

AN14191

如何在MCXN MCU上用SmartDMA实现摄像头接口

第1版 — 2024年1月18日

应用笔记

文档信息

信息	内容
关键词	AN14191、MCXN947、SmartDMA
摘要	本应用笔记介绍了MCXN947上的摄像头并行接口解决方案，包括摄像头接口、特性、API例程和演示等的介绍。



1 介绍

本应用笔记介绍了MCXN947上的摄像头并行接口解决方案，包括摄像头接口、特性、API例程和演示等。MCXN947包含一个协处理器SmartDMA，可用于实现摄像头接口。

2 目标应用

摄像头接口是摄像头的一个重要组成部分，可作以下用途：

- 目标检测
- 手势识别
- 颜色识别
- 二维码扫描

3 摄像头接口简介

典型的摄像头接口支持至少一个并行接口，然而如今许多摄像头接口开始支持MIPI CSI接口。

摄像头并行接口由以下线路组成：

- 数据线（D[0:7]）：这些并行数据线承载了像素数据。在这些线路上传输的数据随着每个像素时钟（PCLK）的变化而变化。
- 水平同步（HSYNC）：这是来自摄像头传感器的特殊信号。HSYNC表示数据帧中的一行已传输完成。
- 垂直同步（VSYNC）：该信号在整个帧传输完成后发送。该信号通常用于表示一个完整的帧已传输完成。
- 像素时钟（PCLK）：该像素时钟依每个像素发生变化。

本应用笔记仅关注数字视频端口（DVP）接口，这是一种并行接口。

4 MCX Nx4x MCU简介

MCX Nx4x MCU具有高达2 MB的闪存、高达512 kB的SRAM、150 MHz系统时钟、SmartDMA FlexIO、QSPI接口和I2C接口。SmartDMA可用于将数据从摄像头接口传输到内部RAM。FlexIO可用于将RAM中的数据传送到LCD接口。QSPI可用于扩展存储器以存储帧数据。内部SRAM可用于存储临时帧数据。I2C接口可用于配置摄像头模块。

本应用笔记重点介绍了摄像头的实现。有关用FlexIO实现的LCD接口，请参阅《使用FlexIO驱动8080总线接口LCD模块》（文档[AN5313](#)）。

5 摄像头接口的特性

支持的格式如下：

- RGB565。
- 对于VGA（640x480）的最大图像传输速率为30 fps。对于小RAM的芯片，请降低图像大小和帧率。
- OV7670是经过测试的摄像头模块。

- 如果其他摄像头模块提供相同的信号时序，则也可以支持。

6 功能描述

本章对各种功能进行介绍。

6.1 SmartDMA的工作方式

SmartDMA可以在一个系统周期内访问GPIO。它从摄像头读取数据并将数据存储存储在RAM中。之后，FlexIO可以将数据发送到LCD屏。

在使用SmartDMA之前，必须进行一些配置，包括引脚配置、时钟使能、专用处理器使能、中断使能等。

SmartDMA与Arm内核共享系统时钟。为了加快处理速度，最好将系统时钟配置为150 MHz。

6.2 摄像头的时钟源

摄像头需要一个约6 MHz的时钟源，该时钟源由MCU的CLKOUT信号提供。不同的时钟源可以获得不同的帧率输出。

6.3 MCU8080 LCD接口

FlexIO可以实现LCD接口。请参阅《使用FlexIO驱动8080总线接口LCD模块》（文档[AN5313](#)）。

6.4 I2C接口

摄像头通过MCU的I2C接口进行配置。在摄像头运行之前，必须由Arm内核通过I2C外设进行配置。

6.5 内存的占用

在本应用笔记中，LCD屏的分辨率为480x320。OV7670摄像头模块的输出分辨率为640x480。SmartDMA可以裁剪图像输出的尺寸，且仅将分辨率为480x320的帧存储在RAM中，以匹配LCD的尺寸。这就体现了SmartDMA的灵活性。因此，所需的存储空间为300kB的RAM。

要实现不同大小的采样，请使用由SmartDMA实现的不同API功能。

乒乓缓冲区可用来存储部分帧数据。这就需要及时处理数据，以免影响后续的数据存储。

此外，为了获得高性能，摄像头引擎的指令代码必须在RAM中运行。该解决方案使用SRAMX来存储摄像头引擎代码。由于SmartDMA的实现与运行代码所在的RAM地址有关，因此，此运行地址必须固定在SRAMX的起始位置。如果在其他的RAM地址运行SmartDMA代码，则需要重新生成代码数组。

6.6 其他支持的摄像头模块

如果其他摄像头模块能提供相同的信号时序，则它们也可以支持。

1. 摄像头模块必须在RGB565模式下配置，时序图如[图1](#)所示。

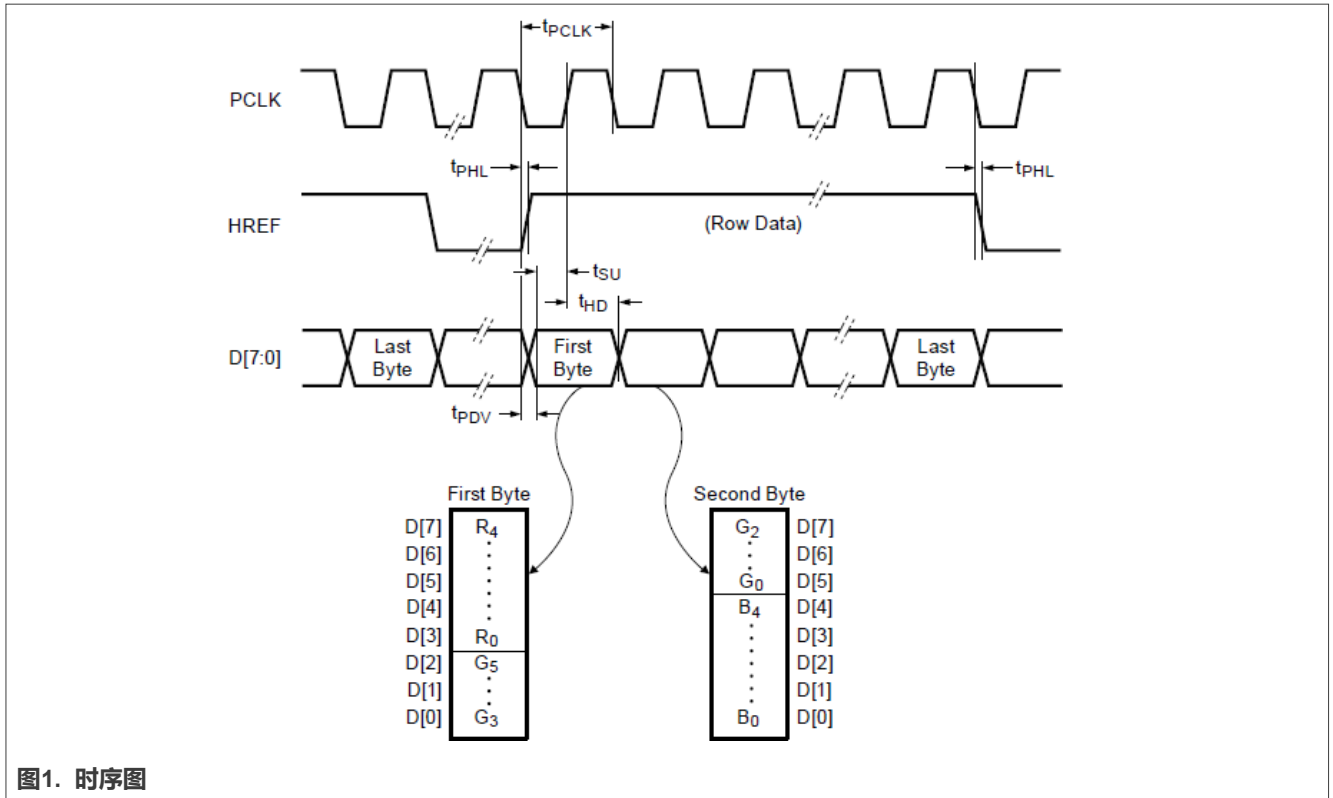


图1. 时序图

- 分辨率可配置为VGA (640x480)、QVGA (320x240)、QQVGA (160x120) 等。必须修改SmartDMA代码以匹配不同的分辨率。

7 引脚说明

本章对相关引脚进行介绍。

7.1 接口连接

SmartDMA可以访问MCXN947中的32个GPIO引脚，同时并行使用8个GPIO引脚来读取摄像头的数据。MCU使用I2C来配置摄像头。VSYNC、HREF、PCLK和PWDN信号可由SmartDMA控制。摄像头模块可以由MCU电路板的3.3 V供电。

7.2 接口要求

D0-D7应连接到SmartDMA D0-D7，以进行数据的字节读取。SIOC和SIOD应连接到MCU的I2C接口进行配置。VSYNC、HREF和PCLK应连接到Port0的引脚，以触发SmartDMA。XCLK应连接到MCU的时钟输出引脚。

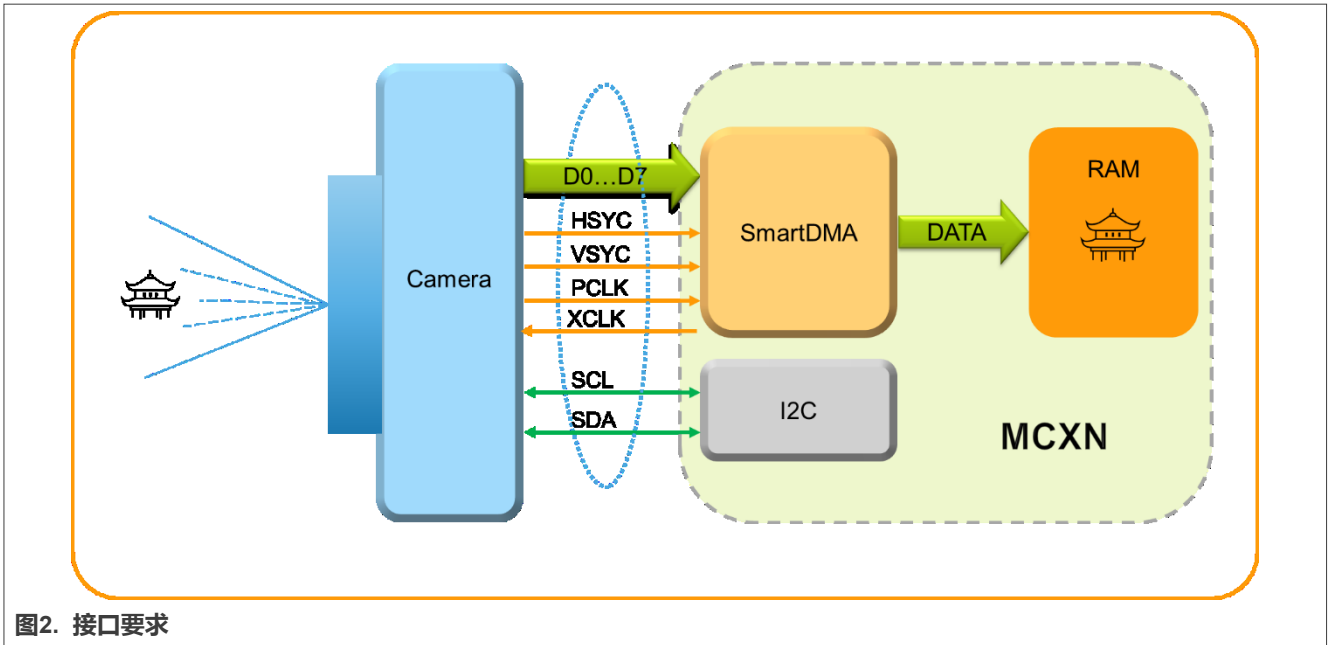


图2. 接口要求

8 软件

本章对软件进行介绍。

8.1 演示代码

本应用笔记随附的软件代码名为“frdm_mcxn947_SmartDMA_camera”。该代码是使用MCUXpresso IDE生成的。

8.2 SmartDMA的代码数组

作为精简指令集的内核，SmartDMA必须执行汇编指令才能执行任务。为了降低用户的学习难度，本应用笔记中的SmartDMA代码以数据数组的形式呈现。只需设置数组的SmartDMA地址即可执行任务。由于存在着绝对地址跳转指令，因此代码数组必须放置在指定的RAM地址。该代码数组可以轻松集成到Keil IDE、MCUXpresso IDE和IAR IDE中。在本应用笔记的软件代码中，该数组的名称为“s_smartdmaCameraFirmware”。

8.3 API例程

API例程的主要用途包括：

- 启用SmartDMA时钟。
- 将IO口配置为摄像头接口。
- 初始化I2C接口。
- 使能SmartDMA中断，以告知Arm内核数据已准备就绪。
- 初始化并启动SmartDMA。
- LCD初始化。
- LCD刷新。

8.4 API例程说明

表1. API例程

例程	说明
SMARTDMA_InitWithoutFirmware	初始化SmartDMA
SMARTDMA_InstallFirmware	安装固件
SMARTDMA_InstallCallback	安装完整的回调函数
SMARTDMA_Boot	启动SmartDMA，运行程序
SMARTDMA_Deinit	取消SmartDMA的初始化
SMARTDMA_Reset	复位SmartDMA
SMARTDMA_HandleIRQ	SmartDMA IRQ
SmartDMA_camera_callback	SmartDMA中断回调
lcd_impl_init	LCD驱动程序初始化
st7796_lcd_init	LCD初始化
st7796_lcd_load	LCD刷新
Ov7670_Init	摄像头模块初始化

8.5 详细的代码说明

本节对代码进行详细的介绍。

8.5.1 系统时钟

当每个像素的边缘出现时，SmartDMA需要在有限的时间内存储数据。若引擎的时钟频率越高，则花费的时间就越少。在此解决方案中，当引擎运行时，系统时钟必须设置为150 MHz。配置系统时钟的代码如下：

```
BOARD_BootClockPLL150M();
```

8.5.2 I2C接口

对于MCXN947，“flexcomm7”用作I2C接口来初始化摄像头。

8.5.3 引脚功能

表2. 引脚功能

MCXN947	功能编号	输入/输出	说明
P1_4	7 (摄像头功能)	输入	摄像头引擎功能DATA0
P1_5	7 (摄像头功能)	输入	摄像头引擎功能DATA1
P1_6	7 (摄像头功能)	输入	摄像头引擎功能DATA2
P1_7	7 (摄像头功能)	输入	摄像头引擎功能DATA3

表2. 引脚函数 (续)

MCXN947	功能编号	输入/输出	说明
P3_4	7 (摄像头功能)	输入	摄像头引擎功能DATA4
P3_5	7 (摄像头功能)	输入	摄像头引擎功能DATA5
P1_10	7 (摄像头功能)	输入	摄像头引擎功能DATA6
P1_11	7 (摄像头功能)	输入	摄像头引擎功能DATA7
P0_4	0 (GPIO功能)	输入	GPIO作为VSYNC输入
P0_11	0 (GPIO功能)	输入	GPIO作为HSYNC输入
P0_5	0 (GPIO功能)	输入	GPIO作为像素时钟输入
P2_2	CLKOUT功能	输出	摄像头的时钟输入
P3_2(FLEXCOM7)	I2C功能	输入/输出	I2C_SDA
P3_3(FLEXCOM7)	I2C功能	输出	I2C_SCL

注：在PORT寄存器中，摄像头功能模式为7。

8.5.4 LCD功能

LCD用于实时显示摄像头拍摄的影像。FlexIO和DMA用于驱动LCD。当SmartDMA完成将摄像头数据在缓冲区的存储后，它会向Arm内核发出中断触发信号。在Arm内核中断中，设置完成标志位。一旦设置了该标志位，就会调用“st7796_lcd_load”例程来刷新LCD。有关LCD实现的更多信息，请参阅其他应用笔记。

8.5.5 SmartDMA_camera_callback

与其他外设处理程序一样，SmartDMA处理程序在SmartDMA完成存储操作后由Arm内核实现。

在初始化阶段，已安装了处理程序的回调例程。在回调例程中，某个标志位会被设为1。在“1”例程中，当标志位变为逻辑1时，可以允许刷新操作。

8.5.6 数据缓冲区

对于一帧视频（480x320分辨率），必须有300 kB的RAM空间，而MCXN947大约有512 kB的RAM。双缓冲区对于全帧模式是不可能的。因此仅有一个缓冲区被使用。由于LCD刷新时间（16 ms）短于数据存储时间（66 ms），所以Arm总是在SmartDMA存储数据之前读取LCD的刷新数据。LCD始终根据RAM中存储的数据来显示前一帧数据。因此，媒体数据不会丢失。

8.5.7 时序

LCD始终显示摄像头的前一帧数据。在显示之前，存储的数据必须由协处理器进行优化，以交换每个像素的高低字节。由于LCD模块显示的速度比摄像头接口读取数据的速度快，因此在该应用中使用了单个数据缓冲区。在存储当前帧数据的同时，LCD会显示前一帧数据。

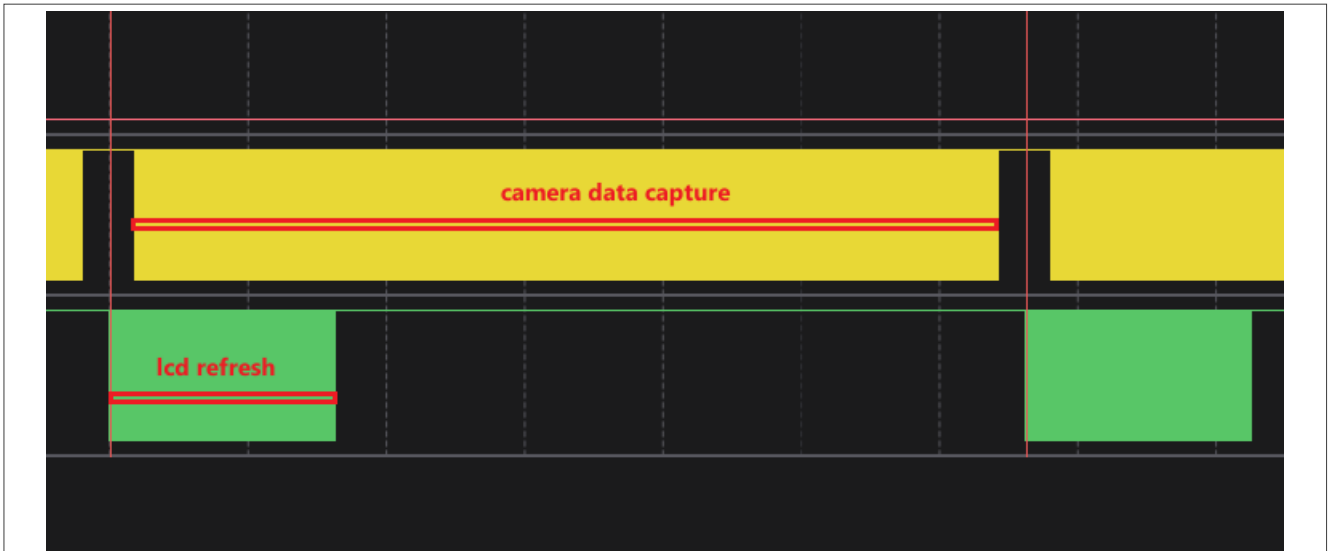


图3. 时序

8.5.8 演示

1. 构建并编译工程。
2. 将USB电缆的一端插入PC的USB端口，另一端插入FRDM-MCXN947电路板的调试器（debug link）端口，然后将固件下载到MCU。断开电缆与FRDM-MCXN947电路板的连接。
3. 参照图4中的接口连接，将摄像头连接到MCU。
4. 将LCD面板连接到FRDM-MCXN947电路板上的FlexIO LCD端口。
5. LCD显示来自摄像头的视频帧，参见图4。

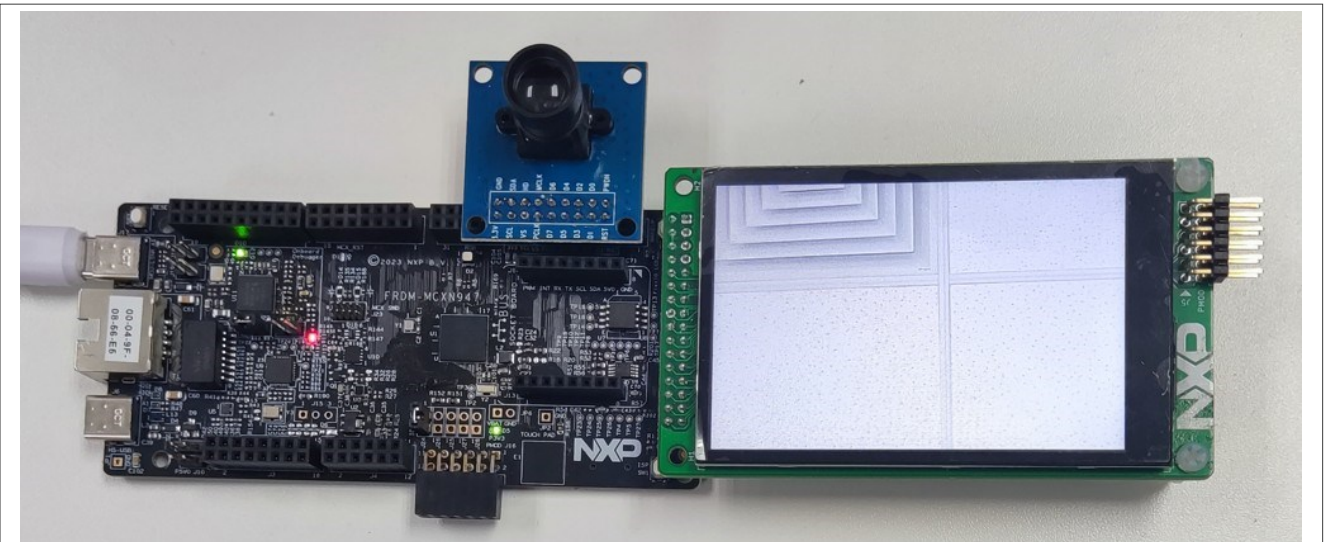


图4. 演示

9 关于本文中源代码的说明

本文中所示的示例代码具有以下版权和BSD-3-Clause许可：

2024年恩智浦版权所有。在满足以下条件的情况下，允许以源代码和二进制文件的形式重新分发和使用本源代码（无论是否经过修改）：

- 重新分发源代码必须保留上述版权声明、这些条件和以下免责声明。
- 以二进制文件形式重新分发时，必须在文档和/或随分发提供的其他材料中必须复制上述版权声明、这些条件和以下免责声明。
- 未经事先书面许可，不得使用版权所有者的姓名或参与者的姓名为本软件的衍生产品进行背书或推广。

本软件由版权所有者和参与者“按原样”提供，不承担任何明示或暗示的担保责任，包括但不限于对适销性和特定用途适用性的暗示保证。在任何情况下，无论因何种原因或根据何种法律条例，版权所有或参与者均不对因使用本软件而导致的任何直接、间接、偶然、特殊、惩戒性或后果性损害（包括但不限于采购替代商品或服务；使用损失、数据损失或利润损失或业务中断）承担责任，无论是因合同、严格责任还是侵权行为（包括疏忽或其他原因）造成的，即使事先被告知有此类损害的可能性也不例外。

10 修订历史

[表3](#)汇总了对本文档所做的更改。

表3. 修订历史

版本号	发布日期	说明
第1版	1月18日	首次对外发布

目录

1	介绍.....	2
2	目标应用.....	2
3	摄像头接口简介.....	2
4	MCX Nx4x MCU简介.....	2
5	摄像头接口的特性.....	2
6	功能描述.....	3
6.1	SmartDMA的工作方式.....	3
6.2	摄像头的时钟源.....	3
6.3	MCU8080 LCD接口.....	3
6.4	I2C接口.....	3
6.5	内存的占用.....	3
6.6	其他支持的摄像头模块.....	3
7	引脚说明.....	4
7.1	接口连接.....	4
7.2	接口要求.....	4
8	软件.....	5
8.1	演示代码.....	5
8.2	SmartDMA的代码数组.....	5
8.3	API例程.....	5
8.4	API例程说明.....	6
8.5	详细的代码说明.....	6
8.5.1	系统时钟.....	6
8.5.2	I2C接口.....	6
8.5.3	引脚功能.....	6
8.5.4	LCD功能.....	7
8.5.5	SmartDMA_camera_callback.....	7
8.5.6	数据缓冲区.....	7
8.5.7	时序.....	7
8.5.8	演示.....	8
9	关于本文中源代码的说明.....	8
10	修订历史.....	9