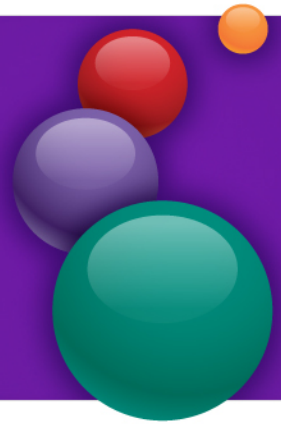
MIPI DisCo and ACPI: Streamlining MIPI Component Integration



www.mipi.org

Rob Gough

MIPI Software WG Chair

2 February 2017

MIPI Overview

www.mipi.org

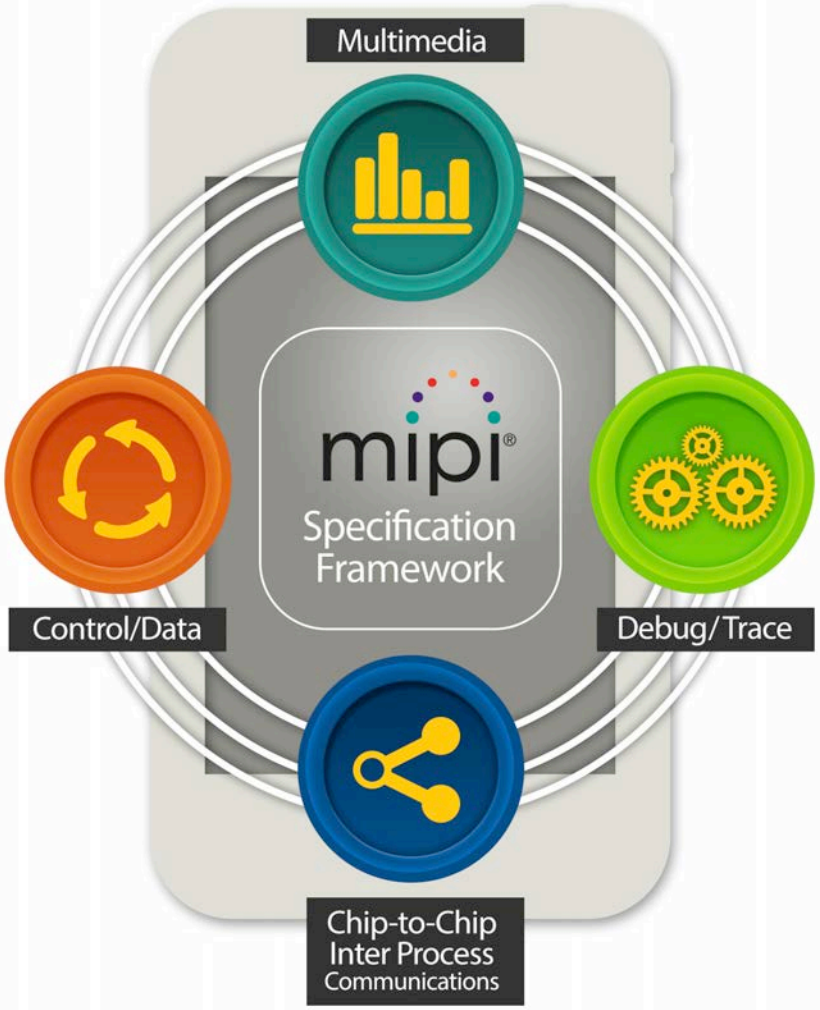
Peter Lefkin
MIPI Alliance
Managing Director

2 February 2017

About MIPI Alliance

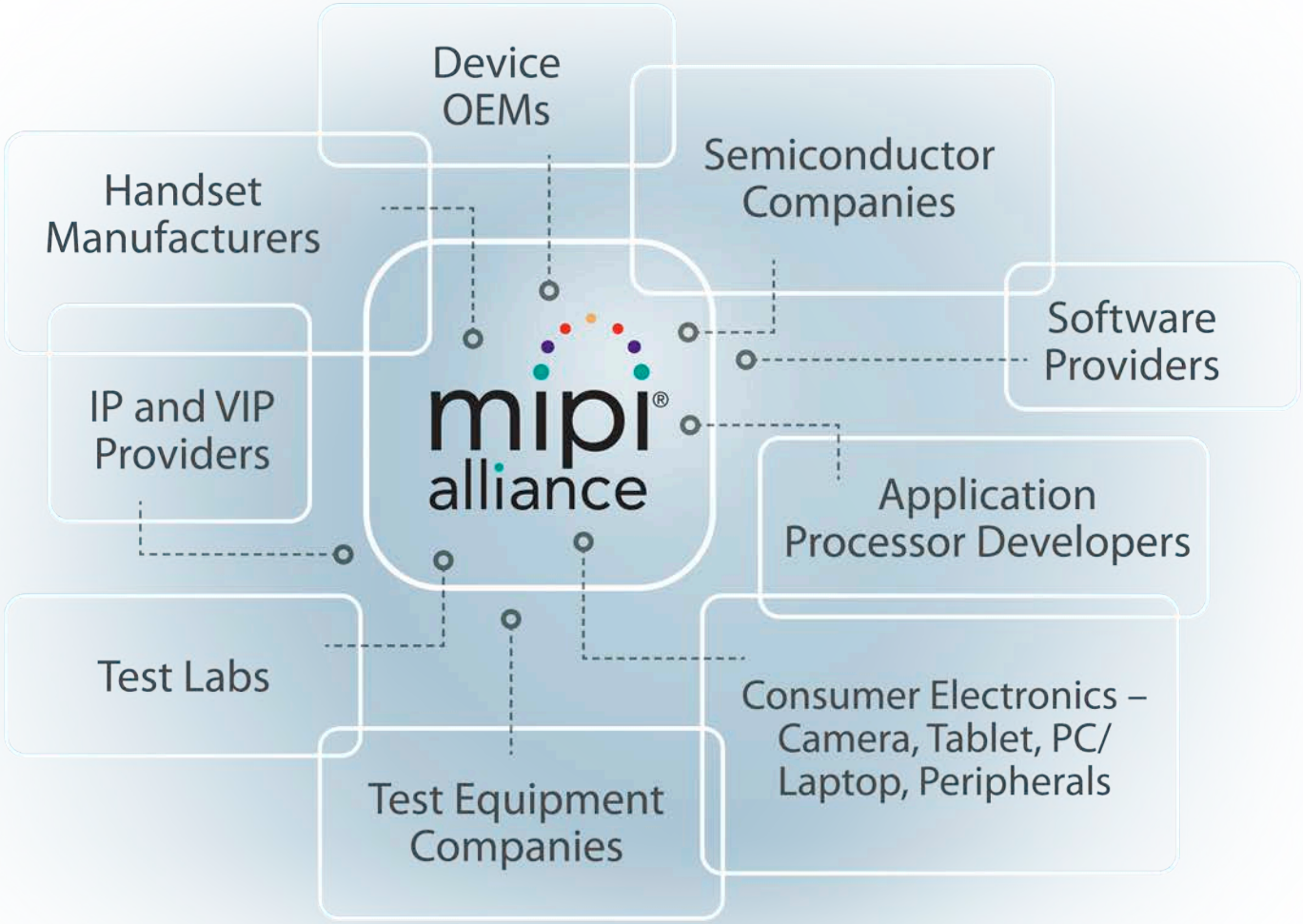
We are a global, collaborative organization comprised of over 280 member companies spanning the mobile and mobile-influenced ecosystems.

MIPI Alliance is leading innovation in mobile interface technology.



© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI Alliance Member Ecosystem



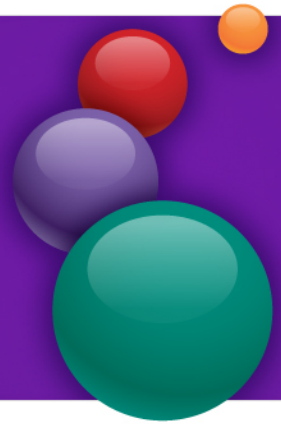
© 2017 MIPI Alliance, Inc. All rights reserved.

Active Technical Working Groups

Camera	Debug	Display	Low Latency Interface
Low Speed Multipoint Link	PHY (C/D/M)	Reduced Input Output	RF Front End
Sensor / I3C℠	Software	Test	UniPro℠

© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI DisCo and ACPI: Streamlining MIPI Component Integration



www.mipi.org

Rob Gough

MIPI Software WG Chair

2 February 2017

Agenda

- MIPI Software Working Group Charter
- ACPI Overview
- MIPI DisCo implementation of ACPI
- MIPI SoundWire example
- Open Property Database
- Benefits of using DisCo-defined property sets
- Streamlining the Platform Integration Process
- Call to Action

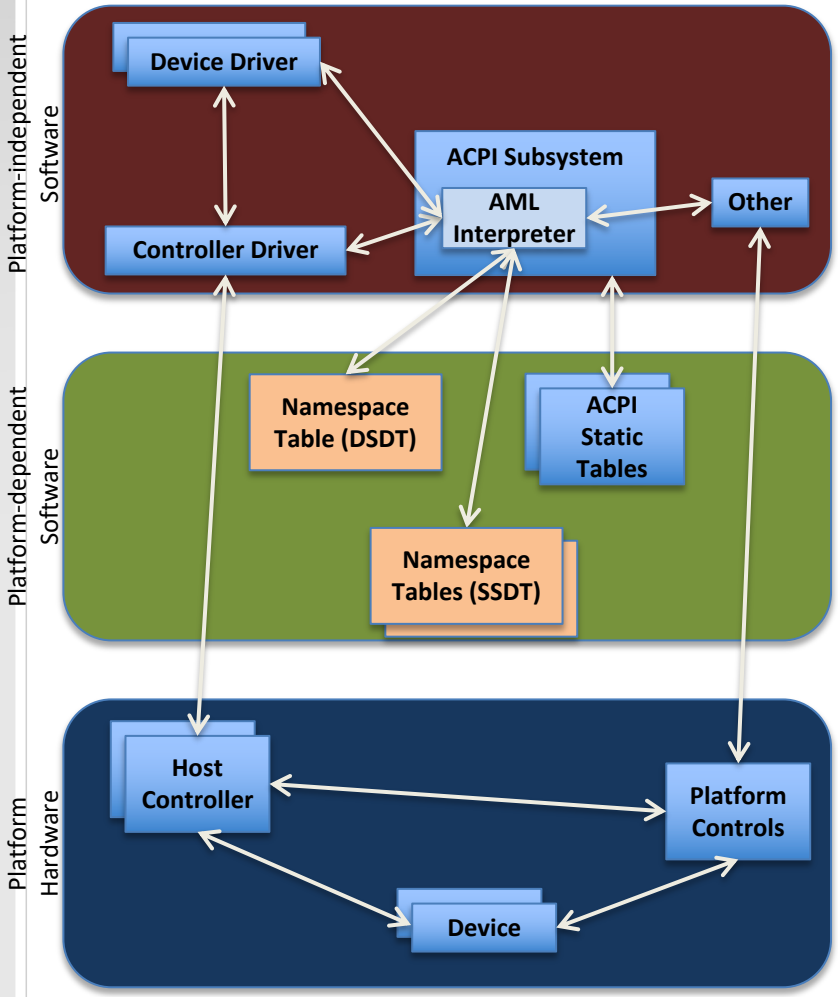
© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI Software Working Group Charter

- Develop uniform hardware description mechanism for components/devices
- Define an extensible framework, flexibly applicable to all MIPI interfaces
- Ensure compatibility with existing mechanisms defined by MIPI interfaces
- Minimize cost for platform silicon (gate count, resource efficiency) and OS (code footprint, responsiveness)
- Provide consistent capabilities, w/o constraints/restrictions on implementation
- Enable OS consistency for capabilities and controls through reference implementation

© 2017 MIPI Alliance, Inc. All rights reserved.

ACPI Overview



- ACPI-compliant platforms present data to the OS via static tables and the Namespace – scope of the MIPI Software specification
- ACPI Namespace is a hierarchical representation of the platform based on bus/controller/device topology
- Namespace is comprised of System Descriptor Tables containing ACPI header and machine language
- A platform has a Root Table, a Differentiated System Descriptor Table (DSDT) and 0 or more Secondary System Descriptor Tables (SSDT)
- AML is a machine-independent interpreted language describing Namespace objects
- MIPI controllers/devices are defined as static container objects (DeviceObject) that include component- and platform-specific data

mipi alliance
© 2017 MIPI Alliance, Inc. All rights reserved.

Discovery and Configuration: MIPI DisCo

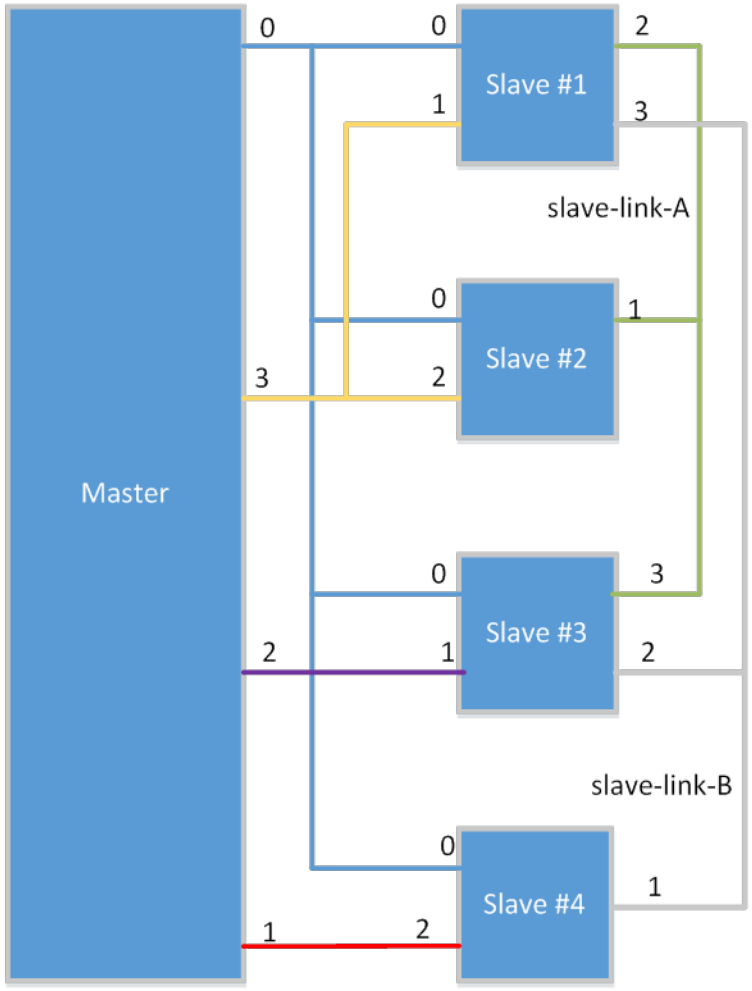
- DisCo mechanisms use ACPI Objects like _DSD (Device-Specific Data) to provide property sets to upper layer software.
- Information encoded in _DSD properties includes:
  - Platform-specific information
    - E.g. Intra-device and host-device connection information
    - Platform design variations
    - Properties are required if applicable; information must be obtained from platform
  - Component-specific information that cannot be discovered
    - E.g. clock rates supported, features available
    - Typically described in device datasheet or implementation guides
    - Properties are optional; component-specific drivers can hard-code information
    - Vendor-specific information can also be described via _DSD
- _DSD properties are portable!
  - Can also be supported in non-ACPI environment e.g. DeviceTree

© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI SoundWire Example: Platform

Property String	Property Data Type	Description
"mipi-sdw-lane-<n>-mapping"	String	Identifier encoded as a string. This property is used by driver software to determine which lanes are connected to lanes on the Master or on other Slaves via a Slave Link. ...

Device	Property Key	Value
Slave #1	"mipi-sdw-lane-1-mapping"	"master-lane-3"
	"mipi-sdw-lane-2-mapping"	"slave-link-A"
	"mipi-sdw-lane-3-mapping"	"slave-link-B"
Slave #2	"mipi-sdw-lane-2-mapping"	"master-lane-3"
	"mipi-sdw-lane-1-mapping"	"slave-link-A"
Slave #3	"mipi-sdw-lane-1-mapping"	"master-lane-2"
	"mipi-sdw-lane-2-mapping"	"slave-link-B"
	"mipi-sdw-lane-3-mapping"	"slave-link-A"
Slave #4	"mipi-sdw-lane-2-mapping"	"master-lane-1"
	"mipi-sdw-lane-1-mapping"	"slave-link-B"



© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI SoundWire Example: Component

Property String	Property Data Type	Description
"mipi-sdw-sw-interface-revision”	Integer	This is a 32-bit value where the upper word contains the major version number of this Specification, and the lower word contains the minor version number. This entry shall be provided if any other property entry within this structure is populated. Example: 0x00010000 equates to Specification v1.0.
“mipi-sdw-max-clock-frequency”	Integer	This value provides the maximum Bus clock in Hz for this master. This is the maximum usable Bus clock frequency for this platform.
“mipi-sdw-clock-frequencies-supported”	Package	A package containing one integer entry for each clock frequency supported. Frequencies are represented in Hz.
“mipi-sdw-supported-clock-gears”	Package	A package containing one integer entry for each supported gear, e.g. {1, 2, 3, 4, 8, 16}. Some Masters may only support a single gear, or powers of two.
“mipi-sdw-data-port-type”	Integer	Type of Data Port. 0: Full Data Port 1: Simplified Data Port 2: Reduced Data Port

© 2017 MIPI Alliance, Inc. All rights reserved.

_DSD Property Database

- _DSD can be used to provide MIPI- and Vendor-defined Property Sets
  - Component vendors can define their own property sets
- _DSD Database:
  - An open repository currently under development
  - Supports publication of property set definitions from component vendors, consumable by driver and firmware developers
  - MIPI SWWG will promote DisCo properties as specifications are released
- Mailing list: https://lists.acpica.org/mailman/listinfo/dsd

© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI DisCo Benefits

- Driver
  - Pre-DisCo, drivers are developed on a single platform, and then re-built or ported to new platforms
    - A driver for a given component on platform A must be modified to work on platform B
    - Many versions of drivers exist (must be supported) for the same component
  - DisCo-compliant drivers for a MIPI component can be written once, consuming ACPI-provided properties for device and platform configuration.
    - No more forking drivers per-platform
- Firmware
  - Information is OS-agnostic
  - Simple presentation of DisCo properties
  - Drop-in property sets from component vendors

© 2017 MIPI Alliance, Inc. All rights reserved.

Platform Component Integration

- Component vendor provides component-specific DisCo property sets
  - Datasheet and/or electronic format
- Platform designer imports vendor-provided properties
- Platform designer adds platform implementation-specific properties as needed
- Existing OS drivers work **as-is**
  - **Drivers for new components developed based on DisCo specifications can be re-used**

© 2017 MIPI Alliance, Inc. All rights reserved.

Call To Action

- Component Vendors:
  - Utilize _DSD properties for components defined in MIPI DisCo Specifications
  - Describe vendor-specific properties through the _DSD database
    - Consider applicability- candidate for a future DisCo spec? Engage SWWG
    - Publish existing vendor-defined DeviceTree property sets to the _DSD database
  - Provide component-specific static _DSD packages to customers
  - **Provide tools to dynamically produce _DSD packages for customers**
  - Provide links to product information in the MIPI product registry: registry.mipi.org
- System Integrators:
  - Utilize _DSD properties for platform configurations defined in MIPI DisCo Specifications
  - Integrate component-specific packages from component vendors
- All:
  - **Participate in DisCo Spec development, property set definition**

© 2017 MIPI Alliance, Inc. All rights reserved.

Legal Disclaimer

The material contained herein is not a license, either expressly or impliedly, to any IPR owned or controlled by any of the authors or developers of this material or MIPI®. The material contained herein is provided on an "AS IS" basis and to the maximum extent permitted by applicable law, this material is provided AS IS AND WITH ALL FAULTS, and the authors and developers of this material and MIPI hereby disclaim all other warranties and conditions, either express, implied or statutory, including, but not limited to, any (if any) implied warranties, duties or conditions of merchantability, of fitness for a particular purpose, of accuracy or completeness of responses, of results, of workmanlike effort, of lack of viruses, and of lack of negligence. ALSO, THERE IS NO WARRANTY OR CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, CORRESPONDENCE TO DESCRIPTION OR NON-INFRINGEMENT WITH REGARD TO THIS MATERIAL.

All materials contained herein are protected by copyright laws, and may not be reproduced, republished, distributed, transmitted, displayed, broadcast or otherwise exploited in any manner without the express prior written permission of MIPI Alliance. MIPI, MIPI Alliance and the dotted rainbow arch and all related trademarks, tradenames, and other intellectual property are the exclusive property of MIPI Alliance and cannot be used without its express prior written permission.

IN NO EVENT WILL ANY AUTHOR OR DEVELOPER OF THIS MATERIAL OR MIPI BE LIABLE TO ANY OTHER PARTY FOR THE COST OF PROCURING SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA, OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, OR SPECIAL DAMAGES WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THIS OR ANY OTHER AGREEMENT RELATING TO THIS MATERIAL, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

© 2017 MIPI Alliance, Inc. All rights reserved.

BACKUP

© 2017 MIPI Alliance, Inc. All rights reserved.

MIPI DisCo Development Process

- MIPI Software WG members work with other WGs to jointly define property sets for MIPI component types (e.g. SoundWire)
- SWWG develops specification, ongoing reviews with other MIPI Working Groups.
- Other WG ratifies content; SWWG owns content and publishes, with board approval.
- Software WG specs are made available to non-MIPI members via http://Software.MIPI.org

© 2017 MIPI Alliance, Inc. All rights reserved.

ASL Example

```
Device(SWC0) {  // SoundWire Controller 0, Full Device Descriptor
 Name(_HID, "VEND0000")   // sample Vendor ID

 Name(_DSD, Package() {
  ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
  Package () {
   Package (2) {"mipi-sdw-sw-interface-revision", 0x00010000}, // v 1.0
   },
  ToUUID("dbb8e3e6-5886-4ba6-8795-1319f52a966b"), // Hierarchical Extension
  Package () {
   Package (2) {"mipi-sdw-link-0-subproperties",         "SWM0"},
  }
 })

 Name(SWM0, Package() { // SoundWire Master 0
  ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
  Package () {
   Package (2) {"mipi-sdw-max-clock-frequency",        9600000},
  }
 })

 Device (SWS0) {   // SoundWire Slave 0
  Name(_ADR, 0x00055AA55AA)
  Name(_DSD, Package() {
   ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
   Package () {
    Package (2) {"mipi-sdw-sw-interface-revision", 0x00010000}, // v 1.0
    // ...
    Package (2) {"mipi-sdw-source-port-list",              0x06},
    Package (2) {"mipi-sdw-sink-port-list",                0x18},
   },
   ToUUID("dbb8e3e6-5886-4ba6-8795-1319f52a966b"), // Hierarchical Extension
   Package () {
    Package (2) {"mipi-sdw-dp-0-subproperties",                  "P0SP"},
    Package (2) {"mipi-sdw-dp-1-source-subproperties",           "S1SP"},
   }
  })

 Name(P0SP, Package() {
     ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
     Package () {
      Package (2) {"mipi-sdw-port-max-wordlength",              16},
      Package (2) {"mipi-sdw-port-min-wordlength",              8},
     }
    }) // End SWC0.SWS0.P0SP

    Name(S1SP, Package() {
     ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
     Package () {
      Package (2) {"mipi-sdw-port-max-wordlength",              16},
      Package (2) {"mipi-sdw-port-min-wordlength",              8},
      Package (2) {"mipi-sdw-data-port-type",                   0},
      // ...
     }
    }) // End SWC0.SWS0.S1SP
   } // End SWC0.SWS1
 } // End SWC0
```


© 2017 MIPI Alliance, Inc. All rights reserved.