Introduction To The PMBus™

Presented By
Robert V. White, Artesyn Technologies;
Chair, PMBus Specification Working Group

May 2005

www.powerSIG.org
What Is PMBus?
What Is PMBus?

A Standard Way To Communicate With Power Converters Over A Digital Communications Bus
Configure 

System 

Control 

Monitor 

PMBus Device 

Control 

Monitor 

Configure
Configure

System

Configure

Control

System Maintenance Processor

– Or –

Spare Gates In An FPGA

– Or –

Laptop Computer

– Or –

Dedicated Controller IC

– Or –

General Purpose Microcontroller

– Or –

Automatic Test Equipment
PMBus Is An Open Standard

• Owned By The System Management Interface Forum (SM-IF)
  – SM-IF Membership Is Open To All
• Royalty Free
• Released Specifications Freely Available
• Works With All Types Of Power Converters
  – AC-DC Power Supplies
  – Isolated DC-DC And Bus Converters
  – Non-Isolated Point-Of-Load Converters
  – Microprocessor Power Converters
PMBus: What It Is Not

• Not A Standard For A Power Supplies Or DC-DC Converters
  – No Form Factor, Pin Out, Efficiency, Etc.
  – Alliances Like POLA And DOSA Will Define

• No Converter-To-Converter Communication
  – Such As Current Share And Analog Voltage Tracking
  – Left To The IC And Power Supply Manufacturers
  – Including These Would Inhibit Future Innovation
Some Basic PMBus Requirements

• PMBus Devices Must Start Up Safely Without Bus Communication
• PMBus Devices Can Be Used With Or Without A Power System Manager/Controller
• PMBus Devices Support “Set And Forget”
  – Can Be Programmed Once At Time Of Manufacture
  – Then Operate Forevermore Without Bus Communication
• Defaults From Either/Or
  – Non-Volatile Memory
  – Pin Programming
Who Is PMBus?
# PMBus Founders And Supporters

## Founders

<table>
<thead>
<tr>
<th>Founders</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artesyn Technologies</td>
</tr>
<tr>
<td>Texas Instruments</td>
</tr>
<tr>
<td>Microchip Technology</td>
</tr>
<tr>
<td>Volterra Semiconductor</td>
</tr>
<tr>
<td>Emerson/Astec</td>
</tr>
<tr>
<td>Intersil</td>
</tr>
<tr>
<td>Summit Microelectronics</td>
</tr>
<tr>
<td>Zilker Labs</td>
</tr>
</tbody>
</table>

## Supporters

- **POLA** - Point-of-Load Alliance
- **DOSA** - Distributed-power Open Standards Alliance
System Management Interface Forum, Inc.
System Management Interface Forum, Inc.

www.powerSIG.org
System Management Interface Forum, Inc.

SM-IF Membership Open To Any And All

www.powerSIG.org
PMBus Specification
PMBus Specifications

• Part I: Transport
  – SMBus V1.1 Plus Extensions
  – Addressing
  – Hardwired Signals
    • Example: CONTROL Signal (On/Off Function)

• Part II: Command Language
  – Configuration
  – Control
  – Status Monitoring
  – Fault Management
  – Information Storage: Inventory, User Data, Etc.
Why SMBus?

- Low Cost Like I²C
- More Robust Than I²C
  - Timeouts Force Bus Reset
- More Features Than I²C
  - SMBALERT# Line For Interrupts
  - Packet Error Checking (PEC)
  - Host Notify Protocol
- Generally Electrically Compatible With I²C
Addressing I²C Shortcomings

• “Noise Sensitivity” – Edge Triggering
  – False START: Timeouts Force Reset
  – False STOP: PMBus Devices Detect Failed Transmissions As Faults

• “Noise Sensitivity” – Corrupt Data
  – Data Rates Permit Digital Filtering
  – Packet Error Checking (PEC)
  – Every Value That Can Be Written Can Be Read
Addressing I²C Shortcomings

• Slave Device Hangs Bus
  – Timeouts Force Device Reset

• Requires Retrieving Device Information By Polling
  – SMBALERT# Line Acts As An Interrupt
  – Automatic Bitwise Arbitration Of Simultaneous Requests

• No More Than 8 Devices Of A Type On One Bus
  – No Central Address Control Bureaucracy
  – Over 100 Device Addresses Available
Other Issues

• Fault Tolerance
  – Physically Failed Devices Are A Problem With Any Bus
  – Must Add Isolating Switches And Multiplexers

• 400 pF Maximum Bus Capacitance Requires Repeaters Or Bridges For:
  – Large Numbers Of Devices
  – Long Distance
PMBus™ Connections

SYSTEM HOST/ BUS MASTER

CONTROL SIGNAL

SERIAL BUS DATA

SERIAL BUS CLOCK

SMBALERT# SIGNAL

UNIT #1

SMBALERT# CONTROL
DATA
CLOCK

WP

WRITE PROTECT

PHYSICAL ADDRESS

UNIT #2

SMBALERT# CONTROL
DATA
CLOCK

WP

WRITE PROTECT

PHYSICAL ADDRESS

UNIT #N

SMBALERT# CONTROL
DATA
CLOCK

WP

WRITE PROTECT

PHYSICAL ADDRESS

©2005 System Management Interface Forum
PMBus™ Connections

SYSTEM HOST/BUS MASTER

- CONTROL SIGNAL
- SERIAL BUS DATA
- SERIAL BUS CLOCK

Data

Clock

UNIT #1

- SMBALERT# CONTROL
- DATA CLOCK WP
- WRITE PROTECT
- PHYSICAL ADDRESS

UNIT #2

- SMBALERT# CONTROL
- DATA CLOCK WP
- WRITE PROTECT
- PHYSICAL ADDRESS

UNIT #N

- SMBALERT# CONTROL
- DATA CLOCK WP
- WRITE PROTECT
- PHYSICAL ADDRESS

©2005 System Management Interface Forum
PMBus™ Connections

Optional CONTROL Signal Is For On/Off Control
Optional SMBALERT# Signal Acts As An Interrupt Line And Activates The Alert Protocol
PMBus™ Connections

SYSTEM HOST/ BUS MASTER

UNIT #1

UNIT #2

UNIT #N

SMBALERT# SIGNAL
CONTROLL SIGNAL
SERIAL BUS DATA
SERIAL BUS CLOCK

WRITE PROTECT
PHYSICAL ADDRESS

WRITE PROTECT
PHYSICAL ADDRESS

WRITE PROTECT
PHYSICAL ADDRESS

Required Hardwired Pins To Set Physical Address

©2005 System Management Interface Forum
PMBus™ Connections

Optional Write Protect Pin To Prevent Unwanted Data Changes
PMBus™ In A Large System

System Power Bus → Bus Converter → POL w/ PMBus → VOUT1

Local/On-Board Power Bus

POL w/ PMBus → VOUT2

POL w/ PMBus → VOUT3

STD POL → VOUT4

STD POL → VOUT5

Host — Power System Communication Bus (e.g. IPMI Or RS-485)

Host System → Power System Interface

PMBus Interface IC

Analog Control Lines (Sense, Enable, Trim, Power Good)
**PMBus™ In A Large System**

Existing System Processor
- Or -
Extra Gates In An FPGA
- Or -
General Purpose Microcontroller

Host — Power System Communication Bus (e.g. IPMI Or RS-485)

Host System

Power System Interface

PMBus

Local/On-Board Power Bus

- VOUT1
- VOUT2
- VOUT3
- VOUT4
- VOUT5

Analog Control Lines (Sense, Enable, Trim, Power Good)

POL w/ PMBus

STD POL

Std POL

PMBus Interface IC
Typical Packet Structure

ADDRESS BYTE COMMAND BYTE DATA BYTE 1

S 7 6 5 4 3 2 1 0 A 7 6 5 4 3 2 1 0 A 7 6 5 4 3 2 1 0 A

DATA BYTE 2 DATA BYTE N OPTIONAL PEC BYTE

7 6 5 4 3 2 1 0 A 7 6 5 4 3 2 1 0 A 7 6 5 4 3 2 1 0 A P

S START Signal From Host System

O READ/WRITE# Bit

A ACKNOWLEDGE Signal From Converter

P STOP Signal From Host System

©2005 System Management Interface Forum
Addressing

• PMBus Devices Use A 7 Bit Address Per The SMBus Specification
  – Provides More Than 100 Possible Device Addresses After Allowing For Reserved Addresses
• No I²C Style Address Control Assignments Or Limitations
• PMBus Users Can Expect Device Addresses To Be Set By A Mix Of:
  – Hardwired Address Pins
  – High Order Address Bits Set By The PMBus Device Manufacturer
Addressing (cont’d)

• PMBus Device Manufacturers Will Trade Off Cost Of Pins Versus Address Flexibility
• Expect Device Makers To Offer Tri-State Pins Or Resistor Value Programming
• Examples Of The Possibilities
  – 3 Tri-State Pins => 27 Addresses
  – 1 Resistor Programmed Pin => 16–32 Addresses
Multiple Output Units And Paging

• Paging Allows One Physical Address To Be Used To Control Multiple Outputs
  – One Address Per Physical Unit
  – One Page Per Output
  – Pages Contain All The Settings Of Each Output

• Paging Process
  – Set Page For Output Of Interest
  – Send Commands
    • Configure, Control, Read Status
Paging: Multiple Output Units
Paging: Non-PMBus Device Adapter

- PMBus Bus Switch
- PMBus To Non-PMBus Device Adapter
  - PAGE 0 Memory
  - Device 0 Interface
  - Non-PMBus Device
  - PAGE 1 Memory
  - Device 1 Interface
  - Non-PMBus Device
  - PAGE 2 Memory
  - Device 2 Interface
  - Non-PMBus Device
- Address Pins
Paging: Non-PMBus Device Adapter

Example Device: Analog Margin/Sequence Controller With PMBus Interface

Example Device: POL Converter With An Analog Interface

PMBus To Non-PMBus Device Adapter

Page 0 Memory

Device 0 Interface

Page 1 Memory

Device 1 Interface

Page 2 Memory

Device 2 Interface

PMBus Bus Switch

Non-PMBus Device

Non-PMBus Device

Non-PMBus Device
Command Language

• Extensive And Comprehensive
• Commands Take Effect Immediately
• Every Value That Can Be Written Can Be Read
Command Language

- Extensive And Comprehensive
- Commands Take Effect Immediately
- Every Value That Can Be Written Can Be Read

Not All Devices Support All Commands!

Devices Will Support Commands Appropriate To Their Intended Application And Price Point
Concept: Setting The Output Voltage

VOUT_MARGIN_HIGH → VOUT_COMMAND → VOUT_MARGIN_LOW → 3:1 Mux → OPERATION Command

VOUT_TRIM → VOUT_CAL → VOUT_DROOP → X → IOUT

VOUT_MAX → Limiter → VOUT_SCALE_LOOP → “Reference Voltage Equivalent”
Memory And Startup Concepts

1. Hard Coded Parameters
2. Pin Programmed Values
3. Operating Memory (Volatile)
4. Default Store (Non-Volatile)
5. User Store (Non-Volatile)

- STORE_DEFAULT
- RESTORE_DEFAULT
- STORE_USER
- RESTORE_USER

BUS COMMUNICATION

©2005 System Management Interface Forum
Memory And Startup Concepts

1. Hard Coded Parameters

2. Operating Memory

3. Conceptual Volatile Memory Store For Device’s Operating Parameters

4. BUS COMMUNICATION
Memory And Startup Concepts

At Device Power Up, Values Hard Coded Into The PMBus Device Are Loaded First Into The Operating Memory
Memory And Startup Concepts

Next, Pin Programmed Values Are Loaded Into Operating Memory. This Overwrites Any Previously Loaded Values.
Memory And Startup Concepts

Default Values

Next, Values From The Non-Volatile Default Store (If Provided) Are Loaded. This Overwrites Any Previously Loaded Values.

Operating Memory (Volatile)

Default Store (Non-Volatile)

User Store (Non-Volatile)

BUS COMMUNICATION
Memory And Startup Concepts

Next, Values From The Non-Volatile User Store (If Provided) Are Loaded. This Overwrites Any Previously Loaded Values.
Memory And Startup Concepts

Bus Communication

Next, Values Sent Via The SMBus Are Loaded. This Overwrites Any Previously Loaded Values.

Operating Memory (Volatile)

1. Hard Coded Parameters
2. Default Store (Non-Volatile)
3. RESTORE_DEFAULT
   - Default Store (Non-Volatile)
4. RESTORE_USER
   - User Store (Non-Volatile)
5. BUS COMMUNICATION

©2005 System Management Interface Forum
Memory And Startup Concepts

Used To Store A Snapshot Of The Device’s Operating State. When Power Removed And Restored, Device Can Resume Operation From Its Last Programmed State.
Setting The Output Voltage

• Two Step Process
• Step 1: Set Or Determine Voltage Command Mode
• Step 2: Send Output Voltage Command

• Output Voltage Command Modes
  – Linear In LSB
  – Popular VIDs
  – “IPMI Like” Equation Mode
On/Off Control

• Two Inputs Control Whether A PMBus Device Is Operating Or Not
  – Hardwired CONTROL Pin (Programmable Polarity)
  – OPERATION Command From The Bus
• On/Off Control Totally Programmable
• CONTROL Pin Options
  – Active High Or Active Low
  – Followed Programmed Sequencing Or Shutdown Immediately
On/Off Control Options

• “Always On”
  – Device Providing Output Power Anytime Input Power Is Present

• Respond To CONTROL Pin, Ignore OPERATION Command

• Respond To OPERATION Command, Ignore Control Pin

• Respond To Both CONTROL Pin And OPERATION Command
  – An “Off” From Either Turns Output Off
### OPERATION Command Data Byte

One Command Used To Set Operation Mode: On/Off/Margin

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>XX</td>
<td>XX</td>
<td>XX</td>
<td>IMMEDIATE OFF (No Sequencing)</td>
<td>N/A</td>
</tr>
<tr>
<td>01</td>
<td>XX</td>
<td>XX</td>
<td>XX</td>
<td>OFF (With Sequencing)</td>
<td>N/A</td>
</tr>
<tr>
<td>10</td>
<td>00</td>
<td>XX</td>
<td>XX</td>
<td>ON</td>
<td>OFF</td>
</tr>
<tr>
<td>10</td>
<td>01</td>
<td>01</td>
<td>XX</td>
<td>ON</td>
<td>MARGIN LOW (Ignore Fault)</td>
</tr>
<tr>
<td>10</td>
<td>01</td>
<td>10</td>
<td>XX</td>
<td>ON</td>
<td>MARGIN LOW (Act On Fault)</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>01</td>
<td>XX</td>
<td>ON</td>
<td>MARGIN HIGH (Ignore Fault)</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10</td>
<td>XX</td>
<td>ON</td>
<td>MARGIN HIGH (Act Of Fault)</td>
</tr>
</tbody>
</table>
Group Commands/Operation

• Used When Multiple Units Need To Execute A Command Simultaneously

• One SMBus Transaction Used To Send Commands To Multiple Addresses
  – Sent In One Large Packet Using Repeated STARTs

• Can Be Same Or Different Commands
  – Example: Command One Unit To Margin Low And All Others To Margin High

• Commands Are Executed When SMBus STOP Condition Received
Interleaving

• INTERLEAVE Command Sets
  – Group Number
  – Number Of Units In The Group
  – Switching Order Within The Group

Example Of INTERLEAVE Command Operation

\[ T_{delay}(Unit \ X) = \frac{Interleave \ Order \ Of \ Unit \ X}{Number \ In \ Group} \times T_s \]
Many Other Configuration Commands

- Maximum Output Voltage
- Maximum Output Power
- Voltage Scale For External Divider Network
- Maximum Duty Cycle
- Switching Frequency
- Turn On/Off Levels For Input Voltage
- Current Scale For Current Sense Resistance
- Current Measurement Calibration
Sequencing: Event Driven

- Event Driven Sequencing Is Closed Loop
- Requires Power System Manager To Close The Loop
Sequencing: Time Driven Commands

- Open Loop: Does Not Require Power System Manager

![Diagram showing time-driven commands with Ton-Rise, Ton-Delay, Ton-Max, Toff-Delay, Toff-Fall, and Toff-Max Fault.]
Fault Management: Input

**Voltage**
- OV FAULT
- OV WARN
- UV WARN
- UV FAULT

**Current**
- OC FAULT
- OC WARN

Related Commands: VIN_ON, VIN_OFF
Fault Management: Input

Faults

Faults Cause Action. Response To A Fault Can Be Programmed

Related Commands: VIN_ON, VIN_OFF
Fault Management: Input

<table>
<thead>
<tr>
<th>Voltage</th>
<th>Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>OV FAULT</td>
<td></td>
</tr>
<tr>
<td>OV WARN</td>
<td></td>
</tr>
<tr>
<td>UV WARN</td>
<td></td>
</tr>
<tr>
<td>UV FAULT</td>
<td></td>
</tr>
</tbody>
</table>

**Warnings**

Warnings Do Not Directly Cause Action. Status Bits Are Set And Host Notified (If Device Supports)

Related Commands: VIN_ON, VIN_OFF
Fault Management: Output

Related Commands:
POWER_GOOD_ON, POWER_GOOD_OFF
Other Fault Management

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Other Faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>OT FAULT</td>
<td>Fan Fault 1</td>
</tr>
<tr>
<td>OT WARN</td>
<td>Fan Fault 2</td>
</tr>
<tr>
<td>UT WARN</td>
<td>Current Share</td>
</tr>
<tr>
<td>UT FAULT</td>
<td>Power Limiting</td>
</tr>
<tr>
<td></td>
<td>Communication</td>
</tr>
<tr>
<td></td>
<td>And more...</td>
</tr>
</tbody>
</table>
Fault Response Programming Byte

### RESPONSE
00 - CONTINUE
01 - DELAYED OFF
10 - SHUTDOWN & RETRY
11 - INHIBIT

### DELAY TIME
XXX - NUMBER OF DELAY TIME UNITS

### RETRY
000 - LATCH OFF
001 - 110: RETRY COUNT
111 - CONTINUOUS

©2005 System Management Interface Forum
Notifying The Host Of A Fault

• Host Can Continuously Poll PMBus Devices
• PMBus Device Can Send An Interrupt
  – SMBALERT# Signal Is Optional
  – See The SMBus Specification For Details
• PMBus Device Can Become A Bus Master And Transmit Notice To System Host
  – Optional
  – Requires A More Sophisticated Host
Handling Unsupported Commands & Bad Data

• Choice 1
  – PMBus Device NACKS Command Or Data Byte
  – Reason Put Into Status Registers

• Choice 2
  – PMBus Device ACKs Everything, Processes Later
  – If An Unsupported Command Or Data Out Of Bounds Is Received:
    • Set CML Bit In STATUS_BYTE
    • Set Appropriate Bit In Status Registers (If Supported)
    • Notify Host (If Supported)
Status Reporting: 3 Levels Of Detail

Level 1:
STATUS_BYTE
Most Critical Info

Level 2:
STATUS_WORD
Adds More Important Info

Level 3:
Status Registers
Detailed Information
# STATUS_BYTE & STATUS_WORD

<table>
<thead>
<tr>
<th>High Byte</th>
<th>Low Byte</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UNKNOWN FAULT OR WARNING</strong></td>
<td><strong>STATUS_BYTE</strong></td>
</tr>
<tr>
<td><strong>Reserved</strong></td>
<td><strong>UNIT IS BUSY</strong></td>
</tr>
<tr>
<td><strong>POWER_GOOD Negated</strong></td>
<td><strong>UNIT IS OFF</strong></td>
</tr>
<tr>
<td><strong>MFR_SPECIFIC²</strong></td>
<td><strong>VOUT_OV_FAULT</strong></td>
</tr>
<tr>
<td><strong>INPUT FAULT OR WARNING</strong></td>
<td><strong>IOUT_OC_FAULT</strong></td>
</tr>
<tr>
<td><strong>IOUT_FAULT OR WARNING</strong></td>
<td><strong>VIN_UV_FAULT</strong></td>
</tr>
<tr>
<td><strong>VOUT_FAULTS OR WARNINGS</strong></td>
<td><strong>TEMPERATURE FAULT OR WARNING</strong></td>
</tr>
<tr>
<td><strong>7 6 5 4 3 2 1 0</strong></td>
<td><strong>COMM, LOGIC, MEMORY EVENT</strong></td>
</tr>
<tr>
<td></td>
<td><strong>OTHER FAULT OR WARNING</strong></td>
</tr>
</tbody>
</table>
Status Registers

<table>
<thead>
<tr>
<th>STATUS WORD</th>
<th>STATUS_BYTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Byte</td>
<td>Low Byte</td>
</tr>
<tr>
<td>UNK. FAULT OR WARNING</td>
<td>UNIT IS BUSY</td>
</tr>
<tr>
<td>Reserved</td>
<td>UNIT IS OFF</td>
</tr>
<tr>
<td>POWERGOOD Negated</td>
<td>VOUT_OV FAULT</td>
</tr>
<tr>
<td>Reserved</td>
<td>IOUT_OC FAULT</td>
</tr>
<tr>
<td>INPUT FAULT OR WARNING</td>
<td>VIN UV FAULT</td>
</tr>
<tr>
<td>IOUT FAULT OR WARNING</td>
<td>TEMPERATURE FAULT OR WARNING</td>
</tr>
<tr>
<td>VOUT FAULTS OR WARNINGS</td>
<td>COMM, LOGIC, MEMORY EVENT</td>
</tr>
<tr>
<td>OTHER FAULT OR WARNING</td>
<td>OTHER FAULT OR WARNING</td>
</tr>
</tbody>
</table>

7 6 5 4 3 2 1 0

STATUS_VOUT Register

STATUS_IOUT Register

STATUS_INPUT Register

STATUS_MFR² Register

STATUS_OTHER Register

STATUS_CML¹ Register

STATUS_TEMPERATURE Register

¹: CML: Communication, Memory, Logic
²: MFR SPECIFIC: Manufacturer Specific
Parametric Information

- Input Voltage
- Input Current
- Output Voltage
- Output Current
- Hold Up Capacitor Voltage
- Temperature
  - Up To 3 Sensors
- Fan Speed
  - Up To 2 Fans
- Duty Cycle
- Switching Frequency
Parametric Information

- Input Voltage
- Input Current
- Output Voltage
- Output Current
- Hold Up Capacitor Voltage
- Temperature
  - Up To 3 Sensors
- Fan Speed
  - Up To 2 Fans
- Duty Cycle
- Switching Frequency

**REMEMBER!**

Not All PMBus Devices Will Support All Commands!
Support Based On Application And Price Point
Manufacturer And User Data

• Manufacturer’s Information
  – Inventory Information (Model Number, Etc.)
  – Ratings Information (Input Voltage Range, Etc.)

• User Data
  – 32 Command Codes For PMBus Device Makers To Support User Inventory And Configuration Data
  – Example: Digital Control Loop Coefficients

• Manufacturer Specific Commands
  – 45 Command Codes Reserved For PMBus Device Makers To Implement Manufacturer Specific Commands
Data Integrity And Security

• Protecting Against Corrupted Transmissions
  – Packet Error Checking Can Be Used

• Unwanted Or Unintentional Data Changes
  – Write Protect Pin
  – WRITE_PROTECT Command
Summary

• PMBus Is A Flexible, Powerful Tool For Digital Power System Management
• Supports Both Embedded, Discrete Converters As Well As Complete, Purchased Converters
• Has The Features Needed By Nearly All Users
• Application Oriented Feature Sets Control Cost

More Information And Specifications At: www.powerSIG.org