Data Communications Issues For Digital Power System Management

Robert V. White
Artesyn Technologies
Westminster, Colorado

David L . Freeman
Texas Instruments
Dallas, Texas

IBM Power And Cooling Symposium
20 September 2005

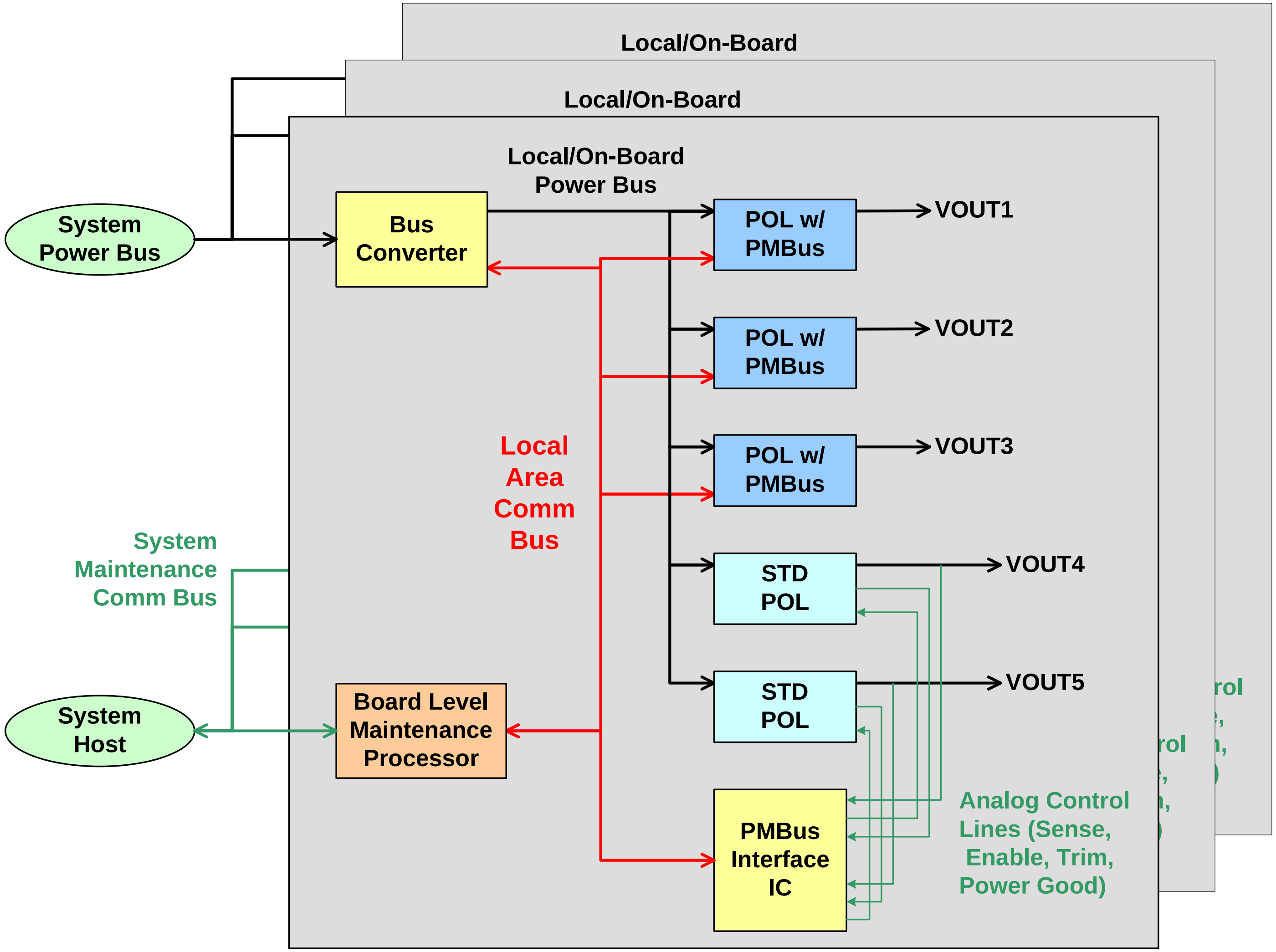
Presentation Overview

- Requirements
  – Physical And Fiscal
  – Data Flow
- Characteristics
  – Types Of Buses
  – Issues And Constraints
- Recommendations
  – By Data Bus
  – By Application

Fundamental Requirements

- Low Cost, Low Cost & Low Cost
  - Component - Low Cost
  - Development - Low Cost
- Robust
  - Carry Data Without Corruption Or Interruption In The Presence Of Noise
  - Does Not Pass the Burden To The Host
- Must Support Time Critical Communication
  - Address The Need For Alarms And Alerts
  - Address The Need For Fast Host Intervention

Additional Requirements: Who Talks To Whom?



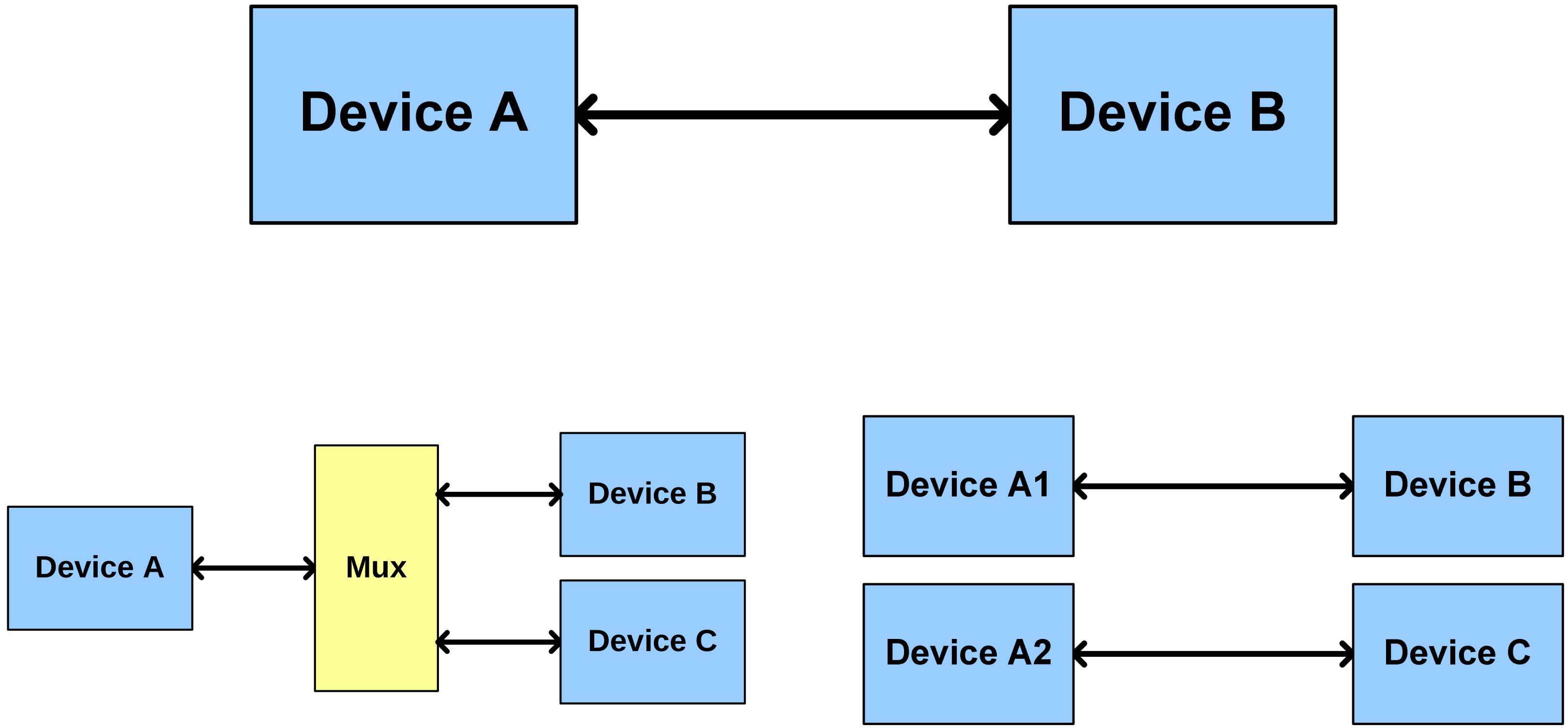
Additional Requirements:
Real Time Data

- Time Critical Information Of Two Types
  - Events
  - Parametric
- Fault Events Can Be Catastrophic
  And Must Be Transmitted With Minimum Delay
- Parametric Data Requires Data Rates
  Of Tens Of Megabits Per Second
- Recommendation
  - Events: Dedicated Signal Lines
  - Parametric: Dedicated, Customized Buses

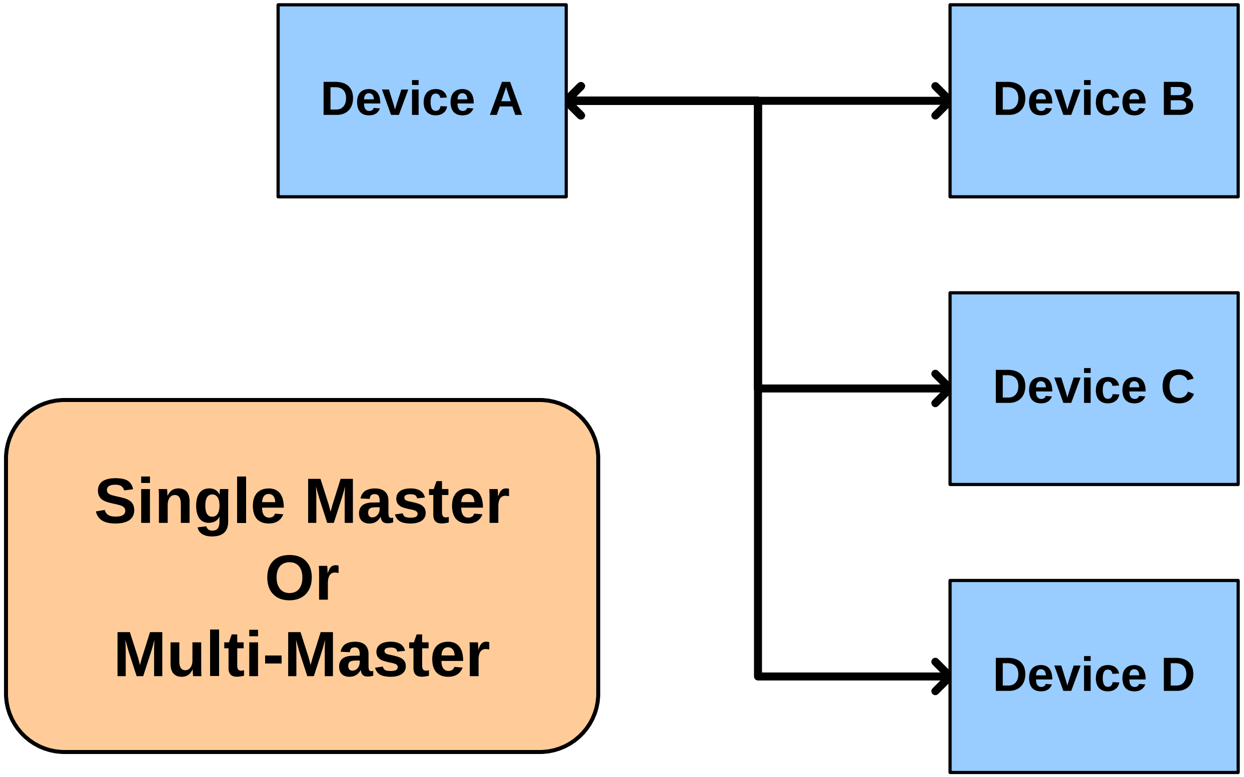
Additional Requirements: Hot Swap

- A Common Requirement
  - Does Not Interrupt Bus Traffic
  - Does Not Require Complex System Response
- More On This Later

Data Communication Characteristics: Connectivity: Point-to-Point



Data Communication Characteristics: Connectivity: Multi-Drop



Data Communication Characteristics: Directing Communication On Multi-Drop Bus

- Chip/Device Select Lines
- Addressing
  - Hard Versus Soft Addresses
  - Allowable Addresses
  - Assuring Unique Addresses
  - Address Ties To Physical Location Or Function
- Address Pins – Not Just Binary
  - Tri-State
  - Resistor Programmable

Data Communication Characteristics: Bus Contention

- Bus Contention In Multi-Drop Buses Is Unavoidable For Multi-Master
- Lossless, Bitwise Arbitration Common For Simultaneous Attempts To Transmit
- Adding Priority To Messages (Like CAN Bus) Does Not Prevent Delayed Messages
  - If Bus Is Busy, Even High Priority Messages Have To Wait Until The Bus Is Clear
  - Continuous Stream Of Higher Priority Messages Can Indefinitely Delay A Lower Priority Message

Data Communication Characteristics: Speed And Timing

- Megabits Per Second Is Not The Whole Story
- How Fast Can Data Get From Sender To Receiver?
  - Over Communication Bus, Not Fast Enough For Most Time Critical Events
- Packet Overhead Reduces Effective Bit Rate
- Time Critical Data Should Be Routed Over Dedicated Buses Between Only The Devices Involved
  - Example: Real Time Digital Current Sharing

Data Communication Characteristics: Polling And Interrupts

- Polling
  - Simpler To Implement
  - Detection Of Failed Or Removed Devices
  - Consumes A Lot Of Resources
  - Possible Delay Time = Refresh Rate
- Alert Or Interrupt Driven
  - More Complicated Code
  - Reduces Burden On Host
  - Quick Notification Of Events
- Good Choice: Blend The Two

Data Communication Characteristics: Range And Number Of Devices

- Range
  - Often Capacitance Limited
  - Short Range, Open Drain Drivers = Low Cost
  - Longer Range Requires More Robust Drivers
- Number Of Devices
  - Like Range, Often Load Limited
  - May Be Address Limited
  - Generally Not A Problem

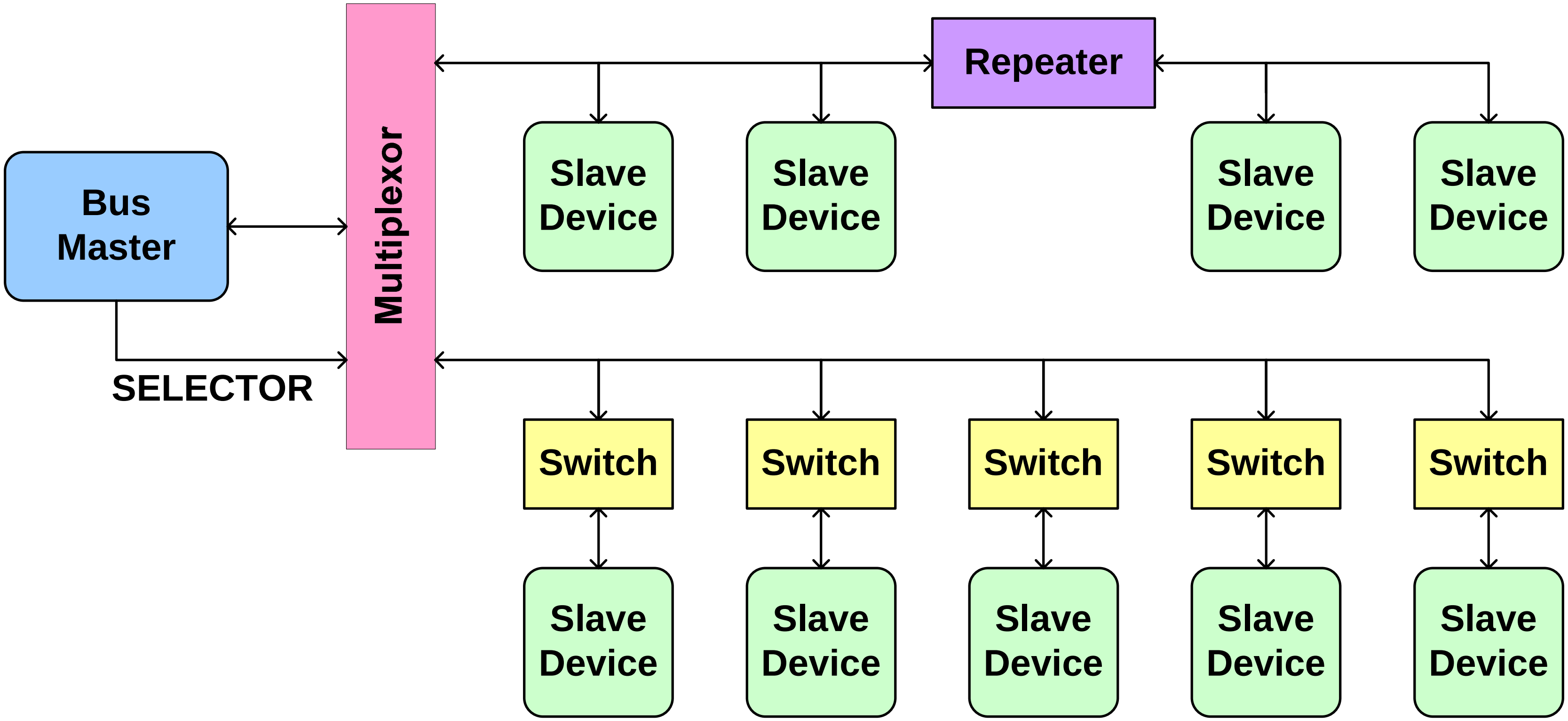
Data Communication Characteristics: Range And Number Of Devices

- Range
  - Often Capacitance Limited
  - Short Range, Open Drain Drivers = Low Cost
  - Longer Range Requires More Robust Drivers
- Number Of Devices
  - … d Limit…
  - … ed
  - … em

Watch Out For The Edges!

Obey The Spec – Or Else!

Data Communication Characteristics: Range And Number Of Devices



Data Communication Characteristics: Clock

- Synchronous
  - – Clock Signal Sent With Data
  - – Receiving Devices Do Not Need An Oscillator
  - – Range Limited
- Asynchronous
  - – No Clock Signal Sent With Data
  - – Each Device Needs Its Own Oscillator
  - – Can Loose Sync On Long Strings Of Ones Or Zeroes
    - Bit Stuffing
    - Fancy Coding

Data Communication Characteristics: Single Ended Or Differential Signaling

- Single Ended Signaling
  - One Wire For Data
  - Lower Cost, Less Complicated
  - More Susceptible To Noise Then Differential
- Differential Signaling
  - Two Wires For Data
    - Opposite Polarity Signals On Each
  - More Immune To Noise
  - More Immune To Ground Voltage Differences
  - Higher Cost

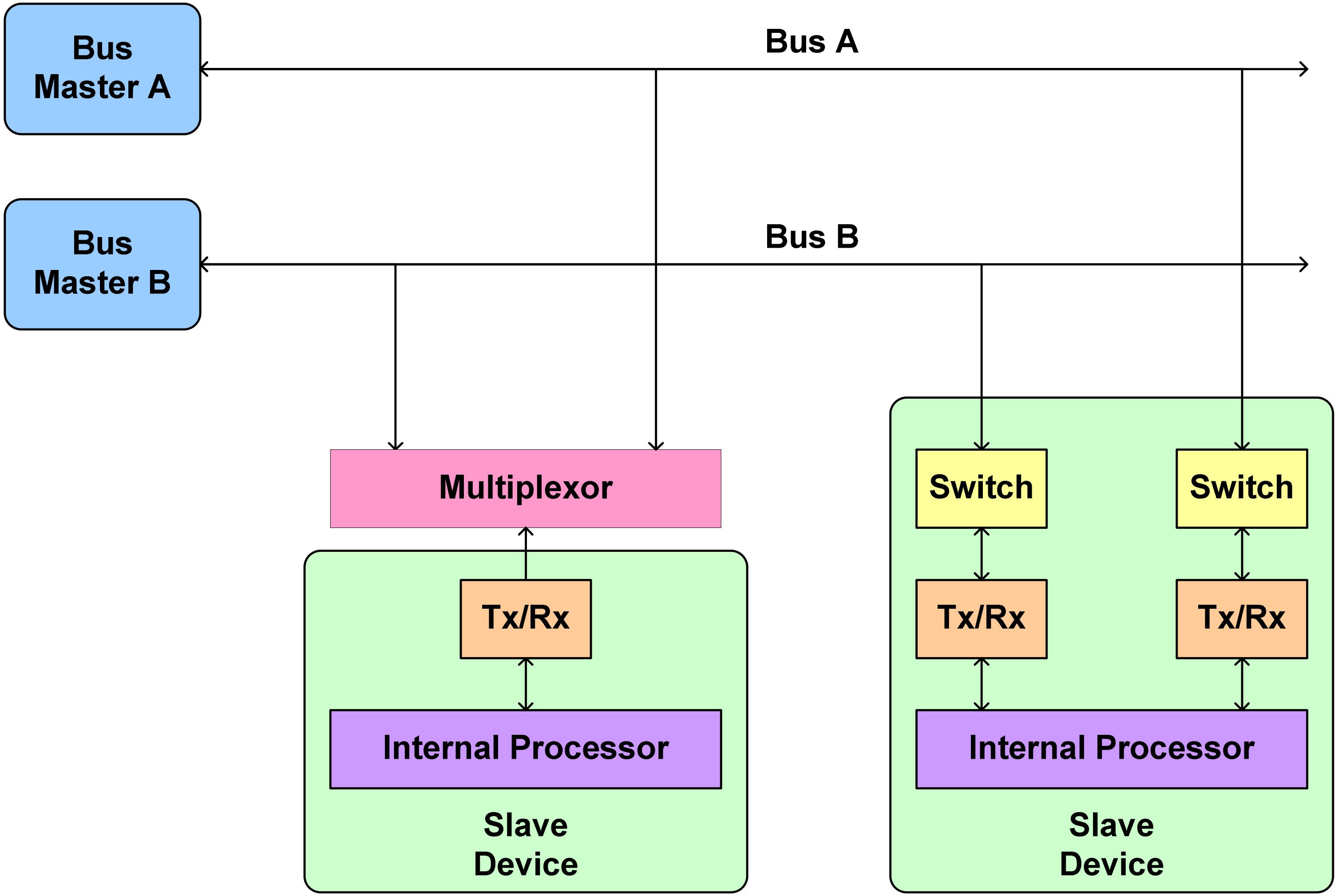
Data Communication Characteristics: Transmission Control And Protocols

- Transmission Control Issues
  - Device Is Busy And Cannot Be Interrupted To Respond To Another Request
  - Device Cannot Accept Data At Current Rate
  - Device’s Buffer Is Full
  - Bus Is Busy And Device Must Wait
- Protocols
  - Read/Write Like
  - Message Based

Data Communication Characteristics: Error Detection And Correction

- Possible Errors
  - Bit Value Changed
  - Beginning Or End Of A Byte Or Bit Sequence Not Recognized
  - Too Many/Too Few Bits In A Frame Or Packet
  - Start Or End Of A Packet Or Message Not Recognized
- Error Detection: Parity Bit, Checksum, CRC
- Error Correction
  - More Complex
  - Lots More Bits

Data Communication Characteristics: Fault Tolerance



Data Communication Characteristics: Hot Swap

- Hot Swap Issues
  - Removal And Insertion Without Disruption
  - How Does System Know If A Unit Has Been Added Or Removed?
- Most Buses Support Hot Swap Fairly Well
- Implementation Issues
  - Making Ground Connection “Last Break & First Make”
  - Preventing Unpowered Devices From Shorting Bus During Insertion Or Removal
  - MODULE_PRESENT Signal To Assist Detection

Data Communication Characteristics: Hardware Implementation

- Software Emulation Using GPIO
  - Possible To Do
  - A Source Of Many, Many Headaches
  - Timing Is Very Difficult Even For “Slow” Buses
- Integrated Solutions
  - Many Low Cost Microcontrollers Have Bus Interfaces Built In
  - Must Have For Complex Buses Like CAN Bus And Ethernet

Data Communication Characteristics: IP Issues

- Standard: De Facto vs. De Jure
- Who Controls?
  - An Organization
  - Single Company
  - No One
- Organization Ownership Preferred
  - Adopter's Agreements
  - Compliance Assurance
- Royalty Free Or Not?

Recommendation By Bus Type
RS-232 & RS-485

RS-232

- Advantages
  - Common Peripheral
  - Simple
  - Relatively Low Cost
- Disadvantages
  - Point-To-Point
  - Oscillator
  - Speed
- Recommended
  - Simple Point-To-Point With Logic Level Interface

RS-485

- Advantages
  - Differential Signaling
  - Long Distance Communication
- Disadvantages
  - Additional Cost Of $1.00 to $1.50 In High Volume
  - All Protocol In Software
- Recommended
  - Longer Range Communication Such As Rack-To-Rack

Recommendation By Bus Type
I²C And SMBus

I²C

- Advantages
  - Common Peripheral
  - Simple
  - Very Low Cost
- Disadvantages
  - Noise Sensitivity
  - Bus Capacitance Limitation
- Recommended
  - SMBus Is Better Choice In Almost Any Case

SMBus

- Advantages
  - Low Cost Like I²C
  - More Robust Than I²C
  - Additional Features
- Disadvantages
  - Bus Capacitance Limitation
- Recommended
  - On-Board And Shelf-Level Communication

Recommendation By Bus Type
SPI Bus And Dallas 1-Wire

SPI Bus

- Advantages
  - Simple
  - Chip Select Lines Eliminate Addressing Concerns
  - Good Speed (1 MHz)
- Disadvantages
  - No Standard
  - Chip Select Lines
- Recommended
  - Local Interconnect Of A Couple Of Peripherals

Dallas 1-Wire

- Advantages
  - 1-Wire
  - Unique ID In Every Device
  - Low Power
- Disadvantages
  - Noise Sensitivity
  - Proprietary
  - Cost
- Recommended
  - Only To Accommodate Legacy Situations

Recommendation By Bus Type
CAN Bus & LIN Bus

<u>CAN Bus</u>

- Advantages
  - Differential Signaling
  - Noise Immunity
  - Fault Tolerance
- Disadvantages
  - Cost
  - Requires Integrated Peripheral
- Recommended
  - Longer Range Communication Such As Rack-To-Rack And Beyond

<u>LIN Bus</u>

- Advantages
  - Single Wire
  - Reasonable Noise Immunity
- Disadvantages
  - Cost
  - Slow Speed
  - Complexity
- Recommended
  - Not Recommended (Use SMBus Or CAN Bus/ RS-485 Instead)

Recommendation By Bus Type
USB & Ethernet

<u>USB</u>

- Advantages
  - Well Supported
  - Hot Swap Friendly
- Disadvantages
  - Requires Hub To Initiate All Communication
  - Relatively Complex Software And Hardware
- Recommended
  - PC To Power System Interface For Service

<u>Ethernet</u>

- Advantages
  - Long Haul Capability
  - Internet Friendly
- Disadvantages
  - Cost And Complexity
  - Very Large Packets
  - Software Support
- Recommended
  - Interface To An Embedded Web Server In A Power System Manager

Recommendation By Application

- On-Board/Single Board Power System
  - SMBus
- Shelf-Level/Chassis-Level Power System
  - SMBus If Capacitance Allows
  - RS-485 If Not
- Shelf-To-Shelf Or Rack-To-Rack
  - RS-485
  - CAN Bus

Recommendation By Application

- Facility Level
  - RS-485 Or CAN Bus
- Campus Level
  - Ethernet
- PC To Power System Manager
  - USB

Summary

- No One "Right" Bus For All Power Communications
  - The Scope and Benefits Extend To Beyond The Local Area
  - Several Well Established Buses To Choose From
- Know Your Application And Its Requirements
- Be Knowledgeable About Your Choices
- Choose The "Right Tool For The Job"
- Be Smart In Your System Design
  - Imitate Successful Designs
  - Understand The Constraints Before You Start Your Design
- **Follow The Specification!**

Thank You
For Your Time
And Attention

Data Communications Issues For Power System Management

Robert V. White
Artesyn Technologies
Westminster, Colorado
bob.white@artesyn.com

Dave Freeman
Texas Instruments
Dallas, Texas
dave_freeman@ti.com

Abstract – **Digital power management, the ability to remotely manage and configure ac-dc power supplies and dc-dc converters, is being adopted at a rapid rate. There are many choices to be made when implementing a digital power management system [1][2]. The choice of transport (data link and physical layers) is very important. Choosing the wrong transport for the application can create endless problems with the system. This paper starts with a brief history of digital power management. It then reviews the requirements for data communications for digital power management. Knowing the requirements, several of the more common implementation issues are discussed along with recommendations for avoiding those problems. From there, a summary of common data communications buses along with the advantages and disadvantages for power system management is given. The paper concludes with recommendations data communications in power system management for on-board power systems, within a single chassis, for inter-chassis communication and for facility or campus level communication.**

I. Evolution Of Digital Power Management

Digital power management is not new. It was being introduced into telephone central office power systems in the early 1980s [3][4]. Indeed, digital power management was even being used over wide areas using the telephone network as the communications means [5]. By the late 1980s, even computer companies such as the Digital Equipment Corporation were using digital power management in both high end systems such as the VAX 9000 and departmental servers such as the DEC 4000 [6].

By the mid 1990s, digital power management had made its way into desktop and even laptop computers [7]. In 1994, notebook computers began to use a standard for power management based on the System Management Bus (SMBus) protocol. This standard is used today to manage battery, adapter, and backlight power. During this time, digital power management techniques were well established in the test and measurement industry [8] and even into the world of high energy particle physics [9][10]. By the late 1990s, digital power management was included in the overall system management for desktops and small servers [11].

The widespread adoption of the intermediate bus architecture (IBA), which creates a local area power system on each circuit module, has driven power management from the system level to the circuit board level. Starting in 2001, IC makers like Primarion [12], Intersil [13], Volterra [14] and Summit Microelectronics [15] started offering products with digital power management functions. In 2004, Power-One, Inc. introduced the Z-Series products which featured digital power management both at the converter and system level [16]. Later in 2004, a group of power supply and semiconductor companies announced an effort to develop an open standard power system management protocol called PMBus™ [17]. The PMBus™ specification was made public in March 2005 [18].

II. Requirements

When choosing a data communications bus for power system management, there are many factors that must be considered. A bus that is an excellent choice for one application may be a terrible choice for another. The first step to choosing the best communications bus for a given system is to clearly identify the system's requirements. Only when the system's needs are clear can all the tradeoffs be made that lead to the best communications for that system.

A. *Fundamental Requirements*

There are three fundamental requirements for the data communications bus used in power system management.

First, it must be very inexpensive. Although no system works without a power supply, the power system is not considered a significant value add. System OEMS are always looking for ways to reduce the cost of the power system.

Second, the communications must be robust. That is, it must be able to carry data without corruption even in the presence of electrical noise. Ideally there is no extra burden on the system to police the reliability of the communication.

Third, the communication must meet the need for transmitting time critical information without delay. Failures in a power system can release uncontrolled amounts of energy that can destroy a power converter or the loads it powers. Also, failures that drive voltages out of tolerance can result in corrupted data. For many abnormal events in a power system, there must be a means to immediately command the power system to stop transferring energy to the outputs and for the system to stop processing data.

B. *Other Requirements*

1. Who Talks to Whom?

A good place to start when considering digital communications buses is connectivity. The simplest power systems have a single power supply. Its outputs provide power to the entire system. The entire power management function may take place between a system or baseboard controller and the power supply.

More complex power systems may have multiple power supplies powering a distribution that is routed to multiple boards and loads in the system. Often many of those boards and loads have dc-dc converters that create the voltage needed by that board. The power system management function may now require communication among the main power supplies, the dc-dc converters and a system controller.

Even more complex power management systems may have local power systems on each board. These individual boards may have a local power system manager that communicates with multiple dc-dc converters on that board. In turn, there may be a system controller that communicates with the local power system manager on each board as well as the main power supplies.

In some cases a the power system, such as a 48 V power system providing power to multiple chassis's, may need to communicate with a data center control room tens of meters away.

The possibilities are essentially endless. The point is that the system engineer must know for each device in the power system:

- With which other devices will it be communicating,
- Will it be receiving information, sending information only in response to requests, or sending information without a request from another device,
- The time criticality of the information being transmitted, and
- How far away physically (and electrically) are the devices with which it will be communicating.

As for the distance question, there are a few important cases:

- All of the devices are on the same circuit board, in close physical proximity and share a good common ground (on-board system),
- All of the devices are in the same shelf or chassis system and share a good common ground (shelf level or sub-rack based system),
- All of the devices are in the same chassis or enclosure but do not share a good common ground (chassis based system),
- The devices are in different shelves or chassis's and do not share a good common ground (inter-shelf or inter-rack system), and
- The devices are spread across a facility, a campus or larger area (wide area system).

This mapping of the connections and distance between the power converters being managed is very important to choosing the right data communications bus for a given application. The right bus for an on-board system, for example, is not the right bus for a facility wide communication bus – and vice versa.

2. Real Time Data

Another important question is whether communication bus chosen for power system management will have to carry real time data. In a power system, real time data come in two forms: events and parametric data.

An example of an event that needs to be communicated in real time is an output voltage out of tolerance condition. In this case, the system needs to stop processing data with the shortest possible to delay to prevent or minimize corruption of the data.

An example of real time parametric information would be used is when multiple units are powering up at the same time. These units might continuously share information about their output voltage so that each voltage tracks all the others until each unit reaches its setpoint value. Another example would units operating in parallel passing information about its output current during the current switching cycle to the other units so that the units can equalize their output currents and maximize transient response.

Passing real time parametric data is a challenge. With switching frequencies routinely at hundreds of kilohertz to megahertz, passing digital data in real time requires passing tens of megabits per second. Data transmission rates this fast require specialized and possible costly hardware and extreme care in design. It is recommended that for now and the near future, system engineers avoid passing real time parametric data between power converters or between a power converter and a power system manager.

It is also recommended that information about real time events also not be passed over a communication bus. As described below, there are too many ways that a signal can be delayed when trying to transmit over a bus. For real time events that require immediate action by other power converters and/or the system host, dedicated signal lines, such as POWER_FAULT signal, are recommended.

3. <u>Hot Swap</u>

A common requirement for a power management system bus is that devices be hot swappable. Hot swapping, even if the chosen data bus nominally supports it, requires care in design to make sure that removing or inserting devices does not disrupt the bus. Hot swapping and how it relates to various data communications buses is discussed below.

III. Data Communications Characteristics

This section reviews the key characteristics of data communication buses against the needs of a power management protocol.

A. Connectivity

1. <u>Point-To-Point Connectivity</u>

One way is for the host device to connect to only one power device (point-to-point). This provides some simplicity by reducing the possibly that the bus is busy when the host or the power wants to send a message. The disadvantage is that for the host system to communicate with to multiple power devices requires multiple bus connections or a means to multiplex the communication bus.

2. <u>Multi-Drop Connectivity</u>

More commonly, there is a single bus connecting all of the elements being managed (multi-drop). This greatly reduces the number of connections and board space required. However, multi-drop has two notable disadvantages. The first is that the host must know the address of each device or have device selection lines to signal the device the communication is directed at that particular device. The second is the potential for bus contention. When a device has information that needs to be transmitted, it may be prohibited from doing so because the bus is already in use by other devices on the bus. Without some form of out-of-band signaling, the device will have no way of notifying the host that that it has information to send. These are discussed below in more depth.

Most digital communication buses, but not all, are capable of multi-drop operation. For example, the simplest UART buses, RS-232 (single ended) [19] and RS-422 (differential) [20] are point-to-point only.

B. Directing Communication To A Specific Device On A Multi-Drop Bus

Most communication buses used in power system management are multi-drop. In multi-drop systems, there must be a way that that a device to know that a message is intended for it.

One way is to use a device select or chip select line. In this method, there are individual connections from the bus master to each device on the bus. When the bus master wants to send a message to a device, it asserts the device select line, notifying the device that the next message on the bus is intended for that device. The SPI bus is an example of a bus that uses a chip select line.

The advantage to device or chip select line is that unique identifiers (addresses) are not needed in the devices on the bus. While removing the need for unique addresses simplifies the devices, a device select line from the master to each device is an additional cost and adds complexity to board layouts. These additional traces must be added in the design to every location where the there might be a device on the bus. For scalable and expandable systems, this adds cost to the entry level system.

The alternative to using a device select line is to have unique identifier (address) for each device on the bus. While this simplifies the hardware by eliminating device select lines, it brings its own set of problems.

How the address is set or programmed is the first problem. Commonly, these are set in hardware by jumpers, DIP switches or hardwired address pins. In some cases, addresses are linked to a physical location, such as in a backplane, and are automatically configured by connecting a device's address pins to ground or not. In some configurations, the high order bits of the address can be programmed into the power device while allowing the low order bits to be determined by the pin connection. In this case the number of address pins can be reduced.

"Soft addressing", the setting the address of multiple devices over the communications bus, is not easy. The problem is that to be able to send a device its address over the bus, it must have already an address to which the new address can be sent. There are schemes that allow for software configurable addresses but they all rely on a unique identifier, such as a serial number in an IC, as the initial address. Such schemes are generally more complex than are needed for power system management protocols.

Given that the device addresses are set in hardware, there are two questions:

- What are the allowable addresses?
- What means is used to assure that all devices have a unique address?

The allowable addresses are a function of the data communication bus specification. Some buses define the addressing in the specification. The I²C [21] and SMBus specifications [22][23], for example, allow seven bits for a device's address. Allowing for the addressed reserved for in the specifications, this still leaves more than 100 addresses to the user. Many other buses, such as the RS-485 [24] and Controller Area Network (CAN) bus [25][26] leave addressing to a higher level of software. In practice, the number of addresses available on a bus is not a limitation in implementing a digital power management system.

In general, the problem of assigning unique addresses to devices on a bus is left to the system designer. One exception is the Maxim/Dallas 1-Wire bus [27] which embeds a 64 bit identifier in every 1-Wire IC.

For the I²C bus, Philips made an interesting tradeoff that both simplified and complicated creating unique addresses. Although the I²C specification allows for seven address bits, Philips assumed control of the four highest order bits. The four highest order bits were a type code that Philips controlled. Each licensee of the I²C bus submitted their device to Philips and Philips assigned the type code. Philips tried, with some success, to assign the type codes to minimize potential address conflicts.

Since the I²C bus patents have expired in recent years, Philips appears to no longer be controlling the four higher order address bits. Previously, with three address bits (pins) available, there could be up to eight of one device on one I²C bus. Different types of devices co-existed fairly well due to Philips control of the higher order address bits. Not, with no one controlling any of the address bits, device makers will assign them as they please. This does mean that system engineers using the I²C or SMBus will have to choose their devices so that:

- All the devices on the bus have different type codes (the four highest order bits) or
- No more than eight devices with the same type code are used on any one I²C bus.

It will be interesting to see how the market sorts this out. One approach would be to make all seven address bits available to the user. However, adding seven pins to a device is unwanted cost and board space. IC manufacturers are offering different ways to set the full seven bit address without using seven pins. For example, using tri-state pins would allow setting up 125 addresses with five pins. Another solution is to use one or more resistors to program the address. A current from inside the IC develops a voltage across the resistor. Using an A/D converter, the IC measures the voltage and decodes that to an address value. This technique can easily allow for eight to ten values per resistor with some implementations allowing for up to 32 values. While this saves money on pins, it is only economical if the IC already has an A/D converter.

This left three bits for setting a device's address.

C. Bus Contention

A multi-drop bus is like an old style party line telephone. When a device has a message to send, there is a chance that the bus may already be in use. If so, the device must wait until the bus is not busy before sending its message. This is a simple fact of multi-drop buses and cannot be avoided. By choosing the bus and protocol so that the average utilization of the bus is low, the chances of a device encountering a busy bus are reduced but not eliminated.

Suppose now that more than one device attempts to access the bus at the same time – perhaps both have been waiting for the bus to become available so they can send their message. There must a means to arbitrate among the devices wanting access to the bus so that one gets to send its message and the others wait again for the bus to be available.

Some buses, like I²C, SMBus and 1-Wire, use a simple bit-wise arbitration. As each device attempts to send a bit, it compares what it is transmitting with what is on the bus. Any device sending a logic low will prevail. A device trying to send a logic high will see that the bus is low, not high. It can therefore detect that the bus is busy and stop trying to transmit. These buses start there transmission with the address of the destination so by default, so the messages sent to the lowest addresses become by default the highest priority messages. This method is no-destructive. Messages that lose the arbitration do not loose or destroy the message, it is simply held until the bus is available.

Adding a means to add a priority to a message, as the CAN bus does, still does not guarantee that a device will not be blocked from sending its message. As long as messages of higher priority are being sent, a device with a low priority message will have to wait. Fixes to this problem have to be made in higher levels of the software. One way is to deliberately add a minimum time after one device sends a message or packet before it is allowed to send another. While this may allow a lower priority message to get through, it may block a more important higher priority message.

D. Speed And Timing

The question of speed always arises when talking about digital power system management. The key to determining the needed speed is to recognize that in power system management there are three kinds of events:

- Non-time critical events such as configuring and monitoring a power converter operation,
- Time critical events such as issuing a shutdown command in the event of an output overvoltage condition,
- Real-time control such as passing switching commands from a master controller to each of the phases in a multi-phase buck converter, and

The first two event categories are communications that might be expected between a system host service controller and power devices. Configuration and monitoring of power devices by a service controller does not typically dictate a high speed bus or critical timing. For example, using SMBus at 100 kHz with error detection, 6 bytes of information is needed to retrieve a 16 bit value. Adding the other setup and hold times, this communications will take a minimum of 500 µs. The data communication can be made more efficient by using commands that send more data at one time. The data may contain current, voltage, temperature, and status values, for a total of four 16 bit values. In this case, the bus will be busy for about 1 ms. The number of devices on the bus may require faster data flow in order to service all device in the desired time interval. History has shown that once a bus is created, the number of devices on the bus increases over time.

Moving to the second event category, like a shutdown command, general configuration and monitoring may establish the delay time between determining that such a command should be sent and the device actually receiving the complete command. Using the 1 ms packet command and assuming the shut-down command has no data and therefore requires only a three byte packet, then the designer must assume that there will be a delay of about 1.2 ms before the power device will respond to the shut-down command. This delay may be too long for many applications.

For the most time critical events, no data communications bus is adequate. Aside from the transmission time, the bus may be busy when the critical message needs to be sent. For time critical events, dedicated signal lines are required. This adds complexity to the devices, the system host and the layout but is a price that must be paid.

The third category of event and control requires real-time information that needs a dedicated communication channel. A command from a master controller to stop a switching cycle cannot be delayed because the bus is busy with a request to read a device's serial number. For this reason, this kind of information is typically not passed over the bus used for the typical power management functions of configuration, on/off control and monitoring.

As alluded to earlier in this section, transfer speed is only one issue, data packet overhead is another. Small packets typical of power system management can be overwhelmed by lots of communications overhead such data frame sequence number and frame synchronization bits, rendering the rate of transfer of useful information much lower than the "clock rate".

A recommended strategy is to make sure that there are degrees of autonomous operation. For example, if two or more power devices are involved with load sharing then the required information for this function should flow between the devices in a manner where other communications can not interfere and the require bandwidth can be maintained. This means the power communication designer may have to deal with multiple communication busses including the sharing of analog information.

E. Polling And Interrupts

Related to the discussions above on bus contention and speed is how the system host gets information about the status of the other devices on the bus. One way, polling, is for the system host to continually request a status report from each device on the bus. The advantage to this is that system will be able to detect a failed or missing device fairly quickly. The disadvantage is that the system host must spend a lot of processor cycles generating the status requests and processing the results. Also, the continuous bus traffic, and the need to poll devices on a frequent basis means the bus speed must be fairly high to avoid long delay times.

Another way for the system host to know the status of the devices on the bus is for each device to tell the system host when its status has changed. It can do this through a signal line that is daisy chained among all of the devices on the bus and connected to an interrupt input on the host. The only common data communication bus that includes such a signal is the SMBus, which has the SMBALERT# signal for just this purpose. While there is a bit of complexity in generating the interrupt in the device and processing it in the host, it is an overall reduction of complexity and system requirements over a polling system.

Actually, the recommended approach is a combination of the two. It is recommended that the data communication scheme include an alert or interrupt line so that devices can notify the system host of a change in status in a timely manner. It is also not a bad idea if the system host infrequently polls the system just to make sure that all the devices it expects to be there are indeed active and responsive. That polling can be infrequent and does not need to be a priority task for the host, simplifying it and reducing its requirements and the speed requirements of the main data bus.

F. Number Of Devices

Another consideration when choosing a data communication bus is the number of devices that can be attached to the bus. The number of addresses available may be a limitation but electrical characteristics are also a consideration.

Buses that use open drain drivers with pull-up resistors offer advantages in terms of cost, tolerance of short circuits, and simple detection of bus conflicts. The I²C and SMBus are buses of this type. The disadvantage is that the total capacitance on the bus must be limited in order to assure that the maximum rise time specification is not violated. Violating the maximum rise time can cause bus devices not to recognize START and STOP conditions as well as corrupting data. It is important when using a bus with open drain drivers that the bus capacitance specification is obeyed.

Capacitance on the bus comes from both the input capacitance of the devices attached to the bus and the capacitance of the signal lines. This means that when using I²C or SMBus, a system that spans a small physical area, such as power system on a single board, can have more devices than a system that stretches across the backplane of rack mounted system. As a practical matter, typical one board systems may have up to thirty devices on the bus if care is taken with the layout. For systems that stretch across the backplane of a rack mounted system, practical values are in the range of 12 to 16 devices.

The RS-485 bus also has an electrical limit on the number of devices on the bus. The specification defines the maximum load a device can present to the bus and required that no more than 32 such loads be attached to any one bus. However, many device manufacturers provide RS-485 interface ICs with one half, one quarter or one eighth unit loads. This enables the bus to support 64, 128 or even 256 devices. Examples of such devices are the SN65HVDxx series from Texas Instruments [28].

G. Range

Power management systems are used over distances ranging from centimeters to kilometers. Matching the right data communications bus to the application is not difficult but it is important. Using a short range bus for a long range application leads to errors and a non-functional bus. Using a long haul bus for a short range application wastes a lot of money.

Buses that are good for short range (centimeters to a meter or so) are I²C, SMBus and SPI. Good medium range buses (1 meter to tens of meters) are the RS-485 and CAN buses. For hundreds of meters and more, Ethernet is a good choice.

If needed, the range of a bus can be extended by using repeaters.

H. Clock

When transmitting data on a bus, the receiving unit needs to know when a bit of data is valid on the bus. One way to do this is for the sending device to also send a clock signal along with the data. This technique, called synchronous data transmission, is used in buses like I²C, SMBus and SPI Bus. The key advantage is that the receiving devices do not need an oscillator, a significant cost savings.

Synchronous mode works well when the transmission distance is limited. For longer distances, asynchronous mode is used. In this mode, the transmission starts with a synchronizing signal. The receivers detect this signal and adjust their local clock signals to be in time with the transmitted data. Then as the sending device puts data on the bus, the receivers read the bus when their local clocks tell them it is time for another bit.

The advantage to this is that another signal line for a clock signal is not needed. The main disadvantage is that each device on the bus needs to have a precision clock. Even with precision clocks, during long strings of the same value (either zero or one), the receiver can lose synchronization with the sender. This causes the receiver to read the bus at the wrong times and data errors are certain.

To avoid this problem in asynchronous mode, bit stuffing is sometimes used. This means that after a fixed number of bits of the same value are sent, a bit of the opposite value is sent just to enable the receiver to maintain clock synchronization. There are also other techniques, such as return-to-zero (RZ) or Manchester coding, that are used to maintain synchronization. Any of these techniques costs in either device complexity, bus bandwidth required, or both.

For power management systems, with their typically limited physical scope and emphasis on low cost, synchronous buses are recommended for their low cost and overall simplicity.

I. Single Ended Or Differential

Data communication buses suitable for power system management are available with both single-ended and differential signaling.

Single ended signaling, requiring only one wire for the data, is simpler, lower cost and requires less space on the board for routing.

The key advantage to differential signaling in power system management is the ability to have a higher immunity to noise. The most common buses with differential signaling are RS-422, RS-485, CAN bus and the Universal Serial Bus (USB) [29].

Another advantage is that most of the differential signaling buses work in the presence of a difference in ground potential among devices attached to the bus. However, single ended buses can use level shifters or optical isolators to overcome differences in ground levels.

For on-board power systems, single ended signaling is adequate. Care must be taken with layout to minimize noise pickup, but this is a routine matter.

For shelf or chassis level communication, singled ended signaling usually works well. Noise pickup is somewhat more of a concern but good design practice usually makes this a non-issue. I²C and SMBus are routinely deployed in this application with low cost and good results.

For some of the common data communication buses such as I²C, the concern with shelf level buses is the total capacitance. If there is a significant length of bus traces on each board, then when several boards are plugged into a backplane the total capacitance may become excessive. In this case, a repeater between the on-card bus and the traces on the backplane that connect all the cards in the shelf will help.

For longer distance communication, such as shelf-to-shelf or rack-to-rack, a bus with differential signaling is more appropriate. The ability to work with a difference in ground potential is especially valuable in this case. The most popular bus used for power system management in these applications is RS-485 although the CAN bus is growing in popularity.

J. Transmission Control

There are many considerations when dealing with transmission control. Some of these are:

- Device Busy – the slave device maybe performing an operation that cannot be interrupted to respond to device request,
- Device can not accept data at the current speed or its buffer is full, or
- The bus is busy and the device must wait.

In the first case listed above there are several strategies that can help mitigate this problem. The strategies depend on the type of communication request. For example, the host has requested that output voltage from a device. A strategy where such a command requires that the device measure and then report the information may hold-up the bus so it may be better to simply report the latest measured value. In this way the host will always receive data that is no older than a fixed amount of time. This method may also have some issues. What happens if the data is being updated at the same time that it is being requested? This assumes that the slave communication peripheral can operate while the slave CPU is gathering data. In this case dual-ported registers can be used or a small hold-off time is required. This is particularly important when requesting 16 bit data from an 8 bit device.

In the second case, the pace of the data flow needs to be moderated to the point the slower device can keep up. In a single master implementation, transmission control is limited to making sure that the slave device is capable of accepting the data or supplying the data. In an asynchronous bus like RS-232, additional control lines are used or the flow control can be done by taking advantage of the full duplex nature to send flow control information.

In synchronous applications like I^2C, then clock stretching can be used to set the transmission pace. Clock stretching has been a point of aggravation for some system designers. During clock stretching, the slave device will pull the clock low while it is deciding how to respond. This technique is used mostly during the Acknowledge bit phase. Clock stretching requires that master device monitor the clock line to determine when the slave has released the line before proceeding. In those cases where clock stretching is used across an isolation boundary, the isolation technique must provide bidirectional communication.

In the event that data is not ready, the slave device must inform the requesting device. However, not acknowledging ("NAKing") a command is not a good solution as it may confuse the requesting the device as to what the issue actually is. NAKing a command may indicate that the command is not valid as well. Since the requesting device does not know which case this might be, this is a problem. The best solution is to avoid the data not ready issue and only use the NAK to inform the host the command is not valid.

In multi-master communication schemes, buffers and sequencing information is used to reduce the need for flow control.

K. Data Transfer Protocols

There are two basic protocols for moving data over a communication bus. In the first, the entire transaction takes place in a single long exchange between the sender and the receiver. The data transfer is very much like writing to a memory location for sending data or reading from a memory location when data is being received. This technique is used by the I^2C, SMBus and SPI buses. This tends to be a very efficient method of transferring data with a minimum of overhead. It is well suited for use in local power system management protocols where the typical amount of data being transferred is one or two bytes.

The second method of data transfer is to use a message based protocol. In a message based protocol, the system host would start be sending out a message that said "Device number 3, this is the system host calling. Send me the value of your output current." This message would propagate over the bus and be received by device number 3. Device number 3 would respond by sending a message on the bus that said "System host, this is device number 3. My output current is 16.2 amperes." This method of communication is best when the devices are not in close proximity or when all of the devices on the bus are intelligent and fairly autonomous. The CAN bus uses a messaging based protocol. Message based protocols are generally too complex and require more overhead than necessary for most power system management protocols.

The UART based protocols, RS-232 and RS-485 can use either depending on how the driver software is written.

L. Error Detection And Correction

There are many possible errors in a data communication bus. Some are at the message level and some are at the bit level. Possible errors include:

- A bit value being changed,
- The beginning or ending of a byte or other sequence of bits not properly recognized,
- Too many bits or bytes in a frame or packet, and
- The beginning or ending of a message or packet not properly recognized.

There are various methods used to detect transmission error. In some cases, parity can be used to detect problems byte by byte. Packet or date frame error detection is usually done using either checksum or CRC, Cyclic Redundancy Check, values to indicate errors. CRC provides the better error detection but it does add slightly more complexity to solution.

Correction of data is certainly more complex and typically requires sending more information such as ECC, Error Correcting Code, in each packet so that bit error can be corrected. The length of the code determines the number of bits of data that can be corrected. For example, 8 bits of ECC can detect and correct 1 bit in 64 bits. This same code can also detect multiple bad data bits without correction.

M. Fault Tolerant Data Communication

Any data communication bus can be disrupted by either faulted signal line (open, shorted to ground, shorted to a supply voltage) or a failed device on the bus (I/O pin open, I/O pin shorted or stuck low, I/O pin stuck high, or an internal device failure that causes the device to stop communicating).

The only way to make the data communication bus fault tolerant is use redundant data buses. The redundancy must generally include a redundant host system device if the system is to be tolerant of all single point failures. At the power devices, there can either be a multiplexer to select between the two buses (lower cost) or redundant transceivers inside the power device (higher cost). The expense and complexity to fully implement a fully fault tolerant power system management bus means that this is rarely done.

One of the features of the CAN bus is its ability to tolerate a fault on one of the two signal lines. In this case the bus can continue operating with limited speed and range.

Most faults in a power system result from a failed or malfunctioning device. This can be handled by using a method of isolating the device from the bus. In some cases, multiplexers are used to fan out the bus to multiple devices. This method is well suited for a single master topology. However, multiplexers add cost and complexity to any solution. If an alert response is required then identifying the alert device is slowed when using a multiplexer.

N. IP Issues

When a bus is considered to be a standard, then it usually has an organization associated with it. In these cases, an adopter's agreement is needed so that the end customer can have faith that the supplier is privy to the specifications. These buses include SMBus, PMBus, and CAN bus.

Other buses may have licensing agreements. These are typically owned by a single company and are not supported by an industry group. The Maxim/Dallas 1-Wire® bus, for example, requires payment of a royalty for each device on the bus. This is required whether one uses a 1-Wire device from Maxim or if the 1-Wire protocol is emulated in a microcontroller.

Of course, given that cost must be minimized in digital power management, royalty free buses are recommended.

O. Hot Swap Capability

In many digital power management applications, it is required that units be hot-swapped. This could be ac-dc front end power supplies removed from or inserted into a shelf with other power supplies. Another possibility is dc-dc converters on a circuit board that is being swapped. When hot-swap is required, there are two key questions:

- Can the devices on the unit being removed (or inserted) detach (or attach) without disturbing the operation of the bus?
- And even if a hot-swap operation is successful, how does the host system know that a unit has been removed or inserted?

Of the common data communication buses, several support hot-swap very well. RS-485, USB, CAN bus and Ethernet all have hot-swap capability built in. With a bit of care in the design, the I^2C, SMBus and SPI buses can be made to work with hot-swap.

The usual problem associated with hot-swapping these buses is the loss of the bus ground return prior to disconnecting the remaining communication lines. This is usually avoided by using a longer pin for the ground connection than for the signal connections.

Another common issue is an unpowered device being attached to the bus. In this case, there can be a path from the signal pin through the device to its supply voltage pin – which is effectively at ground potential. When the device attached to the bus it effectively shorts out the bus until the device is active. In addition, the flow of current through the device in this manner can cause lock-up so that the device never operates properly. This may mean that some form of switch, such as a small signal MOSFET, may be needed in series with the signal line to prevent the device from shorting the bus when it is inserted. The switch can be turned on and the device connected to the bus once the device's power supply is established and the output is well controlled.

Extraction can also cause problems. It is best if the device being removed has the time to gracefully disconnect itself from the bus, such as by setting tri-state outputs to be open or opening the small signal MOSFET switch mentioned above. Long pin/short pin arrangements are often used to provide this notice but need careful attention to tolerances, both electrical and mechanical, to work well. More expensive, but providing more assurance that the bus will not be disrupted during an extraction, is a switch attached to an extractor handle.

The other question of notifying the system host that a device has been removed or inserted typically requires some form of a MODULE_PRESENT signal. This is most often implemented by a pin that connects to ground. When the unit is inserted, the pin is shorted and the host knows that module is present. When the device is removed the signal line is pulled high and the host knows that the device has been removed. This requires that the system monitor addition lines.

For digital power management systems, hot-swap is generally not an issue. For some buses, such as I²C, it is necessary to take some care in the design of each unit to make sure the bus is not disturbed during removal or insertion. Carelessness in this area of design causes problems for many system designers.

P. Hardware Implementation

Depending on the complexity of the bus, the communication protocol can be either integrated into the power device controller or emulated in software. In some cases a portion of the protocol is implemented in hardware while the remainder is performed in software. There are many factors to consider when making this choice. Integrated peripherals typically provide the best performance while decreasing the software overhead. Support for busses like the CAN bus, USB, and Ethernet are done with integrated peripherals. These protocols are considered too complex to perform everything is software.

Simpler buses like I²C and SMBus are typical targets for software emulation, known as "bit banging". However, many system designers have had their projects run very late due to this decision. The attraction is usually lower cost in hardware but the specifications may be difficult to meet without dedicating significant resources to the protocol. For example, start bit detection used in I²C derived protocols requires an interrupt service or a very fast clock service routine. A start bit may occur less than 5 µs after the last clock for the 100 kHz specification.

Many low cost microcontrollers, such as those from TI, Microchip Technology and Atmel, have a generic clocked data peripheral that implements the fast service requirements of a bus while leaving the command service to software.

IV. Recommendations By Bus Type

A. RS-232

1. Advantages

The RS-232 is typical implemented using a UART (Universal Asynchronous Receive Transmit) peripheral although software can also be used with general purpose I/O (GPIO) ports to build the interface. RS-232 provides a simple and relatively low cost point to point communication method. Typical RS-232 implementations provide parity for simple error detection as well as framing error detection which detects a missing stop bit. Other optional levels of sophistication extend to flow control where device with small buffers can hold off data transmission until the previous received data can be retrieved. Typical gate count required for minimal RS-232 communication is about 2000.

2. Disadvantages

Disadvantages include the limitation to point to point communication for most applications. In order to access multiple power devices the communication lines must be either multiplexed or the host controller must have multiple interface circuits. Either of these methods adds cost to the implementation.

The other cost adder is an accurate oscillator. The clock mismatch between the sending and receiving devices should not exceed 3.3%. Most designers use a 2% mismatch limit for the design specification. This mismatch limitation restricts the choice of oscillator to either crystals or high quality resonators. The lowest cost crystal is the type used in the real-time clock circuits. This crystal oscillates at 32.768 kHz and costs less than $0.10 in high volume. The typical UART requires a clock frequency that is 16 times the baud rate. So unless there is a circuit to multiply the clock frequency, only 2k baud can be used.

3. Recommendation

RS-232 is OK for simple point-to-point communications especially when using only a logic level interface. Adding the driver for full EIA specifications will add cost. The oscillators needed in each device will add cost but a practical solution may be to use a common clock source.

B. RS-485

1. Advantages

The RS-485 communication interface adds a two wire differential transceiver to the RS-232 interface and adds the capability for true multi-drop communication. The RS-485 standard defines that these transceivers can be used to interface a total of 32 devices on the bus. The differential two wire interface provides very good noise immunity for high speed and relatively long distance communication. If using RS-485 with the full EIA interface, then the specification also provides some immunity to ground level differences between devices on the bus.

The software driver provides for the multi-point communication and defines the addressing strategy. The simplest implementation is for a single master bus. For long distance communication Category 5 cable can be used for a low cost connection.

2. Disadvantages

The principle disadvantage is the added cost of the transceiver. In very high quantities the price is in the range of $1 to $1.50. The other disadvantages are similar to the RS-232 communication discussed earlier such as the oscillator requirements.

3. Recommendation

RS-485 communications provide a solution for long distance applications requiring good noise immunity and moderate speed. Good applications are for shelf-to-shelf, rack-to-rack communications. RS-485 is also an acceptable choice for longer distance communication such as between a power system controller in a stand-alone 48 V power system and a data center control room.

C. I²C

1. Advantages

I²C is a simple two wire clocked data multi-drop bus. The main advantage is it low cost of implementation. A simple I²C peripheral requires less than 900 gates to implement. The interface may also be implemented using software and GPIO. This communications bus does not require oscillator matching between devices. When an I²C bus is designed to meet the I²C specification requirements, this bus provides a good solution to short haul communications.

2. Disadvantages

The main disadvantage is noise sensitivity. Noise can cause data corruption and erroneous detection of Start or Stop conditions. Bus capacitance can also result in edge detection delay causing data corruption and failure to detect Stop or Start conditions. The main problem with I²C implementations are due to system designs that do not meet the specifications of total bus capacitance, data hold times and start/stop requirements.

For example, a typical I/O pin on a microcontroller has a capacitance of about 10 pF. The maximum allowed total bus capacitance is 400 pF, so no more than 40 devices can be placed on the I²C bus before this specification is violated. In addition, capacitance from PCB traces might add about 1 pF per inch of route in a typical application board which would further reduce the total number of devices that can be added to the bus.

Like any multi-drop bus, an active failure on a communication line may shutdown all bus communication.

Table 1 shows parameters that can cause problems with I²c operation if not properly obeyed in the design.

3. Recommendation

The I²C bus is a very low cost solution for a local area power management such as on-board power systems or at the shelf level. A better choice is the SMBus which offers the same low cost but more robustness.

Table 1. 100 kHz I²C Parameters That May Cause Issues

Parameter	Min	Max
Total Bus Capacitance		400 pF
Start Hold Time Before Clock Generation	4 µs	
Stop Set-Up Time	4 µs	
Data Set-Up Time	250 ns	
Clock And Data Rise Time		1 µs
Clock And Data Fall Time		300 ns

D. SMBus

1. Advantages

This standard adds to I²C specification requiring stronger sink capabilities and minimum clock rate. The standard also calls for bus time-out values that can help detect data line locked low conditions. In multi-master application, the SMBus also calls out the arbitration and maximum bus time before a bus idle condition must occur. The specification also includes an optional Packet Error Checking (PEC) for error detection. The SMBus specification requires that all slave devices acknowledge their address. This requirement is a benefit for power management as it informs the host that the device is present. This bus has been used for more than 10 years in notebook battery management. SMBus needs only about 1700 gates to implement excluding address arbitration.

Table 2 shows improvements the SMBus made that make it a much more robust bus than I²C.

2. Disadvantages

Like the I²C bus, the system must be designed to meet the specification.

3. Recommendation

The SMBus is the best choice for local area power system, either on-board or at the shelf level. It is as low cost as I²C bus but more robust with timeouts, packet error checking and an interrupt line (SMBALERT#).

Table 2. SMBus Timing Parameters Improve Bus Traffic Management

Parameter	Min	Max
SMBus Operating Frequency	10 kHz	100 kHz
Data Hold-Time Prior To Next Clock Generation	300 ns	
Master Cumulative Clock Low Time		10 ms
Slave Cumulative Clock Low Time		30 ms
Maximum Clock Low Time		35 ms

E. SPI Bus

1. Advantages

The SPI Bus is a very simple clocked data multi-drop bus. The addressing of the devices on the bus is accomplished using individual chip select lines to each device. This reduces the circuitry required on each slave device which results in lower cost. The SPI bus can also be used to support higher data transfer speeds due to its simple addressing structure. With the SPI bus, full duplex operation is possible. Traditionally SPI is used to support very simple devices such as external memory and simple peripherals. The typical SPI slave interface requires less than 1000 gates for implementation.

2. Disadvantages

Each device requires bus connections with a total of N+3 bus lines. Although the chip select addressing mode greatly simplifies the addressing of slave devices it complicates buses that require alert functionality.

A key disadvantage of the SPI bus is that there is no formal specification and no means of confirming compliance.

3. Recommendation

The SPI bus provides a simple bus that may be used to connect local devices to a host device or to extend the memory or peripherals.

F. CAN bus

1. Advantages

CAN bus is 2 wire multi-master differential bus. The bus supports up to 1Mbit/s transfers for buses as long as 40m. The specification is allows for high reliability transmission in very noisy environments. The noise and fault tolerance are the principle benefits of the CAN bus. CAN bus is available on many intermediate and high-end microcontrollers.

2. Disadvantages

The CAN bus requires an oscillator with a tolerance better than 1.6%. It is a messaging based protocol and does not have a fixed means of setting device addresses. The data frame is large compared to the amount of data that is typically transferred in digital power management (16 bits) making the bus busier than perhaps necessary. The implementation size is about 4000 gates. The implementation is so complex that it requires integrated control circuitry. Emulating in software ("bit banging") is not considered feasible.

3. Recommendation

The CAN Bus offers a solution for longer distance applications requiring good noise immunity and moderate speed. Good applications are for shelf-to-shelf, rack-to-rack communications. RS-485 is also an acceptable choice for longer distance communication such as between a power system controller in a stand-alone 48 V power system and a data center control room.

G. Local Interconnect Network (LIN) Bus

1. Advantages

LIN bus [30] is a single wire bus supporting single master and multiple slaves. This bus is a lower cost alternative to CAN bus and provides reasonable noise immunity. It is based on the common UART byte interface which allows many common microcontrollers to emulate the LIN bus in software. The LIN bus uses a checksum to detect data errors.

2. Disadvantages

The oscillator tolerance needs to be better than 2% between master and slave during any frame transfer. The maximum transfer speed is 20 kbit/s. The bus interface is open drain and uses a pull-up resistor so all of the cautionary comments about specification limits on bus capacitance apply. Data frame length includes a header that is longer than other simple interface methods. The LIN bus takes about 3000 gates to implement.

3. Recommendation

The LIN bus has no advantage over other lower cost and more available buses for power system management. The SMBus is recommended for simpler, lower cost applications. For longer or more noise immune data communication, use either the RS-485 or CAN buses.

H. USB

1. Advantages

The USB is a well supported interface with considerable hardware and software support. USB provides a simple interface into PC operating systems. The USB is very good at hot swapping.

2. Disadvantages

The USB requires a hub for communications to multiple devices. The USB function needs about 15K gates for implementation on a power device. The USB is a single master bus. All communications are initiated by the hub. Attached devices are not able to independently signal the host system that they need attentions. For applications like power system management, this means that the host must constantly poll all of the attached devices.

Another disadvantage to the USB is that the use of the name or logo requires formal compliance testing, an unwanted expense for most power system management applications.

3. Recommendation

A good use for the USB in power system management is to provide a familiar interface for service people to access power system controllers, such as those found in telephone central office battery plants. This interface has typically been an RS-232 serial port. As serial ports disappear from personal computers, these serial ports are expected to be replaced by USB ports.

I. Ethernet

1. Advantages

Ethernet is very good for long haul and Internet connectivity.

2. Disadvantages

Ethernet packet size is not efficient for the typical message traffic expected for power management. Requires message based transactions, not memory read-write transactions. Ethernet requires about 20K gates to implement and the software support usually is implemented using an embedded operating system.

3. Recommendation

Ethernet is recommended for wide area network access to power system controllers such as those found in central office battery plants or wireless base stations. It is especially recommended for those power system controllers that are hosting embedded Web servers.

V. Recommendations By Application

A. On-Board Power Systems

For on-board power systems, the SMBus is the recommended data communications bus. It offers the low cost of I^2C but with a much greater robustness. The second choice would be the I^2C bus due to its low cost and wide availability of devices with integrated silicon solutions.

B. Shelf Or Chassis Level Power Systems

For shelf or chassis level systems, such as those mounted into 19 inch racks, the SMBus is again recommended if the total bus capacitance allows the specification to be met completely.

If the SMBus is not suitable, then consider SMBus for the on board communication with a bridge to RS-485 for the slot-to-slot or slot-to-host communication. The second choice to the RS-485 for the slot-to-slot or slot-to-host communication would be the CAN bus.

C. Shelf-To-Shelf Or Rack-To-Rack Communication

For power system management protocols that require communication from shelf-to-shelf or from rack-to-rack, RS-485 is recommended. The second choice would be the CAN bus.

D. Facility Or Campus Level Communication

For power system management within a small facility, where the transmission distances are 100 meters or less, either the RS-485 or CAN bus can be considered. At these distances, using Ethernet starts to have some advantages. For example, if the power system manager in the system needs an embedded Web server to provide an HTML interface to the facility level controller, then Ethernet would be the right choice.

E. PC to Power System Manager Communication

For larger power systems, such as those found in a telephone central office or even a wireless base station site, connectivity to a personal computer is often required. A service person might carry a laptop computer that they want to plug into the power system to monitor status or make configuration adjustments. In this case, there are two possible connections.

One is USB. This provides a familiar and hot pluggable interface for the service person's computer. One disadvantage is that the computer will need more complex software to provide the user interface and the communication with the power system manager.

The other choice is Ethernet. If the power system manager includes an embedded Web server, then the service person's computer only needs a Web browser to communicate with the power system. This simplification of the personal computer requirements often overcomes the additional cost of the embedded Web server in the power system manager.

VI. Summary

This paper has provided a comprehensive overview of data communication issues for power system management protocols. The requirements of a power system management communication were reviewed. The attributes of data communication buses were presented and analyzed for their usefulness and appropriateness for power system management applications. Recommendations were then made for the best use of each of several common data communication buses. Also, for several common power system management applications, recommendations were made for the most appropriate bus for those applications.

There is no one "right" data communication bus for all applications. It is always best to have the right tool for the job. That said, for power system management systems that are localized to a single circuit board, shelf level/rack mount system or those in enclosures small enough to have a good common ground, the SMBus is recommended as the data communication bus of choice. It is as low cost as I²C but far more robust. It also supports features like the SMBALERT# signal which can be used as an interrupt line and removes the need for continuous polling. For longer distance communications, such as between systems mounted in different rack, then either RS-485 or Can buses would be a good choice.

REFERENCES

[1] R. White, "Digital Power System Management", Proceedings Of The Applied Power Electronics Conference, APEC 2006, Vol. 1 pp. 176-182

[2] R. White, "Considerations For A Power System Management Protocol", Darnell Digital Power Forum 2004, Presentation 5-1

[3] P. F. Gensinger and E. A. Rosin, "Microprocessors in power control, the time has come", Proceedings Of The Third International Telecommunications Energy Conference, INTELEC 1982, pp. 1-6

[4] T. V. Papathomas and R. Suderi, "Stored program control of dc power plants", Proceedings Of The Third International Telecommunications Energy Conference, INTELEC 1982, pp. 18-23

[5] W. S. Watson, "Distributed energy management systems using the public telephone network", Proceedings Of The Third International Telecommunications Energy Conference, INTELEC 1982, pp. 1-6

[6] Charles Butala, HP, Marlborough, Massachusetts, Private conversation

[7] Intel et al, "System Management Bus Specification", Revision 1.0, September 15, 1995.

[8] P. Birman and S. Nercessian, "Programmable power supplies use switch mode topologies", Electronic Products, March 1995, pp. 33-34.

[9] A. Akiyama, T. Nakamura, M. Yoshida, T. Kubo, N. Yamamoto and T. Katoh, "KEKB Power Supply Interface Controller Module", Proceedings Of The International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPS 1997

[10] T. T. Nakamura, A. Akiyama, T. Katoh, Ta. Kubo, N. Yamamoto and M. Yoshida, "Magnet power supply control system in KEKB accelerators", Proceedings Of The International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPS 1999, pp. 406-408

[11] Intel, NEC, Hewlett-Packard and Dell, "IPMI – Platform Management FRU Information Storage Definition", Revision 1.1, 27 September 1999

[12] R. Zahn, "Digital multi-phase power control for advanced processors", Intel Technology Symposium, Power Track Presentation 9, September 2002

[13] S. Ohr, "Intersil teams with Primarion to power high speed processors", EETimes, 15 October 2001

[14] A. Burstein, "Volterra's 3rd generation integrated switching regulators for Intel microprocessor power", Intel Technology Symposium, Power Track Presentation 8, September 2002

[15] Summit Microelectronics, "Summit Microelectronics Expands Power Management Family With New Sequencing Power Supply Controller for Point-of-Load Applications", Press Release, August 1, 2002

[16] L. Pechi, "Z-One digital IBA – logistic and performance benefits", Proceedings of the Digital Power Forum 2004, Paper 4-1.

[17] "Digital power management protocol", Power Electronics Technology, October 1, 2004

[18] http://www.pmbus.org/news.html

[19] ANSI/TIA-232-F, "Interface Between Data Terminal Equipment And Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange", Telephone Industry Association, 1997

[20] ANSI/TIA/EIA-422-B, "Electrical Characteristics of Balanced Voltage Digital Interface Circuits", Telephone Industry Association and Electronics Industry Association, 1994

[21] "The I²C –Bus Specification", Revision 2.1, Philips Semiconductors, January 2000

[22] "System Management Bus Specification", Revision 1.1, Intel Corporation et al, December 11, 1998

[23] "System Management Bus Specification", Revision 2.0, Intel Corporation et al, August 3, 2000

[24] ANSI/TIA/EIA-485-A, "Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems", Telephone Industry Association and Electronics Industry Association, 1998

[25] "CAN Specification", Revision 2.0, Robert Bosch Gmbh, 1991

[26] ISO 11898-1, "Road Vehicles - Controller Area Network (CAN) - Part 1: Data Link Layer And Physical Signaling", Revision 03, December 1, 2003

[27] http://www.maxim-ic.com/1-Wire.cfm

[28] Kevin Gingerich, "The RS-485 unit load and maximum number of bus connections", Texas Instruments, 2004

[29] "Universal Serial Bus Specification", Revision 2.0 and errata, Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, and Philips, April 27, 2000

[30] "LIN Specification Package", Revision 2.0, LIN Consortium