



内容

1 用途和范围.....	1
2 UART 主机接口.....	1
2.1 UART 要求.....	1
2.2 UART 命令.....	3
2.3 UART 读取命令.....	5
2.4 UART 命令构建器工具.....	6
3 microSD 卡播放.....	6
3.1 microSD 卡概述.....	6
3.2 microSD 卡格式和配置.....	7
3.3 microSD 卡播放操作.....	8
4 图形库.....	8
4.1 图形库概述.....	8
4.2 初始动画.....	8
4.3 microSD 卡中的图像.....	9
4.4 启动界面图像.....	9
4.5 智能家居演示.....	9
5 DSI 模式.....	9
5.1 DSI 模式概述.....	9
6 修订历史记录.....	10

商标

所有商标均为其各自所有者的财产。

1 用途和范围

本指南文档介绍了 MSPM0G3507 MCU 及其与构成 DLPDLCR160CPEVM 的 DLPC3421 芯片组的交互。MCU 用作 DLPC 控制器的宿主。本指南介绍了此 MCU 支持的功能和特性。

2 UART 主机接口

2.1 UART 要求

UART 数字通信是一种控制器/外设通信链路，其中 MSPM0 仅是外设。控制器设置数据传输开始和结束的时间。在控制器发出命令之前，外设不会将数据发送回至控制器。UART 接口上的逻辑 1 值被定义为隐性值 (RXD 引脚上的弱上拉)。UART 接口上的逻辑 0 值被定义为显性值 (RXD 引脚上的强下拉)。

UART 异步模式接口设计用于 2400bps 至 115200bps 的运行数据速率。与 UART 接口运行相关的其他参数包括：

- 从 2.4kbps 到 115.2kbps 的波特率。默认为 9.6kbps
- 8 个数据位
- 1 个起始位
- 1 个停止位
- 无奇偶校验位
- 无流量控制
- 字段间等待时间 (仅 1 个停止位配置需要)

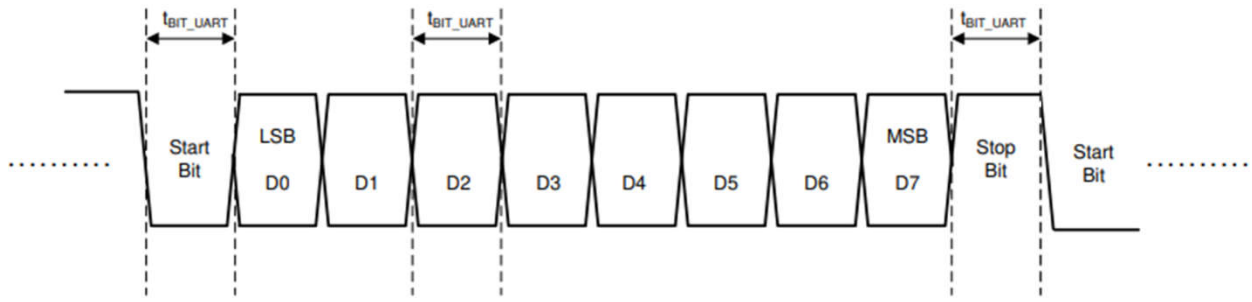


图 2-1. UART 异步接口位时序

数据和控制都采用小端字节序格式。数据以字节大小数据包形式通过 UART 接口进行传输。数据包字段的第一个位是起始位 (显性)。该字段的接下来 8 个位是 UART 接收器要处理的数据位。该字段的最后一个位是停止位 (隐性)。信息字节以及起始位和停止位组合在一起构成了一个 UART 字段。

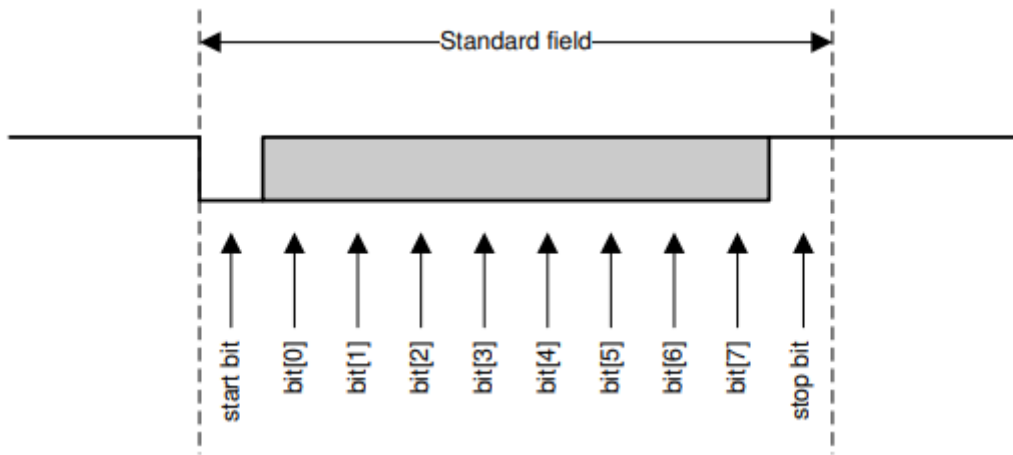


图 2-2. UART 接口数据包字段

TTL-232R-3V3 是 USB 转串行 (TTL 电平) 转换器电缆,可轻松地将 TTL 接口单元连接到 USB。此版本的 FTDI USB 转 TTL 串行适配器电缆具有配置为在 3.3V 电平下运行的 I/O 引脚。这是测试期间使用的一个 USB 转串行设备示例。



图 2-3. USB 转串行设备

2.2 UART 命令

一组字段构成一个传输帧。一个传输帧由在 UART 接口上完成一次传输操作所需的字段组成。图 2-4 展示了一个传输帧中数据传输操作的结构。

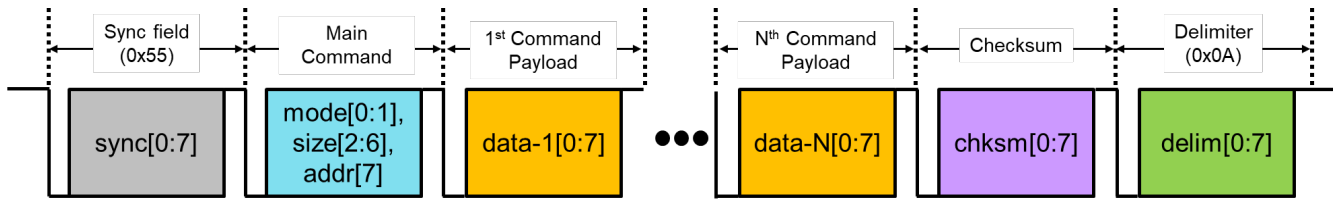


图 2-4. UART 接口传输帧

UART 事件必须依赖中断，而不是轮询，这样当控制器发送 UART 有效载荷时，MSPM0 会进入 UART 中断服务例程 (ISR) 来处理传入的有效载荷。UART 控制器总传入有效载荷：

字节编号	字节字段名称	值	说明
0	同步	0x55	同步字段是控制器发送的每个帧的第一个字段。 MSPM0 器件使用同步字段来确认控制器发送的帧的正确波特率。
1	主命令	Bit0-1：命令模式 Bit2-6：命令大小 Bit7：MSPM0 地址	命令模式： 命令字段是控制器发送的每个帧中的第二个字段。命令字段包含有关如何处理数据以及将数据发送到特定 MSPM0 器件的位置的说明。命令字段还可指示 MSPM0 器件在读取操作期间将数据发送回控制器。要发送的数据字段的数量也由命令字段中的命令决定。 0x00 = DLPC3421 发出命令 0x01 = SSD1963 发出命令 0x02 = MSPM0 发出命令 0x03 = 保留/TI 内部命令 命令大小： 5 个位，用于指示 (主命令) 字节之后有效命令预期的写入字节数 (最多 32 个字节)，但计数中不包括校验和。MSPM0 预计接收到给定数量的字节以确定 UART 控制器传输完成，如果在接收到最后一个字节后的 100ms 内未收到预期数量的字节，则会发生超时以退出 UART 接收函数。 MSPM0 地址： 在 MSPM0 器件中，命令字段的最后一位保留用于 UART 地址信息。命令字段中的地址信息会与 MSPM0 源代码中定义的 UART_ADDR 参数进行比较。接收到命令字段后，MSPM0 器件会检查自地址是否与接收到的地址匹配。如果地址匹配，该器件会根据接收到的命令执行操作。如果地址不匹配，该器件会忽略接收到的帧。MSPM0 器件的默认预编程地址为 0。
2 ... n	命令模式有效载荷	n = 最多 32 个字节	(有关详细信息，请参阅下面的各个命令模式有效载荷表。)
2 ... n+1	校验码	1 字节	Checksum8 模数 256：字节总和和 % 256 同步字段不包含在校验和计算中。
1	分界符	0x0A	定界符字节值为 0x0A，与 ASCII 中的换行符相同，表示整个 UART 有效载荷的末尾。

命令模式：

0x00 = DLPC3421 命令子有效载荷：

字节编号	字节字段名称	值	说明
0	地址 (芯片地址)	1 字节	写入 = 0x36 或 0x3A 读取请求 = 0x36 或 0x3A 读取响应 = 0x37 或 0x3B
1	读取数据长度	1 字节	支持的最大响应长度为 255 (理论值)，但限制为 29 (实际值基于有效载荷结构)。对于写入命令，该值为“无关”。
2	子地址 (命令)	1 字节操作码	支持《DLPC3421 编程人员指南》中列出的所有命令类型，其中包括： <ul style="list-style-type: none"> • 常规运行 • 照明控制 • 图像处理控制 • 常规设置 • DSI 命令 • 管理事务 • 闪存更新
3 ... n	其余数据字节 (写入参数)	n = 最多 8 个或 29 个字节	所有命令所需的写入参数字节数不超过 8 个。根据写入图像裁剪 (10h) 命令，参数字节的最大数量限制为 8。不包括对多达 1024 个字节写入闪存开始 (E1h) 和写入闪存继续 (E2h) 命令的完全支持；由于 UART 控制器有效载荷的主命令大小为 32 个字节，因此这些命令限制为 29 个字节。

有关更多详细信息，请参阅 [DLPC3421 软件编程人员指南](#)。

0x01 = SSD1963 命令子有效载荷：

字节编号	字节字段名称	值	说明
0	命令	1 字节十六进制代码	支持 SSD1963 LCD 控制器数据表中列出的所有命令。
1 ... n	写入参数	n = 最多 9 个字节	所有命令所需的写入参数字节数不超过 9 个。根据 set_dbc_th (D4h) 和 get_dbc_th (D5h) 命令，参数类型的最大数量限制为 9。

有关更多详细信息，请参阅 [SSD1963 LCD 控制器数据表](#)。

0x02 = MSPM0 命令子有效载荷：

字节编号	字节字段名称	值	说明
0-3	MSPM0 32 位地址	4 字节	MSPM0 使用 32 位 (4 字节) 地址长度。字节 0 为 MSB，字节 3 为 LSB。
4-7	寄存器读取数据	4 字节	每个寄存器地址存储一个 32 位数据长度值。字节 4 为 MSB，字节 7 为 LSB。

有关更多详细信息，请参阅 [MSPM0G3507 数据表](#)。

0x03 = TI 内部命令子有效载荷：

字节编号	字节字段名称	值	说明
0 ... n	保留	n = 最多 32 个字节	TI 内部功能

2.3 UART 读取命令

如果 MSPM0 要将数据返回到外部主机控制器 (PC GUI)，则 UART 外设的总传出有效载荷必须与传入有效载荷格式相匹配。传入有效载荷的独有数据会回送至控制器。PC GUI 可对读取/返回数据的预计时间进行编程，并在数据返回到控制器期间向外设发送 null (0xFF) 数据。

UART 外设总传出有效载荷：

字节编号	字节字段名称	值	说明
0	同步	0x55	同步字段是控制器发送的每个帧的第一个字段。MSPM0 器件使用同步字段来确认控制器发送的帧的正确波特率。
1	主命令 (控制器回波)	1 字节	请参阅“传入有效载荷说明”。
2 ... n	命令模式有效载荷	n = 最多 32 个字节	有关详细信息，请参阅下面的各个命令模式有效载荷表。
2 ... n+1	校验码	1 字节	Checksum8 模数 256：字节总和 % 256 同步字段不包含在校验和计算中。
1	分界符	0X0A	定界符字节值为 0x0A，与 ASCII 中的换行符相同，表示整个 UART 有效载荷的末尾。

命令模式：

0x00 = DLPC3421 命令子有效载荷：

字节编号	字节字段名称	值	说明
0	地址 (芯片地址) (控制器回波)	1 字节	写入 = 0x36 或 0x3A 读取请求 = 0x36 或 0x3A 读取响应 = 0x37 或 0x3B
1	读取数据长度 (控制器回波)	1 字节	支持的最大响应长度为 255 (理论值)，但限制为 29 (实际值基于有效载荷结构)。对于写入命令，该值为“无关”。
2	子地址 (命令) (控制器回波)	1 字节	支持 DLPC3421 软件编程人员指南 中列出的所有命令类型，其中包括： <ul style="list-style-type: none"> • 常规运行 • 照明控制 • 图像处理控制 • 常规设置 • DSI 命令 • 管理事务 • 闪存更新
3 ... n	读取参数	n = 最多 29 个字节	根据命令“读取序列头属性” (26h)，返回参数的最大数量预计为 29 个字节。或者，可以考虑完全支持“读取闪存开始” (E3h) 和“读取闪存继续” (E4h) 命令的最多 256 个读取字节；由于 UART 控制器有效载荷的主命令大小为 32 个字节，因此这些命令限制为 29 个字节。

有关更多详细信息，请参阅 [DLPC3421 软件编程人员指南](#)。

0x02 = MSPM0 命令子有效载荷：

字节编号	字节字段名称	值	说明
0	命令 (控制器回波)	1 字节十六进制代码	请参阅“传入有效载荷说明”。
1 ... n	读取参数	n = 最多 9 个字节	所有命令所需的读取参数字节数不超过 9 个。根据 set_dbc_th (D4h) 和 get_dbc_th (D5h) 命令，参数类型的最大数量限制为 9。

有关更多详细信息，请参阅 [SSD1963 LCD 控制器数据表](#)。

0x02 = MSPM0 命令子有效载荷：

字节编号	字节字段名称	值	说明
0-3	MSPM0 32 位地址 (控制器回波)	4 字节	MSPM0 使用 32 位 (4 字节) 地址长度。字节 0 为 MSB，字节 3 为 LSB。
4-7	寄存器读取数据	4 字节	每个寄存器地址存储一个 32 位数据长度值。字节 4 为 MSB，字节 7 为 LSB。

有关更多详细信息，请参阅 [MSPM0G3507 数据表](#)。

0x03 = TI 内部命令子有效载荷：

字节编号	字节字段名称	值	说明
0 ... n	保留	n = 最多 32 个字节	TI 内部功能

2.4 UART 命令构建器工具

使用 [DLPDLCR160CPEVM_Command_Builder.xlsm](#) 可以为主机控制器构建低级 UART 命令有效载荷。

蓝色单元格是用户输入，而黄色单元格是固定的或自动生成的。

	A	B	C	D	E	F	G	H	I	J	K	
1			Main Command					command mode payload				
2			bit 0-1 : Command mode		bit 2-6 : Command size		bit 7 : MSPM0 Address					
3			Select value	-				37 01 0C 37		Checksum	Delimiter	
4	Sync		0	DLPC3421 Command								
5			1	SSD1963 Command								
6			2	MSPM0 Command								
7			3	Reserved / TI Internal Command								
8				0		4	0					
9	55		10							8B	0A	
10												
11												
12	Entire UART command to MSPM0											
13	55 10 37 01 0C 37 8B 0A											
14												
15												
16												
17												
18												
19	User input											
20												
21	Auto calculation											
22	(do not change)											
23												

3 microSD 卡播放

3.1 microSD 卡概述

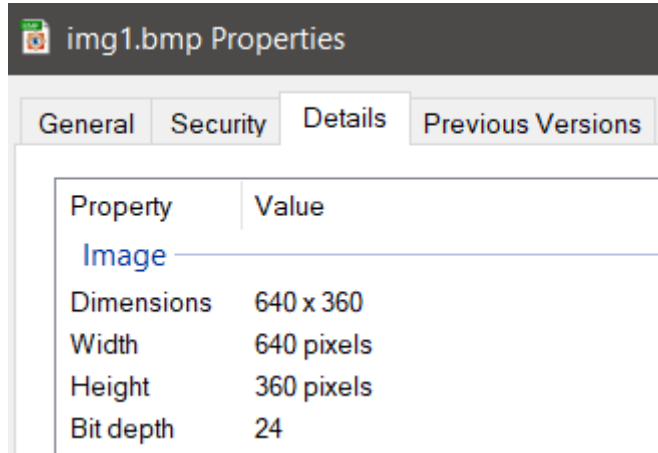
该 EVM 设计用于从 microSD 卡播放图像文件。从 microSD 卡播放的速度大约为每秒 1 帧。

3.2 microSD 卡格式和配置

microSD 卡应格式化为 FAT32，以支持高达 32GB 的 SDHC 容量。更高内存密度的 SD 卡必须从 exFAT 格式化为 FAT32。

每个图像都必须是 nHD 分辨率为 640×360 的 24 位位图 (.BMP) 文件。PNG、JPG 和其他图像文件类型在此 EVM 上不受支持，必须在加载到 microSD 卡之前转换为 BMP 文件。每个生成的 .BMP 文件必须至少为 691 200 字节，并且不能使用任何形式的压缩。

兼容 BMP 图像文件的图像属性示例：



microSD 卡中需要一个名为“conf.txt”的文本文件。conf.txt 文件为每个图像指定一组参数，例如图像文件名、绘制坐标、背景颜色和显示时间。请勿更改 conf.txt 文件的名称，因为 MSPM0 会在启动时专门查找 conf.txt 文件。如果未检测到此文件，或者如果 conf.txt 文件的内容错误，microSD 卡播放将失败。

配置 (conf.txt)文件参数的详细信息：

关键字	参数	说明
boot_up_mode	1	1：启动界面图像 2：SSD (RGB_parallel) 3：DSI
number_of_sdcard_images	>=1	支持的最大图像数量由 512 字符数组除以整个 image_names 行字符长度得出。必须至少为“1”个字符。
image_names	image1_name.bmp, image2_name.bmp, ...	图像名称必须包含“.bmp”扩展名。
sequence_repeat	1	值必须为“1”
number_of_sequence	3 × number_of_sdcard_images	序列编号的指示符。每个图像需要 3 个序列，因此 number_of_sequence 由 #_of_images 乘以 3 来确定。
rgb_fill	R,G,B	每个 R-G-B 值都可以接受一个 8 位值来指定从 SD 卡绘制图像之前显示的特定背景填充颜色。
image	image_index, x_location, y_location	image_index 是要从 image_nams 列表加载的元素。 x/y_location 是指开始绘制位图图像的坐标 (通常为 0,0)
delay	1000	从显示 SD 卡图像的最后一个像素开始到显示图像的延迟，以毫秒为单位。
rgb_fill	R,G,B	
image	image_index, x_location, y_location	
delay	1000	

这是一个包含四个图像的 conf.txt 文件内容示例：






```

boot_up_mode 2
number_of_sdcards_images 4
image_names img1.bmp,img2.bmp,img3.bmp,img4.bmp
sequence_repeat 1
number_of_sequence 12
rgb_fill 0,0,0
image 1,0,0
delay 1024
rgb_fill 0,0,0
image 2,0,0
delay 1024
rgb_fill 0,0,0
image 3,0,0
delay 1024
rgb_fill 0,0,0
image 4,0,0
delay 1024

```

由于 `conf.txt` 文件变量区分大小写，因此 `conf.txt` 文件的所有输入都必须采用小写字母。

文件夹结构要求 `conf.txt` 文件和所访问的 `imgX.bmp` 文件都存储在 microSD 卡的根文件夹中。下面是一个包含四个图像示例的文件夹结构示例：

Name	Date	Type	Size
<input checked="" type="checkbox"/>  conf.txt	11/27/2023 11:50 PM	Text Document	1 KB
 img1.bmp	10/25/2023 2:22 AM	FastStone BMP File	676 KB
 img2.bmp	10/25/2023 2:22 AM	FastStone BMP File	676 KB
 img3.bmp	10/25/2023 2:22 AM	FastStone BMP File	676 KB
 img4.bmp	10/25/2023 2:23 AM	FastStone BMP File	676 KB

3.3 microSD 卡播放操作

若要启用从 SD 卡播放，请按照以下步骤操作：

1. 断开 EVM 的电源。
2. 将 SD 卡插入 EVM。
3. 将 EVM 插入 USB-C 电源。
4. 将 PROJ_ON 开关切换至导通状态。
5. 在显示菜单上，使用物理按钮导航至并选择 microSD 卡选项。注意：点击按钮可在各种菜单选项之间切换。按住按钮至少 2 秒钟可以选择操作模式。
6. 要在显示最后一幅图像之前退出 SD 卡播放模式，请点击按钮。

4 图形库

4.1 图形库概述

MSPM0G3507 支持使用 Rinky-Dink Electronics 提供的图形库。这使得该 EVM 可以投影由 MCU 创建的图像。

4.2 初始动画

当 DLPC3421 成功加电并开始显示内容时，将显示内容的脚本化演示。该演示包括静态图像、动画和菜单，其中两个元素使用本节介绍的图形库。

函数 **StateplayLogo()** 调用函数 **PlayLogo()**，而后者则创建动画，然后显示图像。接下来，使用 **SetupMainMenu()** 函数创建主菜单 GUI。此函数定义显示的文本，并包含直观定义菜单创建方式的函数。这包括形状和颜色。

4.3 microSD 卡中的图像

如节 3 上所述，此 EVM 支持显示 SD 卡中的图像。长按主菜单 GUI 上的按钮来选择 microSD 卡选项。通过长按按钮，输入选择语句 **St_MMENUSDCARD**，并开始设置 EVM 以显示 microSD 卡中的图像。每次读取并显示一个图像，直到显示所有指定的图像。完成此操作后，将出现一条消息来提供进一步的说明。

microSD 卡的必须完全按照节 3 中所述进行格式化，此功能才能正常运行。

4.4 启动界面图像

MSPM0 演示支持显示存储在 EVM 上 DLPC3421 控制器内的启动界面图像。

从主菜单 GUI 开始，用户可以选择闪存图像选项。记录了长按操作后，MSPM0 将进入语句 **St_MMENUFLASH**。此选择语句将创建按钮选择动画，更改输入源选择，并显示启动界面图像。该函数将迭代选择启动界面图像并向用户显示。此过程完成后，显示屏上将显示一条带有说明的消息。按照这些说明返回主菜单。

4.5 智能家居演示

智能家居 GUI 演示完全由 MSPM0 和图形库创建，并未使用任何来自 microSD 卡或固件映像中的图像。

在主菜单上选择“Smart Home demo”会调用选择语句 **St_MMENUSH** 并设置 **StateNow = St_DEMO**。这会调用 **ExecMain()** 函数和 **StateDemo()** 函数。状态演示函数会调用演示中显示的多个图形相关函数。下面列出了相关函数：

- **SetupMenus()**：在显示屏右侧创建图标
- **SetupAlbums()**：在显示屏右上方创建图标
- **SetupWeather()**：在显示屏左下方创建天气显示
- **UpdateClock()**：在显示屏左上方创建时钟视觉效果
- **WeatherNow()**：在显示屏左下方创建天气显示

在显示屏右侧，用户可以使用 EVM 按钮选择四个图标。这些图标包含感叹号、复选标记、日历和房屋图标。使用按钮选择右侧的感叹号图标会调用函数 **MessagesNow()**。此函数会创建一张带有文本的图像。使用按钮选择复选标记会调用函数 **ToDoNow()**。此函数会生成一个动态的事项检查清单。使用按钮选择日历会调用函数 **CalendarNow()**。此函数会生成一张通用日历图像。使用按钮选择房屋图标会退出智能家居演示并返回到主菜单。

在图标之间切换或选择图标时，图标的动画和颜色由以下函数控制。

在图标之间移动和选择图标：

- **SelectHome(int move_button)**
- **SelectAttention(int move_button)**
- **SelectToDo(int move_button)**
- **SelectCalendar(int move_button)**

5 DSI 模式

5.1 DSI 模式概述

显示串行接口 (DSI) 是 DLPDL160CPEVM 上的 DLPC3421 控制器支持的视频接口。这是控制器与 SSD1963 和 MSPM0G3507 连接所用并行接口的替代方案。借助这种方法，该 EVM 能够保持较小的尺寸，因为使用的引脚较少，并且能够在 EVM 上显示外部视频内容。该 EVM 最多支持 4 个 DSI 数据通道。该 EVM 默认配置为 4 个通道。

有关更多信息，请参阅 [DLPC3421 显示控制器](#) 数据表。请查看数据表中的以下部分：DSI 输入数据和时钟表、GPIO 外设接口表和 DSI 主机时序要求。

DSI 设置和调试指南提供了更详细的 DSI 设置指南。该控制器具有一些与上述指南中提到不同的命令，必须执行这些命令才能使用 DSI。这些命令可以添加到 **Autoinit** 批处理文件中，也可以在控制器初始化后发送到设备。以下是要发送的命令示例。所示的命令顺序并不是发送这些命令的唯一方式。有关这些命令的更多详细信息，请参阅 **DLPC3421 软件编程人员指南**。

DsiPortEnable (D7h)

ImageCrop (10h)

InputImageSize (2Eh)

DsiHsClockInput (BDh)

InputSourceSelect (05h)

ExternalVideoSourceFormat (07h)

$$\text{DSI HS Clock} = \frac{\text{Frame Rate} \times V_{\text{Total}} \times H_{\text{Total}} \times \text{bus width}}{\# \text{ of lanes} \times 2 \times 1 \times 10^6} \text{MHz} \quad (1)$$

- $V_{\text{Total}} = V_{\text{Active}} + V_{\text{FrontPorch}} + V_{\text{BackPorch}} + V_{\text{Sync}}$
- $H_{\text{Total}} = H_{\text{Active}} + H_{\text{FrontPorch}} + H_{\text{BackPorch}} + H_{\text{Sync}}$
- 总线宽度：
 - RGB888 = 24
 - RGB666 = 18
 - RGB565 = 16

1. 将 EVM 连接到电源，并将电源开关置于断开位置 (HOST_IRQ 高电平)。
2. 将 DSI 前端连接到 EVM DSI 连接器。
3. 在 LP11 模式下配置 DSI 数据通道。请勿将数据从前端发送到 EVM。
4. 将电源开关移至接通位置并等待控制器完成初始化 (HOST_IRQ 低电平)。
5. 发送上述命令以在 EVM 上启用 DSI 并准备好接收信号。
6. 将视频数据从 DSI 机器发送到 EVM。

备注

本文档中的信息不能替代相应器件数据表中列出的规格。如果数据表与本文档之间存在差异，请以数据表为准。

6 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

日期	修订版本	注释
April 2024	*	初始发行版

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2024，德州仪器 (TI) 公司