PMBus – the Rising Star!

Once upon a time, there was an ugly step-child called "power" that was shoved in a corner where it could be hardly be seen and most definitely not heard. But as applications became more intelligent, the step-child began to grow with it. Now, the world is becoming more energy conscious and more mobile, hence our step-child is emerging into the bright lights of intelligent power management.

By Ramesh Balasubramaniam, Marketing Director, Enterprise Power, International Rectifier Corporation

Active power management in an electronic system is now a central design facet and no longer an afterthought. The challenges of configuring power supplies, sequencing multiple voltage rails, monitoring power supply status, determining faults and reacting to warnings and status are common to all electronic applications regardless of market. From this need, the industry standard Power Management Bus (PMBus™) was conceived. A typical application in a computing system using International Rectifier's PMBus enabled products is shown in Figure 1.

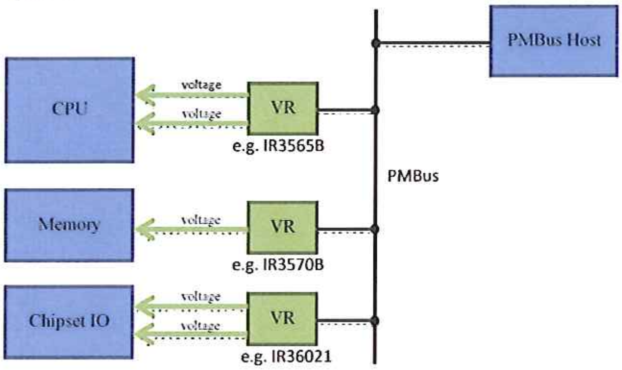

Figure 1: Typical PMBus system

History

The PMBus specification, developed in 2004 and first released at version 1.0 in 2005, is an open standard that is owned and controlled by the System Management interface Forum (SMIF). Adopter companies, such as International Rectifier, bring their market knowledge to these Forums to guide the definition and ratification of the specifications. To minimize implementation cost overhead and to make easy re-use of existing infrastructure, PMBus was conceived as a predefined command set that sits on top of the transport layer protocol of the SMBus™ (System Management Bus). SMBus, first released in 1995 and also owned and controlled by SMIF, is highly adopted in notebook computers and batteries as the industry standard way of controlling and monitoring battery power. SMBus itself, uses the physical and electrical layer protocol of the ubiquitous I2C (Inter integrated Circuit) bus developed by Philips in 1982.

Protocol

I2C is a multipoint, open drain, two wire bus (clock and data). It is generally implemented at low speed (around 100kHz) to ensure compatibility due to its widespread popularity, even though higher speed versions are defined. Multiple masters and multiple slaves may co-exist on the bus, each with their own unique 7-bit address. The I2C protocol consists of a START bit followed by a one byte address (7 bits of address and 1 bit for read/write direction), one or more bytes of data and a STOP bit. Whereas I2C is wide open, SMBus strictly defines the data packets into a few allowable protocols such as "Write Byte/Word", "Block Write/Read" and so on. The advantage is standardization around a few common protocols to ensure compatibility across manufacturers. SMBus also adds robustness to the basic I2C specification with the introduction of a bus timeout and the optional Packet Error Checking (PEC) protocol. A third wire, SALERT#, is added to allow Slaves to quickly signal the Master for faster fault handling. SMBus addresses follow the I2C format, however one address is reserved for an SMBus protocol called Alert Response Address which allows the Master to quickly find out which Slave signaled the SALERT# line.

The current PMBus specification is version 1.2 and was released in 2010. PMBus primarily defines the command set that operates over the SMBus protocol to provide a unified and standardized method of communicating with power management devices (Figure 2). In addition, PMBus adds a further data packet protocol, the "GROUP" command which allows multiple devices to be sent commands in a single large data packet that is the concatenation of address, command and data to each device.

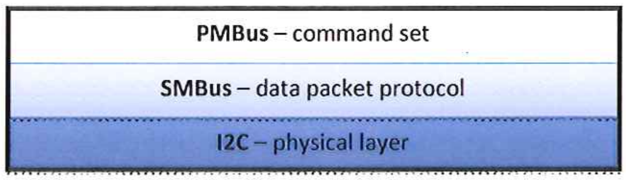

Figure 2: PMBus protocol layers

Command Set

The command set is wide enough to cover a broad range of power management applications such as AC/DC, DC/DC, fan control etc. and the approximately 200 commands can be loosely categorized as:

- Memory related e.g. STORE_DEFAULT_ALL
- On/Off related e.g. OPERATION & ON_OFF_CONFIG
- Output voltage related e.g. VOUT_COMMAND
- Configuration related e.g. FREQUENCY_SWITCH and PHASE

- Warnings and Faults related e.g. IOUT_OC_WARN_LIMIT
- Sequencing related e.g. TON_RISE
- Status related e.g. STATUS_TEMPERATURE
- Telemetry related e.g. READ_VOUT
- Inventory related e.g. MFR_ID
- Manufacturer ratings related e.g. MFR_POUT_MAX

International Rectifier offers products with varying amounts of commands for cost effectiveness as PMBus compliancy does not require all commands to be implemented. As well as standardizing the command set to ensure inter-operatability between manufacturers, an important part of the standardization is to specify the number formats to be used with all values. PMBus version 1.2 allows two universal number formats, Linear mode (most common) and direct mode. The linear data format expresses numbers according to the formula $X = Y.2^N$, where Y is the mantissa and N is the exponent. This is encoded within two bytes of data where the first 5 bits are the exponent (N) and the remaining 11 bits are mantissa (Y). Only VOUT related commands use a higher precision version where the mantissa (Y) is encoded as 16 bits and the 5 bit exponent (N) is set by the VOUT_MODE command. Less commonly used is the Direct mode which has the form $X = 1/m.(Y.10^{-R}-b)$ where m, Y and b are two bytes each and R is one byte.

With power management devices, it is critical that transmission and data errors are handled correctly. To that end, PMBus defines that commands not meeting the correct protocol transmission sequence or reporting a PEC error are flagged and rejected with the appropriate status register bits set. Similarly, invalid data or out-of-range data must be safely rejected, with the proper status bits being set.

Limitations

Although PMBus version 1.2 has been very widely adopted and today provides the backbone for intelligent power management, it is not without its limitations. Wide scale adoption has resulted in increased number of devices on the bus, thus throughput of data (especially telemetry) can be slow. The ALL CALL address can send the same command to every device in a compact form, while the GROUP command can send any command to any device in a long string. However, there is no mechanism to compactly target a small group of devices with the same command. For example, you may only want to turn off the voltage regulators to a particular sub-system and currently the best choice is to use the GROUP command to send the same (turn off) command to the targeted regulators. This requires exactly the same transmission length as sending individual turn off commands with the only advantage being that all regulators will turn off at the same time at the end of the entire transmission (Figure 3). Another common difficulty is deciphering the root cause of a fault on the bus as typically a faulted device causes many other devices on the bus to report faults due to the coupled nature of power components.

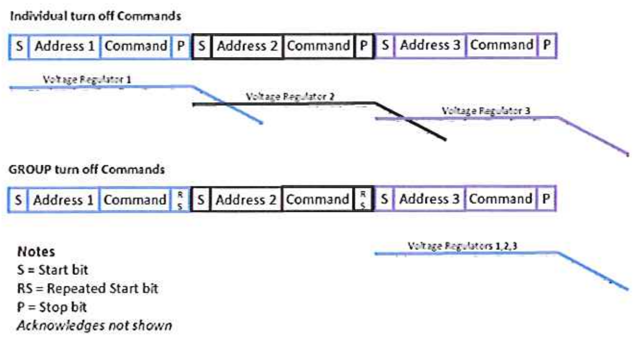

Figure 3: Sending commands to turn off a subset of devices

Looking Forward to PMBus 1.3

SMIF recognized that the success of PMBus had revealed some limitations, so in early 2013 a Working Group was set up. It was tasked to update the specification to address the limitations and encompass emerging usage models. As silicon processes move towards deep submicron geometries, leading edge processors such as CPUs, FPGAs and ASICs require to dynamically control their own voltages to optimize power and performance. A major update in PMBus 1.3 is the inclusion of Adaptive Voltage Scaling (AVS) through a complementary AVSBus (Figure 4). Further enhancements announced in the draft 1.3 specification in September 2013 include:

- Up to 1MHz bus speed for increased data throughput
- Fast Zone read/write protocol to multiple devices for increased data throughput
- Floating point number format for a wider range with higher precision
- Relative output voltage thresholds to allow warning/fault limits to track the output voltage
- AVSBus up to 50MHz for an ASIC to dynamically control its own voltage

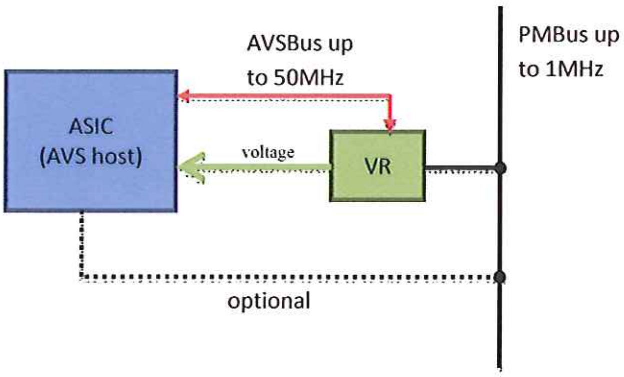

Figure 4: PMBus 1.3 with AVSBus

The final PMBus 1.3 specification is expected to be released around April 2014 and the world waits with bated breath as the little ugly stepchild of long ago finally makes its bow as one of the leading cast.

www.irf.com

PMBus is a registered trademark of SMIF
SMBus is a trademark of Intel
I2C is trademark of Philips semiconductor