zgolee的专栏

记录零碎的知识，期待质变的来临...

MIPI Camera Interface Specifications

Camera Interface Specifications

The Camera Working Group released the CSI-2 v1.0 specification in 2005. A second phase addressed an updated D-PHY specification and JEDEC interleaving.

Currently, the group is developing CSI-3.

Specifications are available to MIPI members only. For more information on joining MIPI, please Go to Join MIPI.

· Camera Serial Interface CSI-2 Active

· Camera Parallel Interface Legacy

· Camera Serial Interface CSI-1 Legacy

MIPI Alliance Standard for Camera Serial Interface CSI-2

Purpose

Demand for increasingly higher image resolutions is pushing the bandwidth capacity of existing host processor-to-camera sensor interfaces. Common parallel interfaces are difficult to expand, require many interconnects and consume relatively large amounts of power. Emerging serial interfaces address many of the shortcomings of parallel interfaces while introducing their own problems. Incompatible, proprietary interfaces prevent devices from different manufacturers from working together. This can raise system costs and reduce system reliability by requiring "hacks" to force the devices to interoperate. The lack of a clear industry standard can slow innovation and inhibit new product market entry. CSI-2 provides the mobile industry a standard, robust, scalable, low-power, high-speed, cost-effective interface that supports a wide range of imaging solutions for mobile devices.

Scope

The Camera Serial Interface 2 specification defines an interface between a peripheral device (camera) and a host processor (baseband, application engine). The purpose of this specification is to identify a standard interface between a camera and a host processor for mobile device applications.

A host processor in this specification means the hardware and software that performs essential core functions for telecommunication or application tasks. The engine of a mobile terminal includes hardware and the functions, which enable the basic operation of the mobile terminal. These include, for example, the printed circuit boards, RF components, basic electronics, and basic software, such as the digital signal processing software.

Overview of CSI-2

The CSI-2 specification defines standard data transmission and control interfaces between transmitter and receiver. Data transmission interface (referred as CSI-2) is unidirectional differential serial interface with data and clock signals; the physical layer of this interface is the MIPI Alliance Standard for D-PHY [2].

Figure 1 illustrates connections between CSI-2 transmitter and receiver, which typically are a camera module and a receiver module, part of the mobile phone engine.

The control interface (referred as CCI) is a bi-directional control interface compatible with I2C standard.

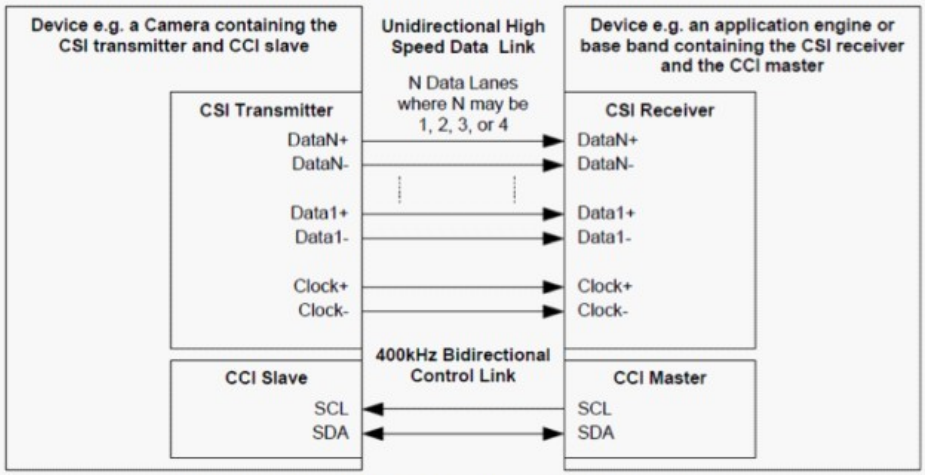


Figure 1 CSI-2 and CCI Transmitter and Receiver Interface

Camera Parallel Interface

Introduction

The Camera Parallel Interface (CPI) is a parallel interface between a digital camera module and a mobile phone engine. This document specifies the requirements on the application processor engine (APE) and on the image sensor in order to allow the interfacing of the two components.

Overview

The Camera Parallel Interface (CPI) shall implement a 8-bit parallel link from image sensor to APE.

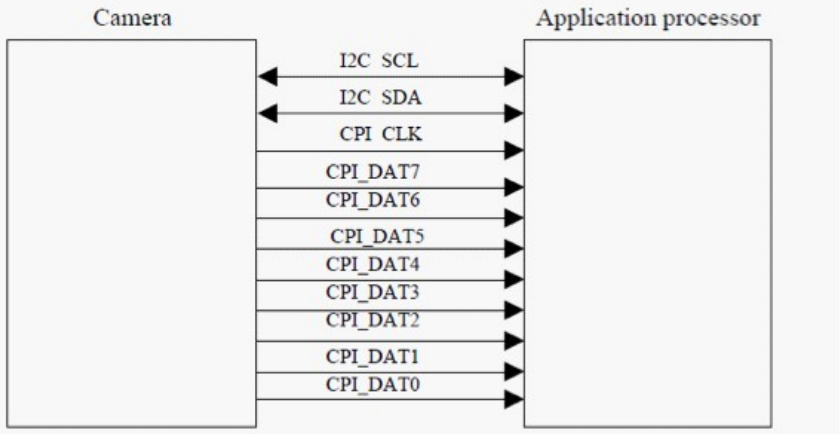


Figure 1. CPI interface between camera and application processor

Camera Serial Interface

Introduction

The Camera Serial Interface (CSI) is a serial interface between digital camera module and mobile phone engine. The purpose of this document is to specify a standard interface between camera and phone engine. The mobile phone engine in this document means the hardware and software that performs essential core functions for telecommunication or application tasks. The engine of a mobile terminal includes hardware and the functions, which enable the basic operation of the mobile terminal. These include, for example, the printed circuit boards, RF components, basic electronics, and basic software, such as the digital signal processing software.

Overview

The interface between CSI transmitter and receiver consists of data transfer and control interface. The data transfer interface (referred as CSI in this document) is a unidirectional differential serial interface with data and clock signals. Figure 1 illustrates the connection between CSI transmitter and receiver, which typically are a camera module and a receiver module, part of the mobile phone engine.

The control interface (referred as CCI) is a bi-directional control interface compatible with I2C standard.

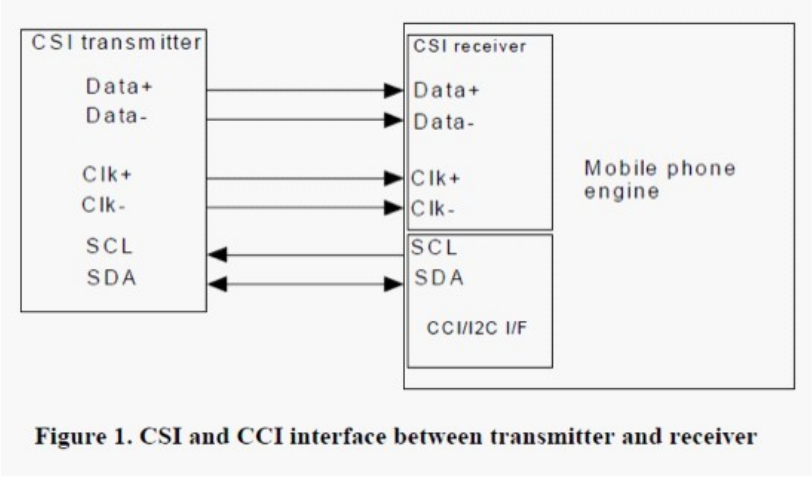


Figure 1. CSI and CCI interface between transmitter and receiver

The physical layer of CSI is based on signaling scheme called SubLVDS, which is current mode differential low voltage signaling method modified from the IEEE 1596.3 LVDS standard for reduced power consumption. Electrical specifications for the SubLVDS I/O's can be found from chapter 10. The use of SubLVDS enables the use of high data rates with low EMI with maximum transfer capacity of 208 Mbps. Thus transferring SVGA images (800 x 600 pixels) in YUV420 format at 30 fps is possible.

Transmission clock rate may vary between 1 MHz to 208 MHz. The maximum clock frequency of CSI should be chosen to be lowest possible for the application / transmitting device in question. It is recommended that CSI transmission clock is generated from host system clock using PLL or clock doubler. This simplifies greatly EMC design. Additional oscillators should be avoided in the camera module.

The CSI transmitter provides the data-qualifying clock to the CSI receiver. The transmission clock rate in practice can vary transmitter implementation. The synchronization to receivers internal clock domain has to be handled inside the receiver. For example, a transmitter may use 13 MHz transmission clock for some type of data and 104 MHz clock for some other type of data. The principle of receiver operation is illustrated in Figure 2.

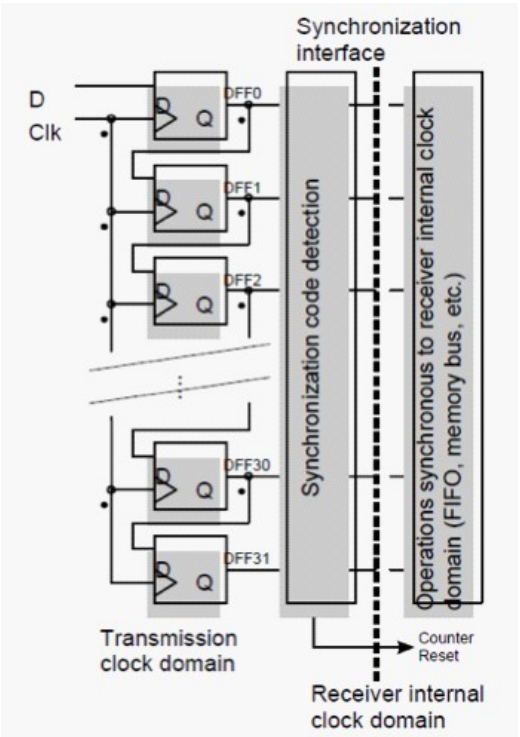


Figure 2. CSI receiver operating principle

- 上一篇Camera图像处理原理及实例分析-重要图像概念
- 下一篇自学经历

camera(8) 硬件(10)

- ·Camera图像处理原理及实例分析-重要图像概念2011-10-30阅读7734
- ·快门与曝光2011-10-30阅读3144
- ·变焦与对焦2011-10-30阅读9950
- ·CCM2011-10-30阅读1553

- ·camera的分辨率2011-10-30阅读2760
- ·景深2011-10-30阅读1962
- ·焦距与光圈2011-10-30阅读6582
- ·Camera硬件结构组成2011-10-30阅读3232

参考知识库

Go知识库

1282 关注 | 713 收录

猜你在找

ArcGIS for javascript 项目实战（环境监测系统）

ArcGIS for JavaScript

Jquery从入门到精通经典视频教程

ArcGIS for JS基础课程

IT革命—Docker技术精解

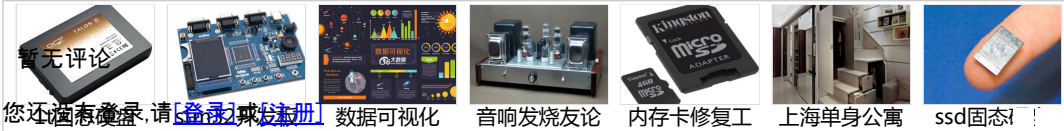
mipiorgMIPI SPECIFICATIONSMIPI规范特点和成立流程简介

extcurllibsinterfaceo undefined reference to symbol

CAMIF camera interface 摄像头接口 简介

S5PV210 三个Camera InterfaceCAMIFFIMC的区别

S5PV210 三个Camera InterfaceCAMIFFIMC的区别



上海单身公寓 ssd固态

查看评论

暂无评论

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack VPN Spark ERP IE10
Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery BI HTML5 Spring Apache .NET API HTML SDK IIS
Fedora XML LBS Unity Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack FTC
coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo Compuware 大数据 aptech Perl
Tornado Ruby Hibernate ThinkPHP HBase Pure Solr Angular Cloud Foundry Redis Scala Django Bootstrap

个人资料

zgolee

- 访问:637042次
- 积分:7015
- 等级:
- 排名:第2366名
- 原创:82篇
- 转载:30篇
- 译文:0篇
- 评论:217条

文章搜索

搜索

文章分类

- android(30)
- android java框架(4)
- dalvik vm(1)
- 非专业(8)

- linux OS(27)
- 硬件(11)
- camera(9)
- wifi(1)
- linux驱动(12)
- 内存管理(1)
- 电源管理(10)
- linux脚本(2)
- c语言(4)
- u-boot(7)

文章存档

- 2014年03月(1)
- 2012年06月(1)
- 2012年05月(4)
- 2012年04月(4)
- 2012年02月(2)

展开

推荐文章

- * 程序员10月书讯，评论得书
- * Android中Xposed框架篇---修改系统位置信息实现自身隐藏功能
- * Chromium插件(Plugin)模块(Module)加载过程分析
- * Android TV开发总结--构建一个TV app的直播节目实例
- * 架构设计：系统存储--MySQL简单主从方案及暴露的问题

最新评论

- android 集成第三方静态库的编译方法
  牙哥: 好贴必须顶另外3楼那种没有 mk 文件的， eclipse 在工程上右键-》android tool...
- select(poll)系统调用实现解析(三)
  这个coder不太冷: 只能先马克一下了..刚开始看还很难理解. 要是有个思维导图就好了= =
- Camera图像处理原理及实例分析-重要图像概念
  novel2008: 不错，谢谢！
- uboot编译过程
  u010416150: 很详细的讲解！
- select(poll)系统调用实现解析(三)
  lhshaoren: 很棒，感谢楼主！
- select(poll)系统调用实现解析(三)
  Holy_Sunrise: 博主非常感谢你的这个系列的文章，特意登录来拜谢！
- Android在标准linux基础上对休眠唤醒的实现(三)
  hoboq: 您好~最近在看release所有的wakelock后，系统没有马上suspend的原因，所以想在re...
- 新版linux系统设备架构中关于电源管理方式的变更
  ilotuo: 我看了sysfs.c 尝试了下#echo disabled > control #cat contr...
- 新版linux系统设备架构中关于电源管理方式的变更
  ilotuo: ls 你好. 文章有点高深需要点时间琢磨我看到我的驱动设备的power接口似乎不止两个.autosu...
- arm处理器异常处理-swi
  loe: 好文章