Agenda

• What is SCMI
• Power Management Models enabled by SCMI
• SCMIv2.0 – FastChannels
• SCMIv2.0 – Agent specific permission Configuration
• SCMIv2.0 – Other Features
• SCMI.next Roadmap - New Features
• Evolution & Roadmap* – High level view
• Status & Next Steps
• Useful Links
What is System Control & Management Interface (SCMI)

Publicly available ARM specification

- Standard messaging interface to a System Controller for
  - System Control
  - Power & Performance Mgmt.

- System Controller may be
  - Firmware in Secure World.
  - Firmware + microcontroller.

- New features in SCMIV2.0:
  - SMC/HVC Based standard SCMI Channels
  - Reset domain management
  - Pre-Notifications
  - FastChannel Support
    - Register or Shared memory-based Channel
    - Agent-specific Permission Configuration & Reset
Power Management Models enabled by SCMI

System Control firmware on a Cortex M based Microcontroller

Normal World

- App
- Operating System
  - HW Mb Driver
  - SCMI Driver

Secure World

- App
- Trusted OS
- EL3 Firmware

System & Power Control uC

1a

System & Power Control uC

SCP

Cortex-Mx

1b

System & Power Control uC

SCP

TF-M

Cortex-M3x

Physical Channel

Physical Channel
Power Management Models enabled by SCMI
System Control firmware in Cortex-A Secure World

Recommended Model

Only recommended if no Secure-EL1 Payload is present
Power Management Models enabled by SCMI

System Control firmware in Virtualized Systems

Normal World

Secure World

Any secure world deployment scenario
Power Management Models enabled by SCMI
System Control firmware in Hardware-assisted Virtualized Systems

Normal World
- Operating System
  - Mb Driver
  - SCMI Driver

Secure World
- Operating System
  - Mb Driver
  - SCMI Driver
- Trusted OS
- EL3 Firmware

- Hypervisor
- vSCP
- System & Power Control uC
- Physical Channel
SCMIv2.0 – FastChannels

- Introduced to reduce latency during certain latency-sensitive operations like DVFS.
- FastChannel is Per-Message Per-Domain dedicated channel for Performance Protocol.
- Different command types cannot be multiplexed over the same FastChannel.
- May be any memory location shared between SCP & AP Domain (e.g. SRAM). May be non-secure.
- Optional Doorbell Support - Location; Register size; Set/Preserve/Clear mask.
- Dynamic discovery. Cpufreq driver can check FastChannel bit per performance domain and discover them via a command.
- No Response back from Channel. Asynchronous Mode of operation (fire-and-forget).
  - Use Shared Stats region to read existing frequency.
  - If Synchronous behavior is required, use normal SCMI channels. Since hardware delays for Synchronous behavior (PLL Lock + Voltage Ramp) is of order of 10’s of us, normal SCMI channel should be sufficient.
SCMlv2.0 – Agent specific permission Configuration

Direct assignment in Virtualized systems

- Hypervisor has access to all channels.
- Each channel has a unique agent-id allocated by System Controller.
  - Discoverable by BASE_DISCOVER_AGENT command
- Hypervisor chooses channels to allocate to VMs.
- Hypervisor configures access permissions of agents
  - BASE_SET_DEVICE_PERMISSIONS
  - BASE_SET_PROTOCOL_PERMISSIONS
  - BASE_RESET_AGENT_CONFIGURATION
  - Only one control channel has permissions to use above commands.
  - Access control imposed at boot by System Controller.
  - Control channel identification is done through firmware tables and is not discoverable.
- Hypervisors maps configured channels into VM space and starts VM.
- VM can use channel mapped to its space.
  - Can send commands to access resources allowed by the hypervisor over the channel.
SCMlv2.0 – Agent specific permission Configuration

How Direct Assignment works

• Hypervisor configures access permissions of agents
  – BASE_SET_DEVICE_PERMISSIONS
    ▪ System Controller Firmware groups resources into Devices.
      – Devices can be logical.
    ▪ Devices uniquely identified by 32 bit device-identifiers.
    ▪ Devices are mapped through firmware tables.
      – Device Dynamic discovery is planned for SCMI.next
    ▪ If agent has access to a device, it gets access to all resources associated with the device.
  – BASE_SET_PROTOCOL_PERMISSIONS
    ▪ Access to specific protocols for a particular device for an agent
  – BASE_RESET_AGENT_CONFIGURATION
    ▪ Reset all permissions of an agent.
    ▪ Reset all device specific configurations done by an agent.

• The VM can send commands to access resources allowed by the hypervisor over the channel.
  – E.g., it can directly access a Power Domain if it is associated with a device the agent has access to.
SCMIv2.0 – Agent specific permission Configuration

Multi-Master Systems

“A+M” Class SoCs
SCMiv2.0 – Other Features

Reset Domain Management & SMC based doorbell

• Reset Domain Management Protocol
  – Reset domains are defined by the System Controller
  – Reset domains can be reset
    ▪ Explicitly by toggling the reset signal low and high.
    ▪ Autonomously by asking for a Reset. Firmware does the reset.
  – Compatible with Linux Reset Framework.

• SMC/HVC Based doorbells
  – SCMv2.0 introduces the possibility of doorbell being SMC/HVC based to enable secure world firmware deployment.
  – Parameters are {SMC FN_ID, 32-bit Mailbox Identifier}
  – Mailbox identifier identifies a memory region shared between normal and secure world which serves as Mailbox.
  – Secure and Normal world need to view the same map of Mailbox identifiers.
    ▪ Mailbox ID discovery is through firmware tables.
SCMI.next Roadmap - New Features

The Device View

Device advertises
- Power State*
- Performance levels
- Reset States

System Controller
- Abstracts and Manages platform specific details
- Creates device attributes from constituent domains
- Manages/Resolves Domain Dependencies for SCMI Device Management calls

*CPU Power Management is still over PSCI

SCMI v1.0
POWER STATE
PERFORMANCE LEVEL
RESET

Notification Support
Sync/Async Support

System Controller Responsibility & View
OS Responsibility & View

13 © 2019 Arm Limited
SCMI.next Roadmap - New Features

PSA-FF-A (previously known as SPCI) Interface based Transport

**PSA-FF-A overview**

- A Secure Partition (SP) is a Secure world sandbox
  - Can host a Trusted OS or a driver stack
  - Exports services to Normal world clients and other partitions

- PSA-FF-A generalizes communication with SPs
  - Describes ABIs to access services in a SP

**SCMI transport over PSA-FF-A**

- Secure World resident SCMI Server in SP
- Fixed-Function hardware or microcontroller-based design

---

**Diagram Description**

- **Normal World**
  - Client Application
  - Operating System Kernel
  - Hypervisor (optional)

- **Secure World**
  - Trusted Application
  - Trusted OS Kernel
  - Secure Partition Manager

- **Isolation Boundary**
  - Secure partition with trusted OS
  - Secure partition with silicon vendor drivers

- **System & Power Control uC**
  - SELJ Firmware

- **Physical Channel**
  - Control & PM Logic
  - Alternate Path

- **System & Power Control Mailbox**

- **SCMI & Support Infrastructure**
  - SCMI Server

---

© 2019 Arm Limited
Status & Next Steps

• Device Management
  • Migration Path (Short Term) and Long-Term kernel framework under analysis.
  • Proof of Concept tentatively targeted at ARM JUNO board.

• SCMIv3.0 (Concept)
  • Device Management Protocol
  • Autonomous Power & Performance Management
  • QoS Support
Useful Links

- SCMI: https://developer.arm.com/architectures/system-architectures/software-standards/scmi
- SCP Reference Firmware: https://github.com/ARM-software/SCP-firmware
- ARM Trusted Firmware: https://github.com/ARM-software/arm-trusted-firmware
Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
תודה