Leveraging App Profiles for Firmware Standardization

Michael Jones
Analog Devices

2/21/18

The Fundamental Firmware Challenge (Today)

- PMBus has a huge command set
- It only takes one command to call a device PMBus
- Finding devices with a common set of commands is difficult
- **Firmware becomes complicated and there are multiple end-user legacy libraries**

The Solution: Application Profiles

- Defined by PMBus Specification Committee with intensive reviews
- Small subset of most useful commands
- Multiple levels of functionality
- Data Sheets will indicate Application Profile Compliance
- **Hardware engineers can select compatible components without software knowledge**

Application Profiles: Who Does What?

- Profile Definition - Committee
- Feature Definition - Market
- **Code Definition – End User Software**

Technical Aspect of Software Opportunity

- Interface
  - It is possible to standardize software interfaces
  - Generic software interfaces can be translated into specific languages
- Syntax vs. Semantics
  - Interfaces define syntax
  - PMBus Specification defines semantics
- Principles
  - Interface should be simple and policy free
  - Additional behaviors belong in a layer over the interface

Functions Layering over Interfaces

- Device Discovery
- Persistence (NVM)
- Numeric Format
- Bit Handling (read before write)
- Page Handling
- Looping
- Grouping
- Write Protect
- Error Handling

Future Committee Work

- Continuous Profile Work
  - Power Controller Profiles
    - AC-DC
    - Module/IC
  - Power Manager Profiles
  - Synchronizing Profiles
    - Maintaining compatibility
    - Consistent API when possible
- Programming File Format

The future is now!