AN12868

Camera Interface in LPC55(S)xx

Rev. 1.0 — 6 August 2025 | Application note

Document information

Information	Content
Keywords	AN12868, LPC55S69, LPC55xx, EZH, Camera
Abstract	This application note introduces a parallel interface for the camera solution for LPC55(S)xx. It includes the introduction of camera interface, features and API routines, and demo.

1 Introduction

This application note introduces a parallel interface for the camera solution for LPC55(S)xx. It includes the introduction of camera interface, features and API routines, and demo.

2 Target application

The camera interface can be used as an important part of camera usage as below:

- Object detection
- Gesture recognition
- Color recognition
- QR code scanning, and so on

3 Introduction of camera interfaces

A typical camera interface supports at least one parallel interface, although nowadays many camera interfaces begin to support the MIPI CSI interface.

The camera parallel interface consists of the following lines:

- Data line (D[0:11]):
  These parallel data lines carry pixel data. The data transmitted on these lines change with every Pixel Clock (PCLK).
- Horizontal Sync (HSYNC)
  This is a special signal that goes from the camera sensor. An HSYNC indicates that one line of the frame is transmitted.
- Vertical Sync (VSYNC)
  This signal is transmitted after the entire frame is transferred. This signal is often a way to indicate that one entire frame is transmitted.
- Pixel Clock (PCLK)
  This pixel clock changes on every pixel.
  The application note only focuses on the Digital-Video-Port (DVP) interface, which is a parallel interface.

4 Features of camera interface

- Supported formats (8-bit): RGB565
- Maximum image transfer rate: 30 fps for QVGA (320 × 240). For small RAM parts, reduce the size of image and frame rate.
- Camera module tested: OV7673
- Other camera modules can be supported as long as they provide the same signal timing.

5 Function description

5.1 Camera interface engine

There is a dedicated processor in the LPC55S69 which can handle the signals of the camera.

It reads the data from the camera and stores the data in the RAM which is accessed by the Arm core. Before using a dedicated processor, some configurations must be made, which includes pin configuration, clock enable, dedicated processor enable, interrupt enable, and so on.

5.2 Camera driver library

The instructions of a dedicated processor use the type of machine code. The code implements the function of the camera interface protocol and is released in lib. Some API routines are provided in this application. User can use API routines to initialize the engine and configure the pins.

5.3 LCD display

This document uses an LCD to display the video stream from the camera in real time. The high-speed SPI port is used for LCD driving port. The max speed on the SPI bus is 50 Mbit/s, so it can display the 320 × 240 resolution LCD up to 30 fps.

5.4 System clock

The camera engine needs a short time to store the data when every pixel edge comes. If the clock frequency of the engine is higher, the time cost is shorter. In this solution, the system clock must be set at 150 MHz when the engine is running. The code to configure the system clock is `BOARD_BootClockPLL150M()`.

5.5 Clock source of camera

The camera needs a 50 MHz clock source, which is provided by CLKOUT signal from MCU.

5.6 I^2C interface

The `flexcomm4` is used as I^2C function for initializing the camera before the video starts.

5.7 Memory usage

The 320 × 240 resolution picture requires 150 K bytes RAM space. The solution uses the space address from `0x20010000` to `0x20035800`.

Also, the instruction code of the camera engine must run in the RAM for high performance. This solution uses the space address from `0x20040000` to `0x20043FFF` to store camera engine code.

5.8 Other supported camera modules

Other camera modules can be supported as long as they provide the same signal timing.

- The camera module must be configured as RGB565 mode with the timing diagram, as shown in Figure 1.

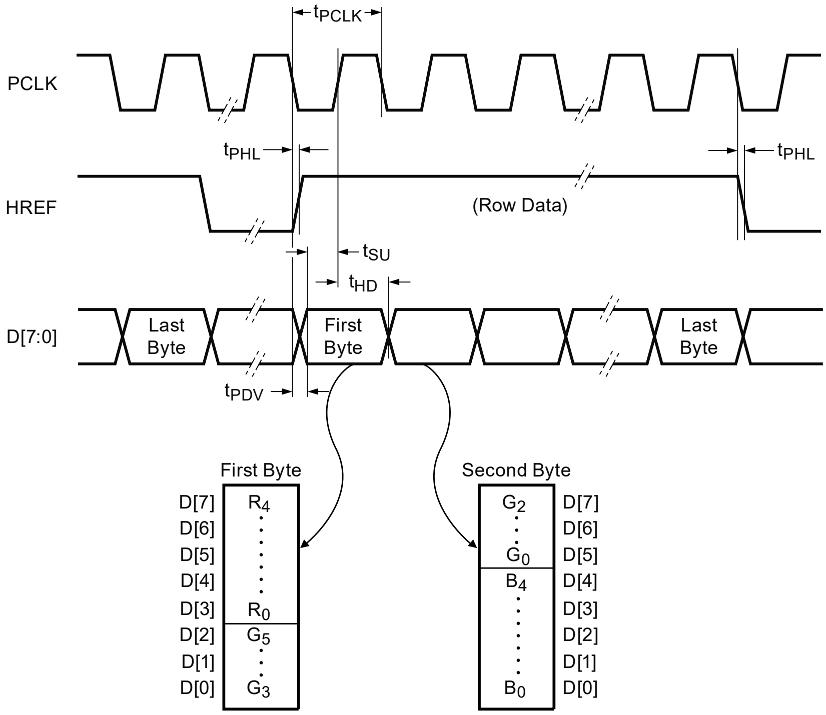


Figure 1. Camera module configuration

- The resolution must be configured as QVGA (320 × 240).

6 Pin description

6.1 Connection of interface

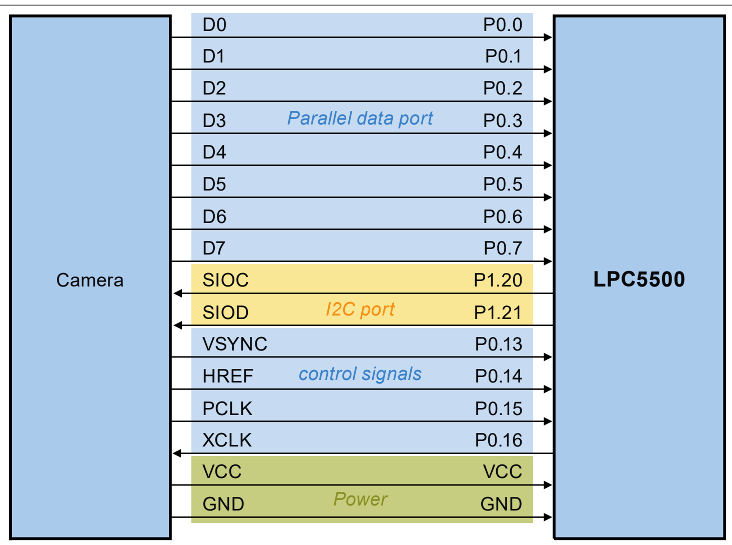


Figure 2. Connection of interface

Note: *Use `P0_16` as a `clkout` pin to provide clock source for the camera.*

6.2 Requirement of interface

- The D0-D7 must be connected to P0.0-P0.7 for byte reading the data.
- SIOC and SIOD must be connected to the I^2C interface of the MCU for configuration.
- The VSYNC, HREF, and PCLK must be connected to pins of `Port0`.
- XCLK must be connected to a clock output pin of the MCU.

7 Library and API routine

7.1 Library

The library is named by camera engine lib. It includes the instructions which have to be handled by a dedicated processor.

The library can support Keil, MCUXpresso IDE, but not IAR.

7.2 API routine

The main purposes of the API routines include:

- Enable the clock of the engine.
- Configure the IO as a camera interface function.
- Initial the I^2C interface.
- Enable the interrupt of the engine for telling the Arm core data is ready.
- Initialize and start the engine.

7.3 API routine description

Table 1. API routine

Routine	Description
`Reserved46_IRQHandler`	Interrupt routine for camera engine
`OV7670_Init`	Camera module initialization
`Camera_Init();`	Camera engine initialization
`Camera_Start();`	Camera engine start running
`LCD_Init();`	LCD module initialization
`LCD_Refresh();`	LCD refresh

7.4 Code detail description

7.4.1 System clock

The camera engine shares the system clock with the Arm core. To speed up the processing time, the system clock must be configured to 150 MHz. For those below 150 MHz, such as a 96 MHz system clock part, reduce the frequency of the pixel clock.

7.4.2 I^2C interface

The camera is configured through the I^2C interface, which can be connected with an I^2C peripheral port in the MCU.

7.4.3 Pin function

Table 2. Pin function

Pin	Function number	Input/output	Description
P0_0	15	Input	Camera engine function
P0_1	15	Input	Camera engine function
P0_2	15	Input	Camera engine function
P0_3	15	Input	Camera engine function
P0_4	15	Input	Camera engine function
P0_5	15	Input	Camera engine function
P0_6	15	Input	Camera engine function
P0_7	15	Input	Camera engine function
P0_18	15	Output	Camera engine function
p0_13	0	Input	GPIO as VSYNC input
P0_15	0	Input	GPIO as Pixel clock input
P0_16	2	Output	CLKOUT
P1_20	5	Input/output	FC4_I2C_SCL
P1_21	5	Output	FC4_I2C_SDA
P1_2	6	Output	LSPI_HS_SCK
P0_26	9	Output	LSPI_HS_MOSI
P1_3	6	Input	LSPI_HS_MISO
P1_1	5	Output	LSPI_HS_SSEL1
p1_11	0	Output	GPIO
P0_29	1	Input	FC0_USART_RX
P0_30	1	Output	FC0_USART_TX

P0_0 to P0_7 are the low 8 bits. They can be read by the engine at one read instruction, which only takes one system clock cycle.

P0_18 is set as camera engine function. It is operated by the engine directly such as set logic high level, clear zero, toggle and so on. P0_18 is toggled by the engine after every VSYNC edge.

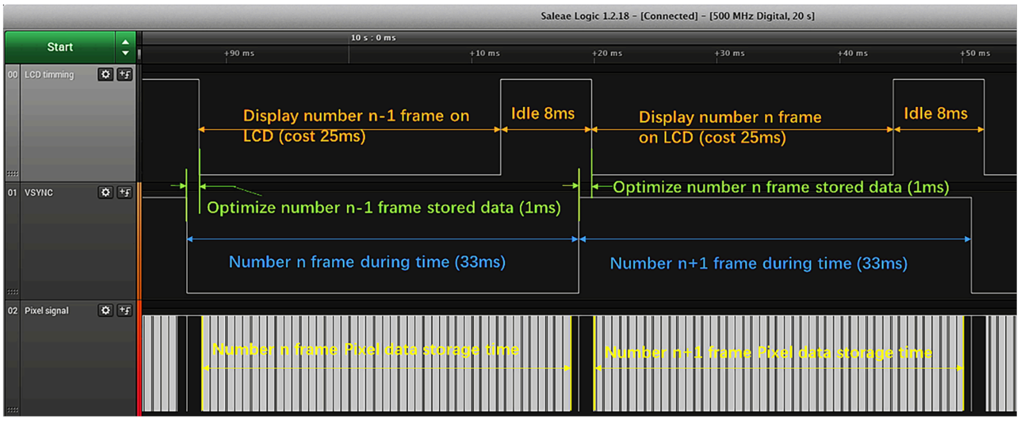
P0_13 and P0_15 are input function pins which can receive the VSYNC and pixel signals. The Pixel clock is 1/4 of the clock source and it is 12.5 MHz.

As clock output pin, P0_16 provides 50 MHz clock to camera as its clock source.

The LSPI_HS is high-speed SPI interface which is used to drive the LCD TFT screen.

7.4.4 LCD function

LCD is used to display the video of the camera in real time. High-speed SPI is used to drive the LCD. The routine named by LCD_Refresh() is a very high efficiency routine. It can refresh a 320 × 240 picture in about 24 ms, up to 40 fps. The bus clock of high-speed SPI can reach up to 50 MHz. The DMA is not used to drive the LCD. Instead, only software is used to drive LCD where the Arm core writes the SPI FIFO data write register once it is empty.



aaa-061955

Figure 3. LCD function

7.4.5 OV7670 camera module

The camera module is set as QVGA RGB565 mode. The 16-bit data is received with two pixel edges (high-byte first and low-byte follows).

At the beginning, the MCU initializes the camera through the I^2C interface. Then, the video stream data is stored in the RAM by the camera engine.

7.4.6 Reserved 46_IRQHandler

Same with other peripheral handler, the camera engine handler is implemented by the Arm core, once the engine finishes the storage operation.

In the handler, a flag is set as one. In the `while(1)` routine, the refreshing operation can be allowed when flag turns to logic one.

7.4.7 Data buffer

150 kB space is required for one frame of video and the LPC55S59 has about 300 kB RAM space. Double buffer is not possible. Only one buffer is used. Because the LCD refresh time (24 ms) is shorter than data storage time (33 ms), the Arm always reads the data for LCD refresh is earlier than storage operation by the engine. Therefore, the media data cannot be lost.

7.4.8 Timing

The LCD always displays the previous frame data from the camera. Before displaying, the data stored must be optimized by the dedicated processor for exchanging the high and low bytes of every pixel. Because the speed of LCD module displaying is higher than the speed of the camera interface reading data, the single data buffer is used in this application. While the current frame data is stored, the LCD displays the previous frame data.

8 Demonstration

1. Build and compile the project.
2. Plug one side of the USB cable in the PC USB port and another side in the debug link port in the EVK board, and then download the image in the MCU.
3. Connect camera to MCU by referring to the connection of interface in Section 6.1.
4. Connect the LCD panel with the Arduino port on the LPCXpresso55s69 EVK board.
5. Connect one side of USB cable to USB power socket and the other to 5 V Power only USB port P5.
6. The LCD displays the video frame from camera as shown in Figure 4.
   Warning:
   If the code cannot execute after downloaded, one of the reasons is that the MCU goes to ISP mode. The root cause is that the camera module generates logic level signal on P0.5 pin during MCU is under reset state. The solutions can be as below:
   a. *Disconnect the pin before reset and connect it after reset.*
   b. *Power down the camera module before reset and power up after reset.*

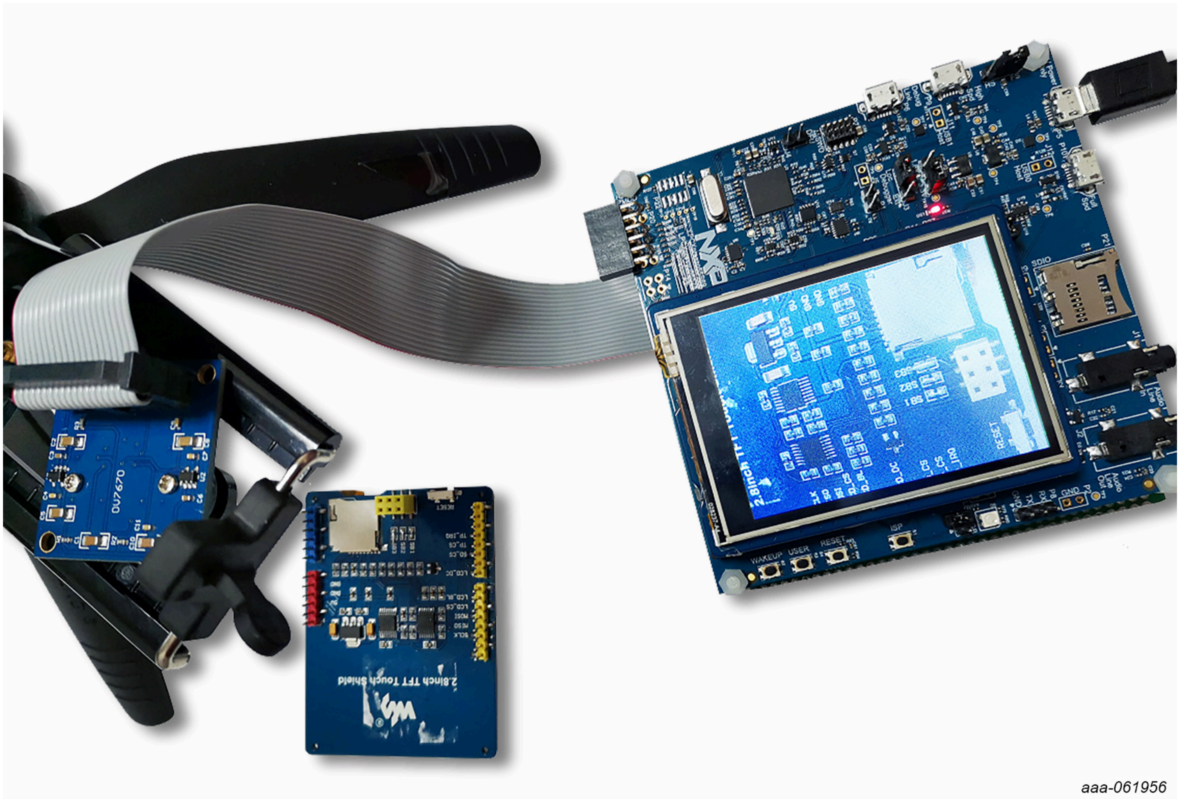


Figure 4. Video frame

9 Purchasing LCD module and camera module used in this demo

You can purchase the LCD board and camera module from the below links:

- https://www.waveshare.com/2.8inch-tft-touch-shield.htm
- http://www.waveshare.net/shop/2.8inch-TFT-Touch-Shield.htm
- https://www.amazon.com/gp/product/B07S66Y3ZQ/ref=ppx_od_dt_b_asin_title_s00?ie=UTF8&psc=1
- https://detail.tmall.com/item.htm?spm=a230r.1.14.10.54a37d76CA0csy&id=554248152327&ns=1&abbucket=12

10 Revision history

Table 3 summarizes the revisions to this document.

Table 3. Revision history

Document ID	Release date	Description
AN12868 v1.0	06 August 2025	Initial public release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, μVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

IAR — is a trademark of IAR Systems AB.

Microsoft, Azure, and ThreadX — are trademarks of the Microsoft group of companies.

Contents