The PMBus 1.3 specification addresses faster alert response, device discovery and system control by introducing Zone Reads and Zone Writes

Michael Jones - February 08, 2014

Previously I discussed the idea of a Global Process Call as a method for getting prioritized data from every PMBus™ device within a system. The industry listened and responded with a request for getting data from a targeted subset of devices. A desire for data writes to all or some devices was also voiced. One system engineers' pain points were being addressed: “The PoL’s just retry and alarm constantly. I can’t get to all of them fast enough to find out what’s going on!” A system architect was also complaining: “I’ve got systems that have modular cards with PoL’s on them. Now I need to tell my system exactly how it is configured in each SKU. I wish the system could figure that out itself, and do it quickly.”

The PMBus spec working group decided to respond. PMBus 1.3 further enhances its communication capability to include not only global calls, but calls for data from subsets of the devices in a system. Also, a write mechanism was added to write data to all or some of the devices synchronously. The capability to read or write to all or a subset of the system’s devices has now been defined as a Zone Read and a Zone Write. (The term Global Process Call will not be used in the PMBus 1.3 specification.)

The PMBus protocol has enabled the electronics industry to standardize communication to their power circuits. Controlling, configuring, and monitoring are now possible across many vendors’ power solutions. The underlying SMBus interface is not without its limitations, however. One might consider replacing the SMBus in its entirety with a cutting-edge high-speed interface. But, there are many concerns when attempting this. A new interface would be more complex and use more power. It would also require substantially more silicon area. Instead, enhancing the current interface could satisfy the need for faster communication while minimizing the impact on the devices. Zone Reads and Zone Writes are enhancements added to PMBus 1.3 to provide faster transactions using the existing SMBus interface.

The need for fast communication

Using the standard SMBus speed and protocol for fault handling and sequencing can cause system design issues. Currently, these functions are bus-access intensive.

Device address discovery is a feature that many system designers deal with when the number of devices on the bus can differ for each system configuration. This situation occurs, for example, when a system consists of optional cards. Figure 1 illustrates this scenario.
Fault handling requires fast communication, and should be dealt with quickly at the system level to minimize circuit damage. Figure 2 shows a PMBus system that needs to be serviced. Today, devices merely have the open-drain SALRT pin to tell the master that there is a problem. The master will not know which device pulled the SALRT pin low. It must, therefore, either poll every device to query its status, or use Alert Response Address to determine which device alerted and then query its status. Only then can the master act on the information that it has. This procedure can take hundreds of milliseconds and is not acceptable in the system implementation.
Figure 2. A PMBus System Master can quickly become overwhelmed with fault handling.

Event-based sequencing requires the system to have two-way communication to report and act on order-of-sequence information. Event-based sequencing is defined by a device starting up in a specific sequence, based on another device’s regulation status. In the existing SMBus implementation, the communication of “powergood” by an individual device is done by either setting its PG pin, which is individually monitored by the host controller, or by setting its SALRT pin and then waiting to be polled by the host controller. The host controller must then communicate to the next device in the sequence to start up. In real applications, a sequence order may require a millisecond device-to-device startup time. The current SMBus transport mechanism does not allow fast enough transactions to achieve this requirement. Also, the user does not want to implement SMBus multimastering to enable two-way communication over the existing bus.

Zone Read: a solution for fast, prioritized data query responses

The PMBus 1.3 Zone Read is a command protocol that uses the Zone Read address 06h from the SMBus 3.0 specification. Zone Read creates a request for a response from all or some of the devices on the bus, or a broadcast request for the highest priority response from a single device. A zone is a subset of the devices on a bus and can include all devices on the bus.

There are three types of transactions for a Zone Read.

1. The first command transaction assigns each device to a zone. PMBus 1.3 defines the command that is used to set the assigned zone as ZONE_CONFIG.
2. The second transaction sets the active zone in all devices using Zone Read address 06h and the ZONE_ACTIVE command. The active zone defines the set of devices that are intended to listen to the Zone Read command.
3. The third transaction is a broadcast of a request for data using Zone Read address 06h, a
Command Control Code, and a PMBus read command. It will create a response from the devices that are configured with a read zone that matches the active zone setting.

Each device will respond with the requested data and its address to identify itself. The device’s responses will be arbitrated by “bit dominant arbitration” allowed by the open-drain topology of SMBus. The order of bits and bytes of the device responses is customizable by the Command Control Code to achieve the desired priority of the responses. The Zone Read can be used to quickly perform device address discovery, priority-based fault reporting, or telemetry and reconfigurable PG-based sequencing.

Devices that implement Zone Read are required to support the repeated start and clock stretching SMBus functions.

**More on the Command Control Code**

The Command Control Code was added to the Zone Read to allow the SMBus master to communicate the type and specifics of the intended transaction to the devices. This code includes group, status, data bit inversion, and data byte order swap bits.

The group bit tells the devices whether or not they should continue to retry their responses when they lose their arbitration. The status bit tells the devices to respond with a status byte or listen for a specific PMBus command. Data bit inversion inverts the data returned by the devices to affect the priority of the arbitration. Data byte order swap also enables manipulation of the priority of the arbitration.

These bits enable specific, priority-based information to be quickly fed back to the system master and each device for efficient system communication.

**Zone Read in a nutshell**

Zone Read was added to PMBus 1.3 to allow the system designer to achieve faster communication for time-critical conditions. These conditions include:

1. Slave address discovery
2. Priority-based fault reporting and fault handling
3. Reconfigurable powergood-based sequencing
4. Priority-based telemetry

The data that each device arbitrates in response can also be read by the rest of the devices on the bus. This effectively provides device-to-device communication. The listening devices can then act on the information, if designed to do so. See PMBus 1.3 for examples and talk to your system design engineers about Zone Read. Critical power device interactions can be performed faster using the Zone Read function.

**Zone Write: a solution for synchronized data execution**

The PMBus 1.3 Zone Write is a command protocol that uses the Zone Write address 07h from the SMBus 3.0 specification. Zone Write causes synchronized execution of written data to all or some of the devices on the bus. Remember, a zone is a subset of the devices on a bus and can include all devices on the bus.

Zone Write also has three transactions associated with it. Like Zone Read, preparing for Zone Write
requires configuring the device with its assigned zone using the PMBus command ZONE_CONFIG. Then the active zone is set using ZONE_ACTIVE. The third transaction is the broadcast write command which is acted upon when a device’s ZONE_CONFIG matches the ZONE_ACTIVE for writes. The command is executed upon the stop bit at the end of the transaction.

**Examples**

As stated previously, Zone Read and Zone Write require three types of transactions with the devices on the bus. The first transaction configures each device to its assigned zone. This is done with the ZONE_CONFIG command shown in Figure 3.

![Figure 3. ZONE_CONFIG command example:](image)

Assign device at address 20h and 34h to read zone 2 and write zone 2, and assign device at address 21h and 33h to read zone 1 and write zone none

During system operation the active zone for a Zone Read and Zone Write is set with the ZONE_ACTIVE command. This command uses the Zone Write address 07h. A ZONE_ACTIVE command can be sent at any time to change the zone targeted for a Zone Read and Zone Write. A device will always respond to a ZONE_ACTIVE command, regardless of the configuration value of its write zone. Figure 4 shows an example of this command.

![Figure 4. ZONE_ACTIVE command example:](image)

Set active zones to read zone 2 and write zone all

The third type of transaction is the Zone Read or Zone Write itself. The following figures show examples of a Zone Read. Figure 5 through Figure 8 show a Zone Read for two devices with the AR, “all respond,” bit set. Figure 9 shows an example of a Zone Write. Modular systems can have a different number of devices on the bus for a given assembly build. The example in Figure 5 shows the bus transactions that would occur to discover all devices on the bus without scanning every address. There is a device at address 20h and 34h to be discovered. The Group and Status bits of the Control Command Code are used. Note that the address bytes are in their normal left-shifted position. This process performs device address discovery 13 times faster than the standard PMBus command method for a bus with 15 devices on it.
Figure 5. Zone Read example: Address Discovery, GR=1, ST=1; known devices at addresses 20h and 34h, PGs good, no fault

For the example in Figure 6, the discovered devices are on the bus at address 20h and 34h. The device at address 20h has no faults pending. The device at address 34h has a Vout fault. When asking for prioritized fault information from a group of devices, the Group, Status and Data Inversion bits of the Command Control Code are used. The status information will be modified to allow the device with the highest priority fault to win the arbitration first.

Figure 6. Zone Read example: Priority Status Request, GR=1, ST=1, DI=1

Known devices at addresses 20h w/ no fault and 34h w/ Vout fault

The next example in Figure 7 describes a Zone Read used to retrieve the output current of every device on the bus. The bus has the same devices at address 20h with an output current of 15A, and at address 34h with an output current of 8A. Only the Group bit of the Control Command Code is set. The order of the reporting devices depends on the arbitration of the low data byte, since low data byte first is the SMBus norm.
If the order and priority of the responded data needs to be controlled, the Control Command Code can be used to do it. Figure 8 shows an example of receiving READ_TEMPERATURE data in “highest value first” order. The Group, Data Inversion, and Data Byte Order bits are used in the Control Command Code. Setting the Data Byte Order bit gets the high byte first and setting the Data Inversion bit causes the highest value to win arbitration. This process performs a highest- or lowest-value read back 11 times faster than the standard PMBus command method for a bus with 15 devices on it.

Synchronizing the enable command to all devices is a key element of bringing up a system. The timing of each device’s soft-start ramp may be very dependent on the beginning of their delay and ramp from the same point in time. Figure 9 shows a Zone Write example using the OPERATION command. Zone Write synchronizes the execution of written data upon assertion of the stop bit at the end.
active write zone set to all; set OPERATION to Enabled, affects all devices

Back to the engineers...

So I explained the new Zone Read and Zone Write features to my system designers. “Wow,” they said, “Great. Both of these enhancements will speed up access to my system. This will be great with the new faster bus in PMBus 1.3 too.” “Bingo,” I said.

PMBus is a trademark of SMIF, Inc.

---

Chris Eckhoff is a Registered Professional Engineer and Senior Principal Member of Technical Staff with Maxim Integrated in Austin, Texas. He was a digital power pioneer with Zilker Labs (now Intersil). Chris's expertise is in system design and interface software. Chris has also practiced engineering at Silicon Labs and AMD.

The PMBusTM Power System Management Protocol Specification was created to provide a standardized method of communication to power conversion and power management devices. A set of power supply oriented commands was defined to provide a common method for configuring, controlling and monitoring. The SMBus serial bus was selected for its cost effectiveness and common use in the market. I2C compatible designs of the SMBus allows for PMBus use in a number of markets that were familiar with and already implementing this form of serial bus. System designs using SMBus or I2C will have a host controller to ‘master’ the bus. The summary of significant differences between PMBus 1.2 and the proposed PMBus 1.3 are:

1. 1 MHz bus speed vs. 400 kHz limit for PMBus 1.2.
2. Floating point data format: supports NaN and +/-Inf, 16 bit number, IEEE 754 half precision and it allows easy conversion to C types.
3. Relative voltage thresholds: allows programming all output voltage related values thresholds as a percentage of the output voltage.
4. Zone Read: enables intelligent queries to all or some devices.
5. Zone Write: enables synchronous data write to all or some devices.