TI Designs

Concurrent Parallel XIP Flash and SRAM Design for Code Download and Execution on High-Performance Microcontrollers Design Guide

TI Designs

TI Designs provide the foundation that you need including methodology, testing, and design files to quickly evaluate and customize the system. TI Designs help *you* accelerate your time to market.

Design Resources

TIDM-TM4CFLASHSRAM	Tool Folder Containing Design Files
EK-TM4C1294XL	Tool Folder
TM4C1294NCPDT	Product Folder
TM4C123GH6PM	Product Folder
TPD4S012	Product Folder
TPS2052B	Product Folder
TPS62177	Product Folder
TPS73733-Q1	Product Folder
SN74LV373A	Product Folder

TI E2E™ Community

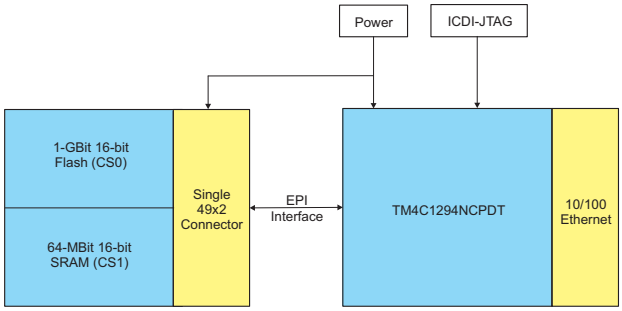
ASK Our E2E Experts
WEBENCH® Calculator Tools

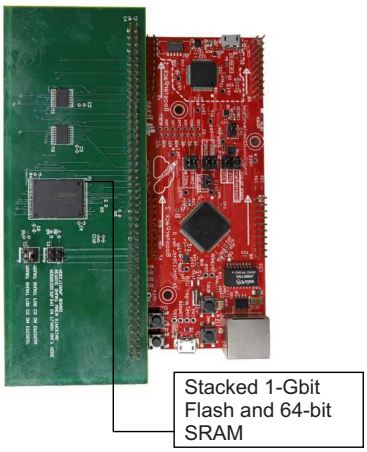
Design Features

- Extend the Useable Memory Space to 1-Gbit 16-bit FLASH and 16-Mbit 16-bit Asynchronous SRAM with the 60-MHz External Peripherial Interface (EPI) for Large Memory Footprint Applications
- Designed for (formerly Tiva MCU) EK-TM4C1294XL Connected LaunchPad™
- Implements Serial Boot Loader for Programming Parallel Flash In Situ Over EPI
- Supports Detection of Flash and SRAM Connected to EPI
- Scalable Flash Footprint Design from 64 Mbit to 1 Gbit
- Source Code Contains Project Examples for Code Composer Studio

Featured Applications

- Interactive Human Machine Interfaces
- Industrial Automation
- IoT Solutions
- Test and Measurement





An IMPORTANT NOTICE at the end of this TI reference design addresses authorized use, intellectual property matters and other important disclaimers and information.

1 System Description

The TM4C129x family can extend its memory space when the internal memory is insufficient. This design guide describes the hardware interface requirements and example software for the TM4C129x microcontrollers from TI. The executable memory region is extended to 1-Gbit 16-bit FLASH and 16-Mbit 16-bit asynchronous SRAM with the 60-MHz External Peripherial Interface (EPI) operating in Host-Bus 16 Mode. Customer applications can leverage extended memory capabilities of TM4C129 product family. The design files include schematics, BOM, layer plot, Altium files, Gerber files, and reference example codes for easy-to-use parallel asynchronous Flash and SRAM with the TM4C1294NCPDT-connected LaunchPad™.

1.1 TM4C1294NCPDT

TM4C1294NCPDT is a 120-MHz high-performance microcontroller with 1MB on-chip Flash and 256KB on-chip SRAM. This microcontroller features an integrated Ethernet MAC+PHY for connected applications. The device has high bandwidth interfaces such as the memory controller and a high-speed USB 2.0 digital interface. With the integration of a number of low- to mid-speed serial communication interfaces like I^2C, UART, and SSI; up to 4 MSPS 12-bit ADC; and motion control peripherals, the TM4C1294NCPDT assists in many applications ranging from industrial communication equipments to Smart Grid & Energy applications.

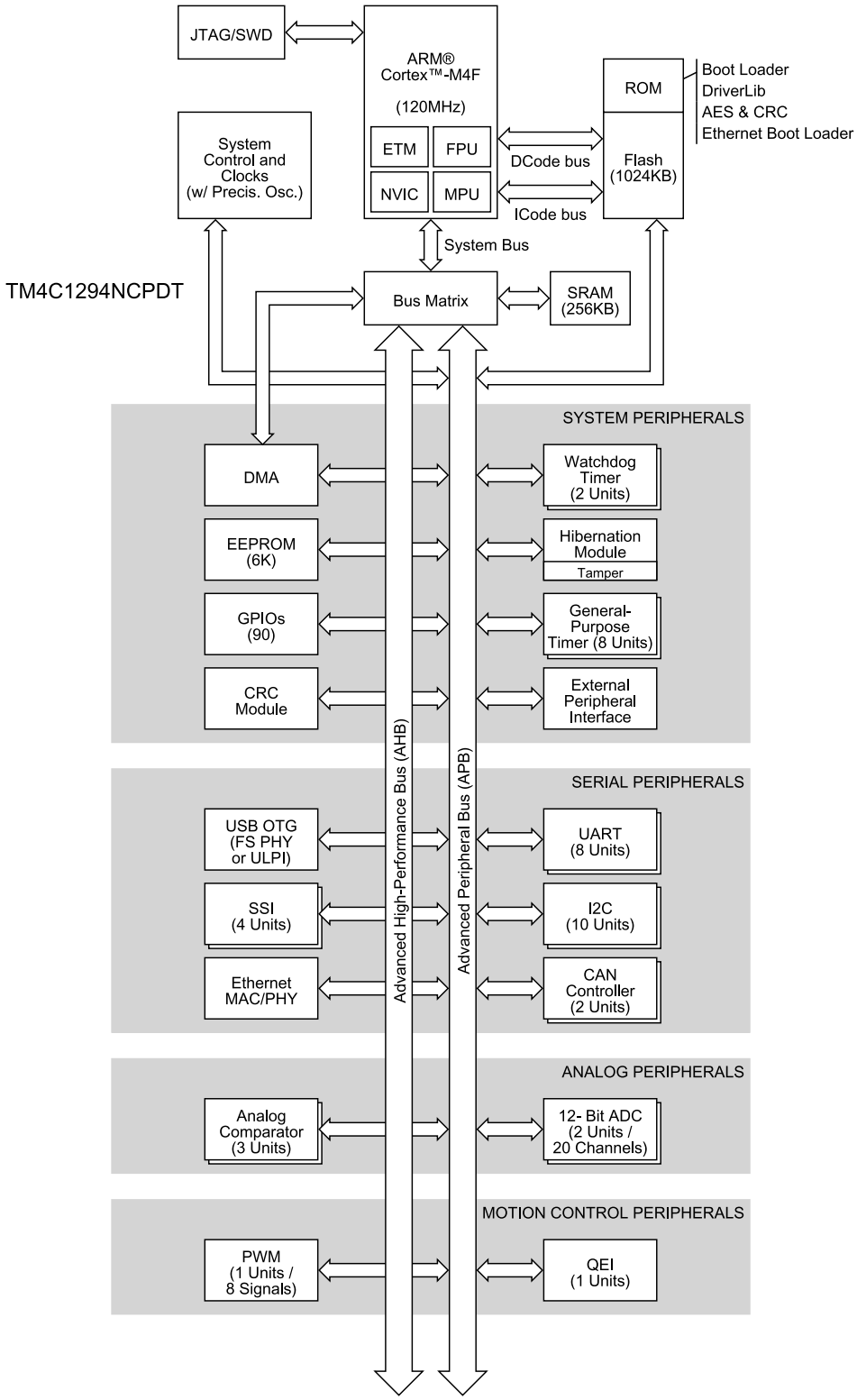


Figure 1. TM4C1294NCPDT Microcontroller High-Level Block Diagram

2 Block Diagram

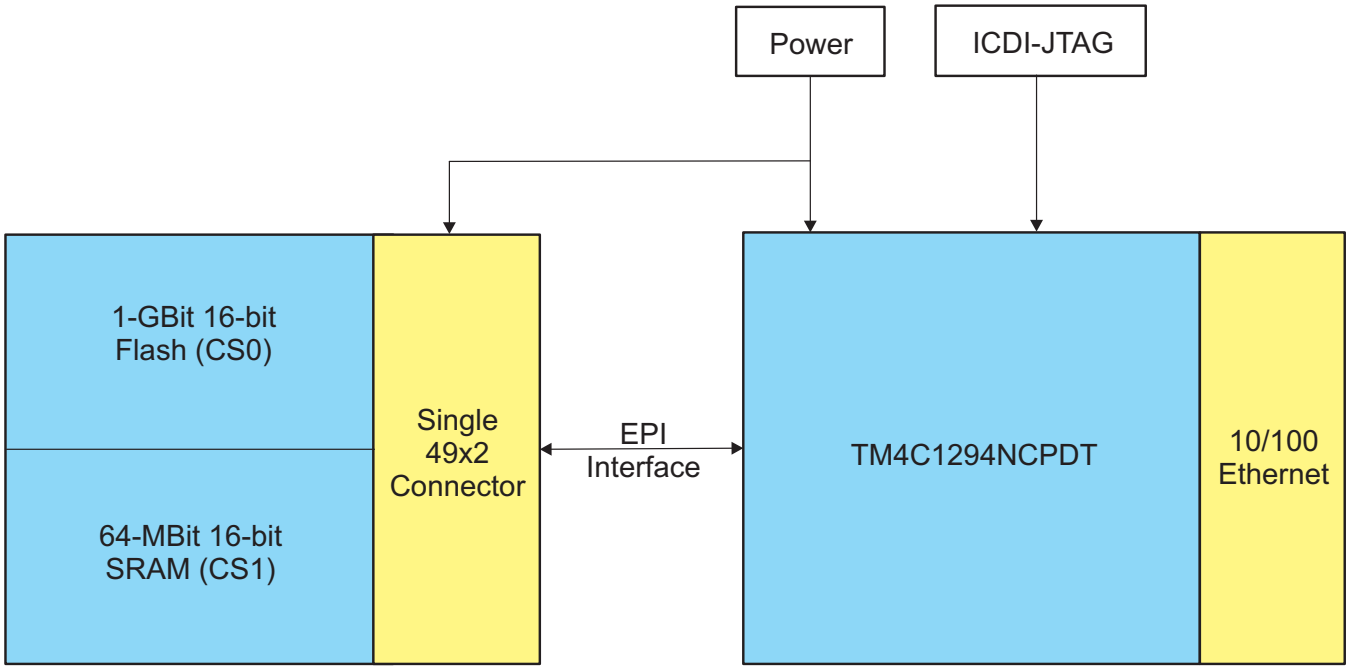


Figure 2. FLASH-SRAM Memory Extender Block Diagram

3 Getting Started Hardware

The interfacing of the Flash-SRAM memory to the TM4C1294NCPDT device on the EK-TM4C1294XL-connected Launchpad™ requires two daughter boards designed to be connected and stacked to the bread board connector X11. The memory cards are designed as independent boards so that the end application can be flexible to interface to either a Flash card, SRAM card, or a stacked solution of Flash and SRAM daughter cards.

The user must note that when using stacked memory, the loading of the pins reduces the effective frequency of access as described in the datasheet. For reference, Table 1 has been reused from the datasheet. Please check the datasheet of the intended product for the most accurate information.

Table 1. EPI Interface Option

INTERFACE	MAXIMUM FREQUENCY
Single SDRAM	60 MHz
Single SDRAM	60 MHz
Single PSRAM without iRDY signal use	55 MHz
Single PSRAM with iRDY signal use	52 MHz
FPGAs, CPLDs, etc using General-Purpose Mode	60 MHz
Memory configurations with 2 chip selects	40 MHz
Memory configurations with 4 chip selects	20 MHz

3.1 Flash Memory Card

The Flash memory card can support from 8-Mbit to 1-Gbit Flash memory in 16-bit mode. The range of memory capacity options is possible as the pin compatibility exists across the TSSOP package.

- TSSOP 56-Pin package: Supports 1-Gbit, 512-Mbit, 256-Mbit, and 128-Mbit memory with access time of 110 ns.
- TSSOP 48-Pin package: Supports 64-Mbit, 32-Mbit, 16-Mbit, and 8-Mbit memory with access time of 70 ns.

Figure 3 shows Pin-1 for each of the package types. The top silkscreen has the marking for each package type clearly marked so that during board assembly, the assembly house can correctly mount the Flash memory.

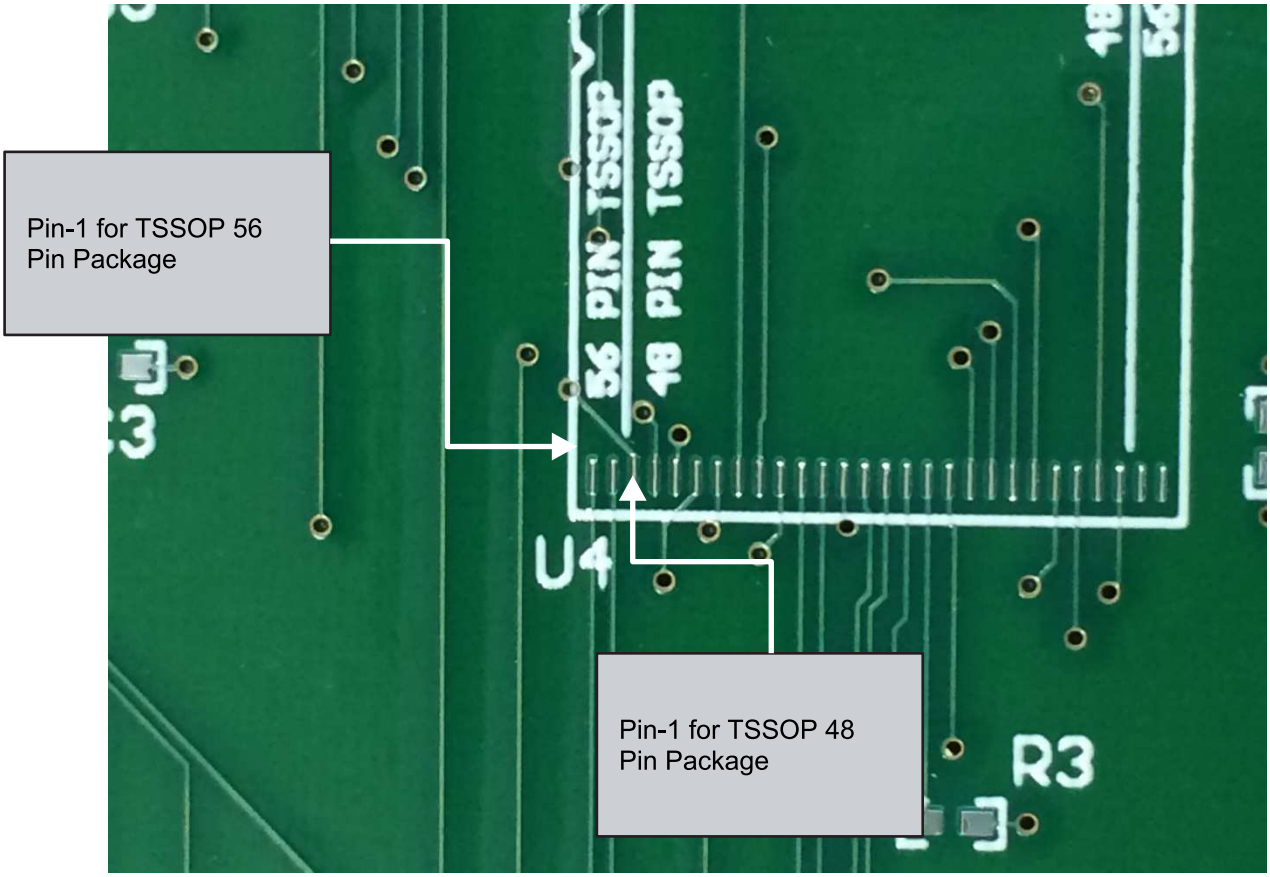


Figure 3. Flash Mounting Option

Figure 4 shows the sample mounting of a 64-Mbit TSSOP48 pin package. Notice that two pins in each corner are not connected, and the marker for 56-Pin TSSOP is visible on top and bottom.

Figure 4. TSSOP48 Mounting Image

The Flash memory card has two sets of 2x2 jumpers. Jumper J2 selects the position of the Chip Select for the Flash memory. The arrow marker indicates the direction in which shunt has to be placed. The shunt must be placed along the length of the board.

CAUTION

Do not connect both shunts for Jumper J2 (Figure 7) or connect the shunts vertically because it may damage the Flash device or the TM4C1294NCPDT on the connected LaunchPad.

The second jumper J3 is used to select the routing of Address Lines A25 and A24 also represented as EPI0S25 and EPI0S24 in the data sheets.

CAUTION

Do not connect the shunts for Jumper J3 (Figure 7) when stacking SRAM because EPI0S25 and EPI0S24 are byte selects for SRAM.

3.2 SRAM Card

The SRAM card can support from 16-Mbit to 256-Mbit SRAM memory in 16-bit mode. Due to lack of options for pin compatibility, the current design is for 16-Mbit asynchronous SRAM memory.

The SRAM card has a single set of 2x2 jumpers. Jumper J1 selects the position of the Chip Select for the memory. The arrow marker indicates the direction in which shunt has to be placed. The shunt is to be placed along the length of the board.

CAUTION

Do not connect both shunts for Jumper J1 (Figure 9) or connect the shunts vertically as it may damage the SRAM device or the TM4C1294NCPDT on the Connected LaunchPad.

4 Getting Started Software

The reference design comes with three example codes that the user can import into Code Composer Studio. The following subsections describe the example code that customers may use as a starting point in their end application.

4.1 Parallel Boot with Flash-SRAM Code Execution

The asynchronous Flash-SRAM boot example code uses the internal Flash of the TM4C1294NCPDT to hold a custom boot loader, and it maps the RW memory to the external SRAM. The boot loader configures the EPI module to enable a 16-bit parallel Flash and SRAM for code download and execution. The boot example code assumes that the user is using a 1-Gbit to 128-Mbit Flash memory because the sector size used for holding the image pointer and the location of the actual application is aligned to a sector size of 128KB (0x20000).

The lowest sector (Sector-0) of Flash memory is used to hold a table indicating the start address of the image, the size of the image, and the validity of the image. This location is updated by the boot loader when the user downloads the binary file to the external Flash memory via UART0. Figure 5 shows the structure of Sector-0 for holding information for the actual application image.

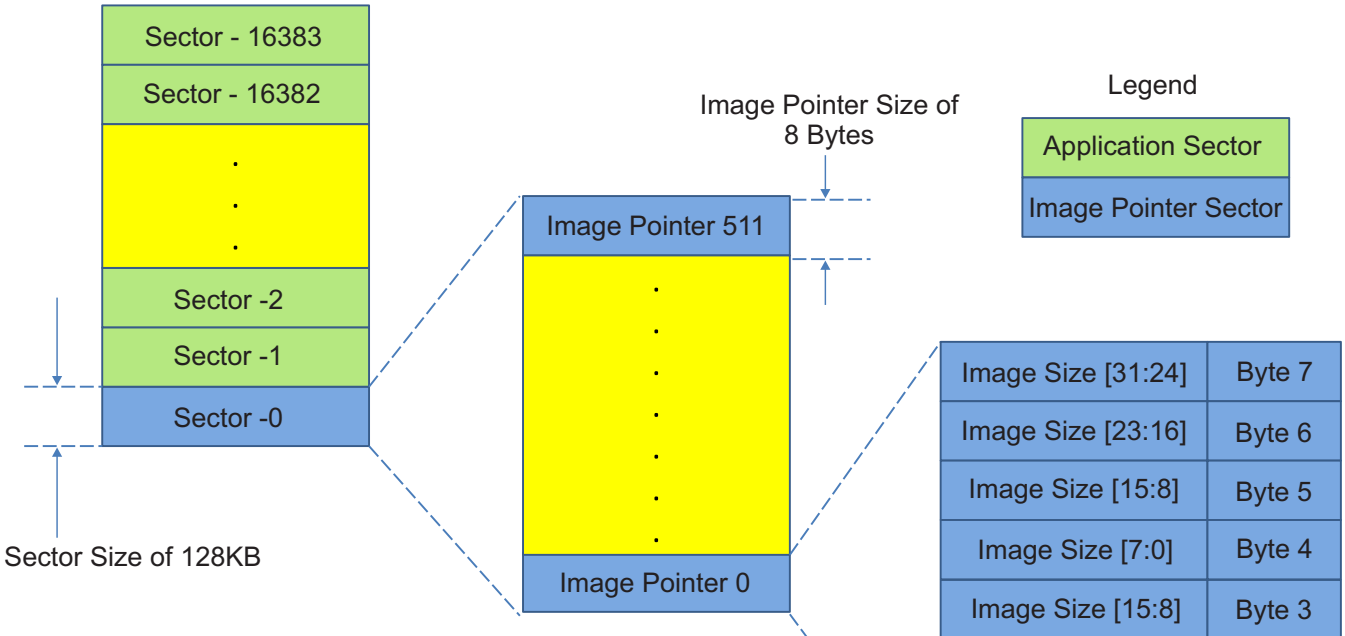


Image Status	Comment
0xFF	No Image
0xFE	Image Download to QSPI In Progress
0xFC	Image Valid

Figure 5. Application Image Pointer

The user can download an image to the parallel Flash memory, which is configured to execute from the EPI address space of 0x60020000. To download a new image from the Serial Console, press USR_SW1 down when powering up or resetting the connected LaunchPad. This action causes the boot loader to go into download mode. If the USR_SW1 is not pressed down, the boot loader will check the Sector-0 for a valid image pointer and execute the latest image if one is available. If no valid image pointer is found, then the boot loader will go back to download mode.

During execution, the boot loader disables the interrupts, updates the NVIC_VTABLE register to map to the external address map, and jumps to the external address space of 0x60020000. All subsequent code execution now happens in the external address space until the user resets the connected LaunchPad. Figure 6 shows how the example code operates.

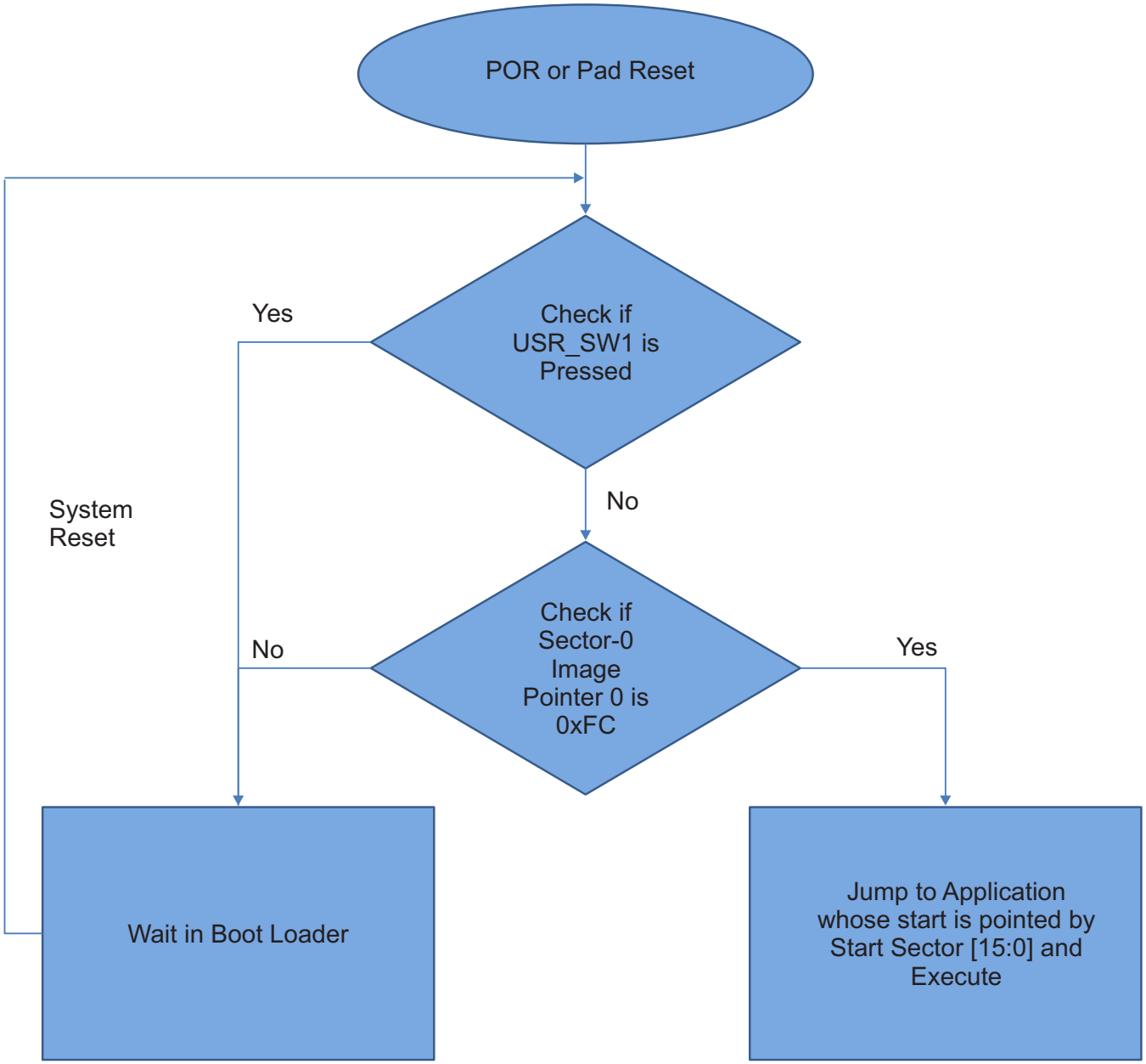


Figure 6. Parallel Flash Boot Flowchart

4.2 Parallel Flash Bare-Metal Example

The parallel Flash bare-metal example code configures the EPI module of the TM4C1294NCPDT to perform a device ID check, sector erase, erase check, program, and program verify operations. The test begins with the example code erasing a sector of 128KB or 64KB (based on the Flash size) and checking for the erase operation before performing programming and read operations. The test is parameterized so that the user defines the size of the external parallel Flash; all test parameters, including checking device ID and sector size, are updated automatically. Once the test completes, the CPU computes its available bandwidth when doing back-to-back access.

4.3 Parallel SRAM Bare-Metal Example

The parallel SRAM bare-metal example code configures the EPI module of the TM4C1294NCPDT to perform a write-and-read check of the external SRAM in byte, half word, and word access. The test performs the three types of accesses in both write-and-read mode, verifies the data, and computes the bandwidth available to the CPU when doing back-to-back access.

5 Test Setup

The test setup involves importing the example code into Code Composer Studio, building the project files, and executing the code on the EK-TM4C1294XL connected Launchpad. The test data section shows the performance results for Flash and SRAM.

5.1 Hardware Setup (Standalone Parallel Flash)

Figure 7 shows the setup in full view. The USB cable that comes with the EK-TM4C1294XL connected LaunchPad (shown on the left) provides power, JTAG for debug, and UART for serial console communication to the connected LaunchPad and the Flash memory extender daughter card. On the top side of the image, the Flash memory extender is connected to the header X11. Note that the Jumper J2 is connected for CS on EPI0S26 and J3 has the address pin A25 connected on EPI0S24.

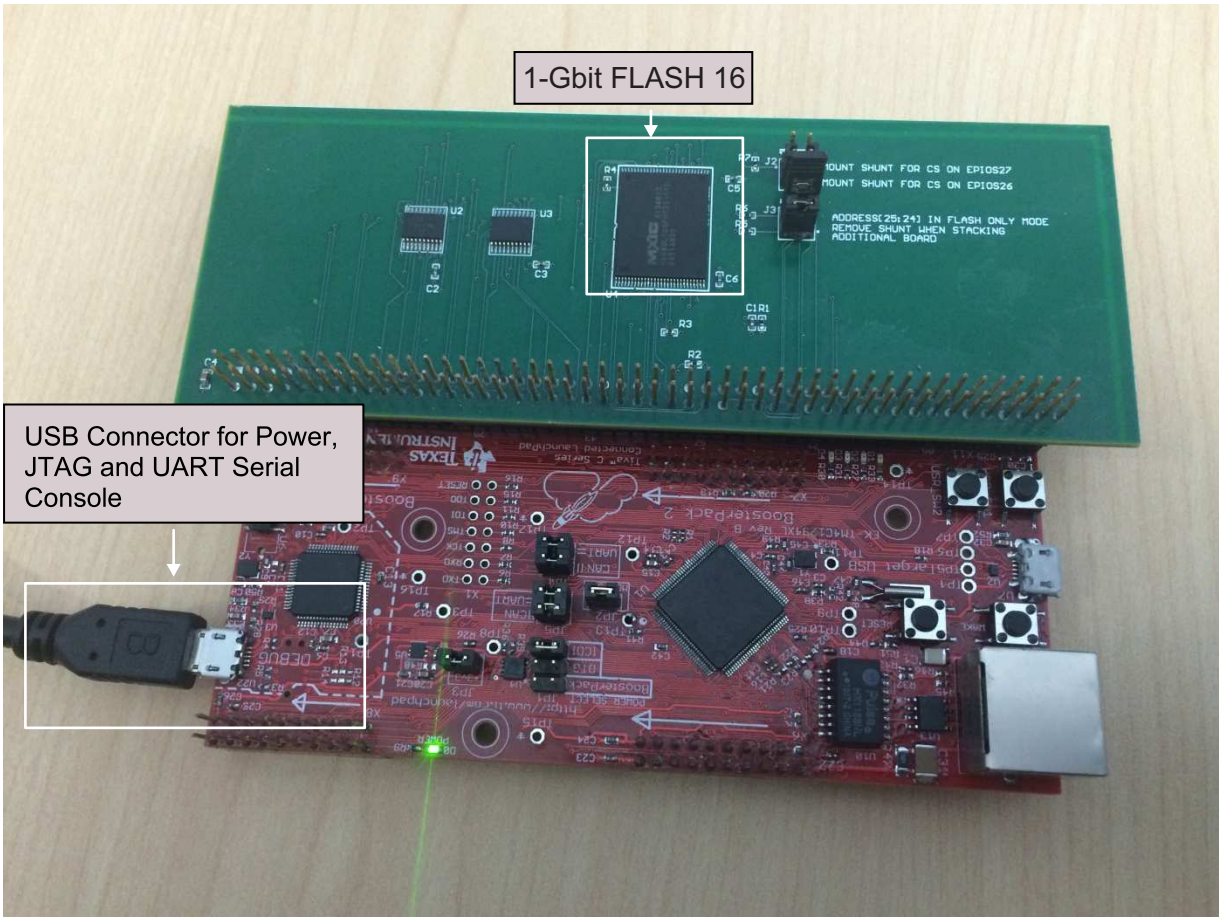


Figure 7. Full Test Assembly (128-Mbit–1-Gbit Flash)

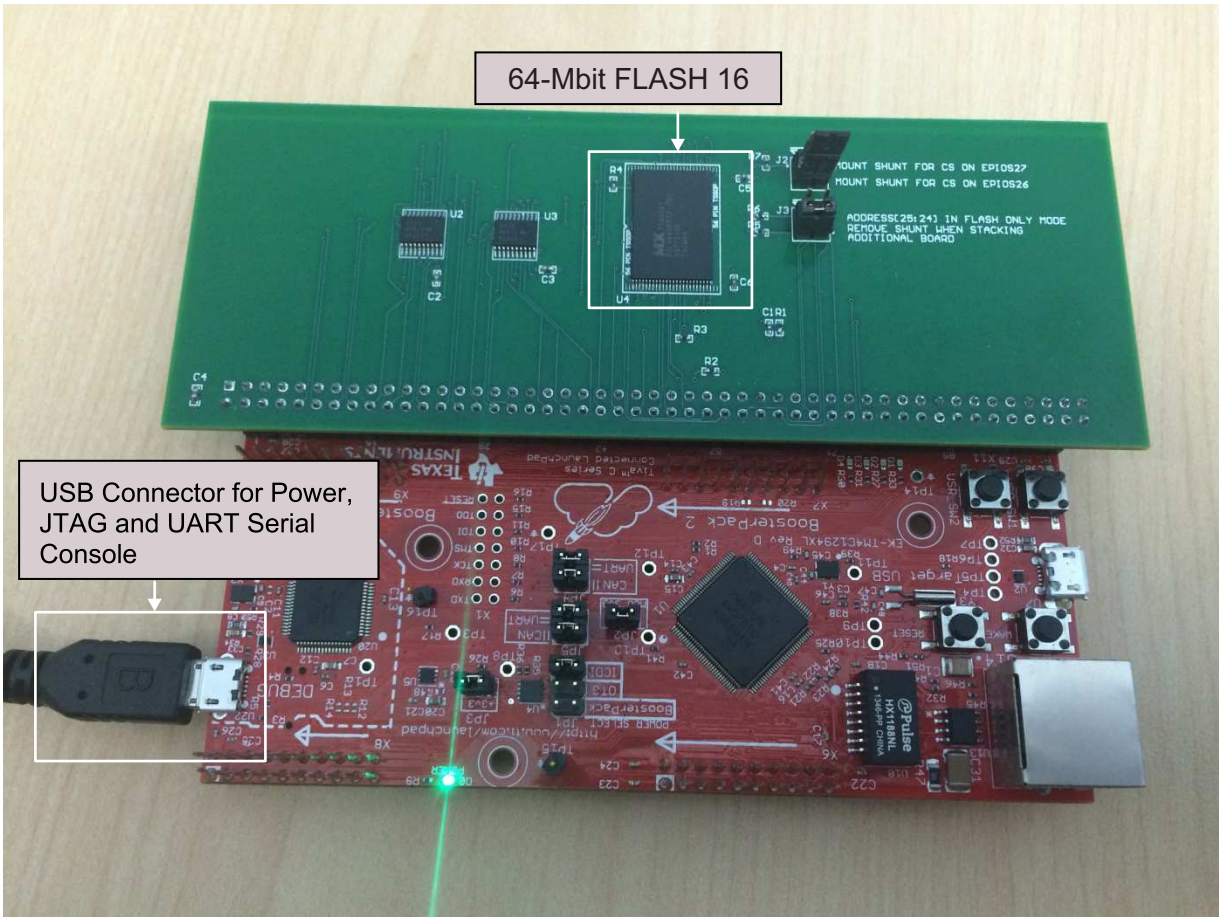


Figure 8. Full Test Assembly (8–64-Mbit Flash)

5.2 Hardware Setup (Standalone SRAM Flash)

Figure 9 shows the setup in full view. The USB cable that comes as a part of the EK-TM4C1294XL connected LaunchPad (shown on the left) provides power, JTAG for debug, and UART for serial console communication to the connected LaunchPad and the SRAM Memory Extender Daughter Card. On the top side of the image, the SRAM memory extender is connected to the headers X11. Note that the Jumper J1 is connected for CS on EPI0S26.

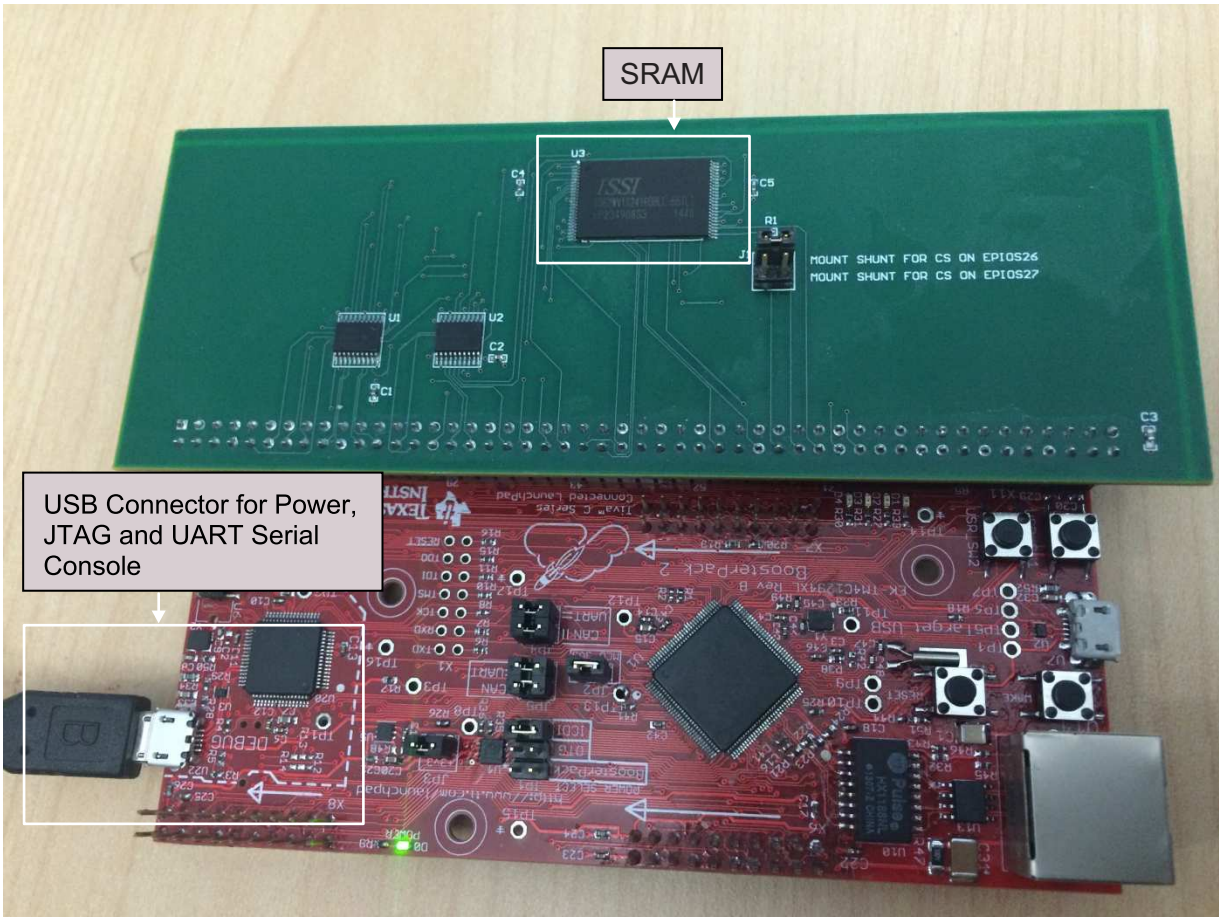


Figure 9. Full Test Assembly (16-Mbit SRAM)

5.3 Hardware Setup (Parallel Flash and SRAM Download and Execute)

Figure 10 shows the setup in full view. The USB cable that comes as a part of the EK-TM4C1294XL connected LaunchPad (shown on the left) provides power, JTAG for debug, and UART for serial console communication to the connected LaunchPad and the Flash-SRAM Memory Extender Stacked Daughter Card. On the top side of the image, the Flash memory extender is connected to the headers X11 and is the first board stacked. Also on the top side of the image, the SRAM memory extender is connected to the headers X11 and is the topmost stacked board.

For this configuration, the Flash board must have J2 shunt mounted for EPI0S26 as CS, J3 shunt mounted for address A24 on EPI0S24; and the SRAM board must have J1 shunt mounted for EPI0S27 as CS.

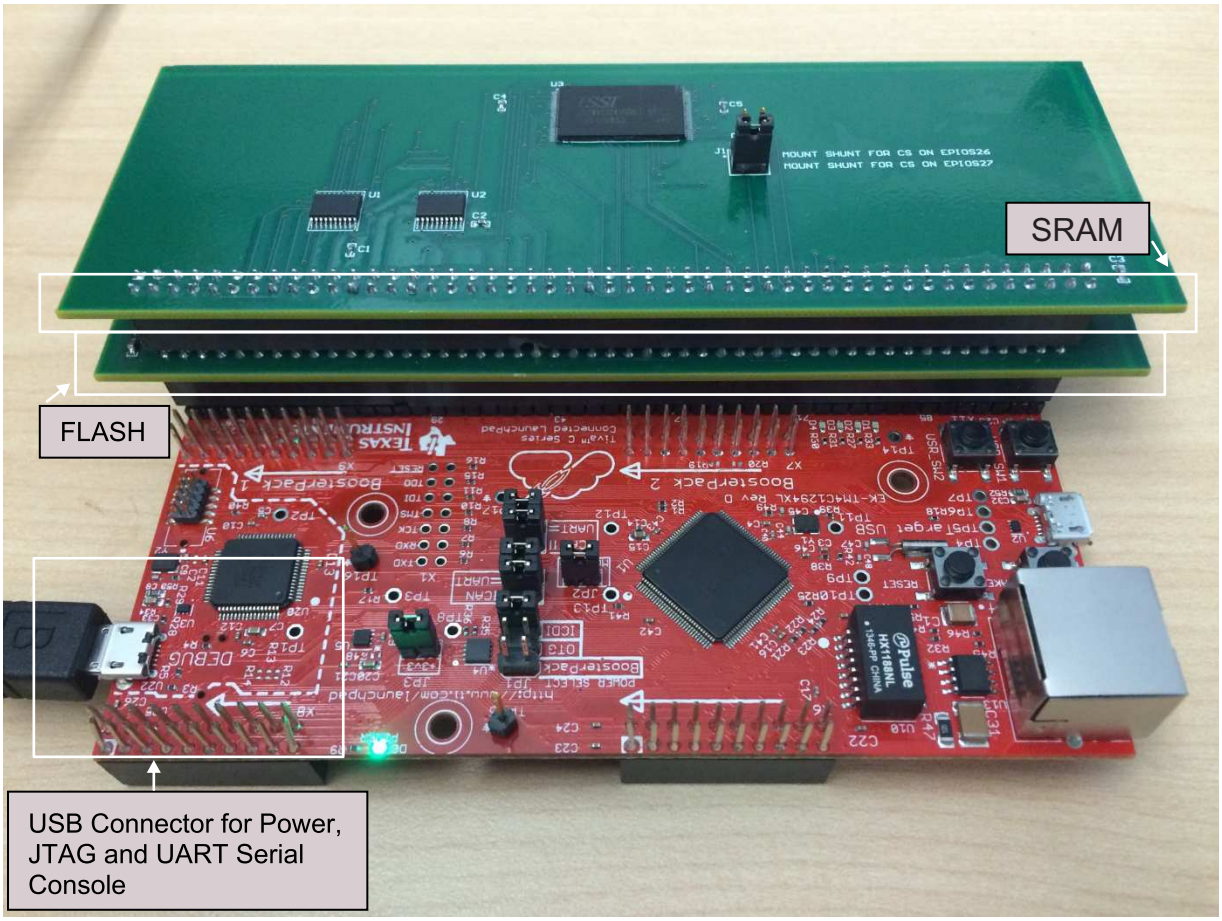


Figure 10. Full Test Assembly (Stacked Flash and SRAM)

NOTE: The usage of the examples assumes that a serial console application (PuTTY, TeraTerm, etc), Code Composer Studio v6.0.1, and TivaWare for C Series v2.1.0-12573 (or higher) have been downloaded and installed on the PC.

5.4 Software Setup (Standalone Parallel Flash)

1. Download the software examples zip package from the TI Design web page, and unzip the same on the local PC.
2. Launch Code Composer Studio v6.0.1 or later. To import the project, click File → Import → CCS Projects, then click “Next”. Browse to the directory where the software examples are kept. Select the project “ektm4c129_epiflash_example”, and then click “Finish”.

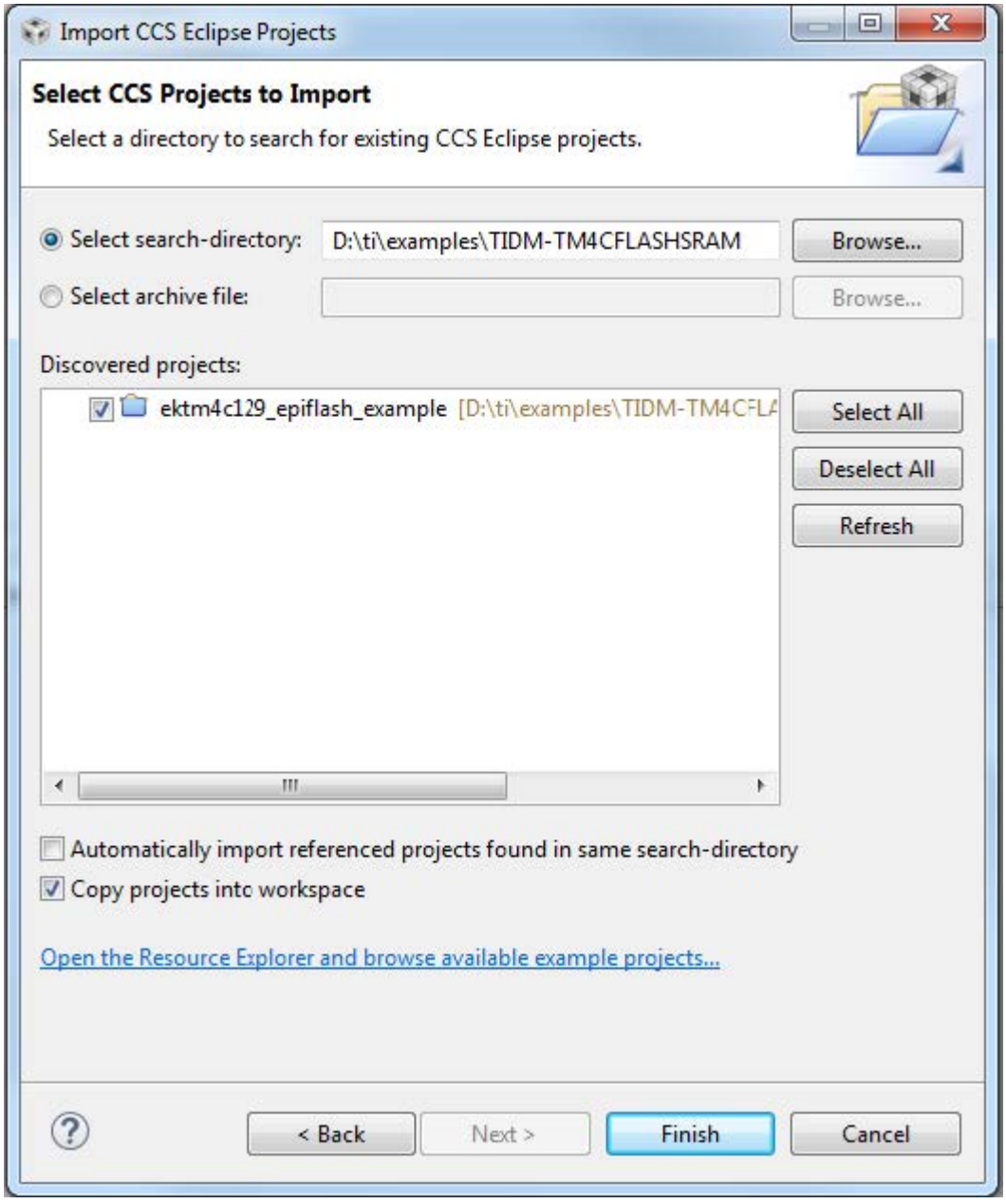


Figure 11. Importing the Software Example (Standalone Parallel Flash)

3. Build the project by right clicking the project and selecting “Rebuild Project”. The projects must compile without any errors. The example code has a DEFINE that by default selects 64-Mbit memory. The user may change the DEFINE FLASH16_MEMSIZE_MBIT to 8, 16, 32, 64, 128, 256, 512, or 1024 according to the size of actual memory assembled on the board; the code will check ID and resize sector size.

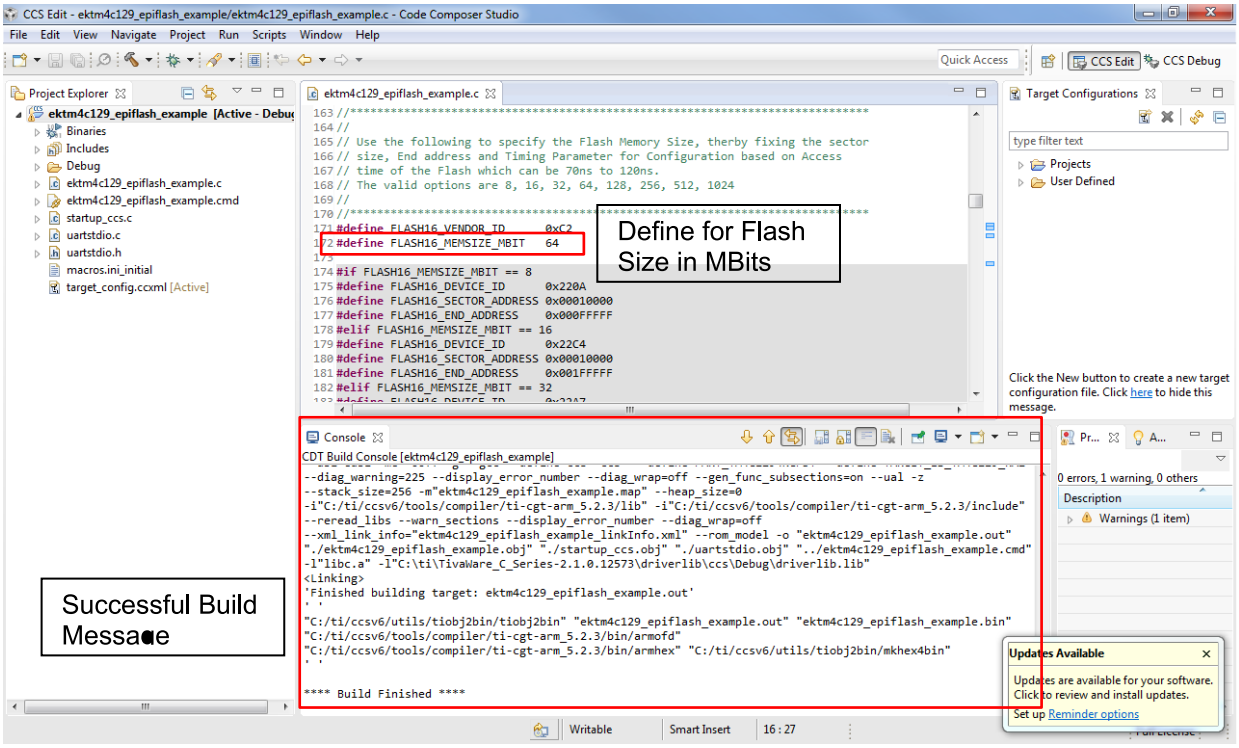


Figure 12. Compiling the Software Example (Standalone Parallel Flash)

4. Run the example “ektm4c129_epiflash_example” by pressing the Debug button, which will load the code into the TM4C1294NCPDT Flash. Press the Play button after the code has loaded. If program-verify checks pass, the result of the ID check will show on the serial console, followed by sector erase, erase-verify, program, and performance numbers.

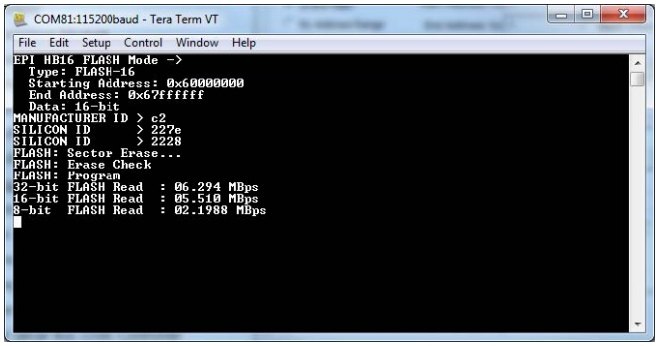


Figure 13. Expected Console Output (Standalone Parallel Flash 1 Gbit)

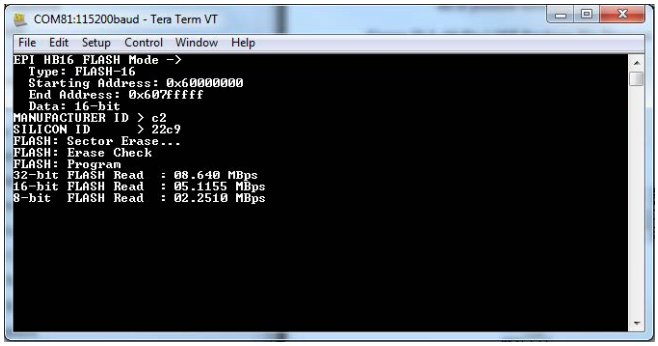


Figure 14. Expected Console Output (Standalone Parallel Flash 64 Mbit)

5.5 Software Setup (Standalone Parallel SRAM)

1. Download the software examples zip package from the TI Design web page, and unzip the same on the local PC.
2. Launch Code Composer Studio v6.0.1 or later. To import the project, click File → Import → CCS Projects, then click “Next”. Browse to the directory where the software examples are kept. Select the project “ektm4c129_episram_example”, and then click “Finish”.

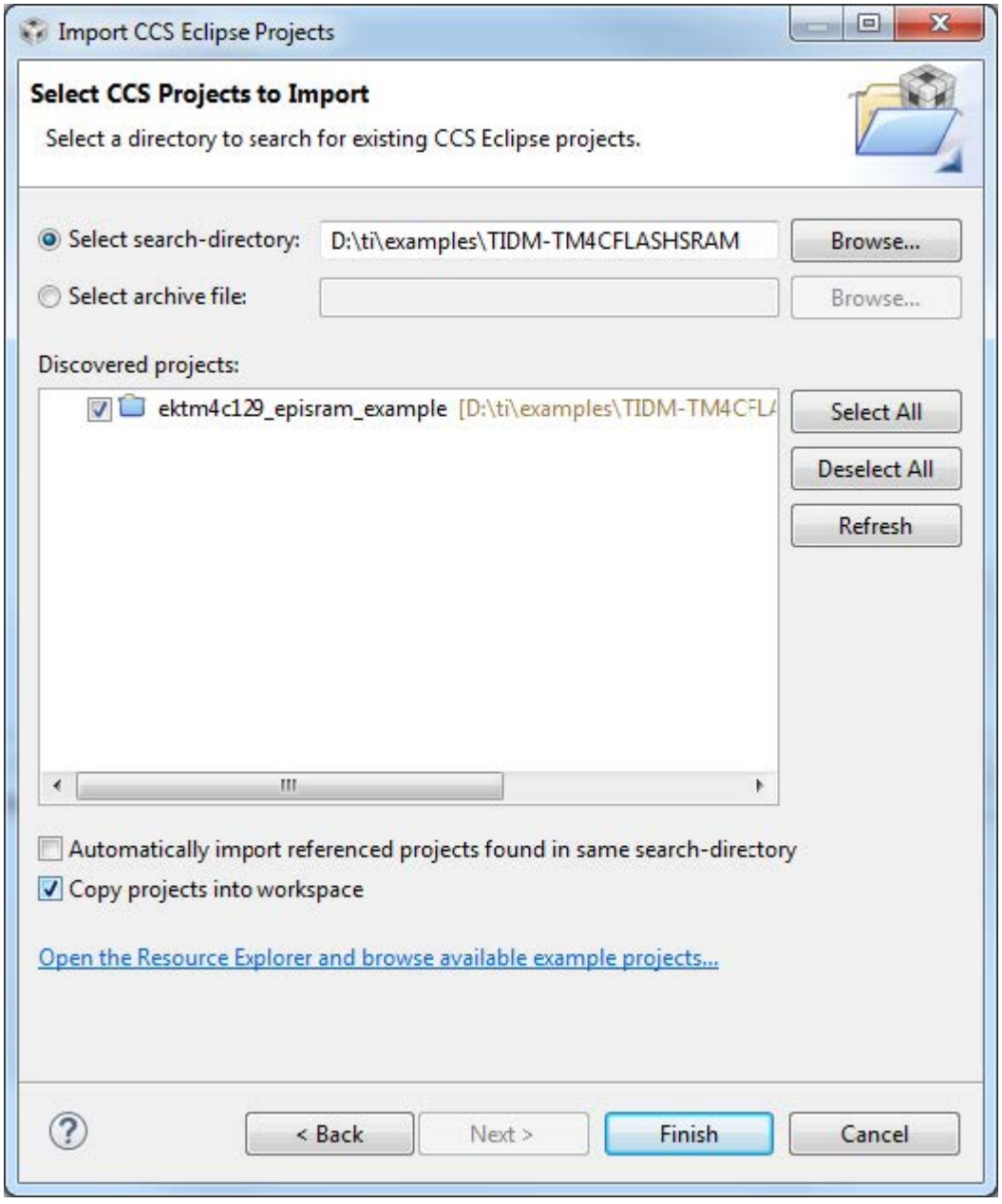


Figure 15. Importing the Software Example (Standalone parallel SRAM)

3. Build the project by right clicking the project and then selecting “Rebuild Project”. The projects must compile without any errors.

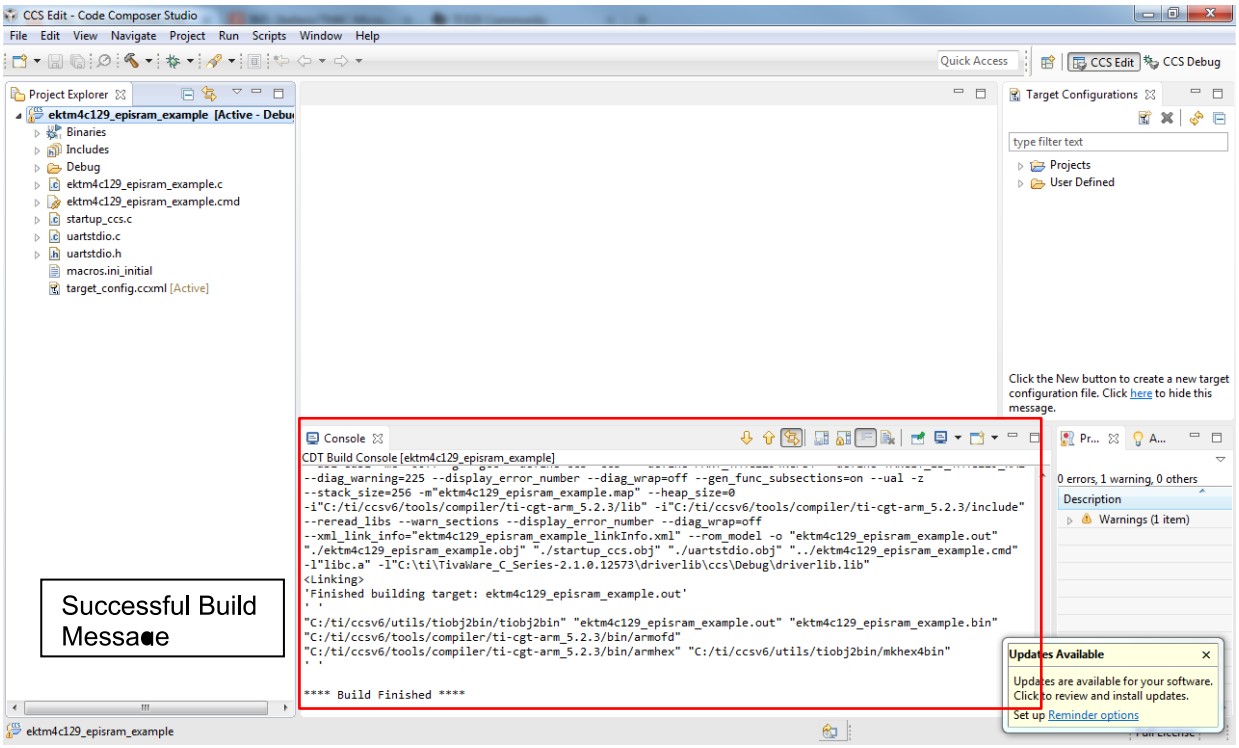


Figure 16. Compiling the Software Example (Standalone parallel SRAM)

4. Run the example “ektm4c129_episram_example” by pressing the Debug button, which will load the code into the TM4C1294NCPDT Flash. Press the Play button after the code has loaded. The performance numbers will show on the serial console if write-and-read checks pass.

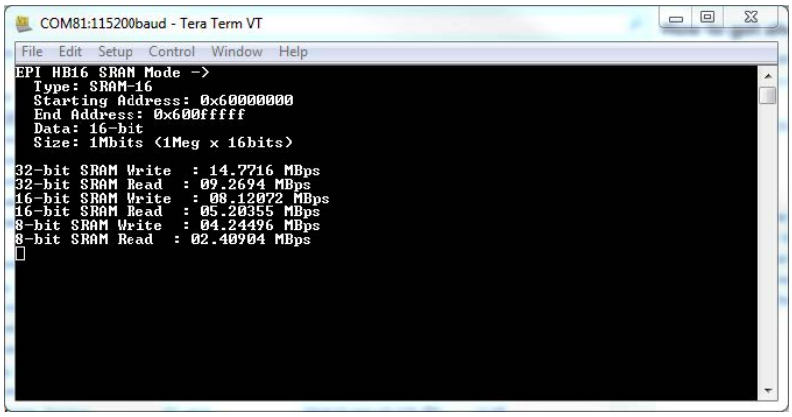


Figure 17. Expected Console Output (Standalone parallel SRAM)

5.6 Software Setup (Parallel Flash and SRAM Download and Execute)

1. Download the software examples zip package from the TI Design web page, and unzip the same on the local PC
2. Launch Code Composer Studio v6.0.1 or later. To import the project, click File → Import → CCS Projects, then click “Next”. Browse to the directory where the software examples are kept. Select the projects “ektm4c129_epiflash_bootloader” and “ektm4c129_epiflash_boot_demo”, then click “Finish”.

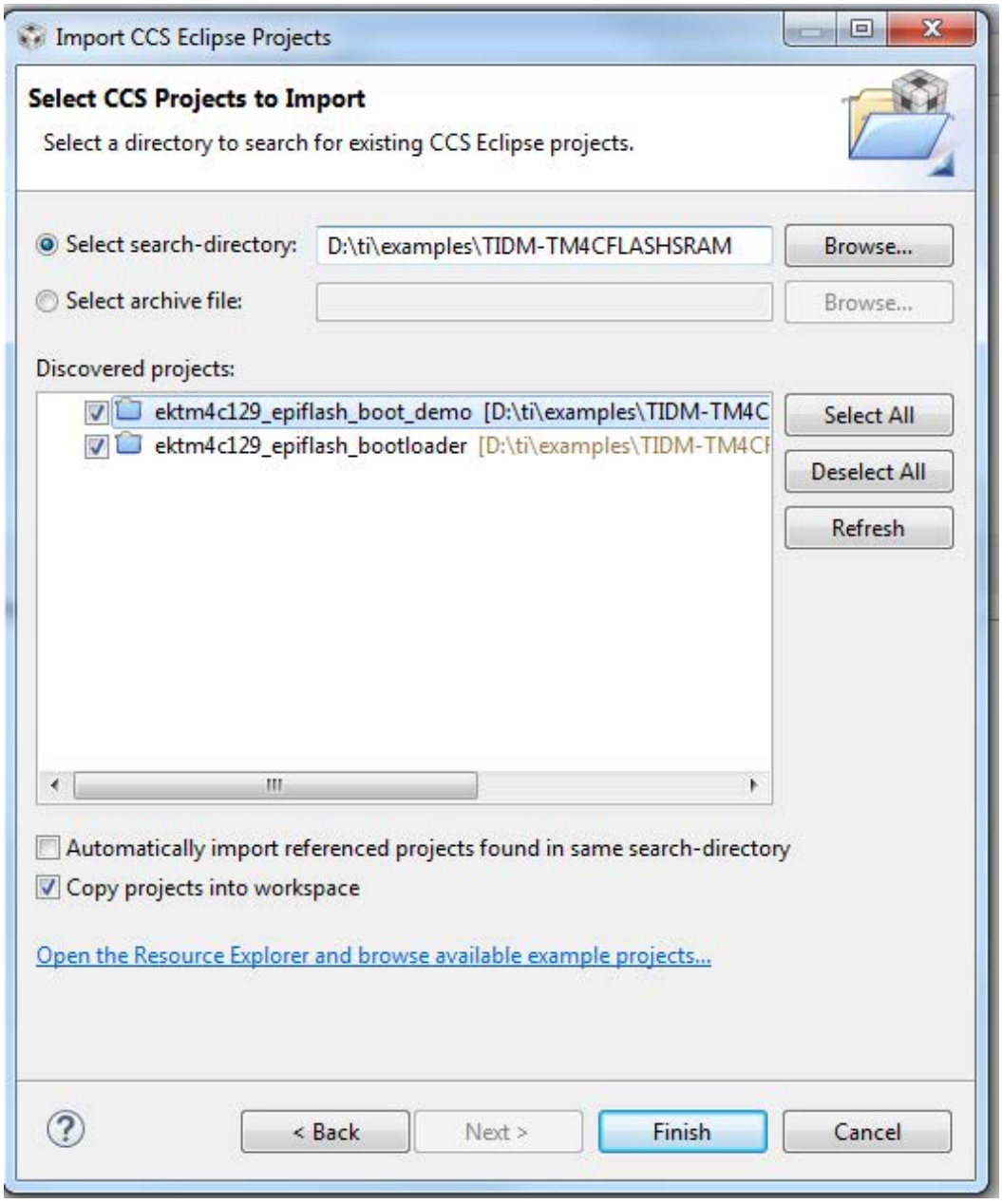


Figure 18. Importing the Software Examples (Parallel Flash and SRAM Download and Execute)

3. Build each of the projects by right clicking the project and then selecting “Rebuild Project”. The projects must compile without any errors.

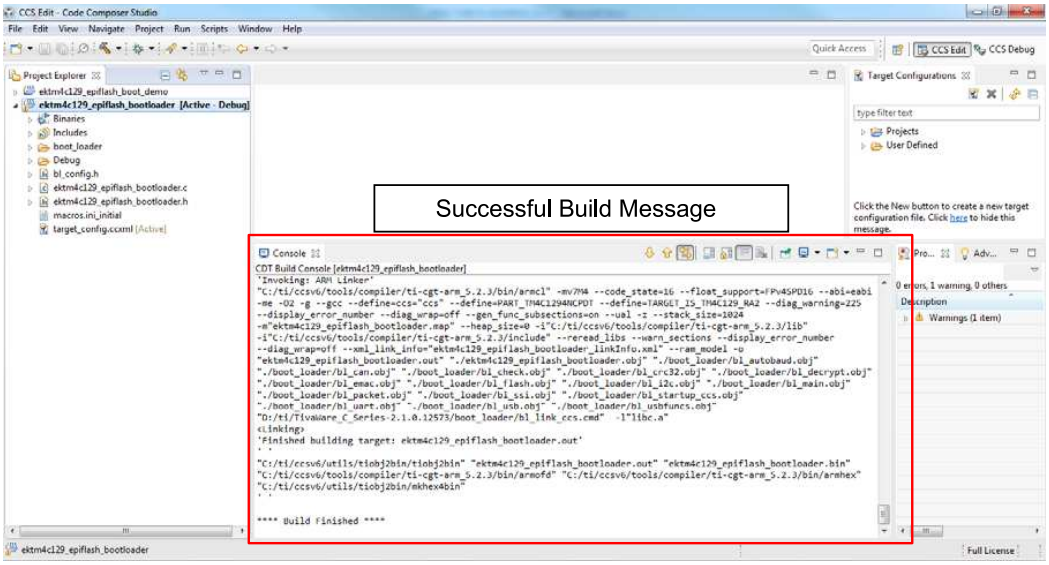


Figure 19. Compiling the Software Examples (Parallel Flash and SRAM Download and Execute)

4. Download the main boot loader example “ektm4c129_epiflash_bootloader” by using the LMFlashProgrammer on a fully erased Flash of the TM4C1294NCPDT connected LaunchPad. After performing checks for Flash and SRAM memory, the boot loader activates UART0 to download an image to the external parallel Flash. Figure 20 and Figure 21 show the setting of the LMFlashProgammer.

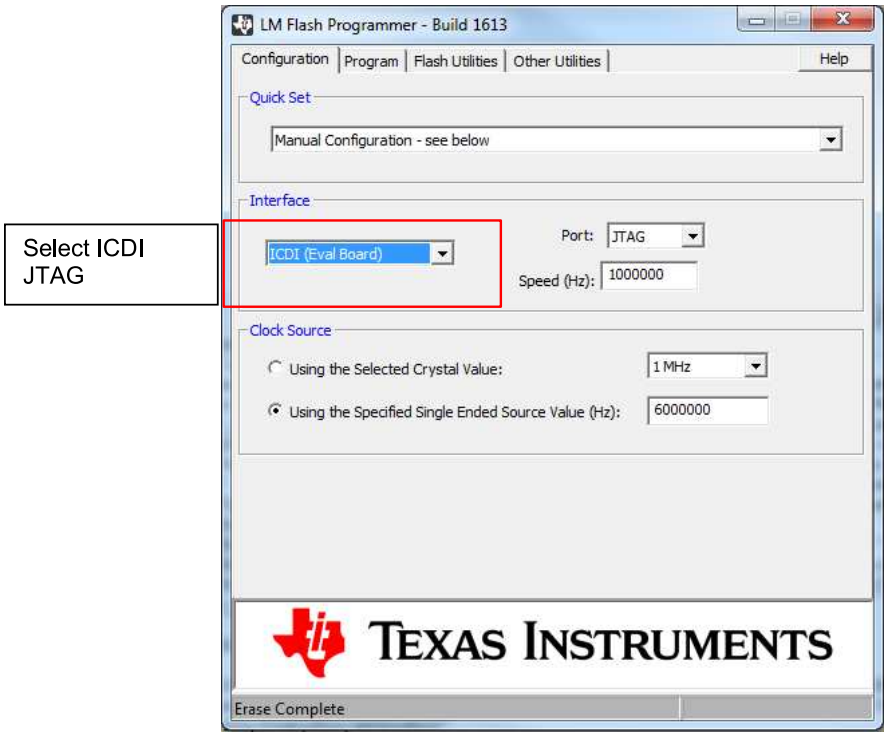


Figure 20. LMFlashProgrammer Configuration Tab for Boot Loader

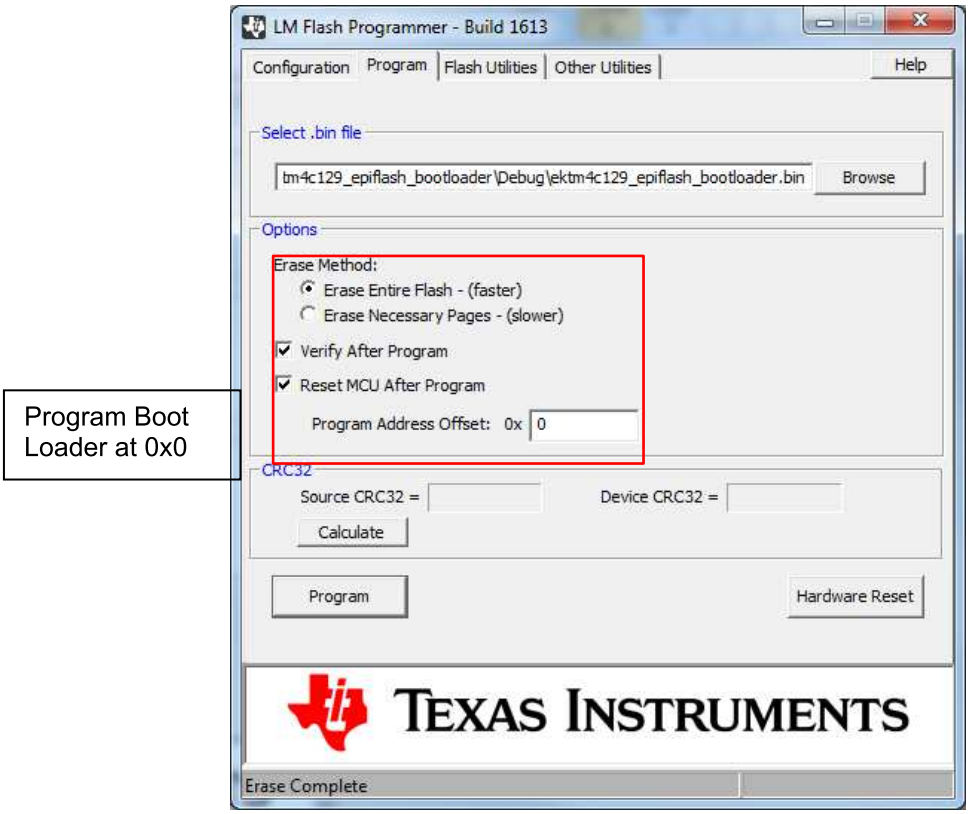


Figure 21. LMFlashProgrammer Program Tab for Boot Loader

5. Download the demo example “ektm4c129_epiflash_boot_demo” by using the LMFlashProgrammer in serial mode. Ensure that the “Disable Auto Baud Support” is checked, the correct COM port is selected, and the transfer size is kept to 64 bytes. On the Program tab, select the Program Address Offset as the start of a sector of parallel Flash other than Sector-0. Figure 22 and Figure 23 show the setting of the LMFlashProgammer for downloading the demo code.

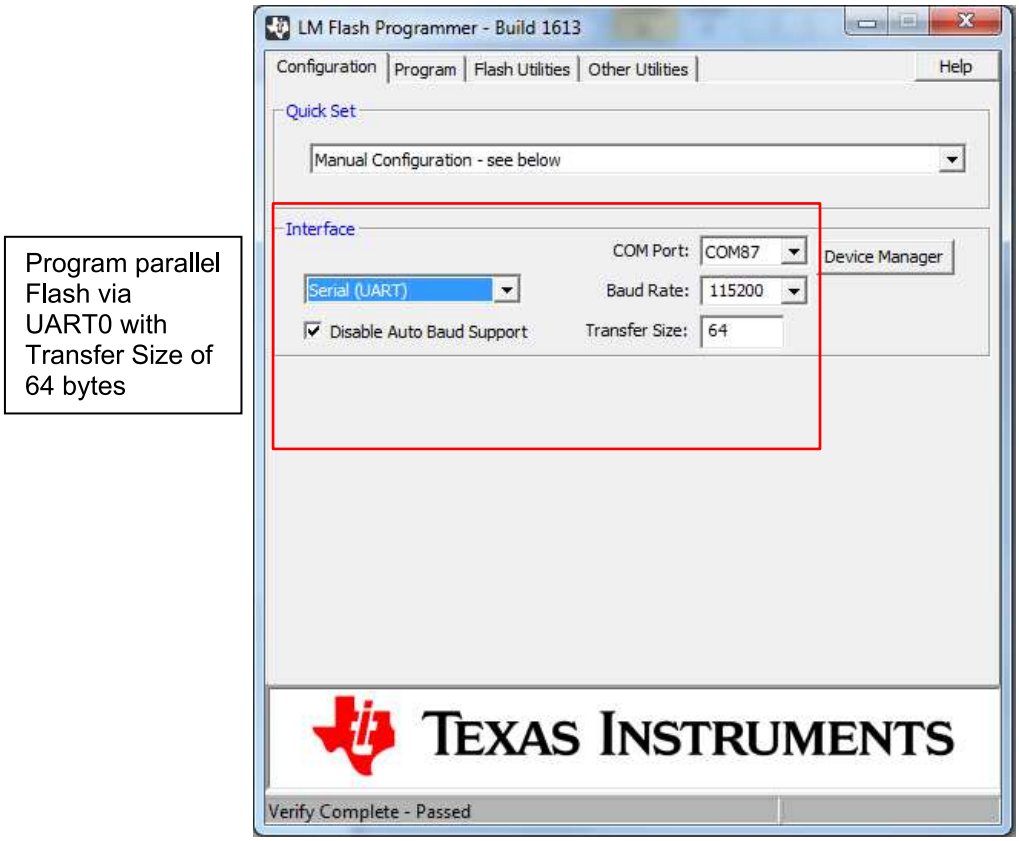


Figure 22. LMFlashProgrammer Configuration Tab for Boot Demo

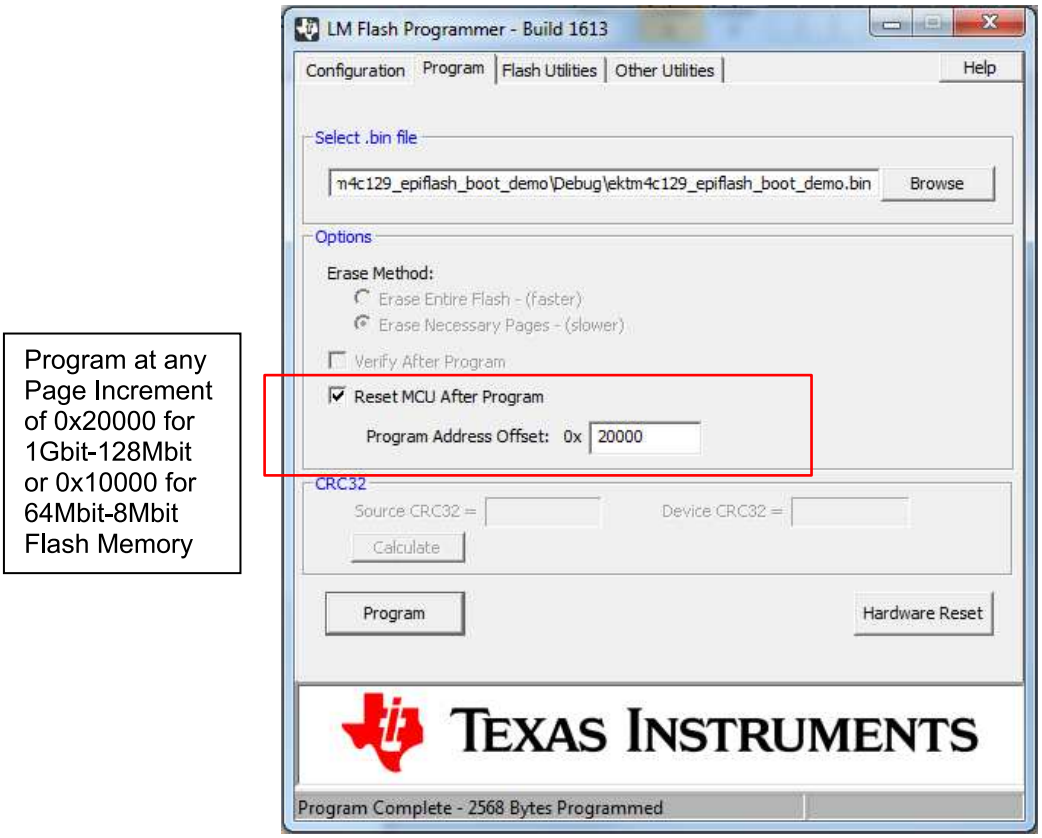


Figure 23. LMFlashProgrammer Program Tab for Boot Demo

6. Once the demo code downloads, the LED D3 will start blinking. Use USR_SW2 to accelerate the blinking rate; use USR_SW1 to reduce the blinking rate.

6 Test Data

The following section highlights the performance data for parallel SRAM in single-chip-select mode.

6.1 Parallel Flash Performance Data

The example code for the Flash performance test is a bare-metal example and may only be used as a reference when all hardware connections are correct.

For the EK-TM4C129X EVM with a system frequency of 120 MHz, and the minimum access time for the Flash device being 70 ns-110 ns (depending on the memory size), the bandwidth for read operations via the CPU is as follows.

Table 2. External Flash Performance

	FLASH READ THROUGHPUT (1 Gbit)	FLASH READ THROUGHPUT (64 Mbit)
8-bit access	2.1988 MBps	2.251 MBps
16-bit access	5.51 MBps	5.1155 MBps
32-bit access	6.294 MBps	8.64 MBps

The number mentioned above are the lowest number. Using uDMA and Read FIFO, the performance of the parallel interfaces can be significantly higher.

6.2 Parallel SRAM Performance Data

The example code for the SRAM performance test is a bare-metal example and may only be used as a reference when all hardware connections are correct.

For the EK-TM4C129X EVM with a system frequency of 120 MHz, and the minimum access time for the SRAM device being 45 ns, the bandwidth for different write-and-read operations via the CPU is as follows.

Table 3. External SRAM Performance

	SRAM WRITE THROUGHPUT	SRAM READ THROUGHPUT
8-bit access	4.2 MBps	2.4 MBps
16-bit access	8.12 MBps	5.2 MBps
32-bit access	14.77 MBps	9.26 MBps

These numbers mentioned in Table 3 are the lowest numbers. Using uDMA and Read FIFO, the performance of the parallel interfaces can be significantly higher.

7 Design Files

7.1 Schematics

To download the Schematics for the board, see the design files at TIDM-TM4CFLASHSRAM.

7.2 Bill of Materials

To download the bill of materials (BOM), see the design files at TIDM-TM4CFLASHSRAM.

7.3 Layer Plots

To download the layer plots, see the design files at TIDM-TM4CFLASHSRAM.

7.4 Altium Project

To download the Altium project files, see the design files at TIDM-TM4CFLASHSRAM.

7.5 Gerber Files

To download the Gerber files, see the design files at TIDM-TM4CFLASHSRAM.

7.6 Software Files

To download the software files, see the design files at TIDM-TM4CFLASHSRAM.

8 References

1. ISSI 16-Mbit SRAM Memory http://www.issi.com/WW/pdf/62-65WV102416DALL-DBLL.pdf
2. Macronix 1-Gbit–8-Mbit Flash Memory http://www.macronix.com/CachePages/en-us-Product-NORFlash-ParallelFlash.aspx

9 About the Author

AMIT ASHARA is an Application Engineer at Texas Instruments, where he is responsible for developing applications for the TM4C12x family of high-performance microcontrollers. Amit brings to this role his extensive experience in high-speed digital and microcontroller system-level design expertise. Amit earned his Bachelor of Engineering (BE) from University of Pune, India.

IMPORTANT NOTICE FOR TI REFERENCE DESIGNS

Texas Instruments Incorporated ("TI") reference designs are solely intended to assist designers (“Buyers”) who are developing systems that incorporate TI semiconductor products (also referred to herein as “components”). Buyer understands and agrees that Buyer remains responsible for using its independent analysis, evaluation and judgment in designing Buyer’s systems and products.

TI reference designs have been created using standard laboratory conditions and engineering practices. **TI has not conducted any testing other than that specifically described in the published documentation for a particular reference design.** TI may make corrections, enhancements, improvements and other changes to its reference designs.

Buyers are authorized to use TI reference designs with the TI component(s) identified in each particular reference design and to modify the reference design in the development of their end products. HOWEVER, NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY THIRD PARTY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT, IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI REFERENCE DESIGNS ARE PROVIDED "AS IS". TI MAKES NO WARRANTIES OR REPRESENTATIONS WITH REGARD TO THE REFERENCE DESIGNS OR USE OF THE REFERENCE DESIGNS, EXPRESS, IMPLIED OR STATUTORY, INCLUDING ACCURACY OR COMPLETENESS. TI DISCLAIMS ANY WARRANTY OF TITLE AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT, QUIET POSSESSION, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS WITH REGARD TO TI REFERENCE DESIGNS OR USE THEREOF. TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY BUYERS AGAINST ANY THIRD PARTY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON A COMBINATION OF COMPONENTS PROVIDED IN A TI REFERENCE DESIGN. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, SPECIAL, INCIDENTAL, CONSEQUENTIAL OR INDIRECT DAMAGES, HOWEVER CAUSED, ON ANY THEORY OF LIABILITY AND WHETHER OR NOT TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, ARISING IN ANY WAY OUT OF TI REFERENCE DESIGNS OR BUYER’S USE OF TI REFERENCE DESIGNS.

TI reserves the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques for TI components are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers’ products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers’ products and applications, Buyers should provide adequate design and operating safeguards.

Reproduction of significant portions of TI information in TI data books, data sheets or reference designs is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards that anticipate dangerous failures, monitor failures and their consequences, lessen the likelihood of dangerous failures and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in Buyer’s safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed an agreement specifically governing such use.

Only those TI components that TI has specifically designated as military grade or “enhanced plastic” are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components that have ***not*** been so designated is solely at Buyer's risk, and Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265