiments:

[0075] High-power switch(es) using suitable connector arrangements.

[0076] Coaxial/RF/Signal switches or multiplexers

[0077] Pulse/frequency counters

[0078] Programmable resistance/loads

[0079] Signal conditioners for accelerometers, strain gauges, pressure, or temperature sensors

[0080] Memory modules (flash ram, SD slots)

[0081] PWM drivers for servo control

[0082] Current loop outputs/sensors

[0083] Specialty Interfaces

[0084] Pattern Generators

[0085] FIG. 8 is a block diagram showing general-purpose I/O (GPIO) architecture features of a PIM **44**. A local DSP **140** controls functionality. A power switch **142** with current and voltage monitoring **144** are included, with programmable trip characteristics. There are also four discrete IO channels DI00-DI03 (may be expandable) for which the drive level may be programmable (LVTTTL thru supply level). Drivers **146** use totem pole output, driven directly by DSP **140**. Pull-ups/downs **148** are used and may be enabled per channel. Direct analog measurements ANO0-ANO3 of outputs via scaler/buffer **150**. Also included are onboard temperature measurement (TEMP) **152** and Flash memory **154** for logging.

[0086] The power switch **142** and monitoring circuitry **144** may feature the following:

[0087] High Side Switch

[0088] Voltage and Current Monitoring

[0089] Overload Protection (current limit, short circuit)

[0090] Overvoltage protection

[0091] Thermal shutdown with restart

[0092] Diagnostic Feedback

[0093] FIG. 9 is a block diagram of key software-implemented elements and their relation to other system elements. The following are high-level objectives and characteristics of the software system:

[0094] Focus on abstracting and generalizing the controller to payload connection.

[0095] Each PIM **44** supports a set of interfaces connected to either a Payload **40** or a Payload Controller **42**.

[0096] PIM software scales with hardware.

[0097] Each PIM **44** converts payload/controller messages to user datagram protocol (UDP) which are transmitted to complementary PIM **44** which converts the message back.

[0098] Primary Datalink signal.

[0099] PIM datalink communication is bi-directional.

- [0100] PIMs depend on software configuration via a configuration file.
- [0101] Each PIM is configured with a unique static IP address.
- [0102] By default, configuration identifies the complementary (far-end) PIM.
- [0103] PIM is interchangeable—a PIM unit can be configured for use in either the ground station 12 or vehicle 30.
- [0104] PIM software monitors and reports network bandwidth usage and recommend action on excess use.
- [0105] PIM software monitors and controls power to the payload and report HW health.
- [0106] A PIM 44 runs an operating system, such as Ubuntu Linux for example, and application-level components including the following:
- [0107] Configuration Reader Object 160—Parse configuration file 170 and initialize the rest of the application and OS settings on start up.
- [0108] Configuration Maintenance Web GUI 162—Provides an interface to view and modify the configuration file 170 over a web browser. Payload Port Handler Object 164—Identifies, opens, listens, and sends on all the payload ports. Messages received on payload ports are handed over to the Datalink Handler Object. Messages received from the Datalink Handler Object 166 have the header removed and are sent out the appropriate payload port via port interface 172 and controller 174.
- [0109] Datalink Handler Object 166—Prepends a header to the message and sends out the Ethernet port to the complementary PIM 176 over the Primary Datalink 178 as a UDP packet according to configured communication scheme. Receives messages from the complementary PIM 176 and hands them over to the Payload Port Handler Object 164. Monitors the network traffic—how many payload message packets go out and come in and reports bandwidth usage statistics to the ground station 12.
- [0110] The prepended header contains info for the complementary PIM 176 including a spooler and which interface port the message should be sent out.
- [0111] PIM Health Monitor 168 (Vehicle Only)—Monitors temperature, power usage, any other sensors, writes log files, listens for payload power commands from the ACE 180 over the CAN bus 182. The vehicle 30 receives power commands over S200 from a ground VSM GUI and provides control to a local power switch 184.
- [0112] PIM application software may be written using C++. Application GUIs, such as the power control dialog, can be written for the DVSM using C++ and the Qt framework. Configuration file 170 can use XML, and the web configuration GUI 162 can use HTML.
- [0113] Hardware Abstraction:
- [0114] Each PIM 44 supports a set of interface ports 172 connected to either the local Payload or Payload Controller.
- [0115] The PIM software listens to these ports for data.
- [0116] Any data received is converted to UDP Ethernet packets which are sent to the complementary PIM 176. Bi-directional Communication is used.
- [0117] The complementary PIM 176 converts the data back and sends it out over the local same-type interface port that the data was received on.
- [0118] FIG. 10 illustrates scalability using expansion modules 46 connected in daisy-chain fashion to a PIM 44 to provide connection to multiple local payloads 40 or payload controllers 42. Adjacent modules are connected by respective Ethernet connections 190 for this purpose. In this example, the single PIM 44 interfaces to a local PL 40 or PC 42 via a serial transceiver
- [0119] (UART) 192. A first expansion module 46-1 interfaces to a local PL 40 or PC 42 via a digital I/O port 194 and Ethernet port 196. A second expansion module 46-2 interfaces to a local PL 40 or PC 42 via a CAN Bus 198.
- [0120] The PIM software scales with hardware I/O expansions. The OS detects I/O ports from expansions. In some embodiments, it may be preferable to increase the number of interface ports on one PIM rather than increase the number of PIMs 44 or expansion modules 46 in the system. A single larger PIM may be easier to manage but would generally consume more physical volume and thus may present placement issues, in which case an approach of using multiple smaller PIMs 44 and/or expansion modules 46 may be preferred.
- [0121] FIGS. 11-13 illustrate PIM-PIM communications according to different configurations which may be suitable for different types of deployments. Generally, each PIM 44 is logically mapped to a complementary PIM 44 on the opposite subsystem.
- [0122] FIG. 11 illustrates one-to-one communication between a PAIR of PIMs 44 using expansion modules 46. Interface ports and sub-slots are logically mapped between corresponding PIMs 44-G, 44-V of the ground station (G-STA) 12 and vehicle (VEH) 30, i.e., data received on sub-slot 2 on a ground PIM 44-G is directed to sub-slot 2 on the corresponding vehicle PIM 44-V.
- [0123] FIG. 12 illustrates one-to-many communication, using one PIM 44-G and expansion modules 46-G on the ground side and a set of PIMs 44-V on the vehicle side. Each slot, or PIM 44-V, on the vehicle 30 maps to one slot, or the single PIM 44-G, at the ground station 12, while each sub-slot on the ground PIM 44-G maps to each slot on the vehicle 30. The ground PIM 44G knows how many vehicle PIMs 44-V it must talk to based on its sub-slot count.
- [0124] FIG. 13 illustrates many-to-many communication. A system can support multiple pairs of PIMs 44-V, 44-G to support multiple payloads simultaneously. Therefore, each PIM 44 is mapped to a single complementary PIM 44 on the opposite subsystem via configuration.
- [0125] The following are additional specific aspects of functionality that may be realized in some embodiments:
- [0126] 1. PIM Detection
- [0127] PIM may know who to talk to by software configuration.
- [0128] Possible hardware solutions for putting PIM in known initial and accessible state:
- [0129] Wire harness modification to provide each PIM with a default configuration based on connection.
- [0130] Physical externally accessible multi-position switch to set a PIM ID prior to installation.
- [0131] Alternative software possibilities
- [0132] Put PIMs into maintenance state via message over CAN from the ACE

- [0133] DHCP server on the vehicle, port-based address allocation. Assign one specific address per port.
- [0134] 2. Software Configuration
- [0135] Every PIM is configured with a slot and communication scheme.
- [0136] Slot—a logical slot doubling as an identifier.
- [0137] Provides the MI's static IP address.
- [0138] One PIM per slot
- [0139] Communication scheme—makes PIMs aware of how they are organized.
- [0140] Provides the complementary PIM's IP address.
- [0141] PIM assumes the complementary PIM is present and alive.
- [0142] Preset communication schemes:
- [0143] One-to-one (FIG. 11)—One big PIM on the GCS maps to one big PIM on the AV
- [0144] One-to-many (FIG. 12)—One big PIM on the GCS maps to multiple little PIMs on the AV and multiple little PIMs on the AV map to one big PIM on the GCS. Every Little PIM knows the slot of the Big PIM, which must be known (e.g., slot 1).
- [0145] Many-to-many (FIG. 13)—Multiple little PIMs on the GCS each map to one of multiple little PIMs on the AV.
- [0146] Other schemes (e.g., multiple Big PIIVs or a mix of Big and Little PIIVs) may also be used.
- [0147] 3. Communication Configuration
- [0148] Slots—Each PIM configured with a logical slot number which doubles as an identifier.
- [0149] Reserved pre-defined static IP address for each slot.
- [0150] Loosely correspond to physical location. Documentation maps logical slot to physical location.
- [0151] Example: Slot 2=left center wing and Slot 3=right center wing for a particular AV payload layout.
- [0152] Each PIM can only occupy one slot. Two PIN/Is in the same slot causes an IP conflict.
- [0153] sub-slot—A set of I/O ports within a slot. PIM SW determines this automatically based on number of I/O
- [0154] The base board of every PIM has one set of I/O ports, so each PIM has at least one sub-slot.
- [0155] Each expansion adds a new set of I/O ports, and a new sub-slot.
- [0156] Port—One individual payload interface port within a sub-slot. From the OS perspective, each port has number used for sub-slot association (e.g., eth0, eth1, eth2=sub-slot 0, 1, 2 respectively)
- [0157] For now, slots, sub-slots, and ports map by ID. That is, the PIM in slot 1 on the GCS communicates with the PIM in slot 1 on the AV and vice versa IAW pre-defined communication schemes. This can be changed by alternative custom communication rules.
- [0158] 4. Other Configuration
- [0159] System Host (ground station 12 or vehicle 30)
- [0160] Payload interface port specific configuration
- [0161] Override port identification, each port is identified by its sub-slot and type (e.g., sub-slot 2, Serial
- [0162] Ethernet—IP override, Port number override, UDP/TCP flag
- [0163] Serial—Baud rate, Parity, Start Bit, Stop Bit, Flow Control, List of Sync Bytes, Length of Packet
- [0164] Custom Communication Rules (e.g., user-created or user-centric)
- [0165] Future rulesets for specifically defining where messages should go, from slot/sub-slot/port to slot/sub-slot/port and under which conditions.
- [0166] May provide rules for specific payload messages based on message bytes.
- [0167] (e.g., A payload message beginning with OxAA, OxBB, OxCC is critical and receipt should be acknowledged by the complementary PIM, else retry some number of times)
- [0168] Web Configuration GUI
- [0169] Each PIM features a configuration GUI 162 which is accessible via web browser and known HTTPS port (e.g., 443).
- [0170] A configuration application may be used in the ground station 12 which queries all the PIMs across the entire system (vehicle 30 must be on with link) for their configuration which is displayed altogether on one interface for quick and easy multi-PIM configuration and verification.
- [0171] Automatically queries every possible PIM IP address.
- [0172] May also be ran from a maintenance laptop (i.e., connected to the SMC maintenance port)
- [0173] The central PIM configuration utility can use SFTP to copy each PIM's configuration file to a central location, read and apply modifications and then SFTP the file back to the PIM to apply configuration changes.
- [0174] While various embodiments of the invention have been particularly shown and described, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the scope of the invention as defined by the appended claims.
- What is claimed is:
1. A remotely piloted vehicle system, comprising:
 - a ground station having a payload controller, a ground data terminal, and a first payload interface module, the first payload interface module having a first interface of a payload-specific type to the payload controller and a first network interface to the ground data terminal, the first payload interface module being configured and operative to convert between signals of the first interface and corresponding messages of the first network interface; and
 - a remotely piloted vehicle having a payload, a vehicle data terminal, and a second payload interface module, the second payload interface module having a second interface of the payload-specific type to the payload and a second network interface to the vehicle data terminal of the vehicle, the vehicle data terminal being communicatively coupled to the ground data terminal for transfer of messages therebetween, the second payload interface module being configured and operative to convert between messages of the second network interface and corresponding signals of the second interface,

the payload interface modules forming respective endpoints of a ground-to-vehicle channel between the payload controller and the payload.

2. The remotely piloted vehicle system of claim 1, wherein each of the first and second interfaces is one of plurality of interfaces of distinct payload-specific types of the respective payload interface modules.

3. The remotely piloted vehicle system of claim 2, wherein the payload-specific types of the plurality of interfaces are serial bus, discrete digital I/O signal, and a second network interface.

4. The remotely piloted vehicle system of claim 1, wherein each of the payload interface modules further includes power circuitry and payload-side connections, the payload-side connections of the second payload interface being connected to the payload to provide switched power thereto.

5. The remotely piloted vehicle system of claim 1, wherein the ground station employs a ground network and the vehicle employs a vehicle network, the ground network being connected to the second payload interface module and to functionally separate ground-control network elements, the vehicle network being connected to the first payload interface module and to functionally separate vehicle network elements.

6. The remotely piloted vehicle of claim 5, wherein each of the payload interface modules includes a respective internal network, a respective port connected to the respective ground or vehicle network, and a respective Ethernet switch providing switched connections including to the respective payload or payload controller.

7. The remotely piloted vehicle system of claim 1, wherein the ground station and remotely piloted vehicle each contain a respective expansion module coupled to the respective payload interface module, each expansion module configured to provide an interface to a respective payload or payload controller respectively and form an endpoint of a second ground-to-vehicle channel between a second payload controller of the ground station and a second payload of the vehicle.

8. The remotely piloted vehicle system of claim 1, wherein each of the payload interface modules is housed in a respective housing for modular placement in proximity to the respective, separately housed, payload or payload controller.

9. The remotely piloted vehicle system of claim 1, wherein the payload interface modules are configurable for distinct communications models including one-to-one, one-to-many, and many-to-many, the one-to-one model being characterized by logical mapping of interface ports and sub-slots of corresponding payload interface modules of the ground station and vehicle in which data received on a given sub-slot of the first payload interface module is directed to a corresponding sub-slot of the second payload interface module.

10. The remotely piloted vehicle system of claim 9, wherein the one-to-many model is characterized by logical mapping of a plurality of payload interface modules of the vehicle to corresponding expansion modules connected to a single payload interface module of the ground station.

11. The remotely piloted vehicle system of claim 9, wherein the many-to-many model is characterized by mapping of each of a plurality of payload interface modules of the ground station to corresponding ones of a plurality of payload interface modules of the vehicle.

12. A payload interface module for use in a remotely piloted vehicle system, comprising a processor, network circuitry, and external connectors for connection to a local data terminal and to a local payload element of a payload-specific type, the connection to the local data terminal providing for transfer of messages to the payload interface module from a paired opposite-end payload interface module via the data terminal, the payload interface module being configured and operative to convert between the messages corresponding signals of the connection to the local payload element, the payload interface modules forming respective endpoints of a ground-to-vehicle channel between the local payload element and a remote payload element connected to the opposite-end payload interface module.

13. The payload interface module of claim 12, wherein the connection to the local payload element includes first and second interfaces each being one of a plurality of interfaces of distinct payload-specific types.

14. The payload interface module of claim 13, wherein the payload-specific types of the plurality of interfaces are serial bus, discrete digital I/O signal, and a second network interface.

15. The payload interface module of claim 12, wherein the payload interface module further includes power circuitry and payload-side power connections for connection to the payload element to provide switched power thereto.

16. The payload interface module of claim 12, wherein the network circuitry includes an internal network, a network port carried by one of the external connectors, and a network switch providing switched connections including to the payload element.

17. The payload interface module of claim 12, wherein the external connectors include an expansion connector for connection to an expansion module, the expansion module being configurable to provide an interface to a respective second payload element and form an endpoint of a second ground-to-vehicle channel between the second payload element and a second remote payload element.

18. The payload interface module of claim 12, housed in a housing for modular placement in proximity to the separately housed payload element.

19. The payload interface module of claim 12, configurable for distinct communications models including one-to-one, one-to-many, and many-to-many, the one-to-one model being characterized by logical mapping of interface ports and sub-slots of corresponding payload interface modules of a ground station and a vehicle in which data received on a given sub-slot of the first payload interface module is directed to a corresponding sub-slot of the second payload interface module.

20. The payload interface module of claim 19, wherein the many-to-many model is characterized by mapping of each of a plurality of payload interface modules of the ground station to corresponding ones of a plurality of payload interface modules of the vehicle.

* * * * *