



说明

此设计借助四分之一视频图形阵列 (QVGA) 显示面板来实现网络摄像头的运行，并利用嵌入式 Web 服务器进行远程监控。凭借用于 Web 服务器的 250KB 内存占用量，TM4C129x 微控制器 (MCU) 上有足够的内存供用户使用。

资源

<a href="#">TIDM-TM4C129CAMERA</a>	设计文件夹
<a href="#">TM4C129XNCZAD</a>	产品文件夹
<a href="#">SN74ALVC244</a>	产品文件夹
<a href="#">SN74LVC2G00</a>	产品文件夹
<a href="#">SN74LVC1G32</a>	产品文件夹
<a href="#">TPS61042</a>	产品文件夹
<a href="#">TPS62177</a>	产品文件夹
<a href="#">TPS2051B</a>	产品文件夹
<a href="#">LM4819</a>	产品文件夹
<a href="#">TMP100</a>	产品文件夹
<a href="#">REF3230</a>	产品文件夹
<a href="#">DK-TM4C129X</a>	工具文件夹

特性

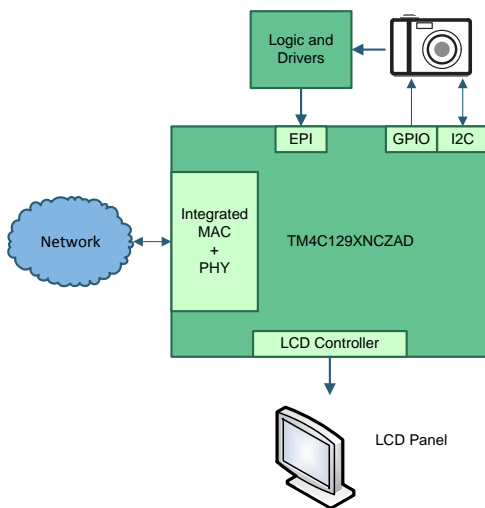
- 支持 QVGA 大小图像采集，使用外部外设接口 (EPI) 和直接存储器存取 (DMA) 传输来降低数据传输所需的 CPU 开销
- 采用嵌入式 Web 服务器，通过定期自动重新加载网页，对图像采集进行远程监控
- 摄像头接口支持在 LCD 面板上进行高达 30 帧/秒的显示
- 使用 TivaWare™ 软件图形库实现具有丰富图形的用户界面和体验

应用

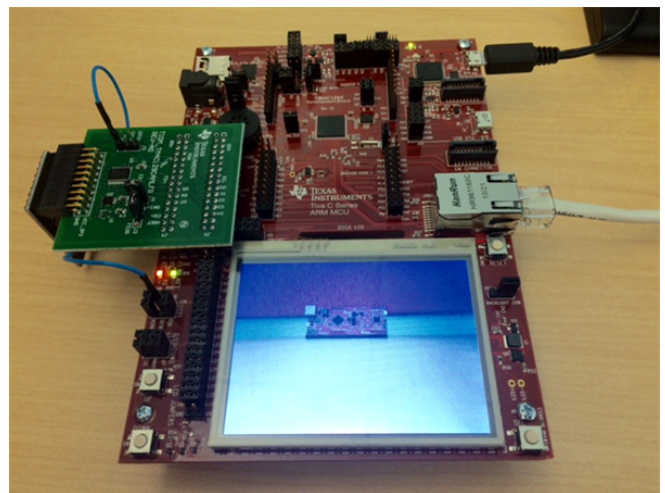
- 家居安全解决方案
- 视频门铃
- 网络摄像头
- 楼宇自动化和监控



咨询我们的 E2E 专家



Copyright © 2016, Texas Instruments Incorporated





该 TI 参考设计末尾的重要声明表述了授权使用、知识产权问题和其他重要的免责声明和信息。

## 1 System Description

The TM4C129x family of MCUs from Texas Instruments feature an integrated Ethernet PHY and MAC with cryptographic modules and a large number of serial and parallel interfaces for control and sensor data acquisition. Integrating a camera would require a dedicated camera interface or an external programmable logic like field programmable gate array (FPGA) and complex programmable logic device (CPLD) to interface a MCU to the camera. This design uses the EPI peripheral with a logic gate for protocol conversion without needing an external FPGA or CPLD. Using EPI with the logic gate significantly reduces the complexity of interfacing the camera to the TM4C129x device family while providing the advantage of integrated PHY that allows network connectivity for a host of video monitoring applications. The design files include schematics, bill of materials (BOM), layer plot, Altium files, Gerber files, and embedded firmware.

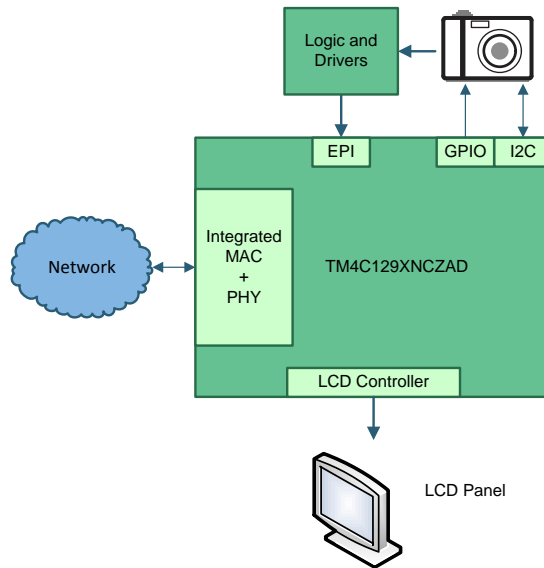
### 1.1 Key System Specifications

表 1. Key System Specifications

PARAMETER	SPECIFICATION
Image capture width	320 pixels
Image capture depth	240 pixels
Image capture format	16 BPP (RGB565 format)
Raw camera clock	20 MHz
Display panel width	320 pixels
Display panel depth	240 pixels
Image frame rate	30 frames per second

## 2 System Overview

### 2.1 Block Diagram



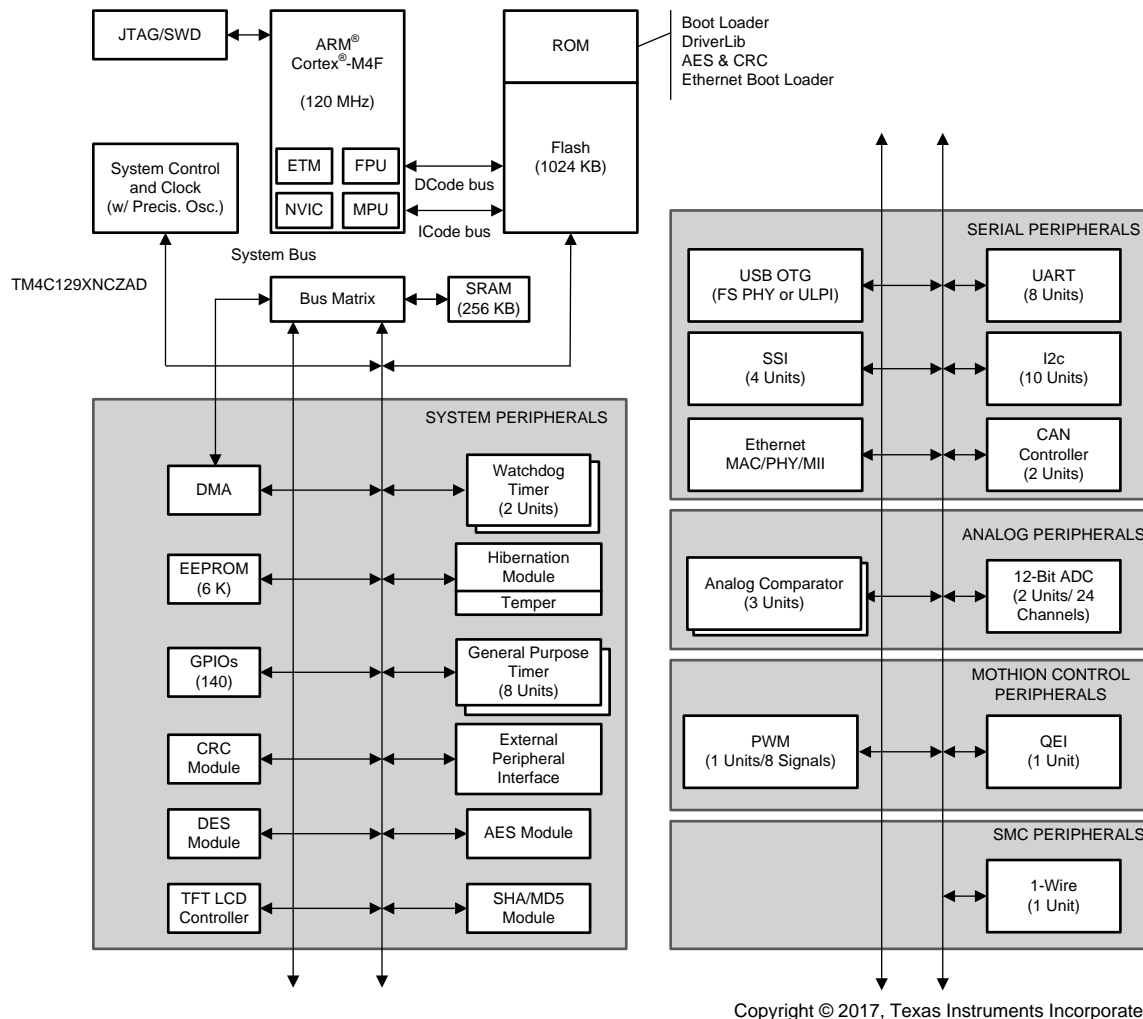
Copyright © 2016, Texas Instruments Incorporated

图 1. Camera Reference Design With TM4C129x Block Diagram

## 2.2 Highlighted Products

### 2.2.1 TM4C129XNCZAD

TM4C129XNCZAD is a 120-MHz high-performance MCU with 1-MB on-chip flash, 256-KB on-chip SRAM integrated Ethernet MAC PHY for connected applications and cryptographic modules such as advanced encryption standard (AES), data encryption standard (DES) and secure hash algorithm (SHA) for encryption, decryption, data validation, and authentication. The device has high-bandwidth interfaces like a memory controller (EPI) and a high-speed USB 2.0 digital interface. With integration of a number of serial communication peripherals, a 12-bit analog-to-digital converter (ADC) capable of up to 4 MSPS and motion control peripherals. The device provides a unique solution for a variety of applications ranging from industrial communication equipment to smart energy and smart grid applications.



Copyright © 2017, Texas Instruments Incorporated

**图 2. TM4C129XNCZAD MCU High-Level Block Diagram**

### 2.2.2 SN74ALVC244

The SN74ALVC244 is organized as two four-bit line drivers with separate output-enable (OEn) inputs. When OEn is low, the device passes data from the A inputs to the Y outputs. When OEn is high, the outputs are in the high-impedance state. This device provides extremely low-propagation delay of 2.8 ns (maximum) when operating at 3.3 V.

### 2.2.3 SN74LVCG00

The SN74LVCG200 is a dual two-input positive NAND gate designed for low-propagation delay over a wide supply range of 1.65 to 5 V.

## 3 System Design Theory

This section will elaborate how the TM4C129x MCU is interfaced to the external camera module using the EPI module.

### 3.1 Camera Module Interface

The camera module interface used for this reference design is the OV9655; however, other camera modules with similar data, configuration, and control interfaces may be used.

#### 3.1.1 Camera Data Interface

The OV9655 has an eight-bit data interface, two additional control signals, and a pixel clock that are provided by the camera. The data and the control signals are timing referenced with respect to the pixel clock. The camera provides the following signals.

1. Pixel clock output
2. Vertical sync output to indicate end of the frame
3. Horizontal reference output to indicate valid data for pixels in a row
4. Eight-bit data output

The following images illustrate the data sent by the camera module for every line (see 图 3) for a 16-BPP QVGA resolution image. The camera data and HREF signal are launched on the falling edge of the pixel clock. The data is valid on every rising edge for sampling by the MCU only when the HREF is logic high. Pixel position R4 and G2 correspond to MSB and pixel position G3 and B0 correspond to LSB.

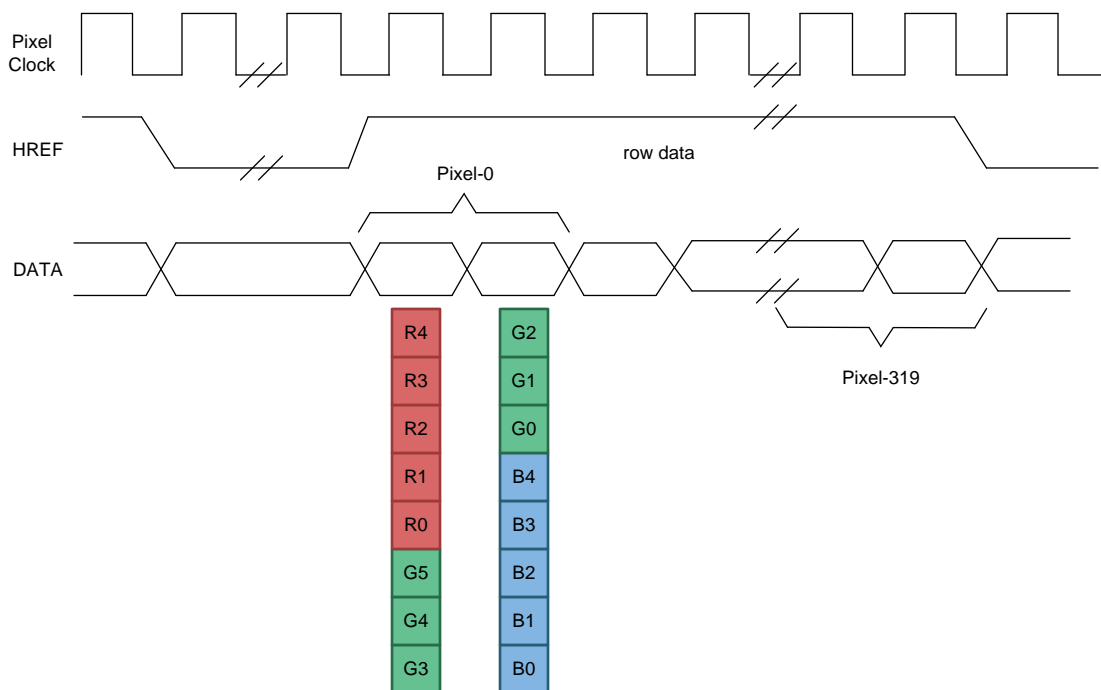


图 3. Line Information From Camera

图 4 shows the HREF and VSYNC for every frame for a 16-BPP QVGA resolution image. The HREF signal is toggled high at 240 times a frame. The time between two valid HREF is referred to as the *horizontal blank period*. The VSYNC is held high only after 240 lines of output from the camera module for a duration referred to as *vertical blank period*.

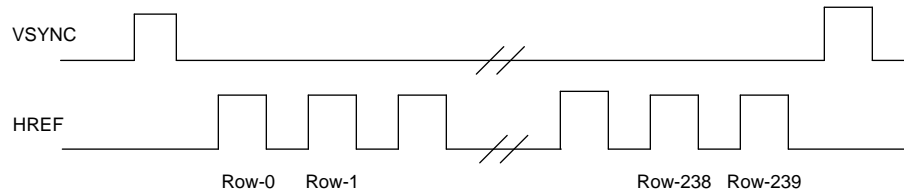


图 4. Frame Information From Camera

### 3.1.2 Camera Configuration Interface

The configuration interface on this camera module is called the serial camera control bus (SCCB). This interface has two signals. The SIO\_C is a clock input signal, and the SIO\_D is a bidirectional data signal. From a protocol perspective the SCCB is compatible with I<sup>2</sup>C. The configuration performed by the TM4C129x MCU is specific to the camera and the user must refer the manufacturer's data sheet.

### 3.1.3 Control Interface

Along with the data and configuration interface, the camera module provides three control signals. The camera includes an active-low reset signal, an active-high power down signal, and a clock, which generates timing signals and for its internal use.

## 3.2 TM4C129x EPI Module

The TM4C129x EPI module provides a high-speed parallel interface for accessing memory devices (SDRAM, SRAM, flash) or programmable devices (FPGA, CPLD). The EPI has multiple modes of operations. To interface the camera module, the host bus 8-bit and 16-bit (HB8 and HB16) mode are used. One of the submodes of the HB8 and HB16 is the XFIFO, which is traditionally used for accessing external FIFO memory and has been leveraged in this reference design for camera access. An additional feature provided by the EPI module is the non-blocking read (NBR) operation which is capable of generating DMA requests through a FIFO. Both the XFIFO and NBRFIFO-DMA features are explained further in 节 3.2.1 and 节 3.2.2.

### 3.2.1 XFIFO Mode Description

The XFIFO mode uses 8- or 16-bits of data, removes address latch enable (ALE), addresses pins, and optionally adds external XFIFO, FFULL, and FEMPTY flag inputs. This submode provides the data side of the normal host-bus interface but is paced by the FIFO control signals. An important consideration is that the XFIFO, FFULL, and FEMPTY control signals may stall the interface and could have an impact on blocking read latency from the processor or  $\mu$ DMA.

For this design the read path for acquiring data from the camera is used, and the XFIFO and FEMPTY signals are used for indicating the EPI when valid data from the camera is available. As shown in 图 5, the read of the data is triggered only when the FEMPTY signal is asserted low. Once the EPI samples the FEMPTY low, a read can be initiated by the MCU (CPU or DMA) to read the data.

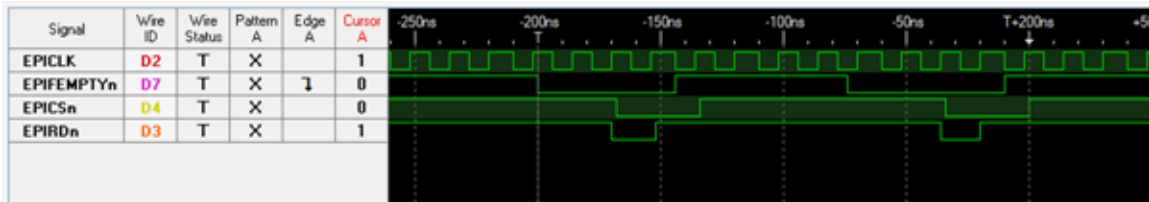


图 5. XFIFO Read Timing Diagram

### 3.2.2 NBRFIFO-DMA Description

The EPI controller supports a special kind of read called a *non-blocking read*. While a normal read stalls the processor or  $\mu$ DMA until the data is returned, a non-blocking read is performed in the background.

A non-blocking read is configured by writing the start address into the EPIRADDR register, the size (data width) of transaction into EPIRSIZE register, and the count of operations into EPIRSTD register. After each read is complete, the result is written into the NBRFIFO, and the EPIRADDR register is incremented by the size (one, two, or four). If the NBRFIFO is filled, the reads pause until space is made available. The NBRFIFO can be configured to interrupt the processor or trigger the DMA based on fullness using the EPIFIFOLVL register. By using the trigger and interrupt method, the DMA can keep space available in the NBRFIFO and allow the reads to continue unimpeded. Another benefit of the mechanism is the low latency of the DMA to transfer the data between the EPI and SRAM.

### 3.3 Logic Block Function

The external logic block is a glue logic that is used to ensure that the signals from the camera and EPI when the two are connected together do not result in bus contention. The logic block ensures that control signals from the camera interface can be converted to a format that the EPI can use. The logic block consists of a NAND gate for creating the FEMPTY signal and an octal driver for controlling the sampling of the data.



图 6. Logic Block Gate

#### 3.3.1 Generating the FEMPTY Signal

The XFIFO mode works by reading data only when the FEMPTY signal is de-asserted to indicate to the EPI that there is data available. To generate the FEMPTY signal, the HREF control signal and pixel clock from the camera are passed through a NAND. 表 2 shows the truth table for the logic and its interpretation by the EPI.

表 2. FEMPTY Signal Truth Table

PCLK	HREF	FEMPTY	INTERPRETATION
L	L	H	Data cannot be sampled
L	H	H	Data cannot be sampled
H	L	H	Data cannot be sampled
H	H	L	Data can be sampled

Because the FEMPTY signal is synchronized by the EPI using its internal clock, the actual FEMPTY used by EPI shall always be a stable signal. Once sampled, FEMPTY can be used to trigger a read request by EPI using the combination of NBRFIFO and DMA allowing a fast response from the MCU to read the data.



### 3.3.2 Data Capture by EPI

The pixel clock is generally a clock in the MHz range. Because there is no frame buffer or FIFO in the data path, every data byte must be sampled correctly. To be able to perform such a high speed capture, the TM4C129x MCU is configured for a system clock of 120 MHz and the EPI configured for a 60-MHz clock. To overcome the absence of a frame buffer, the clock for the camera module is generated by the TM4C129x MCU using the system control clock output. The use of system control clock output provides a synchronous reference (see 图 7) for the EPI to capture the data from the camera. The phase relation can be seen between the XCLK from TM4C129x to the camera and the PCLK from the camera to the TM4C129x.

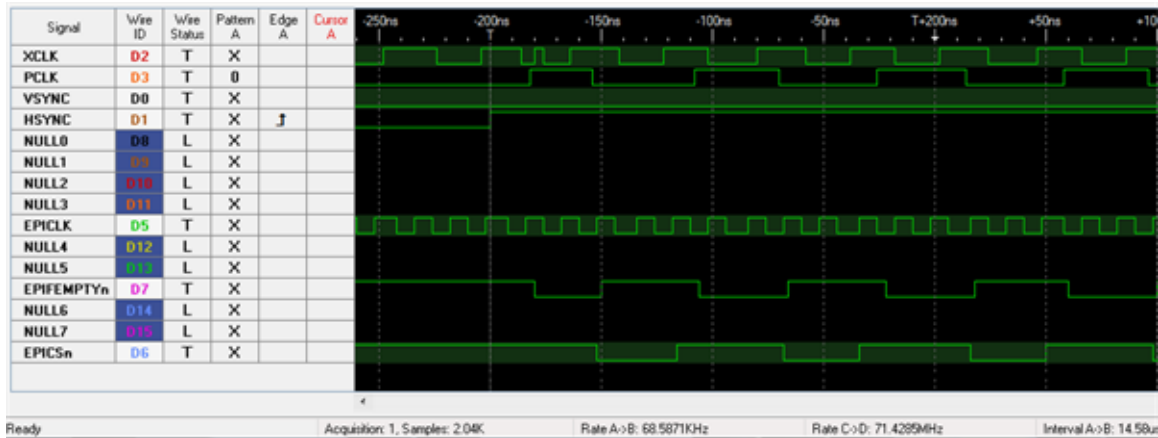


图 7. EPI Data Capture Timing Diagram

The EPI is configured with a non-blocking read with a read count equivalent to the number of pixels that must be captured for every line based on the HB8 or HB16 mode. As seen in the waveform, the HREF and PCLK from the camera, when combined with the NAND gate, forms the FEMPTYn signal for EPI. When this FEMPTY is sampled by the EPI as low, the EPI controller initiates the access to the octal transceiver by asserting the EPICSn and EPIRDn low. The EPICSn allows the data bus from the camera to be read by the EPI data pins.

## 4 Getting Started Hardware and Software

### 4.1 Hardware

To create the solution, the DK-TM4C129x development board is used as the base board to which the camera board can be connected. The USB is used to provide both power and debug for the reference design. To interface the camera board to the DK-TM4C129x, the camera board is connected to the headers J27 and J28 as shown in 图 8.

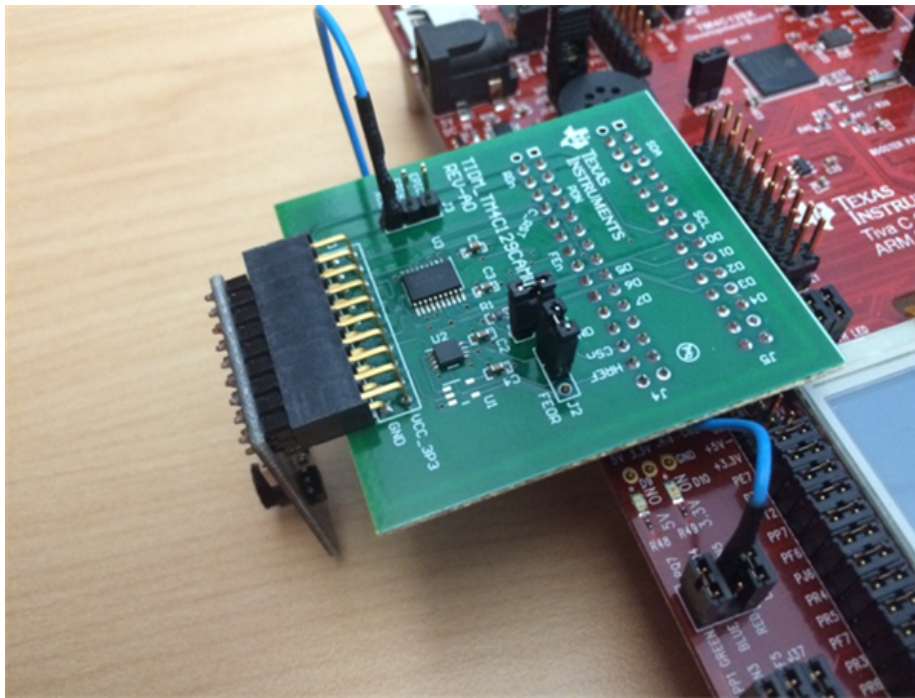


图 8. TIDM-TM4C129CAMERA Assembly

To provide the clock to the camera module, remove the jumper J26 (PQ4) on DK-TM4C129x and connect a wire from the PQ4 marking on the DK-TM4C129x to the XCLK on the J3 header of the camera board. Ensure that the header J2 is connected between pin positions 1 and 2, and the J6 header is mounted on the camera board.

### 4.2 Software

The software for this reference design comes as an installer that the user must install on their PC. It normally takes about a minute for the installer to execute. The following tools are required for rebuilding the project firmware.

1. Download and install TivaWare 2.1.3.
2. Download and install [Code Composer Studio™ \(CCS\)](http://www.ti.com/sc/techlit/CodeComposerStudio) 6.1.1.00022 with ARM® compiler tool chain version 5.2.7.
3. Download and install [LMFlashProgrammer](http://www.ti.com/sc/techlit/LMFlashProgrammer) build 1613.
4. Download and install TIDM-TM4C129CAMERA-Firmware.exe.

When the installer is executed with the default settings, the software will be installed under *C:\Program Files (x86)\Texas Instruments\TM4C\TM4C129CAMERA-1.0*. Three directories will be created by the installer in the above path.

#### 4.2.1 Common Drivers for the Embedded Application

The folder *appdrivers* contains the driver files for the example applications.

#### 4.2.2 Camera With LCD Project

The folder *dktm4c129\_cameralf* contains the CCS project for the camera image capture and displays it on the onboard LCD panel. The software block diagram for the example is shown in 图 9.

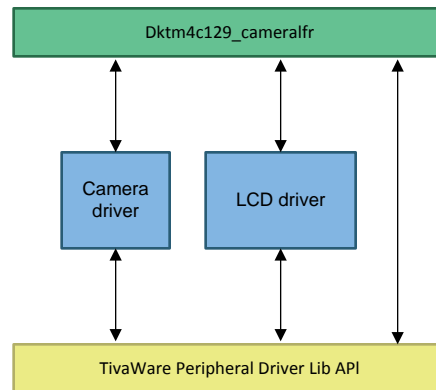


图 9. Software Block Diagram for *dktm4c129\_cameralf*

图 10 shows the data flow diagram for the project. The camera sends the data for every pixel in two bytes. The EPI is configured in HB16 mode, but because the data is received in the lower 8 bits, the EPI controller pads 0 in the upper byte of every 16-bit word. The padding of the upper eight bits has been done because the LCD panel on the DK-TM4C129x is configured to work in LIDD-8 mode. If the byte padding is not performed then the CPU must write the correct byte to the LCD controller, which would reduce CPU bandwidth for any other operation. By performing the automatic padding of data, the CPU can utilize the LCD controller's internal DMA to read the data and send it to the panel. The LCD DMA controller is configured to transfer the data in LIDD-16 mode, which causes the controller to discard the upper byte and only transmit the lower byte and, thus, retain the image information.

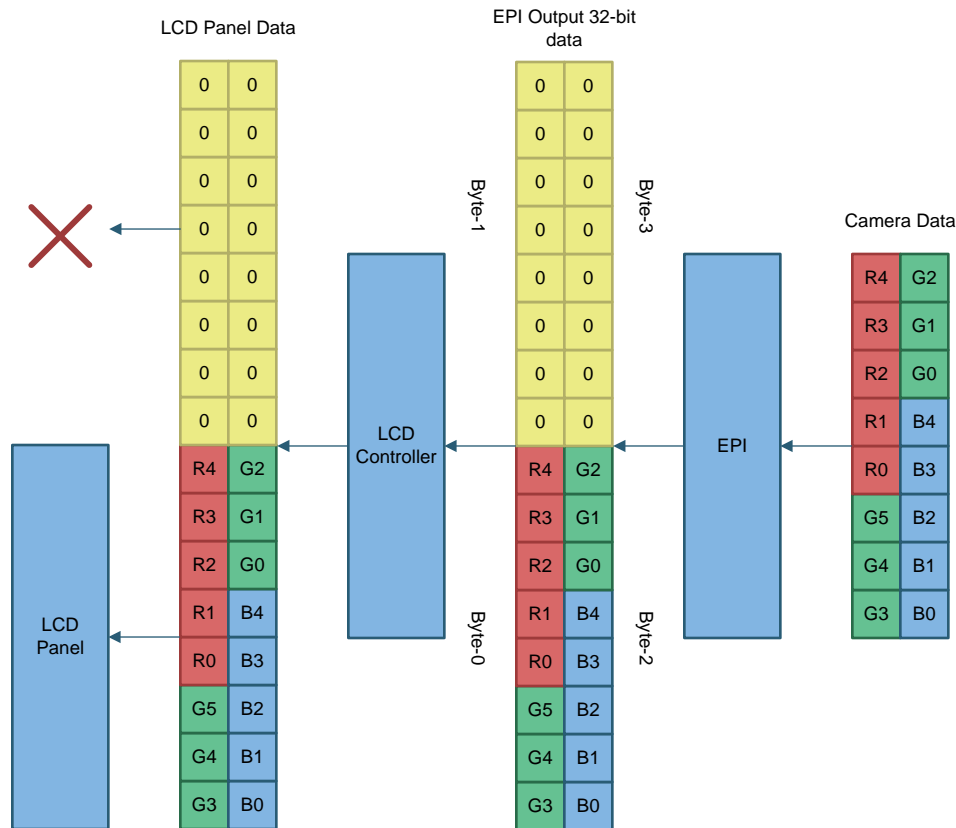


图 10. Pixel Translation During Data Flow

Since the pixel data captured per line is buffered by the LCD controller’s internal FIFO, the SRAM requirement is limited to only one line of data capture, that is, 320 pixels × (2 bytes of pixel information + 2 byte of padded data) or 1280 bytes.

### 4.2.3 Ethernet Web Server With Camera Project

The folder *enet\_camera\_server* contains the CCS project for the camera image capture and viewing it through a web browser. For this application the TM4C129x MCU hosts a web page, which can be opened in a web browser and auto-refreshes every three seconds. The software block diagram for the example is shown in 图 11.

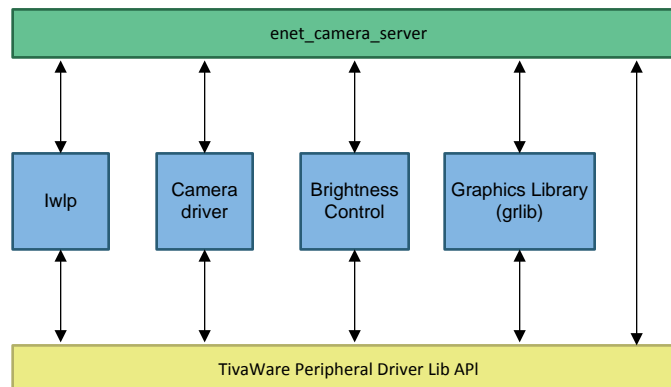


图 11. Software Block Diagram for enet\_camera\_server

For this application the EPI is configured in HB8 mode. The camera sends the data for every pixel in two bytes. The EPI controller now concatenates the bytes in a 32-bit word. Because the web server reads the data as is before sending it to the browser, there is no requirement for the EPI to perform padding operation.

The TM4C129x device hosts the web page from the flash but the image is kept in SRAM. If the user wants to change the web page then there are some steps that must be followed. These steps are given in the ReadMe.txt of the example code.

## 5 Testing and Results

### 5.1 Test Setup

Now that the board has been setup for downloading the application firmware and the user has installed the application software, this section will elaborate on how to import, build, and run the application firmware for voice call.

#### 5.1.1 Create a New Workspace in CCS™

Before building the examples the user must create a new workspace to ensure a clean build of the project.

1. Start CCS.
2. Click on *File* then navigate to *Switch Workspace* → *Others* in the drop down menu.
3. In the pop-up box, provide the name of the new workspace. The created example workspace is called *workspace\_TIDMTM4C129CAMERA* (see [图 12](#)). Click *OK*.

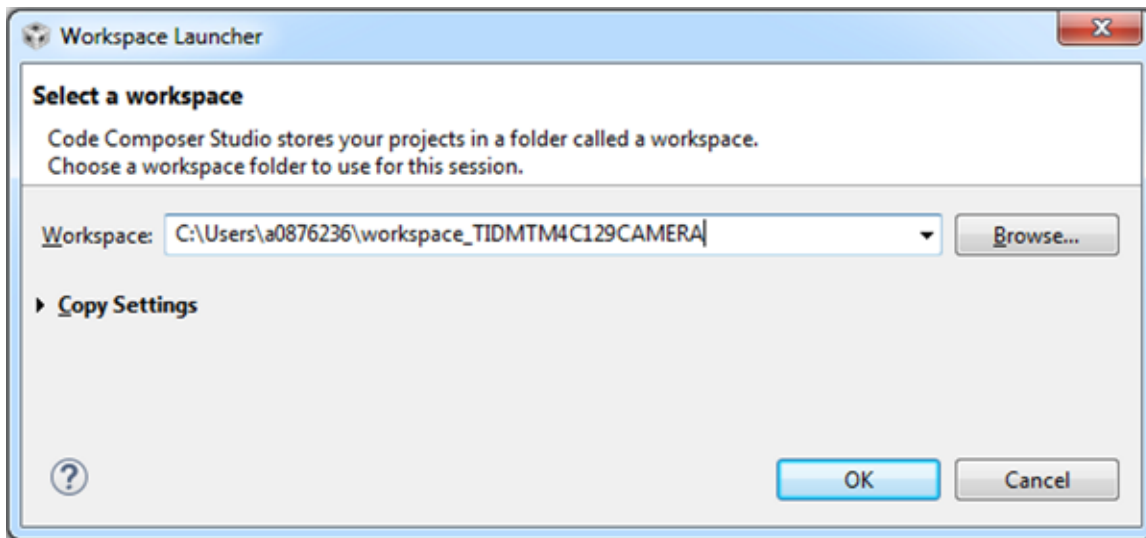


图 12. Create a New Workspace

### 5.1.2 Import and Build Projects in CCS™

To compile the example code, the opus audio codec library must first be built. The example code uses the precompiled output of the opus audio codec library during the linker phase.

1. In the workspace created (see 节 5.1.1), select *File* → *Import*. The import window will display.
2. Expand CCS for more options. Click *CCS Projects*. Click on *Next* (see 图 13).

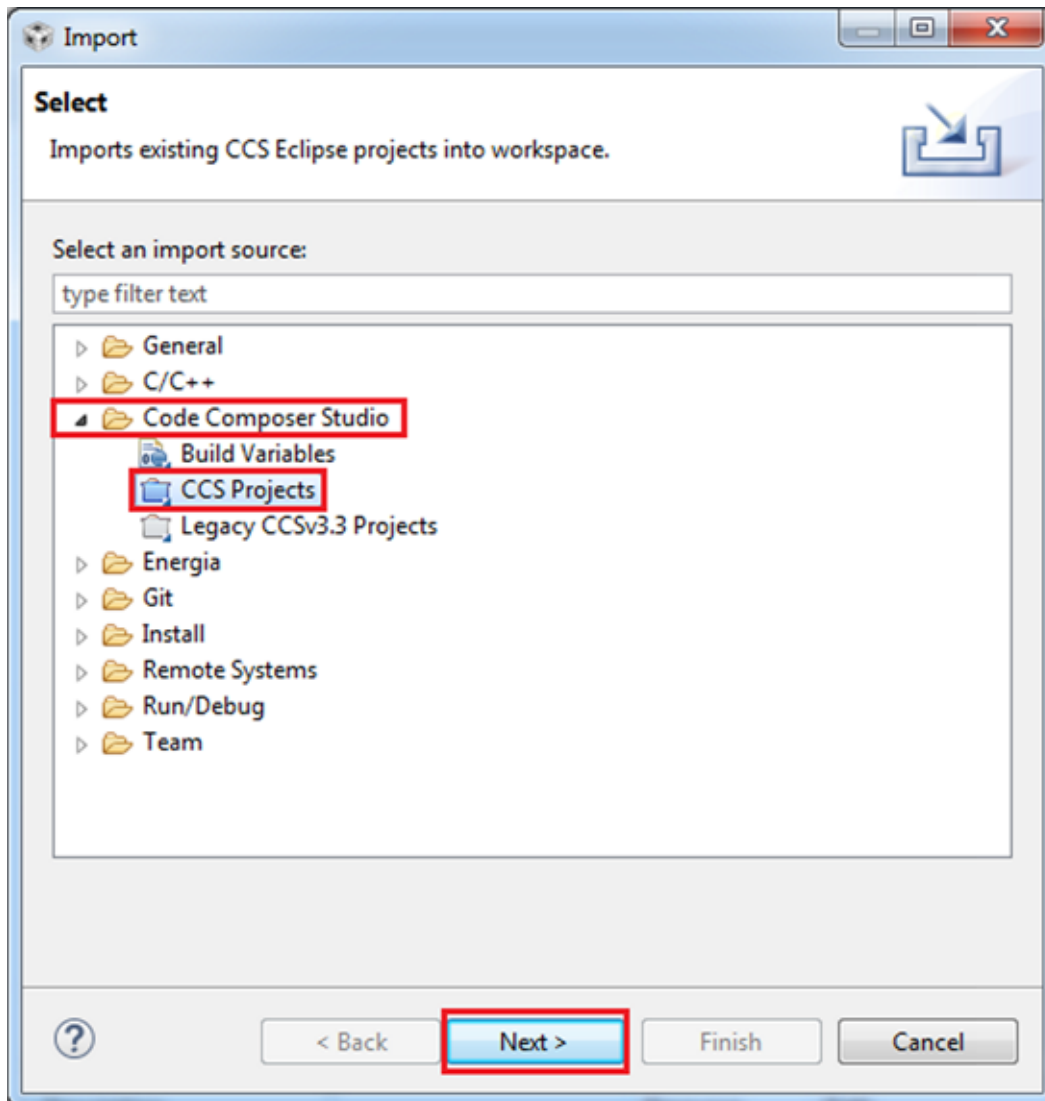


图 13. Import Selection View

- Click the *Browse* button in front of *Select search-directory* option and navigate to the directory where the project collateral has been installed (see 4.2 节). Select all the projects that are listed in the *Discovered projects: pane*. Make sure that the boxes in front of *Automatically import referenced projects found in the same search-directory* and *Copy projects into workspace* are checked (see 图 14). Now click on *Finish* button. The examples will import into CCS.

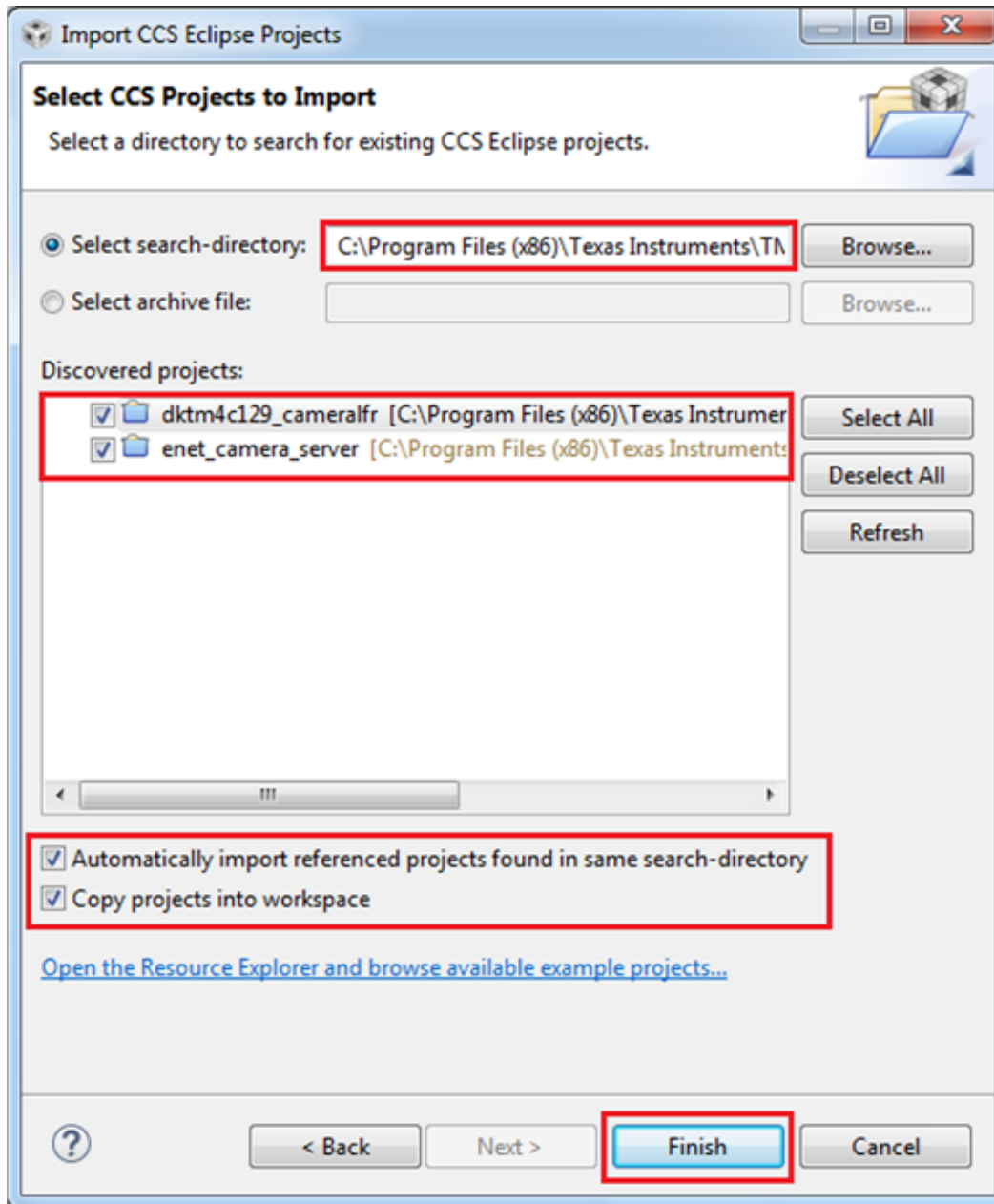


图 14. Import the Projects



- Before building the projects, make sure that the variable `ORIGINAL_PROJECT_ROOT` points to the correct directory where the TivaWare software has been installed and that `TID_ROOT` points to the correct directory where the project collaterals have been installed. To view or change the variable, right-click on the project and click on *Show Build Settings*. In the pop-up window, click on *Build*. Click on the tab *Variables* to view the value of `ORIGINAL_PROJECT_ROOT` or `TID_ROOT`. If the path shown is different compared to the path on the machine where the opus source code has been extracted then select the variable and click on *Edit...* to change the path (see 图 15).

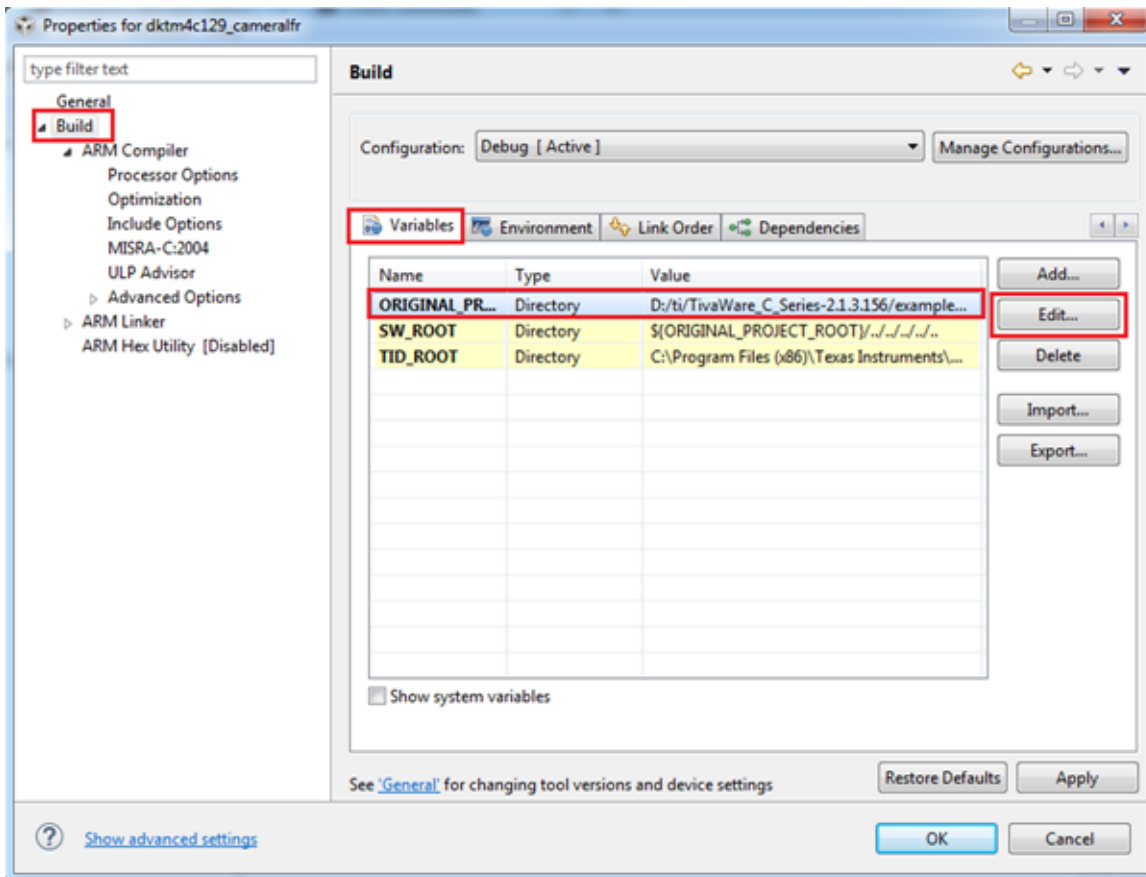


图 15. Update the Environment Variables

5. Right-click the project in *Project Explorer* and click *Build Project*. If building the library for the first time, it may take a few minutes. After the compilation is successful, the CCS console must display the message in 图 16 and 图 17.

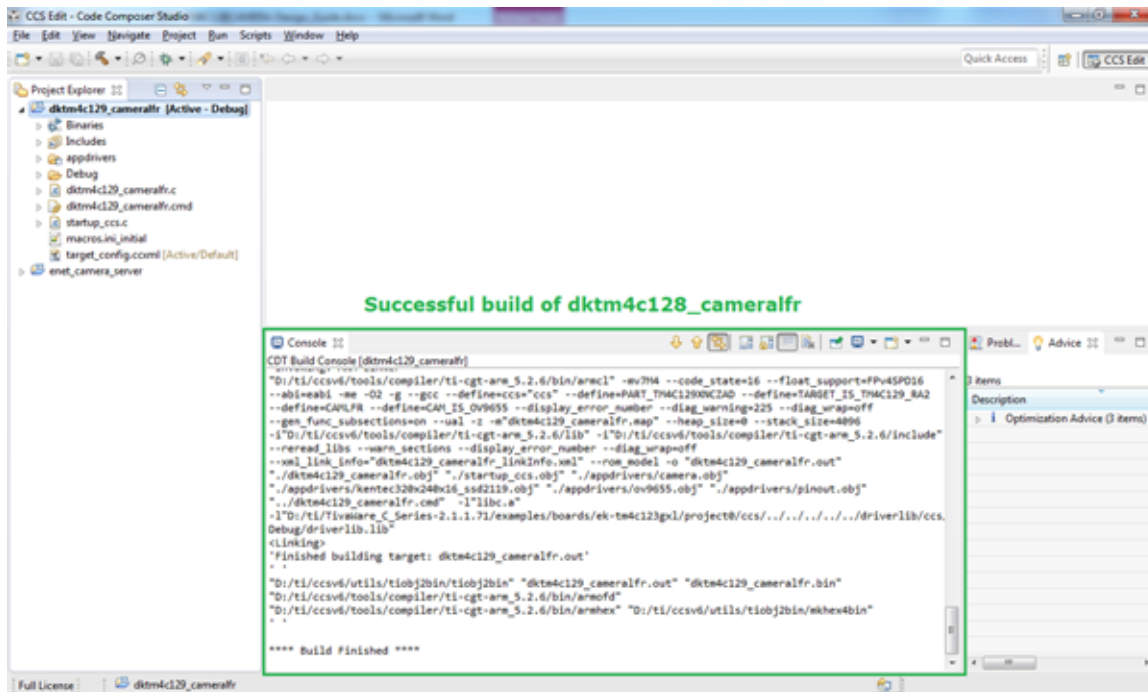


图 16. Compiling dktm4c129\_cameralftr Project

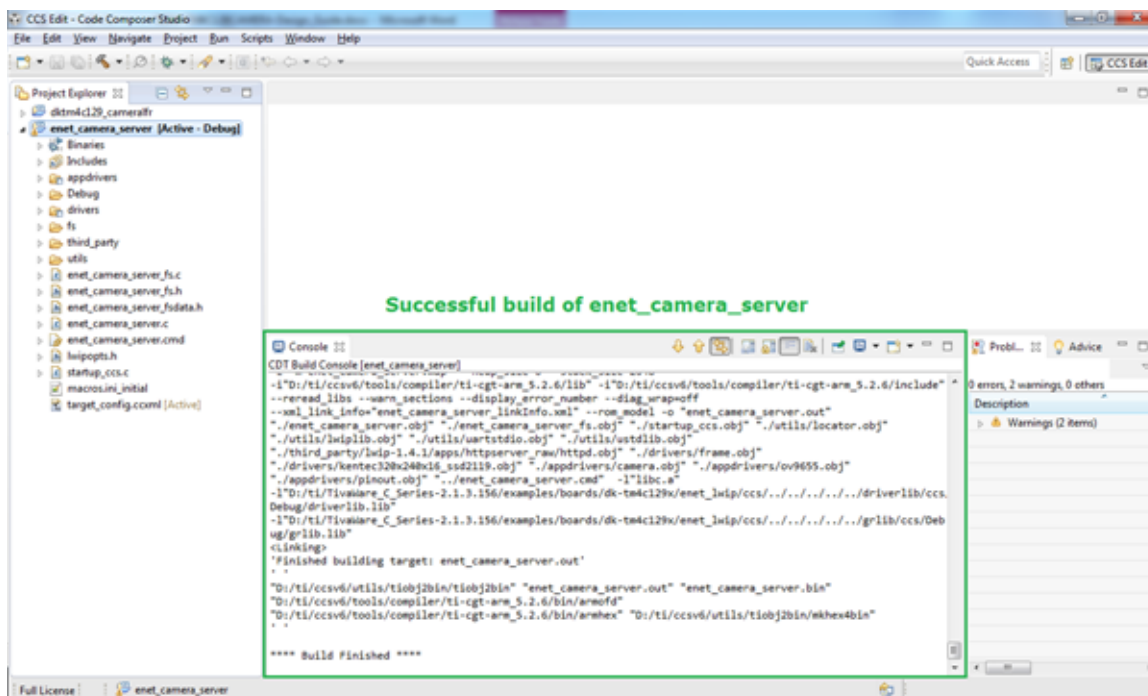


图 17. Compiling enet\_camera\_server Project

### 5.1.3 Test Rig

To display an image captured by the camera either on the LCD or hosted by the web server application, a test rig is set up with an EK-TM4C1294XL connected launchpad shown in [图 18](#).

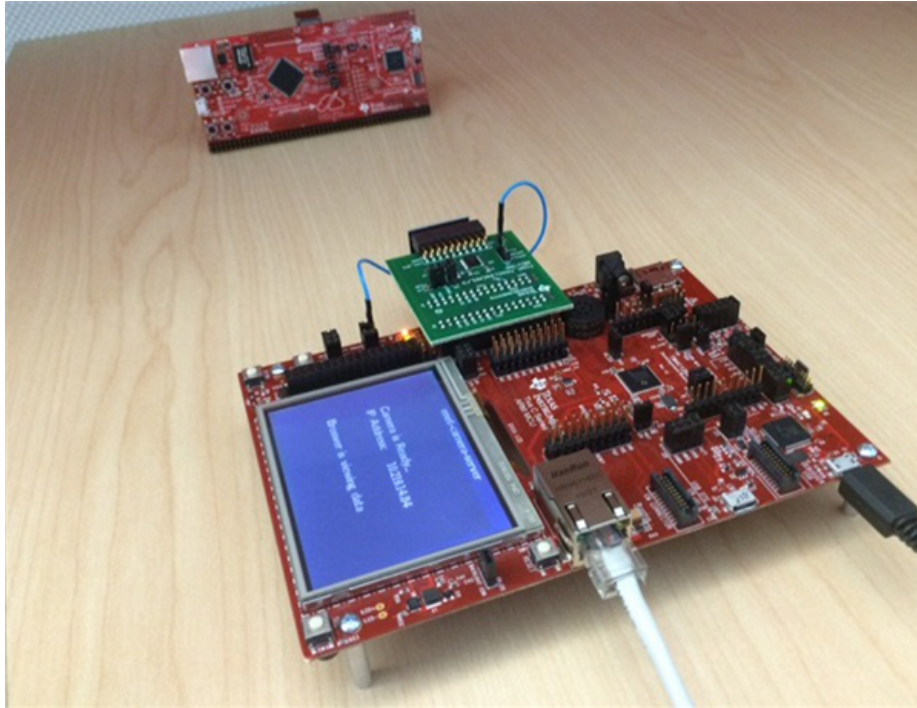


图 18. Test Rig

### 5.1.4 Executing Camera With LCD

Now that the CCS project *dktm4c129\_camera1fr* is built, download the binary image using LMFlashProgrammer or use the debug button in CCS to download the image to the DK-TM4C129x. On running the application, the TM4C129x will start capturing data from the camera and display it on the LCD panel (see [图 19](#)).

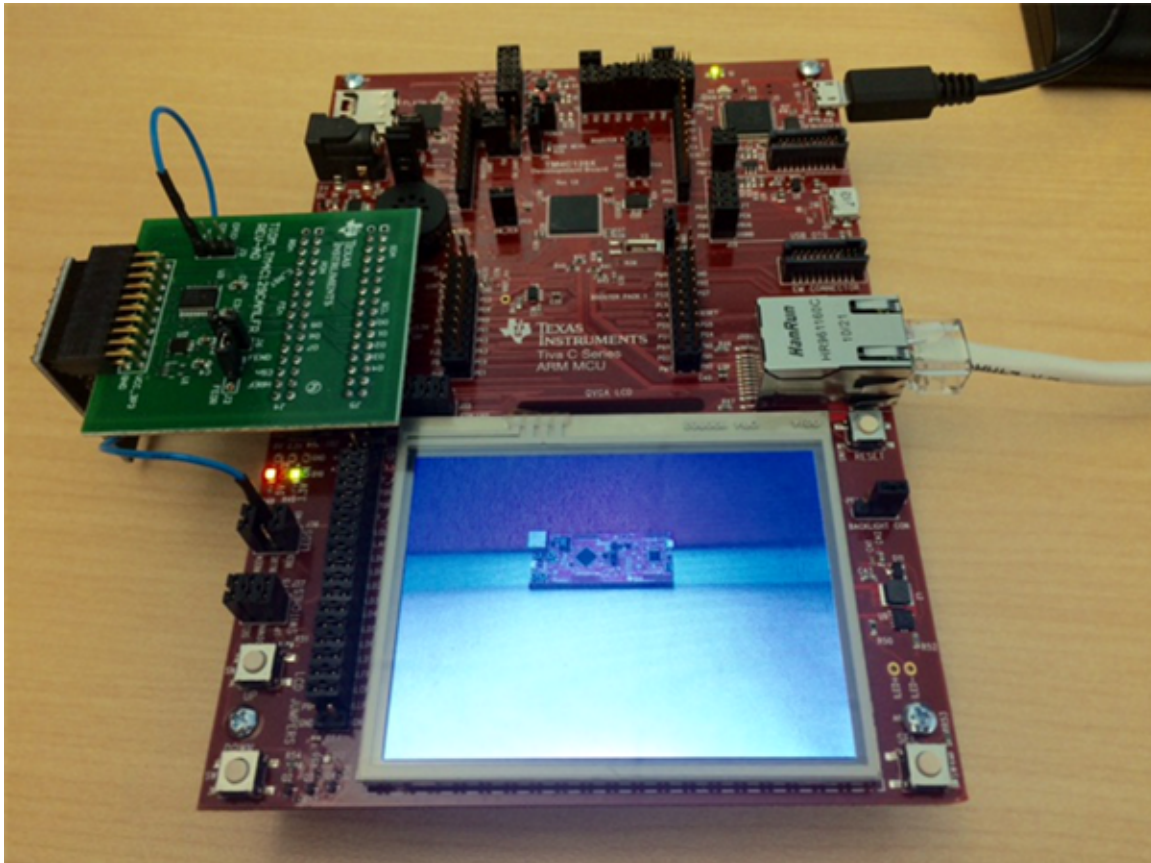


图 19. Board View on Executing *dktm4c129\_camera1fr*

### 5.1.5 Executing Camera With Web Server

Now that the CCS project *enet\_camera\_server* is built, download the binary image using LMFlashProgrammer or use the debug button in CCS to download the image to the DK-TM4C129x.

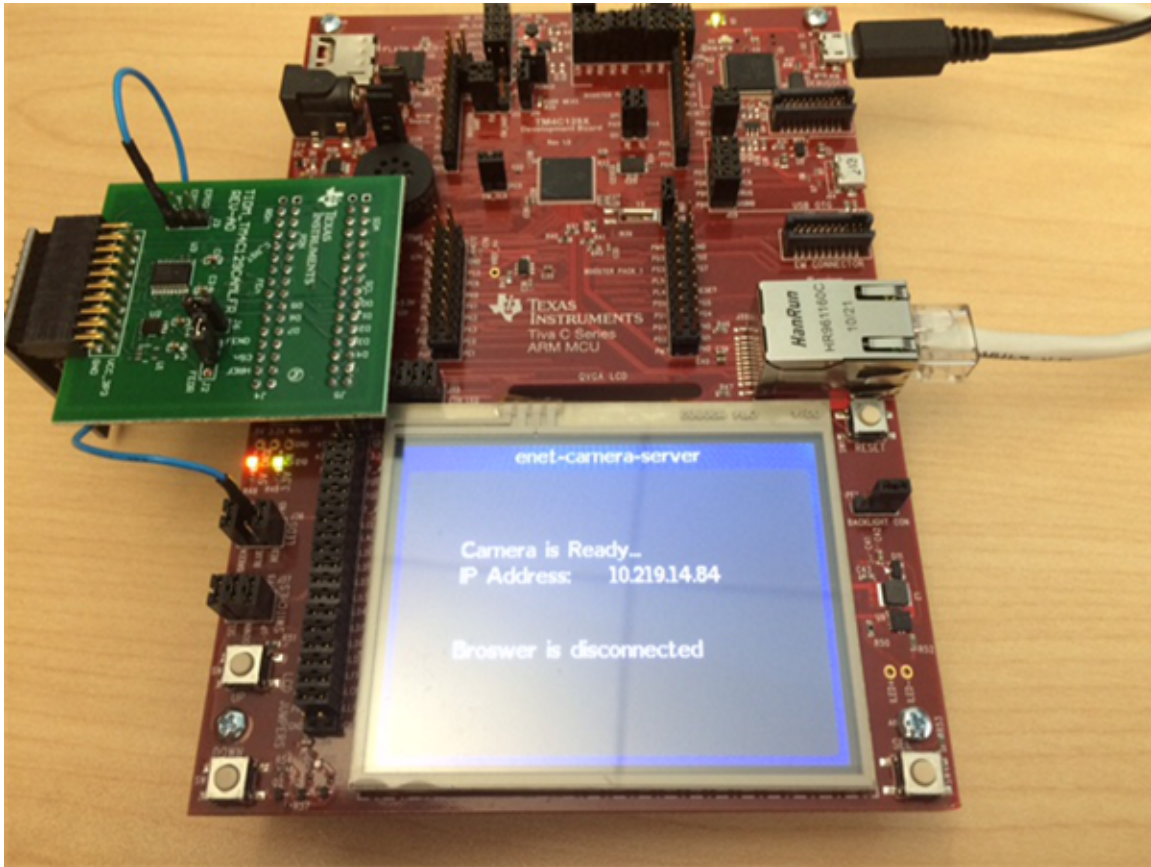
注: The web page hosted by the DK-TM4C129x is configured to auto-refresh the web browser every three seconds using the HTML META tag for reload, which eliminates the requirement for the user to manually reload the page or have a plugin to reload the page automatically.

On running the application, the TM4C129x will first acquire the link and then an IP address (see [图 20](#)).



图 20. Board View When Acquiring IP Address

Once the IP address is acquired, the application informs the user through the LCD panel that the camera is ready and displays the IP address and connection status of a web browser (see [图 21](#)).



**图 21. Board View When Application is Ready**

The user can now open a web browser and type in the IP address. When entering in the IP address, the DK-TM4C129x will send the image captured by the camera to the web browser (see 图 22). At the same time, the status message on the LCD panel changes to indicate that a web browser is connected to the MCU hosting the camera service (see 图 23). If the web browser is closed, the status updates to indicate that the browser is no longer requesting camera data.

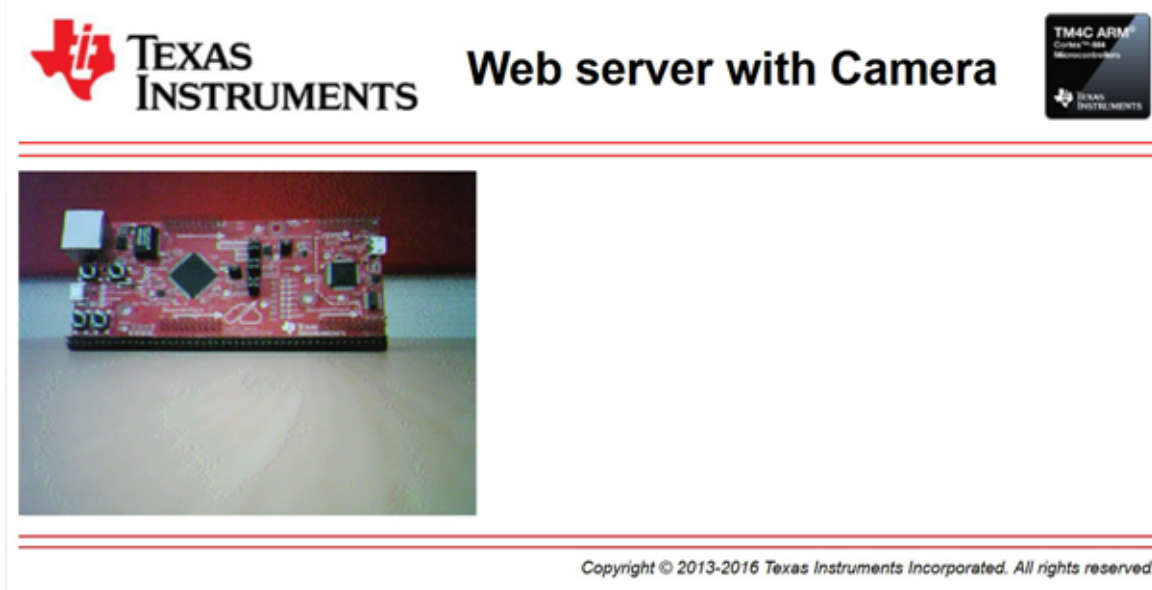


图 22. Web Browser View of Hosted Web Page



图 23. Board View When Web Browser is Viewing Web Page

## 5.2 Test Data

图 24 显示了相机生成的帧和行速率及其转换为 EPI 访问。测量的 XCLK 和 EPICLK 必须忽略，因为以较低的采样率捕获 33.33 ms 信号，而逻辑分析仪缓冲区有限。

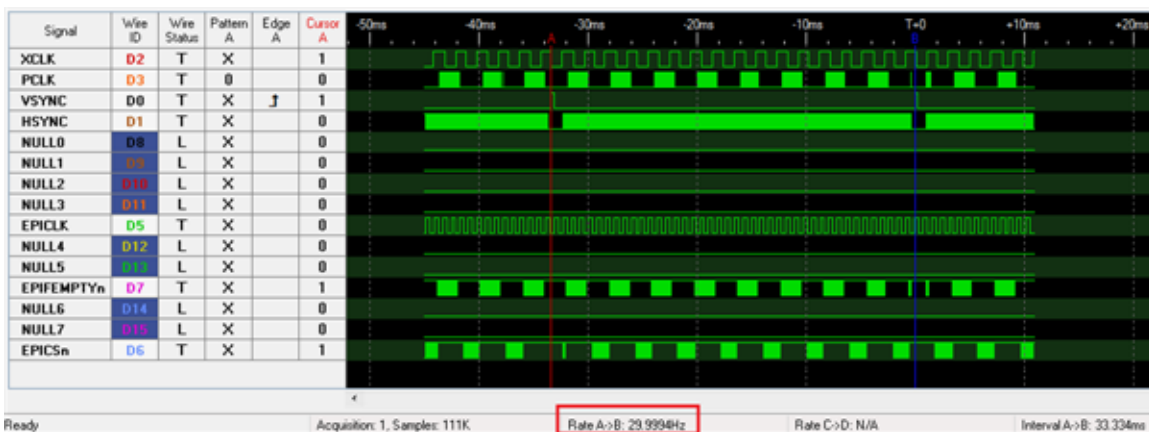


图 24. Frame and Line Rate for Data Capture



## 6 Design Files

### 6.1 Schematics

To download the schematics, see the design files at [TIDM-TM4C129CAMERA](#).

### 6.2 Bill of Materials

To download the bill of materials (BOM), see the design files at [TIDM-TM4C129CAMERA](#).

### 6.3 PCB Layout Recommendations

#### 6.3.1 Layout Prints

To download the layer plots, see the design files at [TIDM-TM4C129CAMERA](#).

### 6.4 Altium Project

To download the Altium project files, see the design files at [TIDM-TM4C129CAMERA](#).

### 6.5 Gerber Files

To download the Gerber files, see the design files at [TIDM-TM4C129CAMERA](#).

### 6.6 Assembly Drawings

To download the assembly drawings, see the design files at [TIDM-TM4C129CAMERA](#).

## 7 Software Files

To download the software files, see the design files at [TIDM-TM4C129CAMERA](#).

## 8 References

1. Wave Share, [0v9655 Camera Board](#).

### 8.1 商标

All trademarks are the property of their respective owners.

## 9 About the Author

**AMIT ASHARA** is an application engineer and Member Group Technical staff member at Texas Instruments. He works on developing applications for the TM4C12x family of high-performance MCUs. Amit brings extensive experience in high-speed digital and MCU system-level design expertise to this role. Amit earned his Bachelor of Engineering (BE) from the University of Pune in India.

## 修订版本 A 历史记录

注：之前版本的页码可能与当前版本有所不同。

<b>Changes from Original (September 2016) to A Revision</b>	<b>Page</b>
• 已更改 location of 1 节 (previously Section 1.1).....	2
• 已更改 location of 1.1 节 (previously Section 1.2).....	2
• 已更改 图 2. ....	4

## 有关 TI 设计信息和资源的重要通知

德州仪器 (TI) 公司提供的技术、应用或其他设计建议、服务或信息，包括但不限于与评估模块有关的参考设计和材料（总称“TI 资源”），旨在帮助设计人员开发整合了 TI 产品的应用；如果您（个人，或如果是代表贵公司，则为贵公司）以任何方式下载、访问或使用了任何特定的 TI 资源，即表示贵方同意仅为该等目标，按照本通知的条款进行使用。

TI 所提供的 TI 资源，并未扩大或以其他方式修改 TI 对 TI 产品的公开适用的质保及质保免责声明；也未导致 TI 承担任何额外的义务或责任。TI 有权对其 TI 资源进行纠正、增强、改进和其他修改。

您理解并同意，在设计应用时应自行实施独立的分析、评价和判断，且应全权负责并确保应用的安全性，以及您的应用（包括应用中使用的 TI 产品）应符合所有适用的法律法规及其他相关要求。您就您的应用声明，您具备制订和实施下列保障措施所需的一切必要专业知识，能够 (1) 预见故障的危险后果，(2) 监视故障及其后果，以及 (3) 降低可能导致危险的故障几率并采取适当措施。您同意，在使用或分发包含 TI 产品的任何应用前，您将彻底测试该等应用和该等应用所用 TI 产品的功能。除特定 TI 资源的公开文档中明确列出的测试外，TI 未进行任何其他测试。

您只有在为开发包含该等 TI 资源所列 TI 产品的应用时，才被授权使用、复制和修改任何相关单项 TI 资源。但并未依据禁止反言原则或其他法律授予您任何 TI 知识产权的任何其他明示或默示的许可，也未授予您 TI 或第三方的任何技术或知识产权的许可，该等产权包括但不限于任何专利权、版权、屏蔽作品权或与使用 TI 产品或服务的任何整合、机器制作、流程相关的其他知识产权。涉及或参考了第三方产品或服务的信息不构成使用此类产品或服务的许可或与其相关的保证或认可。使用 TI 资源可能需要您向第三方获得对该等第三方专利或其他知识产权的许可。

TI 资源系“按原样”提供。TI 兹免除对 TI 资源及其使用作出所有其他明确或默示的保证或陈述，包括但不限于对准确性或完整性、产权保证、无复发故障保证，以及适销性、适合特定用途和不侵犯任何第三方知识产权的任何默认保证。

TI 不负责任何申索，包括但不限于因组合产品所致或与之有关的申索，也不为您辩护或赔偿，即使该等产品组合已列于 TI 资源或其他地方。对因 TI 资源或其使用引起或与之有关的任何实际的、直接的、特殊的、附带的、间接的、惩罚性的、偶发的、从属或惩戒性损害赔偿，不管 TI 是否获悉可能会产生上述损害赔偿，TI 概不负责。

您同意向 TI 及其代表全额赔偿因您不遵守本通知条款和条件而引起的任何损害、费用、损失和/或责任。

本通知适用于 TI 资源。另有其他条款适用于某些类型的材料、TI 产品和服务的使用和采购。这些条款包括但不限于适用于 TI 的半导体产品 (<http://www.ti.com/sc/docs/stdterms.htm>)、[评估模块](http://www.ti.com/sc/docs/sampters.htm)和样品 (<http://www.ti.com/sc/docs/sampters.htm>) 的标准条款。

邮寄地址：上海市浦东新区世纪大道 1568 号中建大厦 32 楼，邮政编码：200122  
Copyright © 2017 德州仪器半导体技术（上海）有限公司