Client Silicon Package

v0.2.0.0

Contents

Chapter 1

1 - Introduction

Purpose

The purpose of this document is to describe the external architecture and interfaces provided in the Client Silicon Package.

Chapter 2

2 - Packages Revision

v1.4.1.0 (BP1410_RP01)

Package(s)	GIT / SVN link(s)
BaseTools, CryptoPkg, FatBinPkg, FatPkg, Intel↵ FrameworkModulePkg, IntelFrameworkPkg, Intel↵ Fsp2Pkg, IntelFsp2WrapperPkg, IntelSiliconPkg, MdeModulePkg, MdePkg, NetworkPkg, PcAt↵ ChipsetPkg, PerformancePkg, SecurityPkg, Shell↵ BinPkg, ShellPkg, SourceLevelDebugPkg, UefiCpu↵ Pkg	GIT: 324a4c9d7d512f3bf78fe782803d3a8a09c69f73 https://github.↵ com/tianocore/edk2/tree/UDK2017
BaseToolsWin32	GIT: 0e088c19ab31fccd1d2f55d9e4fe0314b57c0097 https://github.com/tianocore/edk2-↵ BaseTools-win32.git

v1.4.1.0 (BP1410_RC03_RP01)

Package(s)	GIT / SVN link(s)
BaseTools, CryptoPkg, FatBinPkg, FatPkg, Intel↵ FrameworkModulePkg, IntelFrameworkPkg, Intel↵ Fsp2Pkg, IntelFsp2WrapperPkg, IntelSiliconPkg, MdeModulePkg, MdePkg, NetworkPkg, PcAt↵ ChipsetPkg, PerformancePkg, SecurityPkg, Shell↵ BinPkg, ShellPkg, SourceLevelDebugPkg, UefiCpu↵ Pkg	GIT: adeef88496e74b1a7014ea4008f28d82f43d9683 https://github.↵ com/tianocore/edk2/tree/UDK2017
BaseToolsWin32	GIT: 7b52b1a9bf853b5c498ea61d0a4f17083f543be4 https://github.com/tianocore/edk2-↵ BaseTools-win32.git

v1.4.1.0 (BP1410_RC02_RP01)

Package(s)	GIT / SVN link(s)
BaseTools, CryptoPkg, FatBinPkg, FatPkg, Intel←FrameworkModulePkg, IntelFrameworkPkg, Intel←Fsp2Pkg, IntelFsp2WrapperPkg, IntelSiliconPkg, MdeModulePkg, MdePkg, NetworkPkg, PcAt←ChipsetPkg, PerformancePkg, SecurityPkg, Shell←BinPkg, ShellPkg, SourceLevelDebugPkg, UefiCpu←Pkg	GIT: 86e796a46f3f6bf252482f0b31f669b03c7ffeaf https://github.←com/tianocore/edk2/tree/UDK2017
BaseToolsWin32	GIT: 86ee43a0e874f382081488dc5777cdc4fe641443 https://github.com/tianocore/edk2-←BaseTools-win32.git

Chapter 3

3 - Client Silicon Package Override (CSPO)

Client Silicon Package Override `CSPO-xxxx` tags lists the overrides of the Intel Green H to add / enhance new features or resolve issues to unblock the enabling.

The Active CSPO table lists every issue which is currently resolved with an override. It is a catalog of all such existing overrides.

The Retired CSPO table lists issues which previously required such overrides. It is purely historical as these overrides have all been deleted (are no longer needed).

Active CSPOs in ClientSiliconPkg/Override

Tag	Client HSD	Core HSD	Status / Planned EOL	Description
CSPO-xxxx				
CSPO-0002	22058464		BP1410 RC4	Apply MtrrLib patch from SSG to fix a system hang observed while enabling CPU TraceHub Mode

Retired CSPOs in ClientSiliconPkg/Override

Tag	Client HSD	Core HSD	Actual EOL	Description
CSPO-0001	1504011433			"Enable GCC Link Time Optimization (LTO)" to reduce the GCC build size
CSPO-0002	22058464		BP1410	Apply MtrrLib patch from SSG to fix a system hang observed while enabling CPU TraceHub Mode

Chapter 4

Class Documentation

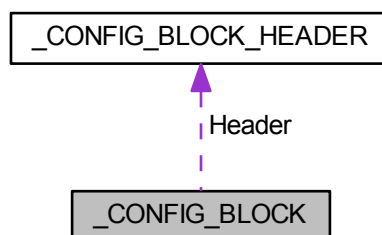## 4.1 _CONFIG_BLOCK Struct Reference

Config Block.

```
#include <ConfigBlock.h>
```

Collaboration diagram for _CONFIG_BLOCK:

Public Attributes

- CONFIG_BLOCK_HEADER Header

    Offset 0-27 Header of config block.

4.1.1 Detailed Description

Config Block.

Definition at line 38 of file ConfigBlock.h.

The documentation for this struct was generated from the following file:

- ConfigBlock.h

4.2 _CONFIG_BLOCK_HEADER Struct Reference

Config Block Header.

`#include <ConfigBlock.h>`

Public Attributes

- EFI_HOB_GUID_TYPE GuidHob

    Offset 0-23 GUID extension HOB header.
- UINT8 Revision

    Offset 24 Revision of this config block.
- UINT8 Attributes

    Offset 25 The main revision for config block.
- UINT8 Reserved [2]

    Offset 26-27 Reserved for future use.

4.2.1 Detailed Description

Config Block Header.

Definition at line 28 of file ConfigBlock.h.

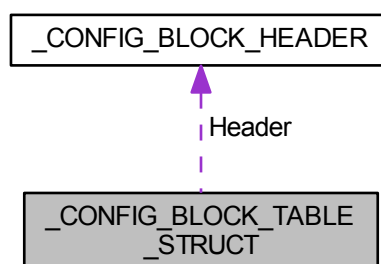The documentation for this struct was generated from the following file:

- ConfigBlock.h

4.3 _CONFIG_BLOCK_TABLE_STRUCT Struct Reference

Config Block Table Header.

`#include <ConfigBlock.h>`

Collaboration diagram for _CONFIG_BLOCK_TABLE_STRUCT:

Public Attributes

- CONFIG_BLOCK_HEADER Header

    Offset 0-27 GUID number for main entry of config block.

- UINT8 Rsvd0 [2]

    Offset 28-29 Reserved for future use.

- UINT16 NumberOfBlocks

    Offset 30-31 Number of config blocks (N)

- UINT32 AvailableSize

    Offset 32-35 Current config block table size.

4.3.1 Detailed Description

Config Block Table Header.

Definition at line 48 of file ConfigBlock.h.

The documentation for this struct was generated from the following file:

- ConfigBlock.h

4.4 FIRMWARE_VERSION Struct Reference

Firmware Version Structure.

```
#include <FirmwareVersionInfoHob.h>
```

4.4.1 Detailed Description

Firmware Version Structure.

Definition at line 26 of file FirmwareVersionInfoHob.h.

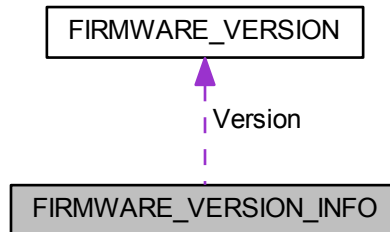The documentation for this struct was generated from the following file:

- FirmwareVersionInfoHob.h

4.5 FIRMWARE_VERSION_INFO Struct Reference

Firmware Version Information Structure.

```
#include <FirmwareVersionInfoHob.h>
```

Collaboration diagram for FIRMWARE_VERSION_INFO:

Public Attributes

- UINT8 ComponentNameIndex

    Offset 0 Index of Component Name.
- UINT8 VersionStringIndex

    Offset 1 Index of Version String.
- FIRMWARE_VERSION Version

    Offset 2-6 Firmware version.

4.5.1 Detailed Description

Firmware Version Information Structure.

Definition at line 36 of file FirmwareVersionInfoHob.h.

The documentation for this struct was generated from the following file:

- FirmwareVersionInfoHob.h

4.6 FIRMWARE_VERSION_INFO_HOB Struct Reference

Firmware Version Information HOB Structure.

```
#include <FirmwareVersionInfoHob.h>
```

Public Attributes

- EFI_HOB_GUID_TYPE Header

  Offset 0-23 The header of FVI HOB.
- UINT8 Count

  Offset 24 Number of FVI elements included.

4.6.1 Detailed Description

Firmware Version Information HOB Structure.

Definition at line 45 of file FirmwareVersionInfoHob.h.

4.6.2 Member Data Documentation

4.6.2.1 Count

```
UINT8 FIRMWARE_VERSION_INFO_HOB::Count
```

Offset 24 Number of FVI elements included.

Definition at line 47 of file FirmwareVersionInfoHob.h.

The documentation for this struct was generated from the following file:

- FirmwareVersionInfoHob.h

4.7 SMBIOS_CACHE_INFO Struct Reference

SMBIOS Cache Info HOB Structure.

```
#include <SmbiosCacheInfoHob.h>
```

Public Attributes

- UINT16 NumberOfCacheLevels

  Based on Number of Cache Types L1/L2/L3.
- UINT8 SocketDesignationStrIndex

  String Index in the string Buffer. Example "L1-CACHE".
- UINT16 CacheConfiguration

  Format defined in SMBIOS Spec v3.0 Section7.8 Table36.
- UINT16 MaxCacheSize

  Format defined in SMBIOS Spec v3.0 Section7.8.1.
- UINT16 InstalledSize

  Format defined in SMBIOS Spec v3.0 Section7.8.1.
- UINT16 SupportedSramType

  Format defined in SMBIOS Spec v3.0 Section7.8.2.
- UINT16 CurrentSramType

  Format defined in SMBIOS Spec v3.0 Section7.8.2.
- UINT8 CacheSpeed

  Cache Speed in nanoseconds. 0 if speed is unknown.
- UINT8 ErrorCorrectionType

  ENUM Format defined in SMBIOS Spec v3.0 Section 7.8.3.
- UINT8 SystemCacheType

  ENUM Format defined in SMBIOS Spec v3.0 Section 7.8.4.
- UINT8 Associativity

  ENUM Format defined in SMBIOS Spec v3.0 Section 7.8.5.

4.7.1 Detailed Description

SMBIOS Cache Info HOB Structure.

Definition at line 29 of file SmbiosCacheInfoHob.h.

The documentation for this struct was generated from the following file:

- SmbiosCacheInfoHob.h

4.8 SMBIOS_PROCESSOR_INFO Struct Reference

SMBIOS Processor Info HOB Structure.

```
#include <SmbiosProcessorInfoHob.h>
```

Public Attributes

- UINT8 ProcessorType

  ENUM defined in SMBIOS Spec v3.0 Section 7.5.1.
- UINT16 ProcessorFamily

  This info is used for both ProcessorFamily and ProcessorFamily2 fields See ENUM defined in SMBIOS Spec v3.0 Section 7.5.2.
- UINT8 ProcessorManufacturerStrIndex

  Index of the String in the String Buffer.
- UINT64 ProcessorId

  ENUM defined in SMBIOS Spec v3.0 Section 7.5.3.
- UINT8 ProcessorVersionStrIndex

  Index of the String in the String Buffer.
- UINT8 Voltage

  Format defined in SMBIOS Spec v3.0 Section 7.5.4.
- UINT16 ExternalClockInMHz

  External Clock Frequency. Set to 0 if unknown.
- UINT16 CurrentSpeedInMHz

  Snapshot of current processor speed during boot.
- UINT8 Status

  Format defined in the SMBIOS Spec v3.0 Table 21.
- UINT8 ProcessorUpgrade

  ENUM defined in SMBIOS Spec v3.0 Section 7.5.5.
- UINT16 CoreCount

  This info is used for both CoreCount & CoreCount2 fields See detailed description in SMBIOS Spec v3.0 Section 7.5.6.
- UINT16 EnabledCoreCount

  This info is used for both CoreEnabled & CoreEnabled2 fields See detailed description in SMBIOS Spec v3.0 Section 7.5.7.
- UINT16 ThreadCount

  This info is used for both ThreadCount & ThreadCount2 fields See detailed description in SMBIOS Spec v3.0 Section 7.5.8.
- UINT16 ProcessorCharacteristics

  Format defined in SMBIOS Spec v3.0 Section 7.5.9.

4.8.1 Detailed Description

SMBIOS Processor Info HOB Structure.

Definition at line 28 of file SmbiosProcessorInfoHob.h.

The documentation for this struct was generated from the following file:

- SmbiosProcessorInfoHob.h

Chapter 5

File Documentation

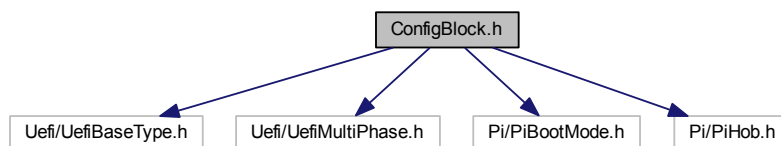## 5.1 ConfigBlock.h File Reference

Header file for Config Block Lib implementation.

```
#include <Uefi/UefiBaseType.h>
#include <Uefi/UefiMultiPhase.h>
#include <Pi/PiBootMode.h>
#include <Pi/PiHob.h>
```
Include dependency graph for ConfigBlock.h:

Classes

- struct _CONFIG_BLOCK_HEADER

    Config Block Header.
- struct _CONFIG_BLOCK

    Config Block.
- struct _CONFIG_BLOCK_TABLE_STRUCT

    Config Block Table Header.

Typedefs

- typedef struct _CONFIG_BLOCK_HEADER CONFIG_BLOCK_HEADER

    Config Block Header.
- typedef struct _CONFIG_BLOCK CONFIG_BLOCK

    Config Block.
- typedef struct _CONFIG_BLOCK_TABLE_STRUCT CONFIG_BLOCK_TABLE_HEADER

    Config Block Table Header.

5.1.1 Detailed Description

Header file for Config Block Lib implementation.

Copyright (c) 2015 - 2016, Intel Corporation. All rights reserved.
This program and the accompanying materials are licensed and made available under the terms and conditions of the BSD License that accompanies this distribution. The full text of the license may be found at http←
://opensource.org/licenses/bsd-license.php.

THE PROGRAM IS DISTRIBUTED UNDER THE BSD LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR REPRESENTATIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED.

5.2 ConfigBlockLib.h File Reference

Header file for Config Block Lib implementation.

Functions

- EFI_STATUS CreateConfigBlockTable (IN UINT16 TotalSize, OUT VOID ∗∗ConfigBlockTableAddress)

  Create config block table.
- EFI_STATUS AddConfigBlock (IN VOID ∗ConfigBlockTableAddress, OUT VOID ∗∗ConfigBlockAddress)

  Add config block into config block table structure.
- EFI_STATUS GetConfigBlock (IN VOID ∗ConfigBlockTableAddress, IN EFI_GUID ∗ConfigBlockGuid, OUT VOID ∗∗ConfigBlockAddress)

  Retrieve a specific Config Block data by GUID.

5.2.1 Detailed Description

Header file for Config Block Lib implementation.

Copyright (c) 2015 - 2016, Intel Corporation. All rights reserved.
This program and the accompanying materials are licensed and made available under the terms and conditions of the BSD License that accompanies this distribution. The full text of the license may be found at http←
://opensource.org/licenses/bsd-license.php.

THE PROGRAM IS DISTRIBUTED UNDER THE BSD LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR REPRESENTATIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED.

5.2.2 Function Documentation

5.2.2.1 AddConfigBlock()

```
EFI_STATUS AddConfigBlock (
            IN VOID * ConfigBlockTableAddress,
            OUT VOID ** ConfigBlockAddress )
```

Add config block into config block table structure.

Parameters

in	ConfigBlockTableAddress	- A pointer to the beginning of Config Block Table Address
out	ConfigBlockAddress	- On return, points to a pointer to the beginning of Config Block Address

Return values

EFI_OUT_OF_RESOURCES	- Config Block Table is full and cannot add new Config Block or Config Block Offset Table is full and cannot add new Config Block.
EFI_SUCCESS	- Successfully added Config Block

5.2.2.2 CreateConfigBlockTable()

```
EFI_STATUS CreateConfigBlockTable (
        IN UINT16 TotalSize,
        OUT VOID ** ConfigBlockTableAddress )
```

Create config block table.

Parameters

in	TotalSize	- Max size to be allocated for the Config Block Table
out	ConfigBlockTableAddress	- On return, points to a pointer to the beginning of Config Block Table Address

Return values

EFI_INVALID_PARAMETER	- Invalid Parameter
EFI_OUT_OF_RESOURCES	- Out of resources
EFI_SUCCESS	- Successfully created Config Block Table at ConfigBlockTableAddress

5.2.2.3 GetConfigBlock()

```
EFI_STATUS GetConfigBlock (
        IN VOID * ConfigBlockTableAddress,
        IN EFI_GUID * ConfigBlockGuid,
        OUT VOID ** ConfigBlockAddress )
```

Retrieve a specific Config Block data by GUID.

Parameters

| in	ConfigBlockTableAddress	- A pointer to the beginning of Config Block Table Address
in	ConfigBlockGuid	- A pointer to the GUID uses to search specific Config Block
out	ConfigBlockAddress	- On return, points to a pointer to the beginning of Config Block Address

Return values

EFI_NOT_FOUND	- Could not find the Config Block
EFI_SUCCESS	- Config Block found and return

5.3 FirmwareVersionInfoHob.h File Reference

Header file for Firmware Version Information.

```
#include <Uefi.h>
#include <Pi/PiHob.h>
```
Include dependency graph for FirmwareVersionInfoHob.h:
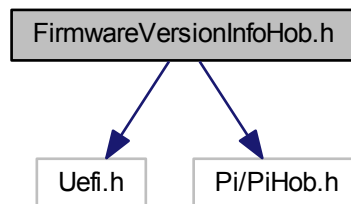
Classes

- struct FIRMWARE_VERSION

  Firmware Version Structure.
- struct FIRMWARE_VERSION_INFO

  Firmware Version Information Structure.
- struct FIRMWARE_VERSION_INFO_HOB

  Firmware Version Information HOB Structure.

5.3.1 Detailed Description

Header file for Firmware Version Information.

Copyright (c) 2015 - 2016, Intel Corporation. All rights reserved.

This program and the accompanying materials are licensed and made available under the terms and conditions of the BSD License which accompanies this distribution. The full text of the license may be found at http←://opensource.org/licenses/bsd-license.php

THE PROGRAM IS DISTRIBUTED UNDER THE BSD LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR REPRESENTATIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED.

5.4 HstiFeatureBit.h File Reference

This file contains various definitions for IHV HSTI implementation including error string definitions.

5.4.1 Detailed Description

This file contains various definitions for IHV HSTI implementation including error string definitions.

Copyright

INTEL CONFIDENTIAL Copyright 2015 - 2016 Intel Corporation.

The source code contained or described herein and all documents related to the source code ("Material") are owned by Intel Corporation or its suppliers or licensors. Title to the Material remains with Intel Corporation or its suppliers and licensors. The Material may contain trade secrets and proprietary and confidential information of Intel Corporation and its suppliers and licensors, and is protected by worldwide copyright and trade secret laws and treaty provisions. No part of the Material may be used, copied, reproduced, modified, published, uploaded, posted, transmitted, distributed, or disclosed in any way without Intel's prior express written permission.

No license under any patent, copyright, trade secret or other intellectual property right is granted to or conferred upon you by disclosure or delivery of the Materials, either expressly, by implication, inducement, estoppel or otherwise. Any license under such intellectual property rights must be express and approved by Intel in writing.

Unless otherwise agreed by Intel in writing, you may not remove or alter this notice or any other notice embedded in Materials by Intel or Intel's suppliers or licensors in any way.

This file contains 'Framework Code' and is licensed as such under the terms of your license agreement with Intel or your vendor. This file may not be modified, except as allowed by additional terms of your license agreement.

Specification

5.5 SmbiosCacheInfoHob.h File Reference

Header file for SMBIOS Cache Info HOB.

```
#include <Uefi.h>
#include <Pi/PiHob.h>
```

Include dependency graph for SmbiosCacheInfoHob.h:
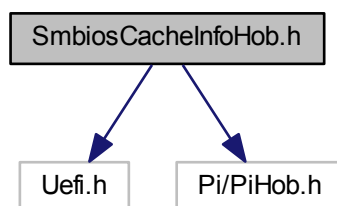
Classes

- struct SMBIOS_CACHE_INFO

  SMBIOS Cache Info HOB Structure.

5.5.1 Detailed Description

Header file for SMBIOS Cache Info HOB.

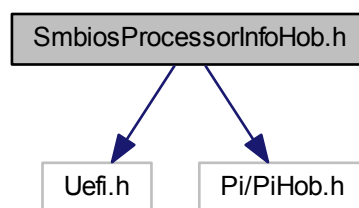Copyright (c) 2015 - 2016, Intel Corporation. All rights reserved.

System Management BIOS (SMBIOS) Reference Specification v3.0.0 dated 2015-Feb-12 (DSP0134) http↩://www.dmtf.org/sites/default/files/standards/documents/DSP0134_3.0.0.pdf

5.6 SmbiosProcessorInfoHob.h File Reference

Header file for SMBIOS Processor Info HOB.

```
#include <Uefi.h>
#include <Pi/PiHob.h>
```
Include dependency graph for SmbiosProcessorInfoHob.h:

Classes

- struct SMBIOS_PROCESSOR_INFO

  SMBIOS Processor Info HOB Structure.

5.6.1 Detailed Description

Header file for SMBIOS Processor Info HOB.

Copyright (c) 2015 - 2016, Intel Corporation. All rights reserved.
This program and the accompanying materials are licensed and made available under the terms and conditions of the BSD License that accompanies this distribution. The full text of the license may be found at `http⤦` `://opensource.org/licenses/bsd-license.php`.

THE PROGRAM IS DISTRIBUTED UNDER THE BSD LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR REPRESENTATIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED.

System Management BIOS (SMBIOS) Reference Specification v3.0.0 dated 2015-Feb-12 (DSP0134) `http⤦` `://www.dmtf.org/sites/default/files/standards/documents/DSP0134_3.0.0.pdf`

Index