



IPM Sentry™ BMR-AVR-AMCm Reference Design Board Management Reference Design for AdvancedMC® Modules “The Quickest Path to Compliant and Interoperable IPM Controllers”

The BMR-AVR-AMCm design is one of a series of Pigeon Point Systems (PPS) IPM Sentry Board Management Reference designs. This member of the series provides a Module Management Controller (MMC) for Advanced Mezzanine Cards (AdvancedMCs or AMC)s and is based on an AVR® micro controller from Atmel.

This reference design is delivered in an IPM Sentry Board Management Starter Kit (which is detailed in a separate Product Brief). The kit includes:

- Schematics for a complete MMC subsystem, ready for integration into the design of your AMC, with adaptation as necessary.
- Firmware for that subsystem, delivered in source form and with development tools—ready for simple and quick adaptation to the specific requirements of your product.
- One-stop support for hardware, firmware and software used in developing and delivering your IPM Sentry BMR-based MMC.

The photo shows the core of a BMR-AVR-AMCm MMC. The active components are:

- An Atmel ATmega64/128 highly integrated micro controller.
- An optional external oscillator to provide the operating frequency.

Specification compliant and interoperability tested

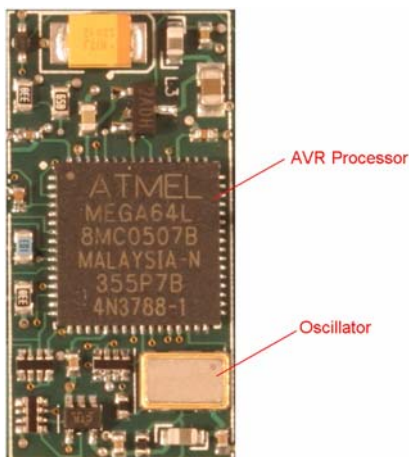
- AMC.0 R1.0, the AdvancedMC Base Specification
- IPMI v1.5, document revision 1.1, plus relevant errata
- Thoroughly tested with other management components at PICMG AdvancedTCA/AMC Interoperability Workshops

Full support for core hardware requirements

- AMC hot swap interfaces (handle and blue LED)
- Geographic address detection from carrier
- Control of E-Keying governed interfaces
- Optional persistence of above controls across MMC resets
- Managed by on-carrier IPMB-L
- UART-based payload and serial debug interfaces
- Thermal sensors (LM60 analog and/or DS75 digital)
- FRU LED management

Small footprint and low power

- Core MMC can fit in 24mm x 12mm footprint
- Active components consume only 30mA of max power



Comprehensive, readily adaptable firmware

- All mandatory and many optional IPMI/ATCA/AMC commands supported over IPMB-L
- Numerous PPS extension commands, primarily used over the payload and debug serial interfaces
- Serial interface protocol based on IPMI Terminal Mode
- Payload alert notifications over payload interface for sensor events and receipt of reset/shutdown commands
- Sophisticated support for firmware upgrades in the field
- Simple—but highly flexible—configuration of firmware features

ASCII-based serial interface protocol supported via UARTs to payload processor and serial debug interface

- Based on IPMI-defined Terminal Mode of the Serial/Modem Interface (and referenced as Serial Interface Protocol Lite—SIPL)
- Same protocol used for both serial interfaces
- Uses ASCII-encoded raw IPMI messages, which are handled by the IPM Controller essentially like IPMB messages
- PPS extension commands implemented as IPMI-compliant OEM messages

Sophisticated support for firmware upgrades in the field

- Firmware upgrades over IPMB-L or over either the payload or debug serial interfaces
- Optional enhanced firmware configuration provides redundant copy firmware, with automatic fallback to backup copy (e.g., if upgrade is interrupted or new firmware is non-operational)

Simple, but powerful, firmware configuration mechanisms

- Configuration variables in a single `config.h` source file parameterize and determine inclusion/exclusion of subsystems during firmware image build
- Binary configuration files for FRU Information and Sensor Data Records (SDR) merged into firmware image
- FRU Information and SDR files produced from textual representations by special supplied compilers

Comprehensive AVR development environment included

- Cross GNU C compiler, GNU GDB and binary utilities and optimized `libc` for AVR architecture
- JTAG-based debugging and firmware download
- Communication utility for Atmel JTAG ICE tool (the latter purchased separately)
- Supported under both Linux and Windows on x86 hosts

Key extensions beyond required IPMI/ATCA/AMC commands and functionality

- Graceful Reboot and Issue Diagnostic Interrupt options in FRU Control command
- Get Sensor Reading Factors
- Get/Set Sensor Hysteresis
- Get/Set Sensor Thresholds
- Get/Set Sensor Event Enable
- Re-arm Sensor Events
- Get Sensor Event Status

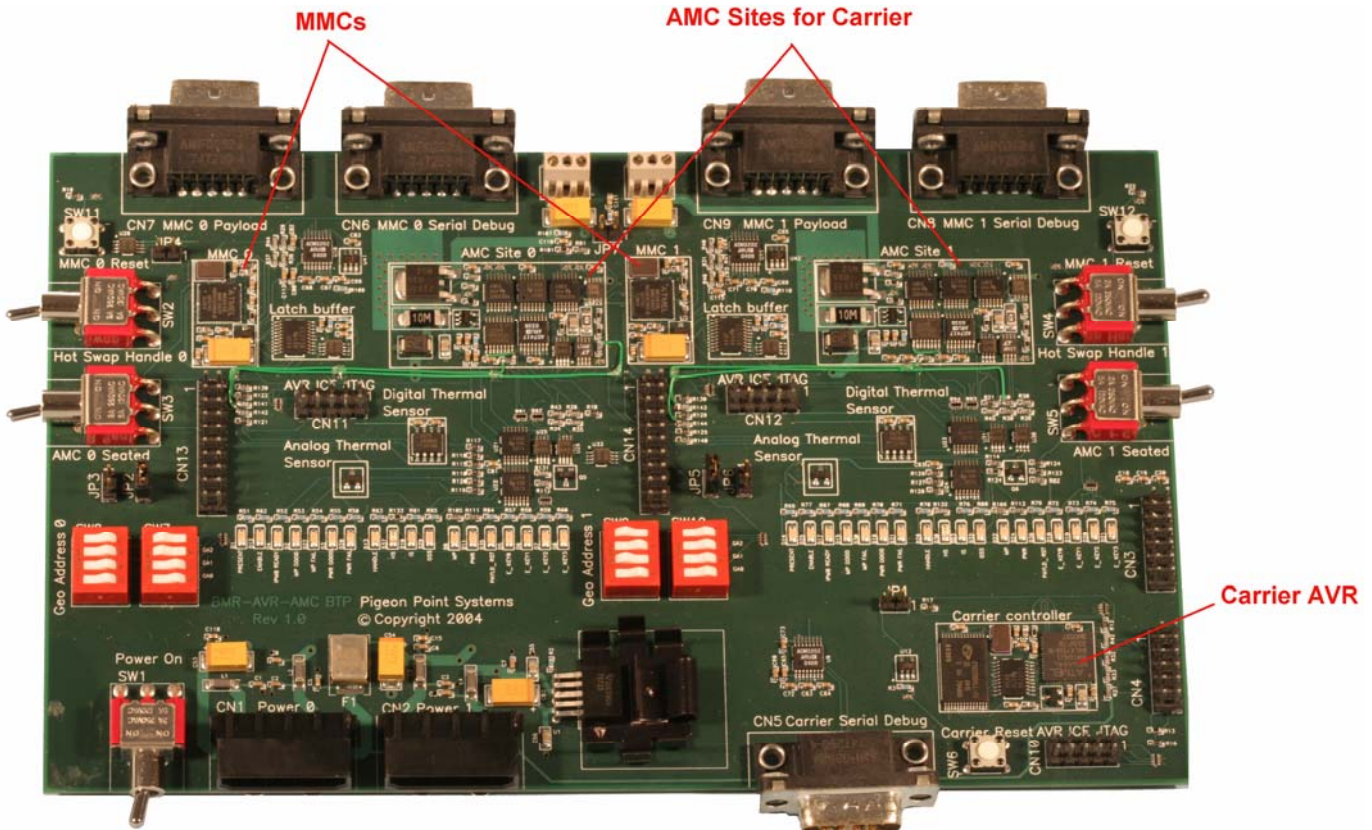
Rich set of PPS extension commands

- Get Status
- Get/Set Serial Interface Properties
- Get/Set Debug Level
- Get/Set Handle Switch
- Get/Set Payload Communication Timeout
- Disable/Enable Payload Control
- Reset MMC
- Hang MMC¹
- Graceful Reset
- Diagnostic Interrupt Results
- Get/Set Payload Shutdown Timer
- Get/Set Geographic Address

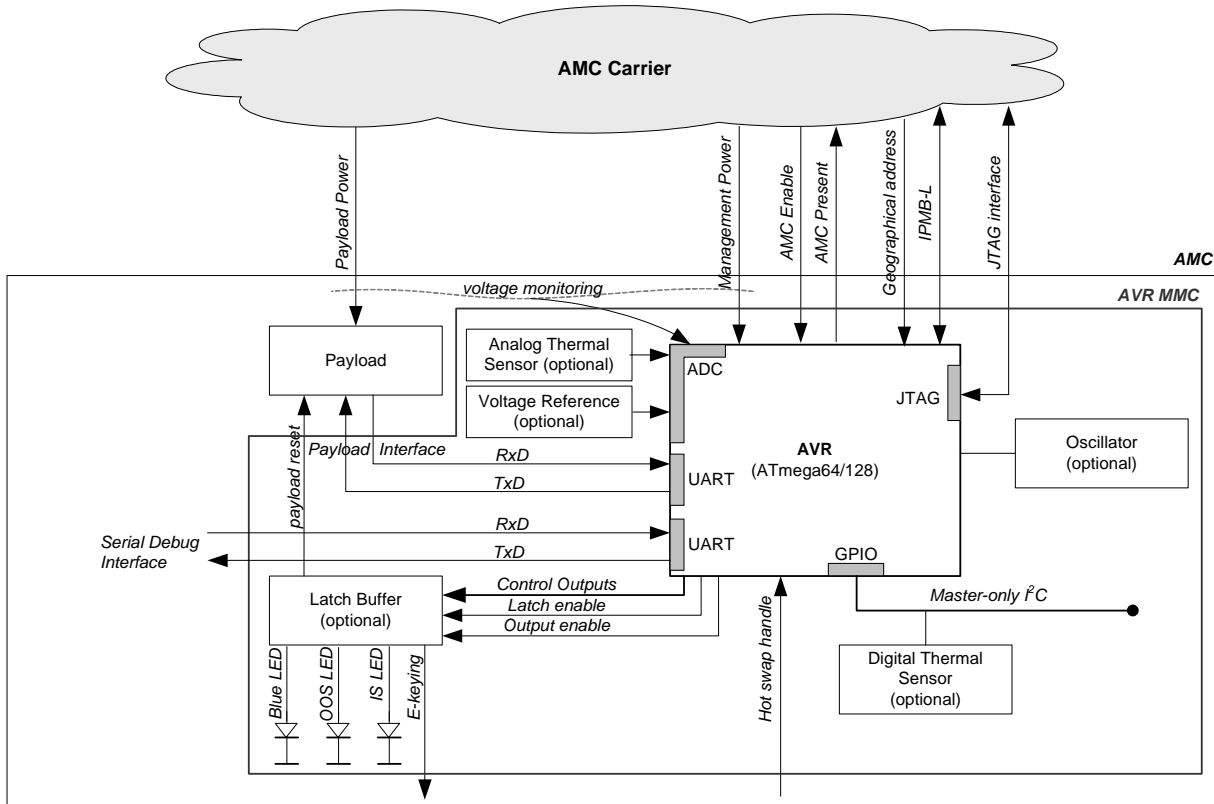
¹ This function is used to test the MMC watchdog.

Reference Implementation

The bench top implementation of this reference design cables to the AVR-based IPM Controller bench top board to form a complete Carrier IPMC, plus two AMC sites. In addition to the necessary carrier-based functionality for each AMC site, the bench top board implements an MMC at each site, based the BMR-AVR-AMCm reference design. Thus, this board serves as a bench top implementation of both the BMR-AVR-AMCc (Carrier) and BMR-AVR-AMCm (Module) reference designs. In addition to the core facilities of those reference designs, the board includes example implementations of the optional controller features and numerous LEDs, switches and headers to allow lab experimentation with the behavior of the controllers, including pre-supported options and any desired firmware customizations. Below the photo is a high level block diagram of the AVR-based Module Management Controller.



BMR-AVR-AMCm IPM Controller Block Diagram



Copyright © 2004-2005, Pigeon Point Systems. All rights reserved. Pigeon Point Systems, IPM Sentry and Monterey Linux are trademarks of Pigeon Point Systems. 2005-05.