Qualcomm Technologies, Inc.

Display Drivers (ACPI and XML)

Configuration Guide

LM80-P0337-1 Rev. B

February 26, 2018

Questions or comments: https://www.96boards.org/product/dragonboard820c/

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Revision history

Revision	Date	Description
A	September 2017	Initial release
B	February 2018	Updated Appendix C and D

Contents

Figures

Tables

1 Introduction

This document provides a description of chipset capabilities. Not all features are available, nor are all features supported in the software.

NOTE: Enabling some features require additional licensing fees.

1.1 Purpose

This document is intended for display developers who are looking to customize the display-specific Advanced Configuration and Power Interface (ACPI) entries for a particular display device. Knowledge of display interfaces, Windows display driver model (WDDM) architecture, and ACPI programming are prerequisites for this document. The XML syntax is common between UEFI and ACPI, therefore this document applies to both configuration methods.

This document supports the APQ8016E and APQ8096 chipsets. Configurations mentioned in this document are specific to these chipset families.

1.2 Acronyms and terms

Table 1-1 Acronyms

Acronym or term	Definition
ACPI	Advanced Configuration and Power Interface
AD	Assertive display
BLLP	Blanking low-power period
CABL	Content adaptive backlight level
EDID	Extended display identification data
ESD	Electrostatic discharge
HBP	Horizontal blanking period
HFP	Horizontal front porch
HS	High speed
HSA	Horizontal sync active
HSIC	High-Speed Inter-Chip Interface
HSP	Horizontal sync period
KMD	Kernel Mode Driver
PBRT	Panel backlight response table
PGCT	Panel gamma correction table
PGMT	Panel gamut mapping table

Acronym or term	Definition
PGRT	Panel gamma response table
PIGC	Panel inverse gamma correction
PPCC	Panel color correction
PPGC	Panel gamma correction
QDI	Qualcomm® display interface
SBC	Smooth backlight control
WDDM	Windows Display Driver Model

2 ACPI configuration for display drivers

2.1 Panel configuration in the boot sequence

The role of the panel configuration in the boot sequence is shown in Figure 2-1.

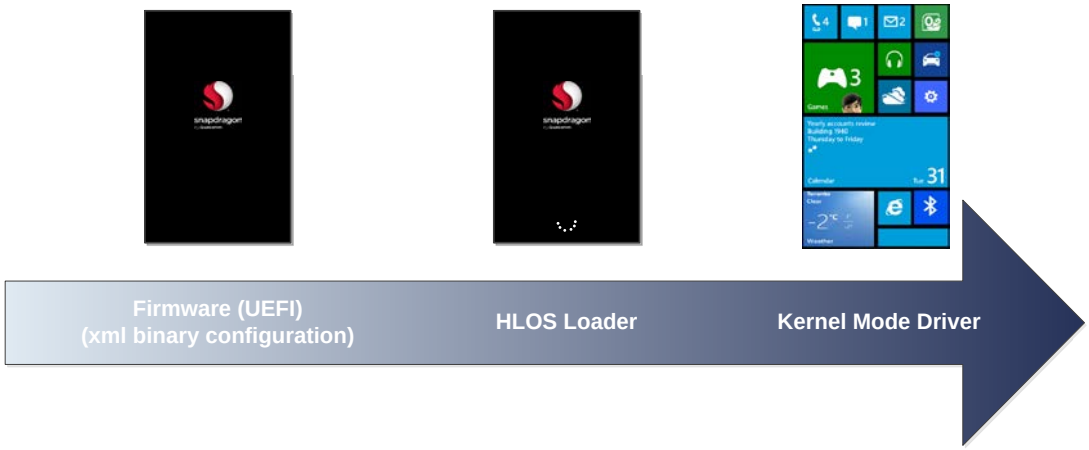


Figure 2-1 Panel configuration in the boot sequence

The following two-stage sequence is specific to HLOSes that use UEFI and ACPI:

- Firmware (UEFI) boot – During UEFI boot, the panel configuration is loaded directly from a panel configuration in the UEFI binary. This configuration is parsed and applied to the display controller.
- OS boot (graphics miniport) – During OS boot, the KMD parses the display configuration from the ACPI (_ROM) method.

Both boot stages use similar XML configurations. However, the data is physically located in two locations even though the data has many configurations that are identical. The reason for this is that the UEFI does not directly process data from the ACPI tables, and therefore the data cannot be shared between both boot sequences. The Figure 2-1 shows the role of panel configuration in the boot sequence

3 Panel configuration format

The panel configuration format is XML-like, only certain data fields are parsed and only certain tags are recognized.

3.1 Tag syntax

The tag syntax is similar to XML using braces for various tags.

```
<?xml version="1.0" encoding="utf-8"?>

<TagName comments=”any comments”>Tag Data</TagName>

<MultiLineTag>
Tag Data
More Data
</MultiLineTag>
```

- All tags must be unique except for certain keyword tags that are ignored
- All tags must end with a </TagName>
- Tags are case-insensitive
- Tags cannot be made from spaces or special characters

3.2 Special keyword tags

- The <?xml xxxxxxxx ?> tag is ignored.
- Tags that use the keyword Group are ignored. However, tags within the context of a group branch are parsed, for example, Group xyz is ignored, but MyTagA is parsed.

  ```
  <Group xyz>
      <MyTagA> data </MyTagA>
  </Group>
  ```

- Comments are added as <!-- Comment -->. However, comments should be avoided because they add extra parsing time and size to the configuration.
- Data appears after the tag name is ignored, but can be used for comments or informational purposes.

  ```
  <MyTag this_data is ignored = “”> tag data </MyTag>
  ```

3.3 Supported data types

Parsing of data supports several data types.

- Integer – Integer data can be in the form of decimal digits or hexadecimal digits. Depending on the integer size, the range can vary; for example, if the size is 1 byte, the range is 0x00-0xFF (0 to 255 decimal). Hexadecimal digits must be prefixed with "0x", for example, these are all valid integer values:
  - <MyIntegerTag>123456</MyIntegerTag>
  - <MyHexTag>0xFFEE23</MyHexTag>
- Integer list – Integer lists are an ordered list of integers separated by spaces. Each entry is interpreted as hexadecimal values regardless if it is prepended with 0x.
  - <MyIntergerList>0x1234 0x456 0xAA 0xBB</MyIntegerList>
- String – String data can be any alphanumeric sequence including spaces, for example:
  - <MyStringTag>Same String Data 1234</MyStringTag>
- Boolean – Boolean data consists of either the true or false keyword tag. The case does not matter.
  - <MyBooleanTag>True</MyBooleanTag>
  - <MyBooleanTag>false</MyBooleanTag>
- GUID – The GUID tag accepts a standard Windows GUID, for example:
  - <MyGUIDTag>{0xf9938f2d, 0x3756, 0x4760, 0xa0, 0x44, 0xcb, 0x29, 0xaf, 0xba, 0x5a, 0x69}</MyGUIDTag>
- Binary – The binary field is a special field that is used to describe raw information, that is, packet data. Each hexadecimal pair byte is separated by spaces and the "0x" prefix is not allowed. Multiple packets can be formed by adding a new byte sequence on each line.

NOTE: This is an important differentiator. Multiple bytes in the same line represent a single sequence, while multiple bytes on multiple lines represent multiple packet sequences.

```
<MyDataSequence>
23 b0 04 29 b3 00 87 29 b6 30 83
</MyDataSequence>
```

```
<MultiPacketSequence>
23 b0 04 29 b3 00
87 29 b6 30 83
</MultiPacketSequence >
```

3.4 Limitations

- ASCII only – Only ASCII characters must be used. Unicode and double byte characters are not supported.

- Limited XML support – The format of the panel configuration is XML-like, and not all XML tags are supported. Avoid using syntax that is not described in this document.
- Configuration size – The panel configuration size is bounded by the amount of memory allocated to store the configuration. Size limitations include all characters, including white spaces, tags, and control characters (line feeds and carriage returns).
  - Firmware – Static structure, no limitations other than compile limitations

4 Supported tag list

Table 4-1 provides a list of all of the currently supported panel configuration tags. This list is subject to change.

Table 4-1 Supported tag list

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
PanelName	String (13)	EDID reporting	No	Yes
PanelDescription	String (128)	EDID reporting	No	Yes
ManufactureID	Integer (2)	EDID reporting	No	Yes
ProductCode	Integer (2)	EDID reporting	No	Yes
SerialNumber	Integer (4)	EDID reporting	No	Yes
WeekofManufacture	Integer (1)	EDID reporting	No	Yes
YearofManufacture	Integer (1)	EDID reporting	No	Yes
EDIDVersion	Integer (1)	EDID reporting	No	Yes
EDIDRevision	Integer (1)	EDID reporting	No	Yes
VideoInputDefinition	Integer (1)	EDID reporting	No	Yes
HorizontalScreenSize	Integer (1)	EDID reporting	No	Yes
VerticalScreenSize	Integer (1)	EDID reporting	No	Yes
DisplayTransferCharacteristics	Integer (1)	EDID reporting	No	Yes
FeatureSupport	Integer (1)	EDID reporting	No	Yes
Red.GreenBits	Integer (1)	EDID reporting	No	Yes
Blue.WhiteBits	Integer (1)	EDID reporting	No	Yes
RedX	Integer (1)	EDID reporting	No	Yes
RedY	Integer (1)	EDID reporting	No	Yes
GreenX	Integer (1)	EDID reporting	No	Yes
GreenY	Integer (1)	EDID reporting	No	Yes
BlueX	Integer (1)	EDID reporting	No	Yes
BlueY	Integer (1)	EDID reporting	No	Yes
WhiteX	Integer (1)	EDID reporting	No	Yes
WhiteY	Integer (1)	EDID reporting	No	Yes
EstablishedTimingsI	Integer (1)	EDID reporting	No	Yes
EstablishedTimingsII	Integer (1)	EDID reporting	No	Yes
ManufacturesTiming	Integer (1)	EDID reporting	No	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
StandardTimings1	Integer (2)	EDID reporting	No	Yes
StandardTimings2	Integer (2)	EDID reporting	No	Yes
StandardTimings3	Integer (2)	EDID reporting	No	Yes
StandardTimings4	Integer (2)	EDID reporting	No	Yes
StandardTimings5	Integer (2)	EDID reporting	No	Yes
StandardTimings6	Integer (2)	EDID reporting	No	Yes
StandardTimings7	Integer (2)	EDID reporting	No	Yes
StandardTimings8	Integer (2)	EDID reporting	No	Yes
SignalTimingInterface	Integer (1)	EDID reporting	No	Yes
HorizontalScreenSizeMM	Integer (1)	EDID reporting	No	Yes
VerticalScreenSizeMM	Integer (1)	EDID reporting	No	Yes
HorizontalVerticalScreenSizeMM	Integer (1)	EDID reporting	No	Yes
Display Hardware Configuration				
InterfaceType	Integer (4)	Display hardware configuration	Yes	Yes
PanelOrientation	Integer (4)	Display hardware configuration	No	No
InterfaceColorFormat	Integer (4)	Display hardware configuration	Yes	Yes
ComponentOrdering	Integer (4)	Display hardware configuration	Yes	Yes
PixelPacking	Integer (4)	Display hardware	Yes	Yes
PixelAlignment	Integer (4)	Configuration	Yes	Yes
HorizontalActive	Integer (4)	Configuration	Yes	Yes
HorizontalFrontPorch	Integer (4)	Display hardware	Yes	Yes
HorizontalBackPorch	Integer (4)	Configuration	Yes	Yes
HorizontalSyncPulse	Integer (4)	Display hardware	Yes	Yes
HorizontalSyncSkew	Integer (4)	Configuration	Yes	Yes
HorizontalLeftBorder	Integer (4)	Display hardware	Yes	Yes
HorizontalRightBorder	Integer (4)	Configuration	Yes	Yes
VerticalActive	Integer (4)	Display hardware	Yes	Yes
VerticalFrontPorch	Integer (4)	Configuration	Yes	Yes
VerticalBackPorch	Integer (4)	Display hardware	Yes	Yes
VerticalSyncPulse	Integer (4)	Configuration	Yes	Yes
VerticalTopBorder	Integer (4)	Display hardware	Yes	Yes
VerticalBottomBorder	Integer (4)	Configuration	Yes	Yes
InvertDataPolarity	Boolean	Display hardware	Yes	Yes
InvertVsyncPolairty	Boolean	Configuration	Yes	Yes
InvertHsyncPolarity	Boolean	Display hardware	Yes	Yes
BorderColor	Integer (4)	Configuration	Yes	Yes
UnderflowColor	Integer (4)	Configuration	No	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
DisplayPrimaryFlags	Integer (4)	Display configuration	No	Yes
DisplayExternalFlags	Integer (4)	Display configuration	No	Yes
ESDDetectionTime	Integer (4)	ESD detection and recovery	No	Yes
ESDDetectionFailureRetry	Integer (4)	ESD detection and recovery	No	Yes
DisplayRecoveryThreshold	Integer (4)	ESD detection and recovery	No	Yes
DSI hardware configuration				
DSIRefreshRate	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDynamicRefreshRates	Integer list (16)	DSI hardware configuration	No	Yes
DSIBitClockFrequency	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDynamicBlankingRefreshRate List	Integer list (16)	DSI hardware configuration	No	Yes
DSIDynamicVFrontPorchList	Integer list (16)	DSI hardware configuration	No	Yes
DSIDynamicVBackPorchList	Integer list (16)	DSI hardware configuration	No	Yes
DSIDynamicVSyncPulseList	Integer list (16)	DSI hardware configuration	No	Yes
DSIDynamicHFrontPorchList	Integer list (16)	DSI hardware configuration	No	Yes
DSIDynamicHBackPorchList	Integer list (16)	DSI hardware configuration	No	Yes
DSIDynamicHSyncPulseList	Integer list (16)	DSI hardware configuration	No	Yes
DSILanes	Integer (4)	DSI hardware configuration	Yes	Yes
DSIChannelId	Integer (4)	DSI hardware configuration	Yes	Yes
DSIVirtualId	Integer (4)	DSI hardware configuration	Yes	Yes
DSIColorFormat	Integer (4)	DSI hardware configuration	Yes	Yes
DSIPacketTransferHS	Boolean	DSI hardware configuration	Yes	Yes
DSIClockHSForceRequest	Integer (4)	DSI hardware configuration	Yes	Yes
DSIPixelXferTiming	Integer (4)	DSI hardware configuration	Yes	Yes
DSIHostLaneMapping	Integer (4)	DSI hardware configuration	Yes	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
DSILP11AtInit	Integer (4)	DSI hardware configuration	Yes	Yes
DSIPhyDCDCMode	Boolean	DSI hardware configuration	Yes	Yes
DSIEnterULPSPowerDown	Boolean	DSI hardware configuration	Yes	Yes
DSIBitClkScalePercent	Integer (4)	DSI hardware configuration	No	Yes
DSIBitClkScalePercent	Integer (4)	DSI hardware configuration	No	Yes
DSIEscapeClockDivisor	Integer (4)	DSI hardware configuration	Yes	Yes
DSIEscapeClockFrequency	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingHSZeroOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingHSZeroValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingHSExitOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingHSExitValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingHSPrepareOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingHSPrepareValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingHSTrailOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingHSTrailValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingHSRequestOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingHSRequestValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingCLKZeroOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingCLKZeroValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingCLKTrailOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingCLKTrailValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingCLKPrepareOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingCLKPrepareValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingCLKPreOverride	Boolean	DSI hardware configuration	Yes	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
DSITimingCLKPreValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingCLKPostOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingCLKPostValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingTASureOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingTASureValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingTAGoOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingTAGoValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITimingTAGetOverride	Boolean	DSI hardware configuration	Yes	Yes
DSITimingTAGetValue	Integer (4)	DSI hardware configuration	Yes	Yes
DSITrafficMode	Integer (4)	DSI hardware configuration	Yes	Yes
DSIHsaHseAfterVsVe	Boolean	DSI hardware configuration	Yes	Yes
DSILowPowerModeInHFP	Boolean	DSI hardware configuration	Yes	Yes
DSILowPowerModeInHBP	Boolean	DSI hardware configuration	Yes	Yes
DSILowPowerModeInHSA	Boolean	DSI hardware configuration	Yes	Yes
DSILowPowerModeInBLLPEOF	Boolean	DSI hardware configuration	Yes	Yes
DSIForceCmdInVideoHS	Boolean	DSI hardware configuration	Yes	Yes
DSILowPowerModeInBLLP	Boolean	DSI hardware configuration	Yes	Yes
DSICMDSwapInterface	Boolean	DSI hardware configuration	Yes	Yes
DSICMDUsingTrigger	Boolean	DSI hardware configuration	Yes	Yes
DSITECheckEnable	Boolean	DSI hardware configuration	Yes	Yes
DSITEUsingDedicatedTEPin	Boolean	DSI hardware configuration	Yes	Yes
DSITEvSyncStartPos	Integer (4)	DSI hardware configuration	Yes	Yes
DSITEvSyncContinueLines	Integer (4)	DSI hardware configuration	Yes	Yes
DSITEvSyncStartLineDivisor	Integer (4)	DSI hardware configuration	Yes	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
DSITEvSyncPosSafetyMargin	Integer (4)	DSI hardware configuration	Yes	Yes
DSITEvSyncBelowSafetyMargin	Integer (4)	DSI hardware configuration	Yes	Yes
DSITEPercentVariance	Integer (4)	DSI hardware configuration	Yes	Yes
DSITEvSyncRdPtrIrqLine	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDataStrengthLP	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDataStrengthHS	Integer (4)	DSI hardware configuration	Yes	Yes
DSIClockStrengthHS	Integer (4)	DSI hardware configuration	Yes	Yes
DSIEnableAutoRefresh	Boolean	DSI hardware configuration	Yes	No
DSIAutoRefreshFrameNumDiv	Integer (4)	DSI hardware configuration	Yes	No
DSIPhyDCDCMode	Boolean	DSI hardware configuration	Yes	Yes
DSIInitSequence	Binary	DSI panel configuration	Yes	Yes
DSITermSequence	Binary	DSI panel configuration	No	Yes
DSIStatusSequence	Binary	DSI panel status check	No	Yes
DSIDisableEoTAfterHSXfer	Boolean	DSI hardware configuration	Yes	Yes
DSIInitMasterTime	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDmaDelayAfterVsync	Integer (4)	DSI DMA scheduling	No	Yes
DSITEvSyncSelect	Integer (4)	DSI hardware configuration	Yes	Yes
DSICmdModeIdleTime	Integer (4)	DSI hardware configuration	Yes	Yes
DSIFBCEnable	Boolean	Display/DSI hardware configuration	Yes	Yes
DSIFBCProfileID	Integer (4)	Display/DSI hardware configuration	Yes	Yes
DSIControllerMapping	Integer List (16)	DSI hardware configuration	Yes	Yes
DSISlaveControllerSkewLines	Integer (4)	DSI hardware configuration	Yes	Yes
DSITransferRetryCntr	Integer (4)	DSI hardware configuration	Yes	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
DSINullpacketInsertionBytes	Integer (4)	DSI hardware configuration	Yes	Yes
DSIFlags	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDSCEnable	Boolean	DSI hardware configuration	Yes	Yes
DSIDSCProfileId	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDSCSliceHeight	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDSCSliceWidth	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDSCMajorVersion	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDSCMinorVersion	Integer (4)	DSI hardware configuration	Yes	Yes
DSIDSCScrVersion	Integer (4)	DSI hardware configuration	Yes	Yes
eDP configuration				
EDPRefreshRate	Integer (4)	eDP hardware configuration	Yes	Yes
EDPTraining	Integer (4)	eDP hardware configuration	Yes	Yes
EDPPixelClockFrequency	Integer (4)	eDP hardware configuration	Yes	Yes
EDPDPCDRead	Boolean	eDP hardware configuration	Yes	Yes
EDPEDIDRead	Boolean	eDP hardware configuration	Yes	Yes
EDPNumberOfLanes	Integer (4)	eDP hardware configuration	Yes	Yes
EDPLinkRate	Integer (4)	eDP hardware configuration	Yes	Yes
EDPASSREnable	Boolean	eDP hardware configuration	Yes	Yes
EDPEnhancedFrameEnable	Boolean	eDP hardware configuration	Yes	Yes
EDPMaxLinkRate	Integer (4)	eDP hardware configuration	Yes	Yes
EDPRGBMap	Integer (4)	eDP hardware configuration	Yes	Yes
EDPLaneMap	Integer (4)	eDP hardware configuration	Yes	Yes
EDPHPDActiveLow	Boolean	eDP hardware configuration	Yes	Yes
EDPDynamicRefreshRates	Integer List	eDP hardware configuration	No	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
EDPStatusSequence	Binary	eDP panel status check	No	Yes
EDPPowerUpWaitInMs	Integer (4)	eDP hardware configuration	Yes	Yes
EDPMaxAuxRetry	Integer (4)	eDP hardware configuration	Yes	Yes
HDMI configuration				
HDMIAviInfoFramePacketsDisable	Boolean	HDMI hardware configuration	No	Yes
HDMIOutVoltageSwingCtrlEnable	Boolean	HDMI hardware configuration	No	Yes
HDMIOutVoltageSwingCtrl	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMaxNumReAuthentication	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMaxModeWidth	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMaxModeHeight	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMaxModeRefreshRate	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMinModeWidth	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMinModeHeight	Integer (4)	HDMI hardware configuration	No	Yes
HDMIMinModeRefreshRate	Integer (4)	HDMI hardware configuration	No	Yes
HDMIInjectedModeList	Integer List	HDMI hardware configuration	No	Yes
HDMIDDCTimeoutInMs	Integer (4)	HDMI hardware configuration	No	Yes
Backlight configuration				
BacklightType	Integer (4)	Backlight configuration	No	Yes
BacklightPmicModel	Integer (4)	Backlight configuration	No	Yes
BacklightPMICNum	Integer (4)	Backlight configuration	No	Yes
BacklightPmicControlType	Integer (4)	Backlight configuration	No	Yes
BacklightPMICBankSelect	Integer (4)	Backlight configuration	No	Yes
BacklightPMICPWMFrequency	Integer (4)	Backlight configuration	No	Yes
BacklightSteps	Integer (4)	Backlight configuration	No	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
BacklightDefault	Integer (4)	Backlight configuration	No	Yes
BacklightLowPower	Integer (4)	Backlight configuration	No	Yes
BacklightPmicAdvancedConfig	Boolean	Advance WLED backlight configuration	No	Yes
BacklightPmicWledInternalModResolution	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledModulationClkSel	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledDimmingMethod	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledOvp	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledIlim	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledFeedbackCtrl	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWlepLoopCompRes	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledVrefControl	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledFullScaleCurrent	Integer (4)	Advance WLED backlight configuration	No	Yes
BacklightPmicWledModulatorSrcSel	Integer (4)	Advance WLED backlight configuration	No	Yes
CABL configuration				
CABLMinUserLevel	Integer (4)	CABL configuration	No	Yes
CABLMinBacklightLevel	Integer (4)	CABL configuration	No	Yes
CABLFilterThreshold	Integer (4)	CABL configuration	No	Yes
Platform hardware configuration				
Display1Reset1Info	String (64)	Platform GPIO configuration	No	Yes
Display1Power1Info	String (64)	Platform GPIO configuration	No	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
Display1Power2Info	String (64)	Platform GPIO configuration	No	Yes
Display1Power3Info	String (64)	Platform GPIO configuration	No	Yes
Display1I2C1Info	String (64)	Platform I2C configuration	No	Yes
Display1I2C2Info	String (64)	Platform I2C configuration	No	Yes
Display1Special1Info	String (64)	Platform GPIO configuration	No	Yes
Display4Reset1Info	String (64)	Platform GPIO configuration	No	Yes
Display4Power1Info	String (64)	Platform GPIO configuration	No	Yes
Display4Power2Info	String (64)	Platform GPIO configuration	No	Yes
Display4Power3Info	String (64)	Platform GPIO configuration	No	Yes
Display4Special1Info	String (64)	Platform GPIO configuration	No	Yes
PMIPowerPmicModel	Integer (4)	PMIC Power configuration	Yes	Yes
PMIPowerPmicNum	Integer (4)	PMIC Power configuration	Yes	Yes
PMIPowerConfig	Integer (4)	PMIC Power configuration	Yes	Yes
Dynamic EDID configuration				
DynamicEDIDEnabled	Boolean	Dynamic EDID configuration	Yes	Yes
DynamicEDIDI2CSlaveAddress	Integer (4)	Dynamic EDID configuration	Yes	No
DynamicEDIDI2CFrequency	Integer (4)	Dynamic EDID configuration	Yes	No
DynamicEDIDI2CGSBIPort	Integer (4)	Dynamic EDID configuration	Yes	No
DynamicEDIDPTM	Integer (4)	Dynamic EDID configuration	Yes	Yes
DynamicEDIDStartAddress	Integer (4)	Dynamic EDID configuration	Yes	Yes
Adaptive brightness				
AdaptiveBrightnessFeature	Integer (4)	Adaptive brightness configuration	No	Yes
CABLEnable	Boolean	Adaptive brightness configuration	No	Yes

Panel name	Type (size in bytes)	Purpose	Firmware (UEFI) supported	Graphics KMD supported
SVIEnable	Boolean	Adaptive brightness configuration	No	Yes
Assertive display (AD)				
ADEnable	Boolean	AD configuration	No	Yes
ADMaxIterations	Integer (4)	AD configuration	No	Yes
ADStrengthLimit	Integer (4)	AD configuration	No	Yes
ADBacklightMin	Integer (4)	AD configuration	No	Yes
ADBacklightMax	Integer (4)	AD configuration	No	Yes
ADAmbientLightMin	Integer (4)	AD configuration	No	Yes
ADCalibrationA	Integer (4)	AD configuration	No	Yes
ADCalibrationB	Integer (4)	AD configuration	No	Yes
ADCalibrationC	Integer (4)	AD configuration	No	Yes
ADCalibrationD	Integer (4)	AD configuration	No	Yes
ADFilterA	Integer (4)	AD configuration	No	Yes
ADFilterB	Integer (4)	AD configuration	No	Yes
ADTFilterControl	Integer (4)	AD configuration	No	Yes
ADAssymetry	Binary	AD configuration	No	Yes
ADColorCorrection	Binary	AD configuration	No	Yes
ADSensorLinearization	Binary	AD configuration	No	Yes
ADPrivateData	Binary	AD configuration	No	Yes
Power saving configuration				
DisplayPowerSavingOverride	Integer (4)	Power saving	No	Yes
Sharpening configuration				
SharpeningEdgeThreshold	Integer (4)	Sharpening	No	Yes
SharpeningSmoothThreshold	Integer (4)	Sharpening	No	Yes
SharpeningNoiseThreshold	Integer (4)	Sharpening	No	Yes
GPIO configuration				
TLMMGPIODefaultLow	Integer list (8)	GPIO configuration	Yes	No
TLMMGPIODefaultHigh	Integer list (8)	GPIO configuration	Yes	No

5 Panel configuration tag descriptions

5.1 Informational fields

The fields listed in Table 5-1 are used for informational purposes only. They are optional fields that may or may not be parsed or used.

Table 5-1 Informational fields

Tag	Description
PanelDescription	Used for debugging purposes

5.2 EDID fields

The graphics KMD needs to report accurate EDID information about all monitors, internal, and external. For details, see the *VESA Enhanced Extended Display Identification Data – Implementation Guide* (Version 1.0 (June 2001)).

Only the first 128-byte block of the EDID is reported to the operating system. No extension blocks are supported.

5.2.1 Static EDID fields

The fields in Table 5-2 are statically defined in the panel configuration and reported by the operating system. The OEM is responsible for populating these fields to meet the panel definition.

Table 5-2 Static EDID fields

Tag	EDID field offset	EDID field description
PanelName	58h	Detailed timing #3 or display descriptor
ProductCode	0Ah	ID product code
SerialNumber	0Ch	ID serial number
WeekofManufacture	10h	Week of manufacture
YearofManufacture	11h	Year of manufacture
EDIDVersion	12h	Version number
EDIDRevision	13h	Revision number
VideoInputDefinition	14h	Video input definition
HorizontalScreenSize	15h	Horizontal screen size in cm of aspect ratio
VerticalScreenSize	16h	Vertical screen size in cm of aspect ratio
DisplayTransferCharacteristics	17h	Display transfer characteristics (gamma)

Tag	EDID field offset	EDID field description
FeatureSupport	18h	Feature support
Red.GreenBits	19h	Red/green low-order bits
Blue.WhiteBits	1Ah	Blue/white low-order bits
RedX	1Bh	Red-x high-order bits
RedY	1Ch	Red-y high-order bits
GreenX	1Dh	Green-x high-order bits
GreenY	1Eh	Green-y high-order bits
BlueX	1Fh	Blue-x high-order bits
BlueY	20h	Blue-y high-order bits
WhiteX	21h	White-x high-order bits
WhiteY	22h	White-y high-order bits
EstablishedTimingsI	23h	Established timings I
EstablishedTimingsII	24h	Established timings I
ManufacturesTiming	25h	Established timings I
StandardTimings1	26h	Standard timing 1
StandardTimings2	28h	Standard timing 2
StandardTimings3	2Ah	Standard timing 3
StandardTimings4	2Ch	Standard timing 4
StandardTimings5	2Eh	Standard timing 5
StandardTimings6	30h	Standard timing 6
StandardTimings7	32h	Standard timing 7
StandardTimings8	34h	Standard timing 8

5.2.2 Detailed timing fields

A single mode is populated in the detailed timing fields based on the EDID configuration given in Table 5-3, which describes how fields can be overwritten from the detailed timing descriptors.

Table 5-3 Detailed timing fields

Tag	Detailed timing offset	Detailed timing description
HorizontalScreenSizeMM	0Ch	Horizontal addressable video image size in mm (lower 8 bits)
VerticalScreenSizeMM	0Dh	Vertical addressable video image size in mm (lower 8 bits)
HorizontalVerticalScreenSizeMM	0Eh	Vertical and horizontal upper 4 bits

5.2.3 EDID information

Correct EDID information is important to the overall user experience. Inaccurate or invalid EDID information could result in an overall unpleasant user experience. Specifically, screen dimensions are important to the overall look and feel of the user interface.

The screen size in both cm and mm must be reported. For example, if the phone dimensions are 5.8 cm and 10.3 cm, as shown in Figure 5-1, the EDID is populated as shown in Figure 5-1.


```
<Group id='EDID Configuration'>
  <HorizontalScreenSize>5</HorizontalScreenSize>
  <VerticalScreenSize>10</VerticalScreenSize>
</Group>

<Group id='Detailed Timing'>
 <HorizontalScreenSizeMM>58</HorizontalScreenSizeMM>
 <VerticalScreenSizeMM>103</VerticalScreenSizeMM>
</Group
```

Figure 5-1 Example phone dimensions and EDID population

5.2.4 Dynamic EDID fields

Only the Preferred Timing mode (see Section 3.10.2 in *VESA Enhanced Extended Display Identification Data Standard* ((Defines EDID Structure Version 1, Revision 4) Release A, Revision 2 (September 2006)) is populated by the driver dynamically, based on the panel configuration. Table 5-4 describes how each field in the detailed timing definition is populated.

Table 5-4 Dynamic EDID fields

Detailed timing field name	Detailed timing specification field offset (size in bytes)	Derived from (panel configuration field)
Pixel clock/10000	0 (2)	DSIRefreshRate/LVDSRefreshRate
Horizontal addressable video (low 8 bits)	2 (1)	HorizontalActive
Horizontal blanking (low 8 bits)	3 (1)	HorizontalFrontPorch+Horizontal BackPorch+HorizontalSyncPulse
Horizontal addressable video (high 4 bits)/ Horizontal blanking (high 4 bits)	4 (1)	Horizontal Active/Horizontal FrontPorch+HorizontalBackPorch+ HorizontalSyncPulse
Vertical addressable video (lower 8 bits)	5 (1)	VerticalActive
Vertical blanking (low 8 bits)	6 (1)	VerticalFrontPorch+ VerticalBackPorch+ VerticalSyncPulse

Detailed timing field name	Detailed timing specification field offset (size in bytes)	Derived from (panel configuration field)
Vertical addressable video (high 4 bits)/ Vertical blanking (high 4 bits)	7 (1)	VerticalActive/VerticalFrontPorch+ VerticalBackPorch+ VerticalSyncPulse
Horizontal Front Porch (HFP) (lower 8 bits)	8 (1)	VerticalFrontPorch
Horizontal sync pulse width (lower 8 bits)	9 (1)	VerticalSyncPulse
Vertical front porch (high 4 bits)/vertical sync pulse width (high 4 bits)	10 (1)	VerticalFrontPorch/VerticalFrontPorch
HFP/ sync pulse/vertical front porch/vertical sync pulse width (upper 2 bits)	11 (1)	HorizontalFrontPorch/HorizontalSync Pulse/VerticalFrontPorch/VerticalSyncP ulse
Horizontal addressable video image size (lower 8 bits)	12 (1)	HorizontalScreenSize
Vertical addressable video image size (lower 8 bits)	13 (1)	VerticalScreenSize
Horizontal addressable video size (upper 4 bits)/vertical addressable video size (upper 4 bits)	14 (1)	HorizontalActive/VerticalActive
Right/left horizontal border	15 (1)	Used to determine the total active window width
Top/bottom vertical border	16 (1)	Used to determine the total active window height
Signal interface type	17 (1)	▪ InvertHsyncPolarity ▪ InvertVsyncPolairty

5.3 Panel timings configuration

All panel timings are derived from the following fields in the panel configuration as shown in Figure 5-2. It is critical that the licensees populate these fields correctly to match their platform configuration.

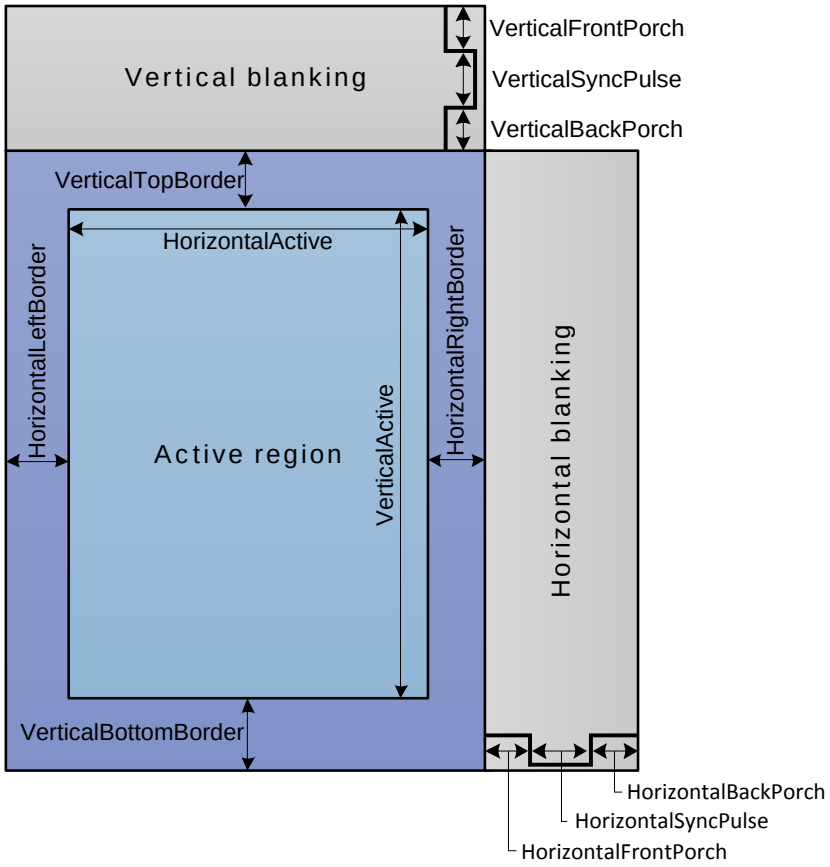


Figure 5-2 Panel timings configuration

Table 5-5 lists the panel timing fields.

Table 5-5 Panel timing fields

Tag	Units	Notes
HorizontalActive	Pixels	–
HorizontalFrontPorch	Pixels	–
HorizontalBackPorch	Pixels	–
HorizontalSyncPulse	Pixels	–
HorizontalSyncSkew	Pixels	Not used
HorizontalLeftBorder	Pixels	Optional, if supported by hardware
HorizontalRightBorder	Pixels	Optional, if supported by hardware
VerticalActive	Lines	–
VerticalFrontPorch	Lines	–

Tag	Units	Notes
VerticalBackPorch	Lines	–
VerticalSyncPulse	Lines	–
VerticalTopBorder	Lines	Optional, if supported by hardware
VerticalBottomBorder	Lines	Optional, if supported by hardware
InvertDataPolarity	Boolean	Inverts the data sync pulse polarity
InvertVsyncPolairty	Boolean	Inverts the vertical sync pulse polarity
InvertHsyncPolarity	Boolean	Inverts the horizontal sync pulse polarity
BorderColor	ARGB8888	Defines the default border color when borders are enabled
UnderflowColor	ARGB8888	Defines the default color when underflow occurs

6 Display hardware tag descriptions

6.1 Common hardware configuration parameters

Table 6-1 lists configuration tags that define the interface information for the display. These settings are critical since they determine the basic interface and mode of communication with the display.

Table 6-1 Common hardware configuration parameters

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>InterfaceType</td><td>Integer defining the interface between the APQ and the panel, based on the enumerated type QDI_DisplayConnectType
<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>0</td><td>None, reserved</td></tr>
<tr><td>1</td><td>EBI2, reserved</td></tr>
<tr><td>2</td><td>LCDC, reserved</td></tr>
<tr><td>3</td><td>MDDI, reserved</td></tr>
<tr><td>4</td><td>MDDI, reserved</td></tr>
<tr><td>5</td><td>TV, analog TV, reserved</td></tr>
<tr><td>6</td><td>MDDI, reserved</td></tr>
<tr><td>7</td><td>DTV, HDMI, reserved</td></tr>
<tr><td>8</td><td>QDI_DISPLAY_CONNECT_PRIMARY_DSI_VIDEO</td></tr>
<tr><td>9</td><td>QDI_DISPLAY_CONNECT_PRIMARY_DSI_CMD</td></tr>
<tr><td>10</td><td>DSI, reserved</td></tr>
<tr><td>11</td><td>DSI, reserved</td></tr>
<tr><td>12</td><td>LVDS, reserved</td></tr>
<tr><td>13</td><td>Frame buffer, reserved</td></tr>
<tr><td>14</td><td>QDI_DISPLAY_CONNECT_DP</td></tr>
<tr><td>15</td><td>DBI, reserved</td></tr>
<tr><td>16</td><td>MHL, reserved</td></tr>
</table>
All other values are invalid or not supported.</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>PanelOrientation</td><td>Integer defining the physical panel orientation

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_ROTATE_NONE</td></tr><tr><td>2</td><td>QDI_ROTATE_180</td></tr></table>
Currently only 0° rotation (none) is supported.</td></tr>
<tr><td>InterfaceColorFormat</td><td>Integer defining the pixel format (pixel depth) between the APQ and the panel, based on the enumerated type QDI_PixelFormatType

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>1</td><td>QDI_PIXEL_FORMAT_RGB_565_16BPP</td></tr><tr><td>2</td><td>QDI_PIXEL_FORMAT_RGB_666_18BPP</td></tr><tr><td>3</td><td>QDI_PIXEL_FORMAT_RGB_888_24BPP</td></tr></table>
All other values are invalid or not supported.</td></tr>
<tr><td>ComponentOrdering</td><td>Integer defining the pixel component ordering between the APQ and the panel, based on the enumerated type QDI_ColorOrderingType

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_COLOR_ORDERING_RGB</td></tr><tr><td>1</td><td>QDI_COLOR_ORDERING_RBG</td></tr><tr><td>2</td><td>QDI_COLOR_ORDERING_GRB</td></tr><tr><td>3</td><td>QDI_COLOR_ORDERING_GBR</td></tr><tr><td>4</td><td>QDI_COLOR_ORDERING_BRG</td></tr><tr><td>5</td><td>QDI_COLOR_ORDERING_BGR</td></tr></table>
All other values are invalid or not supported.</td></tr>
<tr><td>PixelPacking</td><td>Integer defining the pixel packing format between the APQ and the panel; this is based on the enumerated type QDI_ColorComponentsPackingType. Packing is only valid for pixel depths that are not aligned on an 8-bit boundary, such as 18 bpp.

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_COLOR_COMPONENTS_PACKING_TIGHT</td></tr><tr><td>1</td><td>QDI_COLOR_COMPONENTS_PACKING_LOOSE</td></tr></table>
All other values are invalid or not supported.</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>PixelAlignment</td><td>Integer defining the pixel alignment between the APQ and the panel, based on the enumerated type QDI_ColorComponentAlignmentType; alignment is only valid for loose pixel packet format types

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_COLOR_COMPONENT_ALIGNMENT_LSB</td></tr><tr><td>1</td><td>QDI_COLOR_COMPONENT_ALIGNMENT_MSB</td></tr></table>
All other values are invalid or not supported.</td></tr>
</table>

7 Display features

7.1 Display feature flags

Individual display features can be configured based on the feature flags given in Table 7-1. These are the feature flags that can be enabled or disabled.

Table 7-1 Feature flags that can be enabled or disabled

Tag	Description
DisplayPrimaryFlags	Integer defining with each bit representing a feature that can be enabled or disabled on the primary panel. Multiple flags can be combined to enable/disable multiple features.

Value (decimal)	Definition
0x0001	QDI_PANEL_FLAG_FORCE_DISPLAY_REINIT This flag forces the display to be reinitialized during the transition from the firmware (UEFI) to the KMD. The default behavior is to skip initialization when the display is detected as active after the firmware has completed execution.
0x0002	QDI_PANEL_FLAG_DISABLE_SBC This flag disables smooth brightness control. If the licensee chooses not to support this feature, this can be disabled with this flag.
0x0004	QDI_PANEL_FLAG_DISABLE_CABL This flag disables content adaptive backlight leveling.
0x0008	QDI_PANEL_FLAG_DISABLE_HW_CURSOR This flag disables hardware cursors and fallback to OS rendered cursors (software cursors).
0x0010	QDI_PANEL_FLAG_DUAL_MODE_SUPPORT This flag enables the cap to allow the driver to transition between MIPI DSI command and video mode operation when the display mode is being reinitialized.

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td></td><td>
<table>
<tr><td>0x0100</td><td>QDI_PANEL_FLAG_DSI_DCS_POLLING_TRANSFER
This flag disables the interrupted based notification of MIPI DSI DCS packets and reverts to a polling-based mechanism.</td></tr>
<tr><td>0x0200</td><td>QDI_PANEL_FLAG_DSI_DCS_DMA_ONLY
This flag is deprecated; do not use. See DSIFlags for details on DCS configuration.</td></tr>
<tr><td>0x0400</td><td>QDI_PANEL_FLAG_DSI_DCS_FIFO_ONLY
This flag is deprecated; do not use. See DSIFlags for details on DCS configuration.</td></tr>
<tr><td>0x0800</td><td>QDI_PANEL_FLAG_ENABLE_DFS_IDLESCREEN
This flag enables lowering of the refresh rate during idle screen (Vsync off) for video mode panels. This is applicable to configurations that enable dynamic refresh.</td></tr>
<tr><td>0x1000</td><td>QDI_PANEL_FLAG_MIRROR_DUAL_PIPE_CONFIG
This is an internal flag used to swap pixel data between DSI 0 and DSI 1.</td></tr>
<tr><td>0x00002000</td><td>QDI_PANEL_FLAG_DISABLE_POST_PROCESSING
This flag disables the postprocessing pipeline within the display controller. Pixels will not be modified post-blending.</td></tr>
<tr><td>0x00004000</td><td>QDI_PANEL_FLAG_INJECT_DEFAULT_AUDIO_MODE
This flag is used to force the driver to inject a default audio mode (LPCM, 44.1 kHz) when an external display EDID is being parsed. This flag can be used to force overrides for displays that report incorrect or missing audio EDID information.</td></tr>
<tr><td>0x00008000</td><td>QDI_PANEL_FLAG_STEREO_MODE_SUPPORT
This flag is used to force the driver into a VR/AR composition mode that breaks composition into a left and right side frame by hardware layer mixer.</td></tr>
<tr><td>0x01000000</td><td>QDI_PANEL_FLAG_DISABLE_SEAMLESS_SPLASH
This flag disables the display after UEFI is complete. If enabled, this flag disables the display from scanning out after the HLOS image starts.</td></tr>
<tr><td>All other values</td><td>Reserved</td></tr>
</table>
All other values are invalid or not supported.
</td></tr>
<tr><td>DisplayExternalFlags</td><td>Integer defining with each bit representing a feature that can be enabled or disabled on external panel (HDMI). Multiple flags can be combined to enable/disable multiple features.
<table>
<tr><td>Value (decimal)</td><td>Definition</td></tr>
<tr><td>All values</td><td>Reserved</td></tr>
</table>
</td></tr>
</table>

8 Display interface-specific configurations

8.1 DSI interface configurations

8.1.1 Common DSI configurations

Table 8-1 contains the configuration parameters that are common for both DSI Command mode and DSI Video mode.

Table 8-1 Common DSI configurations

Tag	Description
DSIRefreshRate	The requested refresh rate in Q16.16 format. This parameter allows the licensee to configure the rate at which the panel is updated. The driver will be responsible for calculating the necessary DSI timings and clock settings to achieve the requested refresh rate, for example: ▪ 60 Hz = 0x003C0000 ▪ 50.5 Hz = 0x00328000
DSIBitClockFrequency	This is an optional parameter that allows the licensee to configure an exact bit clock frequency in Hz which represents the UI_{inst} timing. ▪ DSI Video mode ▫ This setting overrides any value configured for DSIRefreshRate. The actual DSI refresh rate is recalculated based on the formula: Refresh rate = (#DSILanes x DSIBitClockFrequency)/ (HorizontalTotal x VerticalTotal x DSIBitsPerPixel) ▪ DSI Command mode ▫ This setting allows the bit clock to be independently configured from the actual refresh rate that is, the transfer rate can be faster/slower than the reported panel refresh rate. ▫ If the DSIRefreshRate is specified, the driver reports this value to the OS as the actual panel refresh rate and uses DSIBitClockFrequency to determine the pixel transfer rate to the panel. ▫ If the DSIRefreshRate is 0, the refresh rate will be calculated based on DSIBitClockFrequency using the same equation as in Section 8.1.7.
DSILanes	This integer defines the number of DSI lanes. Valid values are 1 through 4. However, some platforms and APQs may limit the lanes to 1 to 2 lanes.

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSIChannelId</td><td>Integer defining the DSI interface used to communicate with the panel; this must match the desired panel configuration. This setting is based on the QDI_DSIChannelIDType enumeration.
The licensee must select the appropriate engine, either Video or Command mode, depending on the type of DSI interface configured.

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>1</td><td>QDI_DSIChannelID_Video0</td></tr><tr><td>2</td><td>QDI_DSIChannelID_CMD0</td></tr></table>
All other values are invalid or not supported.</td></tr>
<tr><td>DSIVirtualId</td><td>Integer defining the virtual channel ID used when sending command and pixel data; some devices may not support virtual channels. This setting is based on the QDI_DSIDisplayVCType enumeration.

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_DSIDisplayVC_0</td></tr><tr><td>1</td><td>QDI_DSIDisplayVC_1</td></tr></table>
All other values are invalid or not supported.</td></tr>
<tr><td>DSIColorFormat</td><td>Integer defining the color format used to transfer data to the panel; this setting is based on the QDI_DSIColorFormatType enumeration.

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>18</td><td>QDI_DSICOLOR_RGB_565_16BPP</td></tr><tr><td>24</td><td>QDI_DSICOLOR_RGB_666_18BPP</td></tr><tr><td>30</td><td>QDI_DSICOLOR_RGB_666_24BPP</td></tr><tr><td>36</td><td>QDI_DSICOLOR_RGB_888_24BPP</td></tr></table>
All other values are invalid or not supported.</td></tr>
<tr><td>DSIPacketTransferHS</td><td>This Boolean determines how the DMA command packets are sent to the panel. When transferring command data to the panel, the packets can be sent in either High Speed (HS) or Low Power (LP) mode. This setting determines how packets are sent.
The exception to this rule occurs during initialization and termination sequences sent to the panel. In these scenarios, the packets are always in LP mode.
Note: It is recommended that data always be sent in HS mode if the panel supports it.

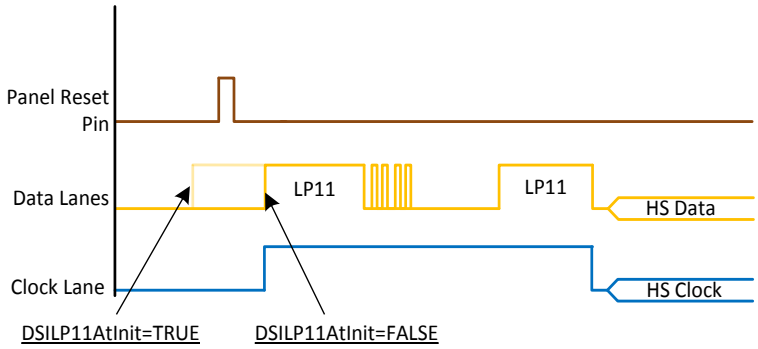
<table><tr><td>Value (Boolean)</td><td>Definition</td></tr><tr><td>FALSE</td><td>Sends DCS commands in LP mode</td></tr><tr><td>TRUE</td><td>Sends DCS commands in HS mode</td></tr></table></td></tr>
</table>

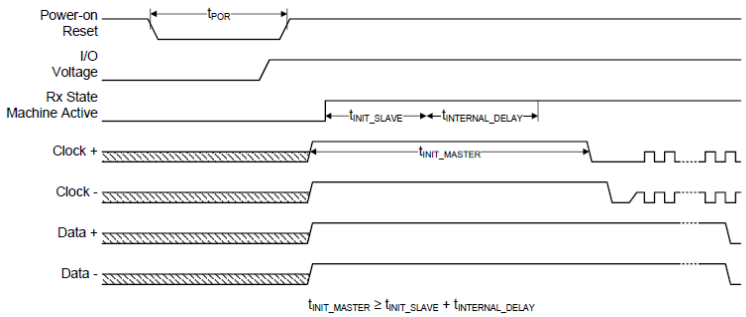
<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSIClockHSForceRequest</td><td>This integer determines the state of the DSI clock lane. Some panels require that the clock lane is always active in order to provide a continual clock signal to the panel.
By default, this should not be used unless the panel requires an always active clock lane.

<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>0</td><td>QDI_DSI_ClockLane_Default
DSI clock lane is in inactive (LP11) state when transfers are not occurring.</td></tr>
<tr><td>1</td><td>QDI_DSI_ClockLane_ForceHS
DSI clock lane is always active regardless of transfers occurring.</td></tr>
<tr><td>2</td><td>QDI_DSI_ClockLane_ForceHS_PostInit
DSI Clock lane is forced to HS after the init sequence has been sent.</td></tr>
</table>
</td></tr>
<tr><td>DSIHostLaneMapping</td><td>This integer defines how the data lanes are mapped to the panel. Depending on the platform, it may be required to map the individual DSI lanes. This configuration is based on the QDI_DSILaneMapType enumeration.
The default mapping should be used unless the platform requires a special lane mapping.

<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>0</td><td>QDI_PANEL_LANE0123MAP</td></tr>
<tr><td>1</td><td>QDI_PANEL_LANE3012MAP</td></tr>
<tr><td>2</td><td>QDI_PANEL_LANE2301MAP</td></tr>
<tr><td>3</td><td>QDI_PANEL_LANE1230MAP</td></tr>
<tr><td>4</td><td>QDI_PANEL_LANE0321MAP</td></tr>
<tr><td>5</td><td>QDI_PANEL_LANE1032MAP</td></tr>
<tr><td>6</td><td>QDI_PANEL_LANE2103MAP</td></tr>
<tr><td>7</td><td>QDI_PANEL_LANE3210MAP</td></tr>
</table>

All other values are invalid or not supported.</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSILP11AtInit</td><td>This Boolean determines the status of the DSI clock and data lanes after hardware initialization. Some panels may require that the DSI clock and lane must enter LP11 prior to powering up and issuing the hardware reset line.
 

<table>
<tr><th>Value (Boolean)</th><th>Definition</th></tr>
<tr><td>FALSE</td><td>DSI data/clock lanes are in LP00 during the power-on reset sequence.</td></tr>
<tr><td>TRUE</td><td>DSI data/clock lanes are in LP11 during the power-on reset sequence.</td></tr>
</table></td></tr>
<tr><td>DSIInitMasterTime</td><td>This integer determines the t_{INIT_MASTER} (as specified in the MIPI Alliance Specification for Display Serial Interface (Version 1.02.00 (June 2010)) before any DSI activity is performed.
 

Minimum time is set to 2 ms. This default value can be extended by overwriting this configuration. Units are given in milliseconds.</td></tr>
<tr><td>DSIEnterULPSPower Down</td><td>This Boolean flag forces the controller to enter ULPS mode prior to powering down (suspend mode). The panel must support ULPS for this feature to be enabled.</td></tr>
<tr><td>DSIDataStrengthLP</td><td>This setting adjusts the drive strength of the data lanes during LP transmission. The driver has already been tuned for the optimal setting, other settings can be overwritten for experimentation. Conformance is not guaranteed if these settings are overwritten. Values range from 0x0 (weakest) to 0xf (strongest).</td></tr>
<tr><td>DSIDataStrengthHS</td><td>This setting adjusts the drive strength of the data lanes during HS transmission. The driver has already been tuned for the optimal setting. The other settings can be overwritten for experimentation. Conformance is not guaranteed if these settings are overwritten. Values range from 0x0 (weakest) to 0xf (strongest).</td></tr>
</table>

Tag	Description
DSIClockStrengthHS	This setting adjusts the drive strength of the clock lanes during HS transmission. The driver has already been tuned for the optimal setting. The other settings can be overwritten for experimentation. Conformance is not guaranteed if these settings are overwritten. Values range from 0x0 (weakest) to 0xf (strongest).
DSIEscapeClockDivisor	The escape clock is used to derive the MIPI DSI timings within the DSI controller. The MIPI DSI specification requires Tlpx time to be a minimum of 50 ns which is a 20 MHz escape clock. The default escape clock configuration is to source from the system CXO clock which is 19.2 MHz. This, however, can be overwritten by adjusting this setting. When DSIEscapeClockDivisor is greater than zero. the escape clock is derived from the DSI PLL frequency (byte clock). The divisor defines how much the PLL frequency is divided by to achieve the escape clock. The exact division value varies by chipset, but it is 1 or 1.5 increments per integer. This field only supports integer values. For example, on some chipsets, a division of 1.5 may use the following configuration: < DSIEscapeClockDivisor>2</ DSIEscapeClockDivisor>
DSIEscapeClockFrequency	When DSIEscapeClockDivisor is used this fileld must be populated with the expected escape clock frequency. This expected clock frequency is used to adjust all of the other DSI D Phy timings. As there is no feedback mechanism for the driver to know the exact escape clock frequency if it is not derived from CXO (19.2 MHz), the user must pass in the exact value in Hz. Incorrectly enabling the escape clock divisor or not properly setting this frequency will result in DSI timings that are out of specification. For a 17 MHz escape clock: < DSIEscapeClockFrequency>17000000</ DSIEscapeClockFrequency>
DSITransferRetryCntr	When sending and receving DCS commands (BTA), the driver will attempt to retry upon failure of the original command. This field allows the OEM to adjust the number of attempts for any read or write command before it gives up and returns an error. The default value is 2 (meaning two total attempts), but this can be disabled or increased based on the requirements.
DSINullpacketInsertionBytes	During pixel data transmission, the controller idles the data lanes (LP00) between pixel tranmissions. Enabling this configuration forces the controller to insert NULL packets to keep the data lanes in HS. The value represents the size of the NULL packets; a value of 0 means disabled.

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSIFlags</td><td>This field is an integer that represents a bitwise mask of DSI configuration options.
The following table lists valid values.

<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>0x0001</td><td>QDI_PANEL_DSI_FLAG_DSI_DCS_FIFO_ONLY
This flag forces the driver to use a FIFO to send DCS packets instead of a DMA buffer. This option reduces the latency for start of packet transmission but limits the maximum packet size that can be sent to 64 bytes total including headers.</td></tr>
<tr><td>0x0002</td><td>QDI_PANEL_DSI_FLAG_DSI_DISABLE_BURST_MODE
This flag disables Burst mode which allows DMA packets to interleave between pixel data transfers. By default the hardware is configured to burst all pixel data together without interruption.
The default configuration is the most optimal and this flag should typically not be needed.</td></tr>
<tr><td>0x0004</td><td>QDI_PANEL_DSI_FLAG_DISABLE_PPS_CMD
This flag disables sending of the PPS packets when MIPI DSC is enabled. When disabled the driver will not automatically send the PPS packets during panel initialization. Some panels do not support PPS packets and this flag should be set for those panels.</td></tr>
<tr><td>All other values</td><td>Reserved</td></tr>
</table>
</td></tr>
</table>

8.1.2 Init sequence timing

To meet the requirements of varying panel hardware, the initialization sequence can be tuned based on particular DSI parameters, shown in Figure 8-1.

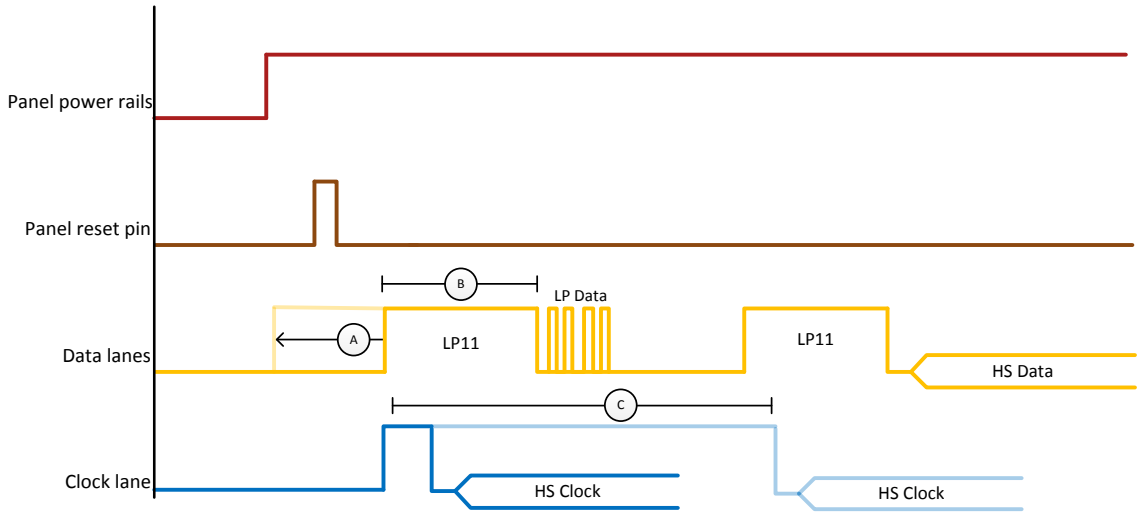


Figure 8-1 Init sequence timing

Init sequence timing depends on the following parameters:

1. DSILP11AtInit – *MIPI Alliance Specification for Display Serial Interface* (Version 1.03.00 (August 2011)) requires the data and clock lanes to be in the LP11 state after the reset pin has been deasserted. In previous revisions of the specification, the lanes were required to be in LP11 before the reset had been asserted. The DSILP11AtInit configuration allows licensees to force the lanes to LP11 before the reset is asserted; the default behavior is to assert LP11 after reset.
2. DSIInitMasterTime – After asserting LP11, the panel IC may require additional delay times to complete initialization before the first DCS command can be recognized. By default the minimum delay is 2 ms. However, the licensee can overwrite this to any value.
3. DSIClockHSForceRequest – Some panel ICs may require the host to drive the clock lanes to HS. Typically, this is done if the panel IC requires an external oscillator. This configuration allows for three options:
   - The clock lane is only transitioned into HS during a HS data transmission (default).
   - The clock lane is always in HS.
   - The clock lane is in HS after the init sequence is complete.

8.1.3 DSI DCS command configurations

During panel initialization and shutdown, the licensee can send custom panel sequences to initialize and shut down the panel. Figure 8-2 shows when these commands are sent.

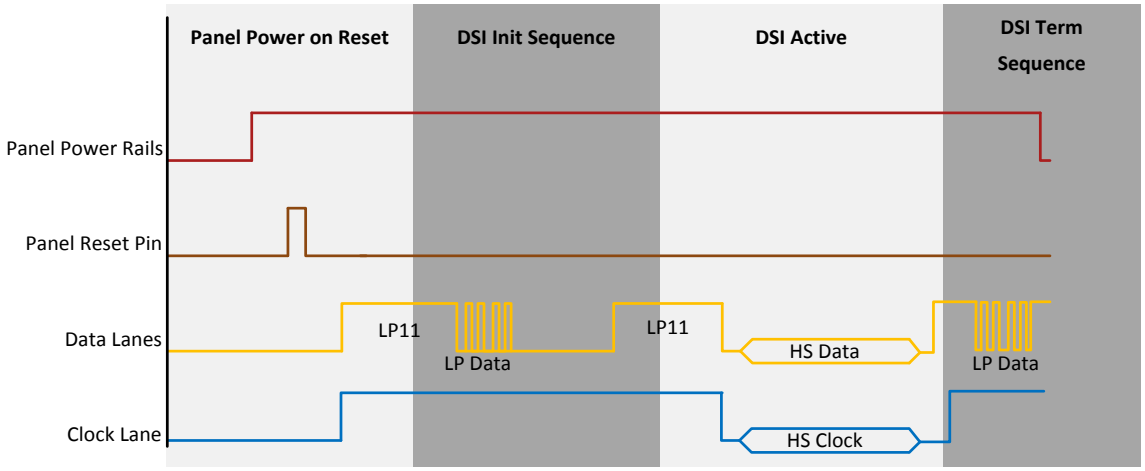


Figure 8-2 DSI DCS command configurations

Table 8-4 shows the DSI DCS command configuration tags.

Table 8-2 DSI DCS command configuration tags

Tag	Description
DSIInitSequence	This is a binary data value that is used to send DCS commands to the panel during panel initialization (display on). One or multiple packets can be sent using this sequence; see the binary data type description of how to build a packet. This data is always sent to the panel in LP mode. For example: `05 11 00 ; DCS command 5h, followed by two payload ; bytes 11h and 00h`
DSITermSequence	This is a binary data value that is used to send DCS commands to the panel during panel termination (display off). One or multiple packets can be sent using this sequence; see the binary data type description of how to build a packet. This data is always sent to the panel in LP mode.

8.1.4 Supported DCS commands

Any valid DCS command and payload can be sent. Table 8-3 provides examples of the valid DCS commands supported. For more details, see the *MIPI Alliance Specification for D-PHY* (Version 1.00.00 (May 2009)).

Table 8-3 Supported DCS commands

Command (hex)	Payload size (bytes)	Definition
03	2	Generic short write, no parameters
13	2	Generic short write, 1 parameter
23	2	Generic short write, 2 parameters
05	2	DCS short write, no parameters
15	2	DCS short write, 1 parameter
29	4	Generic long write

Other commands may not be supported or may be reserved.

8.1.5 Special DCS commands

There are certain reserved commands that can be used and are interpreted by the driver as having a special meaning. These commands are not valid DCS commands but are intercepted by the driver to have a special function as shown in Table 8-4.

Table 8-4 Special DCS commands

Command (hex)	Payload size (bytes)	Definition
FF	1	Stall between commands; payload is used to indicate the number of milliseconds to stall between commands; valid range is 1 to 255. For example: ▪ 05 11 00 – DCS command 5 h, with 11 h as payload ▪ FF 5C – Wait for 92 ms ▪ 05 15 00 – DCS command 5 h, with 15 h as payload
FE	3	Configure MDP to output constant color to clear the display frame buffer for smart panel. The payload is used to indicate the desired constant color in RGB format. Valid range for each color component is 0 to 255. The default color is black if no payload is specified, e.g.: ▪ FE FF FF 00 – Configure MDP to output yellow constant color If this command is used during panel initialization, then constant color will be sent to the panel to clear the random garbage data in the display module buffer. If this command is used during display reset while MDP has already been configured, the last frame in MDP is sent to the panel.

<table>
<tr><th>Command (hex)</th><th>Payload size (bytes)</th><th>Definition</th></tr>
<tr><td>FD</td><td>1</td><td>Broadcast mask used to mask which controllers the command is sent to; this is only valid for dual-DSI applications.

<table>
<tr><th>Mask (hex)</th><th>Definition</th></tr>
<tr><td>0x1</td><td>Command is sent to controller 0; default</td></tr>
<tr><td>0x2</td><td>Command is sent to controller 1</td></tr>
</table>
</td></tr>
</table>

8.1.6 DSI interface timing overrides

The DSI timings are automatically calculated based on the required refresh rate/bit clock frequency. If the panel has specific timing requirements, the adjustment for each timing can be overridden. Table 8-5 lists of DSI interface timing overrides.

Table 8-5 DSI interface timing overrides

Tag	Units	Notes
DSITimingHSZeroOverride	Boolean	See the *MIPI Alliance Specification for D-PHY* (Version 1.00.00 (May 2009)) for details on each timing parameter.
DSITimingHSZeroValue	Integer	
DSITimingHSExitOverride	Boolean	
DSITimingHSExitValue	Integer	
DSITimingHSPrepareOverride	Boolean	
DSITimingHSPrepareValue	Integer	
DSITimingHSTrailOverride	Boolean	
DSITimingHSTrailValue	Integer	
DSITimingHSRequestOverride	Boolean	
DSITimingHSRequestValue	Integer	
DSITimingCLKZeroOverride	Boolean	
DSITimingCLKZeroValue	Integer	
DSITimingCLKTrailOverride	Boolean	
DSITimingCLKTrailValue	Integer	
DSITimingCLKPrepareOverride	Boolean	
DSITimingCLKPrepareValue	Integer	
DSITimingCLKPreOverride	Boolean	
DSITimingCLKPreValue	Integer	
DSITimingCLKPostOverride	Boolean	
DSITimingCLKPostValue	Integer	
DSITimingTASureOverride	Boolean	
DSITimingTASureValue	Integer	
DSITimingTAGoOverride	Boolean	
DSITimingTAGoValue	Integer	
DSITimingTAGetOverride	Boolean	
DSITimingTAGetValue	Integer	

8.1.7 DSI Video mode configuration

Table 8-6 lists configurations specific to DSI Video mode.

Table 8-6 DSI Video mode configuration

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSITrafficMode</td><td>Integer defining the DSI Traffic mode for operation; this value is based on the QDI_DSITrafficModeType enumeration.

<table><tr><td>Value (decimal)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_DSIVideoTrafficMode_NonBurst_VSPulse</td></tr><tr><td>1</td><td>QDI_DSIVideoTrafficMode_NonBurst_VSEvent</td></tr><tr><td>2</td><td>QDI_DSIVideoTrafficMode_Burst</td></tr></table>
All other values are invalid or not supported.</td></tr>
<tr><td>DSIHsaHseAfterVsVe</td><td>This Boolean determines whether the hardware should send horizontal sync pulses during the vertical blanking period.

<table><tr><td>Value (Boolean)</td><td>Definition</td></tr><tr><td>FALSE</td><td>Do not send a horizontal start/end packet after the vertical sync/end</td></tr><tr><td>TRUE</td><td>Send a horizontal start/end packet after the vertical sync/end</td></tr></table></td></tr>
<tr><td>DSILowPowerModeInHFP</td><td>This Boolean determines the DSI lane state during a Horizontal front porch (HFP) blanking period.
Note: It is recommended that this value be set to TRUE for LP mode if the panel supports this feature. Forcing the lanes to HS during this period will consume additional power and prevent DCS commands from being sent during this period.

<table><tr><td>Value (Boolean)</td><td>Definition</td></tr><tr><td>FALSE</td><td>The DSI clock and lanes should stay in HS mode during the HFP blanking period. The hardware will send blanking packets during this period.</td></tr><tr><td>TRUE</td><td>The DSI clock and lanes should enter LP mode during the HFP blanking period.</td></tr></table></td></tr>
<tr><td>DSILowPowerModeInHBP</td><td>This Boolean determines the DSI lane state during the Horizontal blanking period (HBP).
Note: It is recommended that this value be set to TRUE for LP mode if the panel supports this feature. Forcing the lanes to HS during this period will consume additional power and prevent DCS commands from being sent during this period.

<table><tr><td>Value (Boolean)</td><td>Definition</td></tr><tr><td>FALSE</td><td>The DSI clock and lanes should stay in HS mode during the HBP. The hardware sends blanking packets during this period.</td></tr><tr><td>TRUE</td><td>The DSI clock and lanes should enter LP mode during the HBP.</td></tr></table></td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSILowPowerModeInHSA</td><td>This Boolean determines the DSI lane state during Horizontal sync active (HSA).
Note: It is recommended that this value be set to TRUE for LP mode if the panel supports this feature. Forcing the lanes to HS during this period will consume additional power and prevent DCS commands from being sent during this period.

<table><tr><th>Value (Boolean)</th><th>Definition</th></tr><tr><td>FALSE</td><td>The DSI clock and lanes should stay in HS mode during the Horizontal Sync Period (HSP). The hardware will send blanking packets during this period.</td></tr><tr><td>TRUE</td><td>The DSI clock and lanes should enter LP mode during the HSP.</td></tr></table></td></tr>
<tr><td>DSILowPowerModeInBLLPEOF</td><td>This Boolean determines the DSI lane state during the last line of the Blanking low power period (BLLP).
Note: It is recommended that this value be set to TRUE for LP mode if the panel supports this feature. Forcing the lanes to HS during this period will consume additional power and prevent DCS commands from being sent during this period.

<table><tr><th>Value (Boolean)</th><th>Definition</th></tr><tr><td>FALSE</td><td>The DSI clock and lanes should stay in HS mode during the last line of the BLLP. The hardware will send blanking packets during this period.</td></tr><tr><td>TRUE</td><td>The DSI clock and lanes should enter LP mode during the last line of the BLLP.</td></tr></table></td></tr>
<tr><td>DSILowPowerModeInBLLP</td><td>This Boolean determines the DSI lane state during BLLP.
Note: It is recommended that this value be set to TRUE for LP mode if the panel supports this feature. Forcing the lanes to HS during this period will consume additional power and prevent DCS commands from being sent during this period.

<table><tr><th>Value (Boolean)</th><th>Definition</th></tr><tr><td>FALSE</td><td>The DSI clock and lanes should stay in HS mode during the BLLP. The hardware will send blanking packets during this period.</td></tr><tr><td>TRUE</td><td>The DSI clock and lanes should enter LP mode during the BLLP.</td></tr></table></td></tr>
<tr><td>DSIDmaDelayAfterVsync</td><td>This integer is used in the DSI DMA read/write scheduling in Video mode. The software schedules the DSI DMA operations when the DSI controller is busy transferring pixel data, and the DMA packets are sent to the panel when pixel transfer is done by the hardware. This integer decides the number of milliseconds after vsync that the software can trigger the DMA packets transfer.</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSIPixelXferTiming (replacement for DSIForceCmdInVideoHS)</td><td>Integer defining the scenarios to enable/disable the video HS pixel data transfer for initialization and termination sequence. This value is based on the DSIPixelXferTiming enumeration.
Any combination of enumerations can be combined to meet the init and termination sequence requirements. Disabling and enabling bits cannot be combined together.

<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>0</td><td>QDI_DSI_PIXEL_TX_DISABLED_DURING_INIT_TERM
High-speed video data is disabled during the init sequence.</td></tr>
<tr><td>1</td><td>QDI_DSI_PIXEL_TX_ENABLED_DURING_INIT
High-speed video data is enabled during the init sequence.</td></tr>
<tr><td>2</td><td>QDI_DSI_PIXEL_TX_ENABLED_DURING_TERM
High-speed video data is disabled during the termination sequence.</td></tr>
<tr><td>3</td><td>QDI_DSI_PIXEL_TX_ENABLED_DURING_INIT_TERM
High-speed video data is enabled during the termination sequence.</td></tr>
</table>
</td></tr>
<tr><td>DSIForceCmdInVideoHS (to be deprecated)</td><td>Video HS pixel data will be started before the initialization sequence from the OEM/ACPI has been sent to the panel. Video HS pixel data transfer will be active even during the termination sequence from the OEM/ACPI.</td></tr>
<tr><td>DSIDynamicRefreshRates</td><td>The refresh rates supported by the panel other than the default refresh rate; this is a list of refresh rates in Q16.16 format.
For example:
<code><DSIDynamicRefreshRates>0x3C0000
0x320000</DSIDynamicRefreshRates></code>
Adds 48 Hz and 50 Hz respectively to the list of supported refresh rates.
The driver may prune refresh rates from this list based on the capabilities of the display hardware. Not all interfaces and panel configurations support dynamic refresh changes.
Note: The list of frequencies can be in any order, but the default refresh rate must appear first.</td></tr>
<tr><td>DSIDynamicBlankingRefreshRateList</td><td>Refresh rates supported by the panel by adjusting the panel blanking; it is the licensee’s responsibility to calibrate and configure the blanking configuration. This is a list of refresh rates in Q16.16 format.
For example:
<code><DSIDynamicBlankingRefreshRateList>
0x3C0000
0x300000
0x320000
</DSIDynamicBlankingRefreshRateList></code>
Adds 48 Hz and 50 Hz respectively to the list of supported refresh rates that can be achieved by adjusting the panel blanking.
Note: The list of frequencies can be in any order, but the default refresh rate must appear first.</td></tr>
</table>

Tag	Description

DSIDynamicVFrontPorchList	Vertical front porch adjustment is part of the dynamic blanking configuration required to achieve a particular refresh rate.
	For example:
	`<DSIDynamicVFrontPorchList>`
	`0x07`
	`0x09`
	`0x05`
	`</DSIDynamicVFrontPorchList>`
	These vertical front porch values correspond one to one with the refresh rates in DSIDynamicBlankingRefreshRateList and must be in the same order.

DSIDynamicVBackPorchList	Vertical back porch adjustment is part of the dynamic blanking configuration required to achieve a particular refresh rate.
	For example:
	`<DSIDynamicVBackPorchList>`
	`0x06`
	`0x07`
	`0x05`
	`</DSIDynamicVBackPorchList>`
	These vertical back porch values correspond one to one with the refresh rates in DSIDynamicBlankingRefreshRateList and must be in the same order.

DSIDynamicVSyncPulseList	Vsync pulse adjustment is part of the dynamic blanking configuration required to achieve a particular refresh rate.
	For example:
	`<DSIDynamicVSyncPulseList>`
	`0x03`
	`0x04`
	`0x02`
	`<DSIDynamicVSyncPulseList>`
	These vsync pulse values correspond one to one with the refresh rates in DSIDynamicBlankingRefreshRateList and must be in the same order.

DSIDynamicHFrontPorchList	Horizontal front porch adjustment is part of the dynamic blanking configuration required to achieve a particular refresh rate.
	For example:
	`<DSIDynamicHBackPorchList>`
	`0x03`
	`0x04`
	`0x03`
	`</DSIDynamicHBackPorchList>`
	These horizontal front porch values correspond one to one with the refresh rates in DSIDynamicBlankingRefreshRateList and must be in the same order.

Tag	Description
DSIDynamicHBackPorchList	Horizontal back porch adjustment is part of the dynamic blanking configuration required to achieve a particular refresh rate. For example: <DSIDynamicHBackPorchList> 0x03 0x03 0x02 </DSIDynamicHBackPorchList> These horizontal back porch values correspond one to one with the refresh rates in DSIDynamicBlankingRefreshRateList and must be in the same order.
DSIDynamicHSyncPulseList	Hsync pulse adjustment is part of the dynamic blanking configuration required to achieve a particular refresh rate. For example: <DSIDynamicHSyncPulseList> 0x03 0x03 0x01 <DSIDynamicHSyncPulseList> These hsync pulse values correspond one to one with the refresh rates in DSIDynamicBlankingRefreshRateList and must be in the same order.

8.1.8 DSI Command mode configuration

Table 8-7 lists configurations specific to DSI Command mode.

Table 8-7 DSI Command mode configuration

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSICMDSwapInterface</td><td>Reserved Boolean, must be FALSE or leave as undefined</td></tr>
<tr><td>DSICMDUsingTrigger</td><td>Reserved Boolean, must be FALSE or leave as undefined</td></tr>
<tr><td>DSITECheckEnable</td><td>This Boolean enables the Tearing effect (TE) check hardware block within MDP.
<table>
<tr><th>Value (Boolean)</th><th>Definition</th></tr>
<tr><td>FALSE</td><td>Tear check hardware block is disabled</td></tr>
<tr><td>TRUE</td><td>Tear check hardware block is enabled</td></tr>
</table></td></tr>
<tr><td>DSITEUsingDedicatedTEPin</td><td>This Boolean enables the TE via an external GPIO or via an embedded-TE signal.
<table>
<tr><th>Value (Boolean)</th><th>Definition</th></tr>
<tr><td>FALSE</td><td>TE uses an internal TE signal, embedded TE</td></tr>
<tr><td>TRUE</td><td>TE uses a dedicated GPIO, external TE signal</td></tr>
</table></td></tr>
<tr><td>DSITEvSyncStartPos</td><td>Line number from which the TE kickoff condition is evaluated for vertical synchronization</td></tr>
<tr><td>DSITEvSyncRdPtrIrqLine</td><td>This integer configures the scanline number that the DSI pixel transfer will start on. This should be retained at the default value of 0. Adjusting this number to a higher value results in delaying the start of the pixel transfer.</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSITEvSyncContinueLines</td><td>This integer represents the difference in the number of lines between the estimated read and write pointers to allow updating of all the lines except the first line of the frame. This threshold is maintained only when the Tear Check block is enabled.</td></tr>
<tr><td>DSITEvSyncStartLineDivisor</td><td>This integer represents the fraction of the total height from the top, within which the read pointer should fall, so as to allow the first line update of the frame. This threshold is maintained only when Tear Check block is enabled.
For example, if this value is set to 4 and height is 100 lines, then only when the internal read pointer falls between 0 to 25 (=100/4) lines is the first line of the frame updated.
A valid range is from 2 to 100 height. Default value is 4 (25%).
See Section 8.1.9 for more details and examples.</td></tr>
<tr><td>DSITEPercentVariance</td><td>This integer is the percent value by which the refresh rate would be reduced so that the internal estimation of time taken by the panel to read a line is increased. Fine-tuning of this value ensures that external TE comes before the internal timer expires.
A valid range is 0 to 100, representing 0% to 100%, default value is 0 (0%)
See Section 8.1.9 for more details and examples.</td></tr>
<tr><td>DSIEnableAutoRefresh</td><td>This Boolean enables the auto-refresh hardware (if available). The auto-refresh hardware is used to self-trigger the DSI Command mode engine without software interaction.
<table>
<tr><th>Value (Boolean)</th><th>Definition</th></tr>
<tr><td>FALSE</td><td>Auto-refresh hardware is disabled</td></tr>
<tr><td>TRUE</td><td>Auto-refresh hardware is enabled; hardware will automatically trigger periodic updates based on an internal counter</td></tr>
</table></td></tr>
<tr><td>DSIAutoRefreshFrameNumDiv</td><td>This integer defines the interval at which the auto-refresh hardware is triggered. Values range from 1 to 4095.
The rate at which the hardware internal counter triggers the DSI command mode update is based on the formula:
Auto-Refresh Rate = DSIRefreshRate/DSIAutoRefreshFrameNumDiv</td></tr>
</table>

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DSITEvSyncSelect</td><td>This integer defines the external vsync source. Default value is 0, which indicates it is from the dedicated GPIO pin.
<table>
<tr><th>Value (Boolean)</th><th>Definition</th></tr>
<tr><td>0</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_VSYNC_0_GPIO</td></tr>
<tr><td>1</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_VSYNC_1_GPIO</td></tr>
<tr><td>2</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_VSYNC_2_GPIO</td></tr>
<tr><td>3</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_TE_INTF_0</td></tr>
<tr><td>4</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_TE_INTF_1</td></tr>
<tr><td>5</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_TE_INTF_2</td></tr>
<tr><td>6</td><td>HAL_MDP_PINGPONG_VSYNC_SELECT_TYPE_TE_INTF_3</td></tr>
</table></td></tr>
<tr><td>DSICmdModeIdleTime</td><td>This integer defines the number of idle cycles to insert in between pixel data packets. If this configuration is used to enforce a stop state in between pixel data packets in command demo, the delay must take into consideration the time to enter and exit the stop state, which includes the following timings:
<code>t_clk_prepare + t_clk_zero + t_clk_pre + Tlpx + t_hs_prepare + t_hs_zero + t_hs_sync +t_hs_exit + t_hs_trail + t_clk_post + t_clk_trail</code>
Valid range is 0 to 0x1FF clock cycles and the default value is 0</td></tr>
<tr><td>DSIDisableEoTAfterHSXfer</td><td>This Boolean disables the EoT packet after a HS data burst. The default value is FALSE.
<table>
<tr><th>Value (Boolean)</th><th>Definition</th></tr>
<tr><td>FALSE</td><td>EoT packet is appended after a HS burst</td></tr>
<tr><td>TRUE</td><td>EoT packet is not sent</td></tr>
</table></td></tr>
<tr><td>DSIBitClkScalePercent</td><td>In command mode operation, the data transfer rate is typically tuned to meet the requested refresh rate (DSIRefreshRate). In most cases, this is 16.6 ms (60 fps).
Some margin must be added to this transfer rate to account for system variances and leave room to transfer DCS commands. The DSIBitClkScalePercent increases the bitrate by a specific percentage. By default the variance is 5% but can be decreased or increased to the required amount. The valid rate is from 1% to 100%. However, there may be clock limitations that prevent a dramatic increase.</td></tr>
</table>

8.1.9 DSI Command mode TE synchronization

Figure 8-3 shows the DSI Command mode TE synchronization.

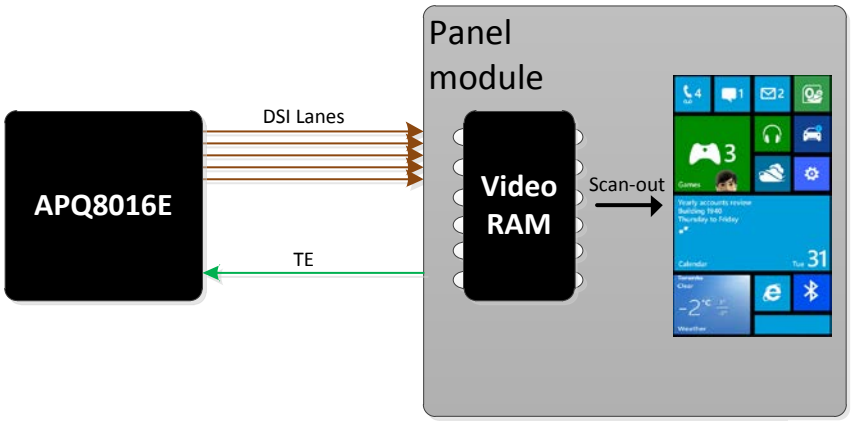


Figure 8-3 DSI Command mode TE synchronization

MIPI DSI Command mode data transfer synchronization requires the TE signal from the panel module. The TE signal is used to synchronize the internal data transfer to ensure a safe region where an update can occur without causing tearing on the display.

Figure 8-4 shows the DSI data transmission with respect to the panel scanline.

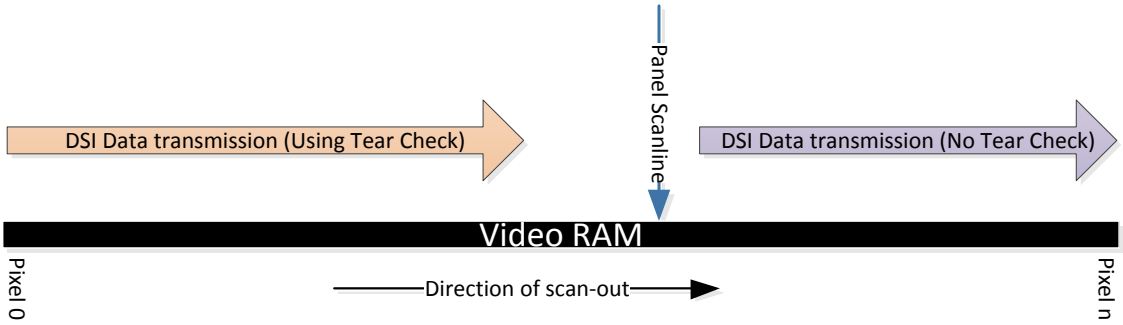


Figure 8-4 DSI data transmission with respect to panel scanline

Two basic methods handle this synchronization:

- Data transfer behind the scanline (not recommended) – This method requires the DSI link rate (bitclock) to be configured to the panels refresh rate or faster. The tear check block must be enabled in order to throttle the data transfer to ensure the write never crosses the current scanline. The data transfer ahead of the scanline is recommended because it provides a much simpler solution and increased efficiency on the bus.
- Data transfer ahead of the scanline – This method requires the DSI link rate (bitclock) to be configured equal to the panel's refresh rate or faster. The teach check block must be disabled in this configuration. The increased link rate will guarantee that the scanline never catches up to the data transfer.

8.1.10 Tear check threshold parameters

DSITEPercentVariance

The DSITEPercentVariance represents the percent value by which refresh rate is to be reduced for the internal timer to ensure that the external TE comes before the internal timer expires. This can be used for fine-tuning the TE under the conditions when the frequency of the external TE varies.

8.1.11 Dual DSI configuration

NOTE: Dual DSI only applies to chipsets with multiple DSI ports.

DSIControllerMapping

The DSIControllerMapping field configures the mapping from the logical DSI port to the physical DSI port. In a single DSI 4-lane configuration, this allows for mapping of the primary DSI to the secondary DSI port. For example:

```
<DSIControllerMapping>0x1</ DSIControllerMapping>
```

Without this configuration, the default configuration maps the primary display to the primary DSI port.

For dual DSI configurations, the mapping informs the driver that two DSI ports are needed and how those ports should be mapped. The ID order applied in the mapping tag indicates what DSI port should assigned to the primary and slave DSI ports.

For example, a typical 8-lane dual DSI solution should map the master to DSI port 0 and the slave to DSI port 1.

```
<DSIControllerMapping>0x0 0x1</ DSIControllerMapping>
```

DSISlaveControllerSkewLines

In dual DSI configurations, the master and slave ports will send their pixel stream at the same time. For a specific configuration, a skew may be needed. This configuration allows the controllers to skew the transfers.

The field represents how many lines the slave should be skewed from the master.

To skew (delay) the slave tramission by 2 lines, the following configuration may be used:

```
<DSISlaveControllerSkewLines>2</DSISlaveControllerSkewLines>
```

To skew (delay) the master tramission by 2 lines, a negative value may be used:

```
<DSISlaveControllerSkewLines>-2</DSISlaveControllerSkewLines>
```

8.2 DSI Command mode configuration

8.2.1 Scenario A – Free run transfer (no tear check block)

Panel transfer should start at the start of the TE. It is desired to run the data transfer faster than the TE, within 14 ms, to guarantee that the update is completed ahead of the TE. No tear check block

is required in such a configuration. Tearing was visible if the transfer was started at line 1, so the transfer was adjusted to start on line 760.

Configuration	Value
Mode	Command code
Lanes	4
Resolution	768x1280
Bit depth	24 bpp
Refresh rate	60 Hz
Bitclock rate	▪ 354 MHz – for 16.6 ms transfer ▪ 419 MHz – for 14 ms transfer

ACPI configuration

```
<DSIRefreshRate units='integer Q16.16'>0x3C0000</DSIRefreshRate>
<DSIBitClockFrequency>419991600</DSIBitClockFrequency>
<DSITECheckEnable units='Bool'>False</DSITECheckEnable>
<DSITEUsingDedicatedTEPin units='Bool'>True</DSITEUsingDedicatedTEPin>
<DSITEvSyncRdPtrIrqLine>760</DSITEvSyncRdPtrIrqLine>
```

8.2.2 Scenario B – Using tear check block (transfer throttling)

Panel transfer uses the tear check block to throttle the transfer. The bitclock is calculated by the driver and is designed to complete the transfer in 16.6 ms (60 Hz).

The internal counter runs 10% slower (percent variance) than 60 Hz. The transfer must fall within the first quarter of the panel height (start line divisor) and the transfer can be started after the TE has reached line 2. The transfer should be 3 lines behind the TE (sync continue lines).

Configuration	Value
Mode	Command mode
Lanes	4
Resolution	768x1280
Bit depth	24 bpp
Refresh rate	60 Hz

ACPI configuration

```
<DSITECheckEnable units='Bool'>True</DSITECheckEnable>
<DSITEvSyncStartPos units='int'>2</DSITEvSyncStartPos>
<DSITEvSyncContinueLines units='int'>3</DSITEvSyncContinueLines>
<DSITEvSyncStartLineDivisor units='int'>4</DSITEvSyncStartLineDivisor>
<DSITEPercentVariance units='integer Q16.16'>0xa0000</DSITEPercentVariance>
```

9 Backlight configuration

One of the responsibilities of the Graphics KMD is to communicate with the backlight hardware to set a specific backlight level. Depending on the hardware interface used to control the backlight, various configurations are required to support backlight adjustment. Figure 9-1 shows how backlight is controlled in the WDDM model.

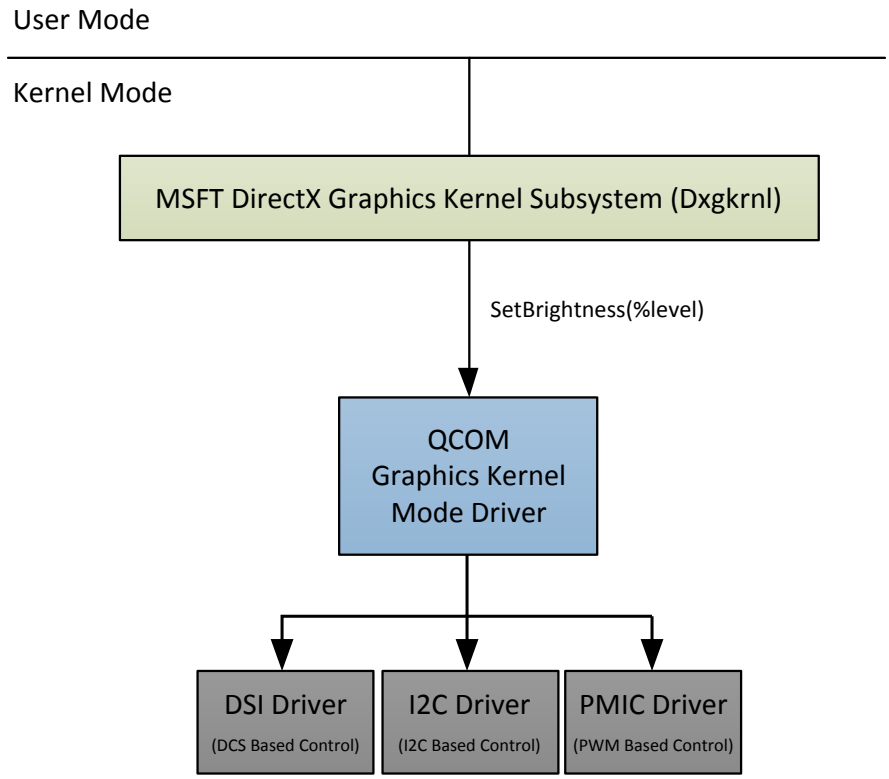


Figure 9-1 Backlight control under WDDM model

9.1 Common backlight configuration parameters

Table 9-1 shows common backlight configuration parameters.

Table 9-1 Common backlight configuration parameters

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>BacklightType</td><td>This integer is used to define the interface used to control the backlight on the platform. This value is based on the QDI_Panel_BacklightType enumeration.

<table><tr><td>Value (integer)</td><td>Definition</td></tr><tr><td>0</td><td>QDI_PANEL_BACKLIGHTTYPE_NONE</td></tr><tr><td>1</td><td>QDI_PANEL_BACKLIGHTTYPE_PMIC</td></tr><tr><td>2</td><td>QDI_PANEL_BACKLIGHTTYPE_I2C</td></tr><tr><td>3</td><td>QDI_PANEL_BACKLIGHTTYPE_DSI</td></tr><tr><td>4</td><td>QDI_PANEL_BACKLIGHTTYPE_ACPI</td></tr></table>
All other values are invalid or unsupported.</td></tr>
<tr><td>BacklightSteps</td><td>This integer defines the number of discrete backlight levels that are supported on this platform. This information is reported back up to the OS and is used to control the granularity in the UI controls for backlight. Valid values are 1 to 100; examples are:
▪ 1% backlight granularity = 100 steps
▪ 10% backlight granularity = 10 steps</td></tr>
<tr><td>BacklightDefault</td><td>This field is reserved and should not be used.</td></tr>
<tr><td>BacklightLowPower</td><td>This field is reserved and should not be used.</td></tr>
</table>

9.2 PWM (PMIC)-based backlight control

If the platform uses a PWM signal generated from the PMIC to control the backlight level, the fields listed in Table 9-2 are used to control the PMIC.

Table 9-2 PWM (PMIC)-based backlight control

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>BacklightPmicModel</td><td>This integer is used to identify the QTI PMIC used to generate the PWM signal on the platform. Currently, only QTI PMICs are supported. The values are based on the QDI_PMICDeviceIdType enumeration.

<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>1</td><td>QDI_PMIC_DEVICEID_8921</td></tr>
<tr><td>2</td><td>QDI_PMIC_DEVICEID_8038</td></tr>
<tr><td>3</td><td>QDI_PMIC_DEVICEID_8941</td></tr>
<tr><td>4</td><td>QDI_PMIC_DEVICEID_8841</td></tr>
<tr><td>5</td><td>QDI_PMIC_DEVICEID_8026</td></tr>
<tr><td>6</td><td>QDI_PMIC_DEVICEID_8926</td></tr>
<tr><td>7</td><td>QDI_PMIC_DEVICEID_8084</td></tr>
<tr><td>8</td><td>QDI_PMIC_DEVICEID_8962</td></tr>
<tr><td>9</td><td>QDI_PMIC_DEVICEID_8994</td></tr>
<tr><td>10</td><td>QDI_PMIC_DEVICEID_8994I</td></tr>
<tr><td>11</td><td>QDI_PMIC_DEVICEID_8916</td></tr>
<tr><td>12</td><td>QDI_PMIC_DEVICEID_8110</td></tr>
<tr><td>13</td><td>QDI_PMIC_DEVICEID_8909</td></tr>
</table>

All other values are invalid or not supported.</td></tr>
<tr><td>BacklightPMICNum</td><td>In systems that use more than one PMIC, this specifies the PMIC that is being controlled; all settings apply to only that PMIC.</td></tr>
<tr><td>BacklightPmicControlType</td><td>This field is used to specify the type of PMIC control interface used to support backlight control.
The values are based on the QDI_PmicModuleControlType enumeration.

<table>
<tr><th>Value (decimal)</th><th>Definition</th></tr>
<tr><td>1</td><td>QDI_PMIC_MODULE_CONTROLTYPE_LPG</td></tr>
<tr><td>2</td><td>QDI_PMIC_MODULE_CONTROLTYPE_WLED</td></tr>
</table>

All other values are invalid or not supported.</td></tr>
</table>

Tag	Description
BacklightPMICBankSelect	This integer is used to specify the PMIC bank used for LPG-based backlight control or the strings to use in WLED type control. The interpretation of this field is based on the value set in BacklightPmicControlType. ▪ LPG (PWM)-based control – This number represents the PMIC LPG bank number. ▪ See the PMIC documentation for bank number to GPIO mapping. ▪ WLED-based control – This number represents the LED strings to be controlled. ▫ Bit 0 – String 1 ▫ Bit 2 – String 2 ▫ Bit 2 – String 3 ▫ And so on. Multiple strings can be controlled by combining bits, e.g., a value of 0x7 = String 0 | String 1 | String 2.
BacklightPMICPWMFrequency	This integer is used to determine the PWM frequency in Hz for controlling the backlight. Some PMICs support a range of PWM frequencies, and check the backlight specification on the range of valid PWM frequencies. If the exact frequency is not available, the closest frequency will be selected, e.g.: ▪ 2 kHz PWM frequency ▪ < BacklightPMICPWMFrequency >2000</BacklightPMICPWM Frequency> This value configures either the WLED or PWM frequency depending on the BacklightPmicControlType.
BacklightPmicAdvancedConfig	This configuration enables all advanced configuration parameters. Depending on the PMIC there are several advanced settings; for more information, see the Advanced WLED configuration and/or PMIC documentation.

9.3 Sample backlight configurations

9.3.1 WLED configuration (1 WLED string)

```
<BacklightType units='QDI_Panel_BacklightType'>1</BacklightType>
<BacklightPmicModel units='QDI_PMICDeviceIdType'>5</BacklightPmicModel>
<BacklightPmicControlType
units='QDI_PmicModuleControlType'>1</BacklightPmicControlType>
<BacklightPmicNum units='int'>0</BacklightPmicNum>
<BacklightPmicBankSelect units='int'>1</BacklightPmicBankSelect>
<BacklightPmicPWMFrequency units='kHz'>800000</BacklightPmicPWMFrequency>
<BacklightSteps units='Percentage'>100</BacklightSteps>
<BacklightDefault units='Percentage'>80</BacklightDefault>
<BacklightLowPower units='Percentage'>40</BacklightLowPower>
<BacklightPmicAdvancedConfig
units='Bool'>True</BacklightPmicAdvancedConfig>
<BacklightPmicWledInternalModResolution
units='int'>0</BacklightPmicWledInternalModResolution>
```

```
<BacklightPmicWledModulationClkSel
units='int'>3</BacklightPmicWledModulationClkSel>
<BacklightPmicWledDimmingMethod
units='int'>0</BacklightPmicWledDimmingMethod>
<BacklightPmicWledOvp units='int'>0</BacklightPmicWledOvp>
<BacklightPmicWledIlim units='int'>5</BacklightPmicWledIlim>
<BacklightPmicWledFeedbackCtrl
units='int'>0</BacklightPmicWledFeedbackCtrl>
<BacklightPmicWlepLoopCompRes units='int'>3</BacklightPmicWlepLoopCompRes>
<BacklightPmicWledVrefControl units='int'>2</BacklightPmicWledVrefControl>
<BacklightPmicWledFullScaleCurrent
units='mA'>25</BacklightPmicWledFullScaleCurrent>
<BacklightPmicWledModulatorSrcSel
units='int'>0</BacklightPmicWledModulatorSrcSel>
<BacklightPmicWledCabcEnable
units='Bool'>False</BacklightPmicWledCabcEnable>
```

9.3.2 LPG configuration

```
<BacklightType units='QDI_Panel_BacklightType'>1</BacklightType>
<BacklightPmicModel units='QDI_PMICDeviceIdType'>3</BacklightPmicModel>
<BacklightPmicControlType
units='QDI_PmicModuleControlType'>1</BacklightPmicControlType>
<BacklightPMICNum units='int'>0</BacklightPMICNum>
<BacklightPMICBankSelect units='int'>7</BacklightPMICBankSelect>
<BacklightPMICPWMFrequency units='kHz'>800000</BacklightPMICPWMFrequency>
<BacklightSteps units='Percentage'>100</BacklightSteps>
<BacklightDefault units='Percentage'>80</BacklightDefault>
<BacklightLowPower units='Percentage'>40</BacklightLowPower>
```

9.4 I2C-based backlight control

If the platform uses I2C to control the backlight level, the BLCP ACPI method is used for controlling the commands sent to the backlight.

9.5 DSI DCS-based backlight control

If the platform uses a DSI DCS command to control the backlight level, the BLCP ACPI method is used for controlling the commands sent to the backlight.

9.6 ACPI-based backlight control

If the licensee wishes to control backlight directly from ACPI, this method should be selected. The BLCP ACPI method is used directly to control the backlight on the platform. The licensee would be responsible for handling the incoming request within the ACPI BLCP method. An example configuration is using the ACPI method to directly modify registers to adjust the backlight.

9.7 CABL configuration

This section of configuration determines how the CABL algorithm behaves. CABL helps reduce backlight level. However, it retains the apparent brightness of the screen by manipulating the pixel values. This saves power.

The CABL algorithm is only used for the primary display.

Supplying these tags requires deep understanding on CABL. Documentation on CABL is outside the scope of this document. The tags in Table 9-3 are optional. Acceptable default values are used when not supplied.

Table 9-3 CABL configuration

Tag	Description
CABLMinUserLevel	This defines the lowest level that can be configured as original backlight level (before CABL algorithm processing) and still keep CABL active; any level lower than this will deactivate CABL. The range is 1 to 255, where 255 is 100% backlight level. This item is important so that CABL can be configured when to deactivate itself (basically turns off) to save processing power when CABL savings are not significant. The default value is 0, meaning algorithm default 75 (30%).
CABLMinBacklightLevel	This defines the lowest backlight level CABL algorithm that will output even when other calculations give a lower level. The range is 1 to 255, where 255 is 100% backlight level. This item limits CABL so that it does not go too low, which can cause quality degradation. This number should be lower than CABLMinUserLevel. The default value is 0, meaning algorithm default of 30 (12%).
CABLFilterThreshold	This configures the maximum backlight change that can happen on each CABL iteration during regular CABL operation. This prevents huge backlight changes that might be noticeable to the user. The range is 1 to 255. The default value is 0, meaning algorithm default 10 (4%).

9.8 Advanced backlight configuration

Table 9-4 defines the PMIC WLED parameters.

Table 9-4 Backlight configuration

Tag	Description
BacklightPmicAdvancedConfig	Set the advanced PMIC configuration for backlight control
BacklightPmicWledInternalModResolution	Internal digital modulator configured
BacklightPmicWledModulationClkSel	Digital modulator input clock frequency control
BacklightPmicWledDimmingMethod	Modulator dimming mode control
BacklightPmicWledOvp	Control for the over voltage protection threshold
BacklightPmicWledIlim	Current limit after soft start is complete
BacklightPmicWledFeedbackCtrl	Forces selection of LED output as feedback node for the boost
BacklightPmicWlepLoopCompRes	Control to select the compensation resistor
BacklightPmicWledVrefControl	Reference voltage for boost feedback
BacklightPmicWledFullScaleCurrent	Select current sink for the LED
BacklightPmicWledModulatorSrcSel	▪ 1 – External PWM ▪ 0 – Internal digital modulator
BacklightPmicWledCabcEnable	Enable content adaptive backlight control for current sinks
BacklightPmicOLEDWledAvddVoltage	Configures the AVDD voltage for the WLED module in AMOLED mode

10 ESD detection and recovery configuration

ESD Detection and Recovery is configured using the tags given in Table 10-1 and Table 10-2.

Table 10-1 ESD Detection and Recovery tags

Tag	Description
ESDDetectionTime	This integer (in milliseconds) is used to schedule how fast the ESD detection and recovery will be running. The default value in ACPI is 0, which means the ESD Detection and Recovery feature is disabled.
ESDDetectionFailureRetry	This integer is used as a retry count if ESD detection returns failure before recovery is called. The default value is 5 if this parameter is omitted.
DSIStatusSequence	This is a binary data value that is used to send DCS read commands to the panel to decide panel status. Syntax of each command is: `<data type> <DCS command> <Expected Result>` Currently, only short DCS command with no parameter is supported. For example: 06 0a 1c: Short DCS command to get panel power mode and expected panel response is 0x1C.

Table 10-2 Display recovery threshold

Tag	Description
DisplayRecoveryThreshold	This configuration will set up the threshold for recovery configuration. The value set in the field will determine the threshold beyond which display recovery action will be triggered. If the value is set to 0, the recovery feature is disabled. Unit is in number of vsyncs. It is recommended that this should be less than 120 (2 sec) to avoid a TDR from resetting the display before it can self-recover.

11 Platform-specific GPIO configuration

A majority of the power rails and GPIOs (both TLMM (APQ) and PMIC) are configured by the PEP component of Windows. However, panels often require specific timing and control of GPIOs during boot. Table 11-1 gives details on additional configuration operations available.

Figure 11-1 shows panel power-on reset.

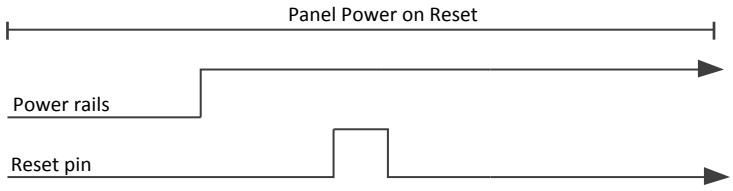
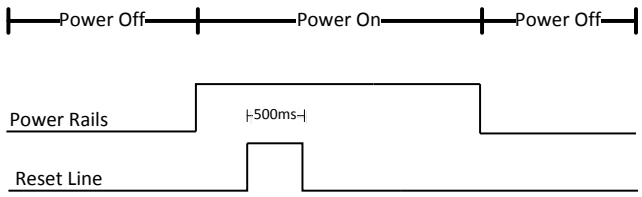
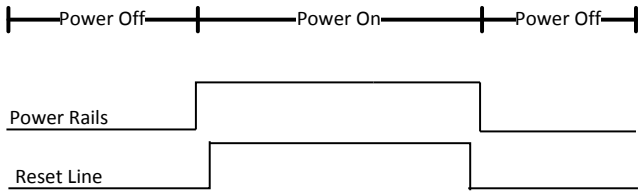


Figure 11-1 Panel power-on reset

During the panel power-on reset sequence, additional GPIOs can be configured. All of these configurations are optional. If PEP is sufficient to control these resources, these configurations must be disabled.

Table 11-1 Panel resources

Tag	Description
Display1Reset1Info	This string is used to configure the reset pin of all internal panel (QDI_DISPLAY_PRIMARY, QDI_DISPLAY_SECONDARY, so on). The following information is parsed in a comma-delimited format: "ResourceName, Assert State, Pulse Width, Assert on Power Down" ▪ Resource name – This is a string matching the resource (GPIO) name assigned to the graphics KMD in graphics.asl. ▪ Assert state – This is the state of the GPIO pin when being asserted. This can be either of the following values: ▫ 0 – Active low; the GPIO is pulled low during assertion of the reset ▫ 1 – Active high; the GPIO is pulled high during the assertion of the reset ▪ Pulse width – This determines the pulse width of the reset in microseconds (µs) when the reset is asserted. If the pulse width is 0, no pulse is created and the line is directly deasserted when active. Active-on power down – This Boolean allows for the reset to be asserted during power-down. Depending on the panel configuration, this may be required. Examples are DISPLAY_RESET_PIN, 0, 500, 0. ▪ Primary panel configuration is specified by DSI_PANEL_RESET and other internal configuration specified by DSI_PANELX_RESET, where X = 2, 3, .., N. For example: N = 2 for second panel, 3 for third panel, and so on. Reset pin configuration examples for primary panels are ▫ DSI_PANEL_RESET, 0, 500, 0  ▫ DSI_PANEL_RESET, 0, 0, 1 

Tag	Description
Display1Power1Info Display1Power2Info Display1Power3Info Display4Power1Info Display4Power2Info Display4Power3Info	This string is used to configure the state of GPIOs during power-on/off of the internal panels (Display1Power1Info is used for QDI_DISPLAY_PRIMARY, QDI_DISPLAY_SECONDARY etc). Other resources can be associated with external displays by using other tags; Dislpay4Power1Info is for (QDI_DISPLAY_EXTERNAL, QDI_DISPLAY_EXTERNAL2, and so on). The following information is parsed in a comma-delimited format: "ResourceName, AssertState, RefCount, ActiveDelay, InactiveDelay" Up to three resources can be controlled, Power1, Power2, and Power3. These are all optional and the licensee should remove these settings if they are not used. ▪ Resource name – This is a string matching the resource (GPIO) name assigned to the graphics KMD in graphics.asl. ▪ Assert state – This is the state of the GPIO pin when being asserted; this can be one of the following values: ▫ 0 – Active low, the GPIO is pulled low during power-on ▫ 1 – Active high, the GPIO is pulled high during power-on ▪ RefCount – If a resource is shared between one or multiple displays, the resource is automatically ref counted by the driver. If the licensee wants to specify a default refcount during boot, this field is used to set the default refcount. A default refcount > 0 set during boot means the resource will never be deasserted during power-off. ▪ Active delay – This integer allows the driver to delay for a specific number of microseconds after asserting this resource during power-on. ▪ Inactive delay – This integer allows the driver to delay for a specific number of microseconds after deasserting this resource during power-off.
Display1Special1Info Display4Special1Info	This string is used to configure special states of GPIOs during power-on of the internal panels (Display1Special1Info=QDI_DISPLAY_PRIMARY, QDI_DISPLAY_SECONDARY etc). Other resources can be associated with external displays by using other tags; Display4Special1Info is for (QDI_DISPLAY_EXTERNAL, QDI_DISPLAY_EXTERNAL2, so on). The following information is parsed in a comma-delimited format: "ResourceName, ChipId, OutBufferCfg, Voltage Source, Source, BufferStrength" ▪ Resource name – This is a PMIC GPIO number that needs the special configuration. ▪ ChipId – This value is specific to the PMIC. ▪ Out Buffer Cfg – This value is specific to the PMIC. ▪ Voltage source – This value is specific to the PMIC. ▪ Source – This value is specific to the PMIC. ▪ Buffer strength – This value is specific to the PMIC.
Display1I2C1Info Display1I2C2Info	This parameter defines I2C connections used. There can be maximum of two I2C connections. The following information is parsed in a comma-delimited format: "ResourceName, I2CSlaveType" ▪ 1 – The I2C slave is for the backlight driver ▪ 2 – The I2C slave is for the panel EDID EEPROM

11.1 GPIO configuration

Table 11-2 show the tags that allow for a default GPIO configuration to be applied prior to initializing the display.

NOTE: This configuration is available only in boot loaders (UEFI).

Table 11-2 GPIO configuration tags

Tag	Description
TLMMGPIODefaultLow	This integer list provides the list of APQ GPIO numbers that should be defaulted to high. Only confirgure GPIOs related to the display. **Note:** This list is in hexadecimal, for example, GIO 91 should be written as 5B.
TLMMGPIODefaultHigh	This integer list provides the list of APQ GPIO numbers that should be defaulted to high. Configures GPIOs related to the display. **Note:** This list is in hexadecimal, for example, GIO 91 should be written as 5B.

12 Power-saving configuration

The optional configuration shown in Table 12-1 is used for display power saving when a device enters idle/static screen (vsync interrupt is masked off). There are various scenarios when a device can enter idle screen, for example, when a user reads webpage content.

There are different modes in which the power-saving feature operates. One of the modes is to allow command data transfer between the panel and the APQ even though the device has entered the power-saving configuration. The other mode is to block the command data transfer between the APQ and the panel when the device enters the power-saving configuration.

If an OEM wants to configure the hardware or transmit commands to the panel using the OEM panel driver, it is highly advisable to *not* configure the power-saving configuration to block data. In addition to these modes, there are different power-saving levels that can be configured as well. A higher power-saving level means greater savings can be attained.

Currently, the power-saving feature supports three power-saving levels. At power saving level 0, there are no power savings and the device would be functioning normally with the power saving configuration disabled. When the device transitions to level 2, maximum power savings can be attained. By default, the device is allowed to transfer data when the device enters the power-saving configuration with the power-saving level configured for maximum power savings.

Table 12-1 lists the power saving configuration parameters.

Table 12-1 Power saving configuration

<table>
<tr><th>Tag</th><th>Description</th></tr>
<tr><td>DisplayPowerSavingOverride</td><td>The following optional ACPI configuration field is used to override the default power saving mode and levels; flags can be combined where it makes sense.
<table>
<tr><th>Value (in hex)</th><th>Definition</th></tr>
<tr><td>0x1</td><td>This allows data transfer while power saving is enabled. If both allow data and block data bits are set, then allow data transfer takes precedence over block data transfer; by default, data transfer is allowed while power saving is enabled.</td></tr>
<tr><td>0x2</td><td>The power-saving level is configured to level 0. No other power-saving level transitions are allowed when configured at level 0. This is equivalent to power saving being disabled.</td></tr>
<tr><td>0x4</td><td>Power-saving level is configured to level 1. This allows transitions between level 0 and level 1 power-saving levels. This power-saving level provides conservative power saving with lower latency. If both level 0 and level 1 power-saving levels are configured, then level 1 power saving takes precedence.</td></tr>
<tr><td>0x8</td><td>Power-saving level is configured to level 2. This allows transitions between all the power-saving levels. This power-saving level provides aggressive power saving with higher latency. If all three power-saving levels are configured, then level 2 power-saving level takes precedence. This is the default power-saving level.</td></tr>
<tr><td>0x10</td><td>This is block data transfer during power saving.</td></tr>
</table>
</td></tr>
</table>

Table 12-2 Power saving levels supported by the driver

Level	Description
Level 0 (no power savings)	In level 0, the driver is expected to perform normal operation. By default, when the display is turned on it transitions to level 0. Display power saving can take effect any time by transitioning to either level 1 or level 2. Vsyncs can be turned on/off.
Level 1 (conservative power savings)	In level 1, there is no actual data transfer happening between the APQ and the panel. Vsyncs are turned off, but it is still necessary to maintain phase. Latency to enter and exit from this mode is kept to a minimum. Expected latency for transitioning to level 1 should be less than 2 ms. However, transitioning from level 1 to other levels should be less than 5 ms. Due to this restriction, there is a limit on what can be turned off. Entering this state will lower clocks and possibly gate off, reducing bandwidth votes.
Level 2 (aggressive power savings)	In level 2, maximum display power saving can be achieved. Vsyncs are turned off, and there is no need to maintain phase. Much higher latency is allowed compared to level 1. Expected latency for transitioning to level 2 should be less than 2 ms. However, transitioning from level 2 to other levels should be less than 16 ms. This limit gives flexibility to save more power by turning off additional hardware components. Entering this state includes all power savings from level 1, plus turning off PLLs and enabling Deep Sleep states for peripherals, for example, ULPS, where applicable.

A Specifying the I2C connection in graphics.asl

To add the I2C connection resource in graphics.asl, the resource information is added in platform.xml. The GfxXMLToACPI.pl script is used to generate the updated graphics.asl.

The example entry in platform.xml to specify the I2C connection configuration between the APQ and the panel EEPROM for reading EDID is shown below.

```
<!--  Dynamic EDID I2c Slave   -->
<Resource>
  <Owner>DISPLAY</Owner>
  <Name>I2C_DYNAMICEDID_SLAVE</Name>
  <Type>I2C</Type>
  <AddressingMode>AddressingMode7Bit</AddressingMode>
  <ConnectionSpeed>100000</ConnectionSpeed>
  <SlaveAddress>0x50</SlaveAddress>
  <SlaveMode>ControllerInitiated</SlaveMode>
  <ResourceSource>\\_SB.IC10</ResourceSource>
  <Comment>I2C Slave used to read EDID information</Comment>
</Resource>
```

Table A-3 lists the XML tags and their corresponding descriptions.

Table A-3 XML tags and descriptions

Tag	Description
Owner	Display is the owner of this resource; this field should be used as is
Name	Name of the I2C connection resource; this field should be used as is
Type	Resource type
AddressingMode	To specify 7-bit or 10-bit addressing mode; refer to the ACPI Specification for details
ConnectionSpeed	Maximum connection speed in Hz
SlaveAddress	I2C bus address for this connection
SlaveMode	Can be either Controller initiated or Device initiated. See the *Advanced Configuration and Power Interface Specification* (Revision 5.0 (December 2011)) for more information
ResourceSource	Indicates the GSBI port

B Sample ACPI configuration for DCS-based backlight control

```
///
// BLCP Method - Backlight control method, returns a
//               command buffer for a specific backlight level
//
// Input Parameters
//    Backlight level - Integer from 0% to 100%
//
// Output Parameters
//
// Packet format:
//   +--32bits--+-----variable (8bit alignment)--+
//   |  Header  |       Packet payload           |
//   +----------+--------------------------------+
//
//  For DSI Command packets, payload data must be in this format
//
//  +-- 8 bits-+----variable (8bit alignment)----+
//  | Cmd Type |           Packet Data           |
//  +----------+---------------------------------+
//
//  For I2C Command packets, payload data must be in this format
//
//  +-- 16 bits-+----variable (8bit alignment)----+
//  |  Address  |         Command Data            |
//  +-----------+---------------------------------+
//
//  All packets must follow with a DWORD header with 0x0
//
Method (BLCP, 1, NotSerialized) {

   // Create Response buffer
   Name(RBUF, Buffer(0x100){})
```

```
   // Store the current byte index offset in local0
   Store(0x0, LOCAL0)

   // Create the packet header field
   CreateField(RBUF, Multiply(LOCAL0, 8), 32, PKHR)    // Create the packet
header
   Add(LOCAL0, 4, LOCAL0)                              // Increment the data
pointer

   // Create the packet payload field
   CreateField(RBUF, Multiply(LOCAL0, 8), 32, PKPL)    // Create the packet
payload


   // Backlight programming packet
   If (LEqual(Arg0, 0))
   {
       Name (BOFF,                  // Backlight off
             Buffer() {0x15,        // Command 15
             0x53, 0x00})           // Turn off backlight

       Store(Sizeof(BOFF), PKHR)    // Store the size of the buffer in the
header
       Store(BOFF, PKPL)            // Store the packet payload
       Add(LOCAL0, Sizeof(BOFF), LOCAL0)    // Increment the offset
   }
   Else
   {

       If (LLessEqual(Arg0, 16))      // Level 0-16%
       {
           Name (BL1,
                 Buffer() {0x15,
                 0x51, 0x0C})

           Store(Sizeof(BL1), PKHR)   // Store the size of the buffer in the
header
           Store(BL1, PKPL)           // Store the packet payload
           Add(LOCAL0, Sizeof(BL1), LOCAL0)   // Increment the offset by the
packet size
       }
       ElseIf (LLessEqual(Arg0, 28))  // Level 17-28%
       {
          Name (BL2,
                Buffer() {0x15,
                0x51, 0x19})
```

```
            Store(Sizeof(BL2), PKHR)   // Store the size of the buffer in the
header
            Store(BL2, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL2), LOCAL0)   // Increment the offset by the
packet size
        }
        ElseIf (LLessEqual(Arg0, 40))  // Level 18-40%
        {
            Name (BL3,
                  Buffer() {0x15,
                  0x51, 0x33})

            Store(Sizeof(BL3), PKHR)   // Store the size of the buffer in the
header
            Store(BL3, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL3), LOCAL0)   // Increment the offset by the
packet size
        }
        ElseIf (LLessEqual(Arg0, 52))  // Level 41-52%
        {
            Name (BL4,
                  Buffer() {0x15,
                  0x51, 0x4C})

            Store(Sizeof(BL4), PKHR)   // Store the size of the buffer in the
header
            Store(BL4, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL4), LOCAL0)   // Increment the offset by the
packet size
        }
        ElseIf (LLessEqual(Arg0, 64))  // Level 53-64%
        {
            Name (BL5,
                  Buffer() {0x15,
                  0x51, 0x6D})

            Store(Sizeof(BL5), PKHR)   // Store the size of the buffer in the
header
            Store(BL5, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL5), LOCAL0)   // Increment the offset by the
packet size
        }
        ElseIf (LLessEqual(Arg0, 76))  // Level 65-76%
        {
            Name (BL6,
                  Buffer() {0x15,
                  0x51, 0x99})
```

```
            Store(Sizeof(BL6), PKHR)   // Store the size of the buffer in the
header
            Store(BL6, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL6), LOCAL0)  // Increment the offset by the
packet size
        }
        ElseIf (LLessEqual(Arg0, 88))  // Level 77-88%
        {
            Name (BL7,
                  Buffer() {0x15,
                  0x51, 0xD3})

            Store(Sizeof(BL7), PKHR)   // Store the size of the buffer in the
header
            Store(BL7, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL7), LOCAL0)   // Increment the offset by the
packet size
        }
        Else                           // Level 89-100%
        {
            Name (BL8,
                  Buffer() {0x15,
                  0x51, 0xFF})

           Store(Sizeof(BL8), PKHR)   // Store the size of the buffer in the
header
            Store(BL8, PKPL)           // Store the packet payload
            Add(LOCAL0, Sizeof(BL8), LOCAL0)   // Increment the offset by the
packet size
        }

        // Add additonal ON command
        Name (BON,                     // Backlight on
              Buffer() {0x15,          // Command 15
              0x53, 0x24})             // Manual backlight control

        // Create the packet header field
        CreateField(RBUF, Multiply(LOCAL0, 8), 32, PKH2)    // Create the packet
header
        Add(LOCAL0, 4, LOCAL0)                               // Increment the
data pointer

        // Create the packet payload field
        CreateField(RBUF, Multiply(LOCAL0, 8), 32, PKP2)    // Create the packet
payload

        Store(Sizeof(BON), PKH2)           // Store the size of the buffer in
the header
```

```
        Store(BON, PKP2)                    // Store the packet payload
        Add(LOCAL0, Sizeof(BON), LOCAL0)    // Increment the offset by the packet
size
    }

    // Add the End of Packet marker
    CreateDWordField(RBUF, Multiply(LOCAL0, 8), EOP)
    Store(0x0, EOP)

    // Return the packet data
    Return(RBUF)
}
```

C Enabling 720p video output

If you need an HDMI display with 720p resolution, place the panelcfg file in an SD card and reboot the device with sdcard inserted. (panelcfg.xml).

panelcfg.xml needs to be copied over to root of FAT32 formatted SD card and inserted into SD card slot of the 8016 IOT device.

Copy the following file into a Notepad and rename the file to panelcfg.xml first before adding the SD card:

```
<?xml version='1.0' encoding='utf-8'?>
<PanelName>ADV7533</PanelName>
<PanelDescription>ADV7533 DSI2HDMI (1280x720 24bpp)</PanelDescription>
<Group id='EDID Configuration'>
  <ManufactureID>0xAF0D</ManufactureID>
  <ProductCode>0x0011</ProductCode>
  <SerialNumber>0x000000</SerialNumber>
  <WeekofManufacture>0x09</WeekofManufacture>
  <YearofManufacture>0x13</YearofManufacture>
  <EDIDVersion>1</EDIDVersion>
  <EDIDRevision>3</EDIDRevision>
  <VideoInputDefinition>0x80</VideoInputDefinition>
  <HorizontalScreenSize>0x16</HorizontalScreenSize>
  <VerticalScreenSize>0x0D</VerticalScreenSize>
  <DisplayTransferCharacteristics>0x78</DisplayTransferCharacteristics>
  <FeatureSupport>0xA</FeatureSupport>
  <Red.GreenBits>0xCF</Red.GreenBits>
  <Blue.WhiteBits>0x45</Blue.WhiteBits>
  <RedX>0x90</RedX>
  <RedY>0x59</RedY>
  <GreenX>0x57</GreenX>
  <GreenY>0x95</GreenY>
  <BlueX>0x29</BlueX>
  <BlueY>0x1f</BlueY>
  <WhiteX>0x50</WhiteX>
  <WhiteY>0x54</WhiteY>
  <EstablishedTimingsI>0x0</EstablishedTimingsI>
  <EstablishedTimingsII>0x0</EstablishedTimingsII>
  <ManufacturesTiming>0x0</ManufacturesTiming>
  <StandardTimings1/>
  <StandardTimings2/>
  <StandardTimings3/>
```

```
  <StandardTimings4/>
  <StandardTimings5/>
  <StandardTimings6/>
  <StandardTimings7/>
  <SignalTimingInterface/>
</Group>
<Group id='Active Timing'>
  <HorizontalActive units='Dot Clocks'>1280</HorizontalActive>
  <HorizontalFrontPorch units='Dot Clocks'>110</HorizontalFrontPorch>
  <HorizontalBackPorch units='Dot Clocks'>220</HorizontalBackPorch>
  <HorizontalSyncPulse units='Dot Clocks'>40</HorizontalSyncPulse>
  <HorizontalSyncSkew units='Dot Clocks'>0</HorizontalSyncSkew>
  <HorizontalLeftBorder units='Dot Clocks'>0</HorizontalLeftBorder>
  <HorizontalRightBorder units='Dot Clocks'>0</HorizontalRightBorder>
  <VerticalActive units='Dot Clocks'>720</VerticalActive>
  <VerticalFrontPorch units='Lines'>5</VerticalFrontPorch>
  <VerticalBackPorch units='Lines'>20</VerticalBackPorch>
  <VerticalSyncPulse units='Lines'>5</VerticalSyncPulse>
  <VerticalSyncSkew units='Lines'>0</VerticalSyncSkew>
  <VerticalTopBorder units='Lines'>0</VerticalTopBorder>
  <VerticalBottomBorder units='Lines'>0</VerticalBottomBorder>
  <InvertDataPolarity>False</InvertDataPolarity>
  <InvertVsyncPolairty>False</InvertVsyncPolairty>
  <InvertHsyncPolarity>False</InvertHsyncPolarity>
  <BorderColor>0x0</BorderColor>
</Group>
<Group id='Display Interface'>
  <InterfaceType units='QDI_DisplayConnectType'>8</InterfaceType>
  <InterfaceColorFormat
units='QDI_PixelFormatType'>3</InterfaceColorFormat>
</Group>
<Group id='DSI Interface'>
  <DSIChannelId units='DSI_Channel_IDType'>1</DSIChannelId>
  <DSIVirtualId units='DSI_Display_VCType'>0</DSIVirtualId>
  <DSIColorFormat units='DSI_ColorFormatType'>36</DSIColorFormat>
  <DSITrafficMode units='DSI_TrafficModeType'>0</DSITrafficMode>
  <DSILanes units='integer'>3</DSILanes>
  <DSIHsaHseAfterVsVe units='Bool'>False</DSIHsaHseAfterVsVe>
  <DSILowPowerModeInHFP units='Bool'>False</DSILowPowerModeInHFP>
  <DSILowPowerModeInHBP units='Bool'>False</DSILowPowerModeInHBP>
  <DSILowPowerModeInHSA units='Bool'>False</DSILowPowerModeInHSA>
  <DSILowPowerModeInBLLPEOF units='Bool'>False</DSILowPowerModeInBLLPEOF>
  <DSILowPowerModeInBLLP units='Bool'>False</DSILowPowerModeInBLLP>
  <DSIRefreshRate units='integer Q16.16'>0x3C0000</DSIRefreshRate>
  <DSIPhyDCDCMode units='Bool'>False</DSIPhyDCDCMode>
</Group>
<DSII2CSlaveAddressList units='int'>0x39 0x3C</DSII2CSlaveAddressList>
<DSII2CI2CNumOfSlaves units='int'>0x2</DSII2CI2CNumOfSlaves>
```

```
<DSII2CFrequency units='int'>0x190</DSII2CFrequency>
<DSII2CGSBIPort units='int'>0x4</DSII2CGSBIPort>
<DisplayPlatformID>0x3</DisplayPlatformID>
<PanelID>0x2D0</PanelID>
<I2CInitSequence>
  39 41 10
  ff 05
  39 d6 48
  ff 05
  3c 03 89
  39 16 20
  39 9a e0
  39 ba 70
  39 de 82
  39 e4 c0
  39 e5 80
  3c 15 d0
  3c 17 d0
  3c 24 20
  3c 57 11
  39 af 06
  39 40 80
  39 4c 04
  39 49 02
  39 0d 40
  3c 1c 30
  39 17 02
  3c 16 00
  3c 27 CB
  3c 28 67
  3c 29 20
  3c 2a 02
  3c 2b 80
  3c 2c 06
  3c 2d e0
  3c 2e 0d
  3c 2f c0
  3c 30 2e
  3c 31 e0
  3c 32 00
  3c 33 50
  3c 34 00
  3c 35 50
  3c 36 01
  3c 37 40
  3c 27 CB
  3c 27 8B
  ff 05
```

```
    3c 27 CB
    ff 64
    3c 55 00
    3c 03 09
    ff 05
    3c 03 89
  </I2CInitSequence>
```

D Enabling 1080p video output

If you need an HDMI display with 1080p resolution, place the panelcfg file in an SD card and reboot the device with sdcard inserted.

panelcfg.xml needs to be copied over to root of FAT32 formatted SD card and inserted into SD card slot of the 8016 IOT device (please use the file as is, and do not rename it).

Reboot the device for changes to take place.

NOTE: The panelcfg.xml file with sd card needs to be inserted all the time, between reboots. If you do not insert the SD card during a bootup it will bootup with default config(HDMI 720p).

To revert the display back to HDMI, just remove the sdcard that has the dsi panelcfg.xml. Device will enable hdmi display on next boot.

Copy the following file into a Notepad and rename the file to panelcfg.xml first before copying to the SD card:

```
<?xml version='1.0' encoding='utf-8'?>
<PanelName>ADV7533</PanelName>
<PanelDescription>ADV7533 DSI2HDMI (1920x1080 24bpp)</PanelDescription>
<Group id='EDID Configuration'>
  <ManufactureID>0xAF0D</ManufactureID>
  <ProductCode>0x0011</ProductCode>
  <SerialNumber>0x000000</SerialNumber>
  <WeekofManufacture>0x09</WeekofManufacture>
  <YearofManufacture>0x13</YearofManufacture>
  <EDIDVersion>1</EDIDVersion>
  <EDIDRevision>3</EDIDRevision>
  <VideoInputDefinition>0x80</VideoInputDefinition>
  <HorizontalScreenSize>0x16</HorizontalScreenSize>
  <VerticalScreenSize>0x0D</VerticalScreenSize>
  <DisplayTransferCharacteristics>0x78</DisplayTransferCharacteristics>
  <FeatureSupport>0xA</FeatureSupport>
  <Red.GreenBits>0xCF</Red.GreenBits>
  <Blue.WhiteBits>0x45</Blue.WhiteBits>
  <RedX>0x90</RedX>
  <RedY>0x59</RedY>
  <GreenX>0x57</GreenX>
  <GreenY>0x95</GreenY>
  <BlueX>0x29</BlueX>
  <BlueY>0x1f</BlueY>
```

```
    <WhiteX>0x50</WhiteX>
    <WhiteY>0x54</WhiteY>
    <EstablishedTimingsI>0x0</EstablishedTimingsI>
    <EstablishedTimingsII>0x0</EstablishedTimingsII>
    <ManufacturesTiming>0x0</ManufacturesTiming>
    <StandardTimings1/>
    <StandardTimings2/>
    <StandardTimings3/>
    <StandardTimings4/>
    <StandardTimings5/>
    <StandardTimings6/>
    <StandardTimings7/>
    <SignalTimingInterface/>
  </Group>
  <Group id='Active Timing'>
    <HorizontalActive units='Dot Clocks'>1920</HorizontalActive>
    <HorizontalFrontPorch units='Dot Clocks'>88</HorizontalFrontPorch>
    <HorizontalBackPorch units='Dot Clocks'>148</HorizontalBackPorch>
    <HorizontalSyncPulse units='Dot Clocks'>44</HorizontalSyncPulse>
    <HorizontalSyncSkew units='Dot Clocks'>0</HorizontalSyncSkew>
    <HorizontalLeftBorder units='Dot Clocks'>0</HorizontalLeftBorder>
    <HorizontalRightBorder units='Dot Clocks'>0</HorizontalRightBorder>
    <VerticalActive units='Dot Clocks'>1080</VerticalActive>
    <VerticalFrontPorch units='Lines'>4</VerticalFrontPorch>
    <VerticalBackPorch units='Lines'>36</VerticalBackPorch>
    <VerticalSyncPulse units='Lines'>5</VerticalSyncPulse>
    <VerticalSyncSkew units='Lines'>0</VerticalSyncSkew>
    <VerticalTopBorder units='Lines'>0</VerticalTopBorder>
    <VerticalBottomBorder units='Lines'>0</VerticalBottomBorder>
    <InvertDataPolarity>False</InvertDataPolarity>
    <InvertVsyncPolairty>False</InvertVsyncPolairty>
    <InvertHsyncPolarity>False</InvertHsyncPolarity>
    <BorderColor>0x0</BorderColor>
  </Group>
  <Group id='Display Interface'>
    <InterfaceType units='QDI_DisplayConnectType'>8</InterfaceType>
    <InterfaceColorFormat
  units='QDI_PixelFormatType'>3</InterfaceColorFormat>
  </Group>
  <Group id='DSI Interface'>
    <DSIChannelId units='DSI_Channel_IDType'>1</DSIChannelId>
    <DSIVirtualId units='DSI_Display_VCType'>0</DSIVirtualId>
    <DSIColorFormat units='DSI_ColorFormatType'>36</DSIColorFormat>
    <DSITrafficMode units='DSI_TrafficModeType'>0</DSITrafficMode>
    <DSILanes units='integer'>4</DSILanes>
    <DSIHsaHseAfterVsVe units='Bool'>False</DSIHsaHseAfterVsVe>
    <DSILowPowerModeInHFP units='Bool'>False</DSILowPowerModeInHFP>
    <DSILowPowerModeInHBP units='Bool'>False</DSILowPowerModeInHBP>
```

```
  <DSILowPowerModeInHSA units='Bool'>False</DSILowPowerModeInHSA>
  <DSILowPowerModeInBLLPEOF units='Bool'>False</DSILowPowerModeInBLLPEOF>
  <DSILowPowerModeInBLLP units='Bool'>False</DSILowPowerModeInBLLP>
  <DSIRefreshRate units='integer Q16.16'>0x3C0000</DSIRefreshRate>
  <DSIPhyDCDCMode units='Bool'>False</DSIPhyDCDCMode>
</Group>
<DSII2CSlaveAddressList units='int'>0x39 0x3C</DSII2CSlaveAddressList>
<DSII2CI2CNumOfSlaves units='int'>0x2</DSII2CI2CNumOfSlaves>
<DSII2CFrequency units='int'>0x190</DSII2CFrequency>
<DSII2CGSBIPort units='int'>0x4</DSII2CGSBIPort>
<DisplayPlatformID>0x3</DisplayPlatformID>
<PanelID>0x0</PanelID>
<I2CInitSequence>
  39 41 10
  ff 05
  39 d6 48
  ff 05
  3c 03 89
  39 16 20
  39 9a e0
  39 ba 70
  39 de 82
  39 e4 c0
  39 e5 80
  3c 15 d0
  3c 17 d0
  3c 24 20
  3c 57 11
  39 af 06
  39 40 80
  39 4c 04
  39 49 02
  39 0d 40
  3c 1c 40
  39 17 02
  3c 16 00
  3c 27 CB
  3c 28 89
  3c 29 80
  3c 2a 02
  3c 2b c0
  3c 2c 05
  3c 2d 80
  3c 2e 09
  3c 2f 40
  3c 30 46
  3c 31 50
  3c 32 00
```

```
    3c 33 50
    3c 34 00
    3c 35 40
    3c 36 02
    3c 37 40
    3c 27 CB
    3c 27 8B
    ff 05
    3c 27 CB
    ff 64
    3c 55 00
    3c 03 09
    ff 05
    3c 03 89
  </I2CInitSequence>
```

E References

Title	Number
Resources	
VESA Enhanced Extended Display Identification Data – Implementation Guide	Version 1.0 (June 2001)
VESA Enhanced Extended Display Identification Data Standard	(Defines EDID Structure Version 1, Revision 4) Release A, Revision 2 (September 2006)
MIPI Alliance Specification for D-PHY	Version 1.00.00 (May 2009)
MIPI Alliance Specification for Display Serial Interface	Version 1.02.00 (June 2010)
MIPI Alliance Specification for Display Serial Interface	Version 1.03.00 (August 2011)
Advanced Configuration and Power Interface Specification	Revision 5.0 (December 2011)

EXHIBIT 1

PLEASE READ THIS LICENSE AGREEMENT ("AGREEMENT") CAREFULLY. THIS AGREEMENT IS A BINDING LEGAL AGREEMENT ENTERED INTO BY AND BETWEEN YOU (OR IF YOU ARE ENTERING INTO THIS AGREEMENT ON BEHALF OF AN ENTITY, THEN THE ENTITY THAT YOU REPRESENT) AND Qualcomm Technologies, Inc. ("QTI" "WE" "OUR" OR "US"). THIS IS THE AGREEMENT THAT APPLIES TO YOUR USE OF THE DESIGNATED AND/OR ATTACHED DOCUMENTATION AND ANY UPDATES OR IMPROVEMENTS THEREOF (COLLECTIVELY, "MATERIALS"). BY USING, ACCESSING, DOWNLOADING OR COMPLETING THE INSTALLATION OF THE MATERIALS, YOU ARE ACCEPTING THIS AGREEMENT AND YOU AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. IF YOU DO NOT AGREE TO THESE TERMS, QTI IS UNWILLING TO AND DOES NOT LICENSE THE MATERIALS TO YOU. IF YOU DO NOT AGREE TO THESE TERMS YOU MUST DISCONTINUE AND YOU MAY NOT USE THE MATERIALS OR RETAIN ANY COPIES OF THE MATERIALS. ANY USE OR POSSESSION OF THE MATERIALS BY YOU IS SUBJECT TO THE TERMS AND CONDITIONS SET FORTH IN THIS AGREEMENT.

1.1 **License.** Subject to the terms and conditions of this Agreement, including, without limitation, the restrictions, conditions, limitations and exclusions set forth in this Agreement, Qualcomm Technologies, Inc. ("QTI") hereby grants to you a nonexclusive, limited license under QTI's copyrights to use the attached Materials; and to reproduce and redistribute a reasonable number of copies of the Materials. You may not use Qualcomm Technologies or its affiliates or subsidiaries name, logo or trademarks; and copyright, trademark, patent and any other notices that appear on the Materials may not be removed or obscured. QTI shall be free to use suggestions, feedback or other information received from You, without obligation of any kind to You. QTI may immediately terminate this Agreement upon your breach. Upon termination of this Agreement, Sections 1.2-4 shall survive.

1.2 **Indemnification.** You agree to indemnify and hold harmless QTI and its officers, directors, employees and successors and assigns against any and all third party claims, demands, causes of action, losses, liabilities, damages, costs and expenses, incurred by QTI (including but not limited to costs of defense, investigation and reasonable attorney's fees) arising out of, resulting from or related to: (i) any breach of this Agreement by You; and (ii) your acts, omissions, products and services. If requested by QTI, You agree to defend QTI in connection with any third party claims, demands, or causes of action resulting from, arising out of or in connection with any of the foregoing.

1.3 **Ownership.** QTI (or its licensors) shall retain title and all ownership rights in and to the Materials and all copies thereof, and nothing herein shall be deemed to grant any right to You under any of QTI's or its affiliates' patents. You shall not subject the Materials to any third party license terms (e.g., open source license terms). You shall not use the Materials for the purpose of identifying or providing evidence to support any potential patent infringement claim against QTI, its affiliates, or any of QTI's or QTI's affiliates' suppliers and/or direct or indirect customers. QTI hereby reserves all rights not expressly granted herein.

1.4 **WARRANTY DISCLAIMER.** YOU EXPRESSLY ACKNOWLEDGE AND AGREE THAT THE USE OF THE MATERIALS IS AT YOUR SOLE RISK. THE MATERIALS AND TECHNICAL SUPPORT, IF ANY, ARE PROVIDED "AS IS" AND WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS OR IMPLIED. QTI ITS LICENSORS AND AFFILIATES MAKE NO WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THE MATERIALS OR ANY OTHER INFORMATION OR DOCUMENTATION PROVIDED UNDER THIS AGREEMENT, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR AGAINST INFRINGEMENT, OR ANY EXPRESS OR IMPLIED WARRANTY ARISING OUT OF TRADE USAGE OR OUT OF A COURSE OF DEALING OR COURSE OF PERFORMANCE. NOTHING CONTAINED IN THIS AGREEMENT SHALL BE CONSTRUED AS (I) A WARRANTY OR REPRESENTATION BY QTI, ITS LICENSORS OR AFFILIATES AS TO THE VALIDITY OR SCOPE OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT OR (II) A WARRANTY OR REPRESENTATION BY QTI THAT ANY MANUFACTURE OR USE WILL BE FREE FROM INFRINGEMENT OF PATENTS, COPYRIGHTS OR OTHER INTELLECTUAL PROPERTY RIGHTS OF OTHERS, AND IT SHALL BE THE SOLE RESPONSIBILITY OF YOU TO MAKE SUCH DETERMINATION AS IS NECESSARY WITH RESPECT TO THE ACQUISITION OF LICENSES UNDER PATENTS AND OTHER INTELLECTUAL PROPERTY OF THIRD PARTIES.

1.5 **LIMITATION OF LIABILITY.** IN NO EVENT SHALL QTI, QTI'S AFFILIATES OR ITS LICENSORS BE LIABLE TO YOU FOR ANY INCIDENTAL, CONSEQUENTIAL OR SPECIAL DAMAGES, INCLUDING BUT NOT LIMITED TO ANY LOST PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL DAMAGES, ARISING OUT OF THE USE OR INABILITY TO USE, OR THE DELIVERY OR FAILURE TO DELIVER, ANY OF THE MATERIALS, OR ANY BREACH OF ANY OBLIGATION UNDER THIS AGREEMENT, EVEN IF QTI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE FOREGOING LIMITATION OF LIABILITY SHALL REMAIN IN FULL FORCE AND EFFECT REGARDLESS OF WHETHER YOUR REMEDIES HEREUNDER ARE DETERMINED TO HAVE FAILED OF THEIR ESSENTIAL PURPOSE. THE ENTIRE LIABILITY OF QTI, QTI's AFFILIATES AND ITS LICENSORS, AND THE SOLE AND EXCLUSIVE REMEDY OF YOU, FOR ANY CLAIM OR CAUSE OF ACTION ARISING HEREUNDER (WHETHER IN CONTRACT, TORT, OR OTHERWISE) SHALL NOT EXCEED US$10.

2. **COMPLIANCE WITH LAWS; APPLICABLE LAW.**

Any litigation or other dispute resolution between You and Us arising out of or relating to this Agreement, or Your relationship with Us will take place in the Southern District of California, and You and QTI hereby consent to the personal jurisdiction of and exclusive venue in the state and federal courts within that District with respect any such litigation or dispute resolution. This Agreement will be governed by and construed in accordance with the laws of the United States and the State of California, except that body of California law concerning conflicts of law. This Agreement shall not be governed by the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded.

3. **CONTRACTING PARTIES.** If the Materials are downloaded on any computer owned by a corporation or other legal entity, then this Agreement is formed by and between QTI and such entity. The individual accepting the terms of this Agreement represents and warrants to QTI that they have the authority to bind such entity to the terms and conditions of this Agreement.

4. MISCELLANEOUS PROVISIONS. This Agreement, together with all exhibits attached hereto, which are incorporated herein by this reference, constitutes the entire agreement between QTI and You and supersedes all prior negotiations, representations and agreements between the parties with respect to the subject matter hereof. No addition or modification of this Agreement shall be effective unless made in writing and signed by the respective representatives of QTI and You. The restrictions, limitations, exclusions and conditions set forth in this Agreement shall apply even if QTI or any of its affiliates becomes aware of or fails to act in a manner to address any violation or failure to comply therewith. You hereby acknowledge and agree that the restrictions, limitations, conditions and exclusions imposed in this Agreement on the rights granted in this Agreement are not a derogation of the benefits of such rights. You further acknowledges that, in the absence of such restrictions, limitations, conditions and exclusions, QTI would not have entered into this Agreement with You. Each party shall be responsible for and shall bear its own expenses in connection with this Agreement. If any of the provisions of this Agreement are determined to be invalid, illegal, or otherwise unenforceable, the remaining provisions shall remain in full force and effect. This Agreement is entered into solely in the English language, and if for any reason any other language version is prepared by any party, it shall be solely for convenience and the English version shall govern and control all aspects. If You are located in the province of Quebec, Canada, the following applies: The Parties hereby confirm they have requested this Agreement and all related documents be prepared in English.