ELECTRONICS (HTTPS://CIRCUITDIGEST.COM/ELECTRONICS-PROJECTS)

Getting Started with STM8S using STVD ar Cosmic C Compiler (/microcontroller-projects/getting-started-with-stm8s-using-stvd-and-cosmic-c-compiler)

By (page_author.html)Aswinth Raj (/users/aswinth-raj) Jun 15, 2020

Getting Started with STM8S using STVD and Cosmic C Compiler

The **STM8** is a series of **8-bit Microcontrollers** from **STMicroelectronics** that has become a common choice of Microcontroller for cost-sensitive product development. I have previously work with AVR (https://circuitdigest.com/avr-microcontroller-projects), PIC (https://circuitdigest.com/pic-microcontroller-projects), and a few other ARM (https://circuitdigest.com/tags/arm-7) Cortex Microcontrollers, but surely in some applications, they become overkill and increase the BOM cost considerably. Recently, after tearing down a few cheap Chinese products, I found that most of them had an **STM8 microcontroller** inside it. The IC is not only cheap but also packed it with lots of features and options making it suitable to be used in many different applications. For instance, the **STM8S** series of controllers (which we will learn in this tutorial series) is a general-purpose controller but there are other series like the **STM8A** for Automotive Designs and **STM8L** for Low Power battery-powered designs which extends the application of these controllers.

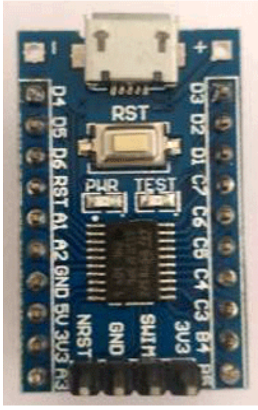
In this series of tutorials, we will learn how to program the **STM8S microcontrollers**, more specifically the **STM8S103F3P6**. The reason for getting started with this specific part number is just for its price to performance factor. Like!! just look at it just for 0.25$ (approx. 20 Rupees) we get an 8-bit microcontroller with 16Mhz Internal Oscillator, 8kB Flash, 10-bit ADC, UART, SPI, and I2C. Who would not be intrigued? Also, the STM8S103F3P6 is widely popular, and hence as a beginner, you will find enough support if you need help outside the tutorials. Starting from this, we will have a sequence of tutorials, similar to the PIC Programming Tutorials (https://circuitdigest.com/forums/embedded/recommendation-pic-microcontroller-get-started) we will learn how to program this controller from a basic blink to various display and sensor inter You can also check out the STM32F103C8 (https://circuitdigest.com/tags/stm32f103c8) Tutor projects if you are interested in 32-bit STM Microcontrollers.

Selecting the Hardware and Software Development Tools

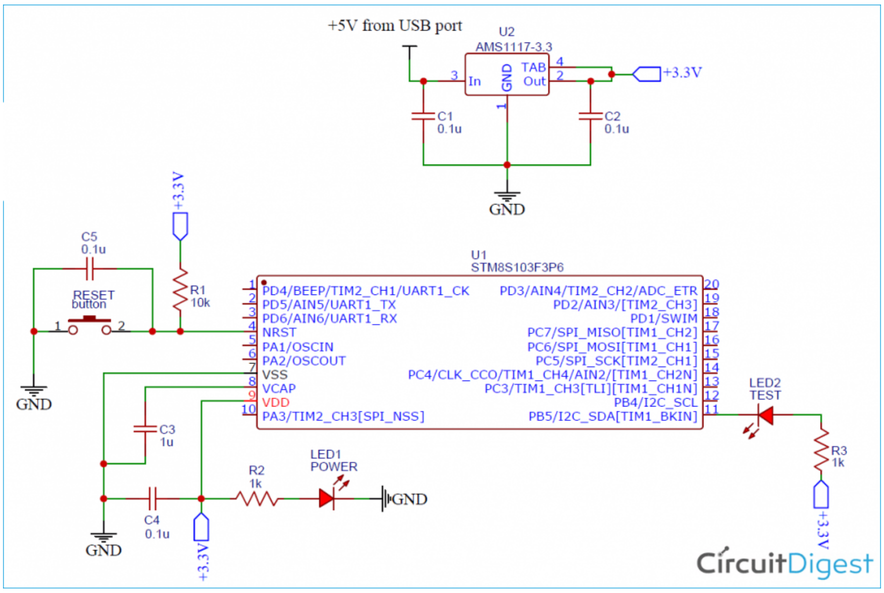
Now that we have deicide the microcontroller, we will have to select hardware and so development tools to begin our learning process. There are many options, here I have selected th that are free to use and easily available to everyone.

STM8S103F3P6 Development

The official development board for ST's 8-bit Microcontroller is called the STM8 Discovery kit, will use another low-cost development board that you can easily find with a local supplier or or store. The image of the **STM8S103F3P6 Development board** that we will use in this tutorial is below.

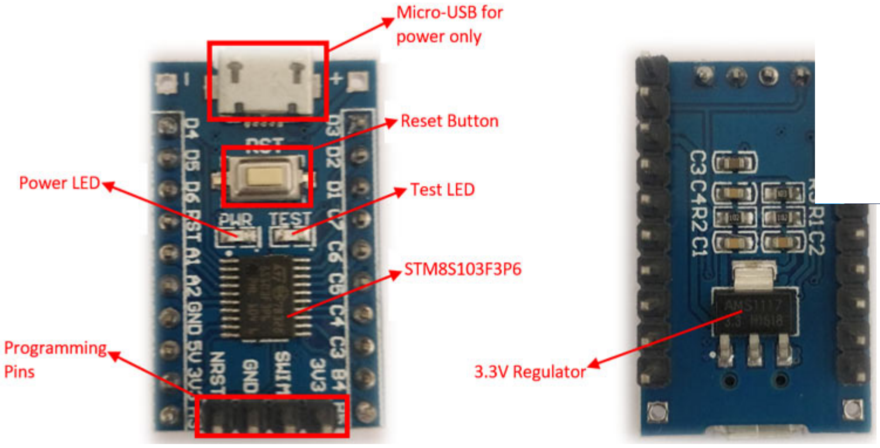

RST
PWR
TEST
D4 D5 D6 RST A1 A2 GND 5V 3U3 A3
D3 D2 D1 C7 C6 C5 C4 C3 B4 B5
NRST
GND
SWIM
3U3

The board only consists of the bare minimum components that are required to begin development. It is breadboard friendly and suitable for compact applications. The internal circuit diagram for the development board is shown below.



(/fullimage?i=circuitdiagram_mic/STM8S103F3P6-Board-Circuit-Diagram.png)

Taking a closer look, you will notice, apart from the STM8S103F3p6 controller, we have a reset bu reset the controller. A power led and a test led connected to PB5 (Port B pin 5) and an AMS1117 \ regulator that converts the 5V from a USB port to 3.3V for the controller. The controller can a made to operate on 5V if required. The different parts on the controller are marked below. We als 4 programmer pins namely 3.3V, GND, SWIM, and NRST which can be used to easily program and our microcontroller.

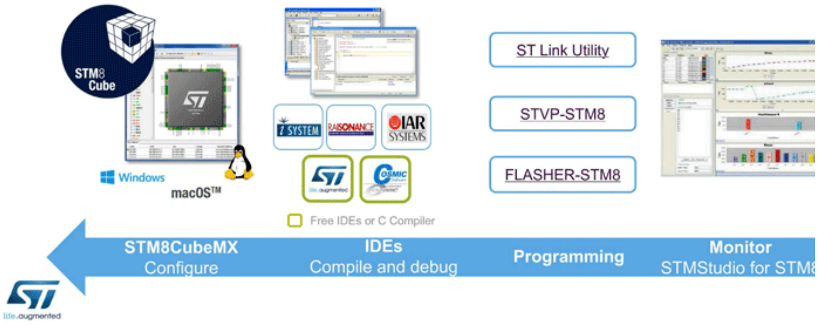


STM8S103F3P6 Programmer

For programming our controller, we will be using the **ST-LINK v2** which is cheap and easily available online. There are many variants (metal, plastic, gold, pink, bare board) of this ST-LINK v2 board and all serve the same purpose. Mine is shown below but you can use any type of **ST-LINK programmer**, the pinout will differ, so just make sure you read the pinouts on the casing correctly. There is also ST-LINK V3 from ST-Microelectronics itself which can be used for some serious debugging. We won't be using that for now since it's a bit expensive, we will save it for the future.

STM8S103F3P6 IDE and Compiler

Selecting the right IDE and Compiler for STM8S103F3P6 is perplexing, simply because there are a lot of options to choose from. The official tools that are available for the STM8 Microcontroller are shown in the below image.



On the Configuration side, we have the **STM8CubeMx**, I tried the software and it was a comp down for me. Unlike the **STM32Cube,** the STM8Cube cannot auto-generate code. It can only g pin mapping for visualization purposes and might come in handy if you are porting your code fr controller to another. Other than that, I would not use it. So in this tutorial series, we won't be usin

For the IDE and Compiler, we have a lot of options to choose from. The top two options are **IAR workbench** and **ST Visual Develop (STVD),** both the software feels like they are from t but after some time playing with it, I found the STVD is a good choice only because it is total Similarly, for the Compiler, we will be using the **Cosmic C Compiler**, again only because it is tota Another reason for choosing this IDE and Compiler is that, once we get familiar with the environ should be easy to use any other 8-bit Microcontroller from ST without much effort. We will get in to install and set-up STVD with Cosmic C compiler later in this article.

For Flashing, we will be using **ST Visual Programmer (STVP)**, this tool will automatically get ir when we install STVD. It will be integrated in the IDE itself, helping us to quickly program and debug. The final software will be **STMStudio** which is an STM8 Monitoring software. The software can help with real-time debugging of STM8 and has some cool features like monitoring variable value, plotting graph, etc. I have not experimented enough with this software. And, at least for the first few tutorials, we won't be using this software since we won't need big debugging requirements.

STM8S Standard Peripheral Library: ST Microelectronics provides a set of libraries to make code development easier for the STM8S Microcontrollers, this library is called the "**Standard Peripheral Library**" or SPL in short. The library is cool except for that it is written by keeping in mind all the possible controllers in the STM8S/A 8-bit family and not just the STM8S103F3P6 controller which we will be using. Hence, we have to make some tweaks here and there to make it work (which I will share later). But still, I believe it worth giving it a try because it can make development very fast, and hence, we will be using it in our tutorial.

If you do not want to use the library, you have to directly access the Registers of the controller or do bare metal assembly programming. Both of these are fun, provided you have the time to learn and use it. My idea is to use the SPL library wherever it works well and then also work on register and assembly level if required. Let's try not to re-invent the wheel!

STM8S with Arduino using Sduino

Discussing the software options will never be complete if I do not mention the **Arduino IDE** support STM8S. Yes, the same STM8S103F3P6 board can be programmed directly from Arduino IDE using Sduino (https://github.com/tenbaht/sduino), thanks to Michael Mayer (https://github.com/tenbaht). But, the project is still in the development stage and would likely take some time and community support for Sduino to provide complete platform support. However, just within a few minutes of playing with Sduino, I started to like it and hence decide to also do a separate article on how to program STM8S Microcontroller with Arduino. I will link the article here when it is ready. That article will discuss why and why not you should use the Arduino IDE for programming your STM8S Microcontrollers.

So, these are my choices for Software and Hardware, let me know in the comment section if you think some other software would have been a better choice and why. I would like to explore other options as well.

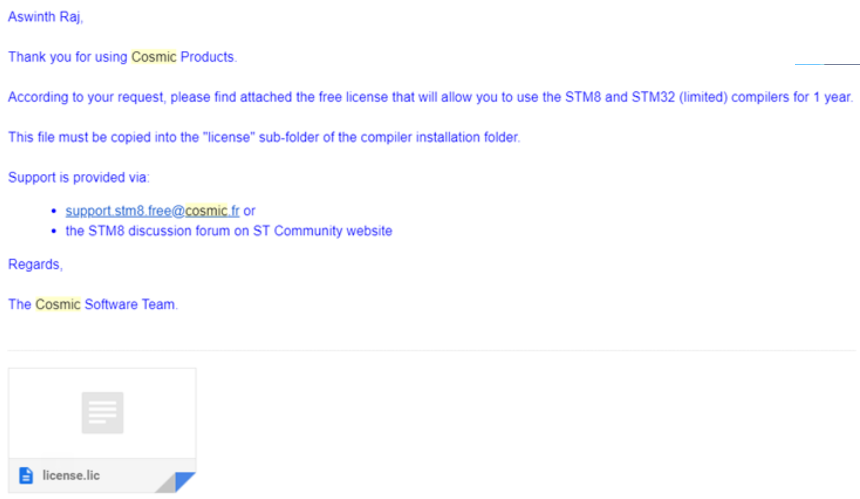
Downloading Required Software's

Now that we have decided on the software will we use for this tutorial, let's go ahead and download them using the following link. All the software is free to download and use, you would just l register a free account on ST and Cosmic if you had not already.

- ST Visual Develop (STVD) (https://www.st.com/en/development-tools/stvd-stm8.html)
- Cosmic C Compiler (https://www.cosmicsoftware.com/download_stm8_free.php)
- Standard Peripheral Library (https://github.com/CircuitDigest/STM8S103F3_SPL)
- STM8Cube Mx (https://www.st.com/en/development-tools/stm8cubemx.html) (Optiona
- STMStudio for STM8 (https://www.st.com/en/development-tools/stm-stm8.html) (Optional)

Setting up the STVD and Cosmic C Compiler

After you have downloaded both the software, just follow the on-screen instructions to install would recommend to leave them at the default directories to avoid later confusion. The STVD I install without any effort. For the Cosmic C compiler, you have to get a free license key dur installation processes. You just have to provide company information with E-mail ID, if you are a s just mention that. The processes of getting the license key are instantaneous and automatic, al the set-up instructions say that it might take a day or two, I received the license key automatically E-Mail ID as soon I submitted it, just make sure you check the SPAM. My Confirmation E-Mail is below.



Aswinth Raj,

Thank you for using Cosmic Products.

According to your request, please find attached the free license that will allow you to use the STM8 and STM32 (limited) compilers for 1 year.

This file must be copied into the "license" sub-folder of the compiler installation folder.

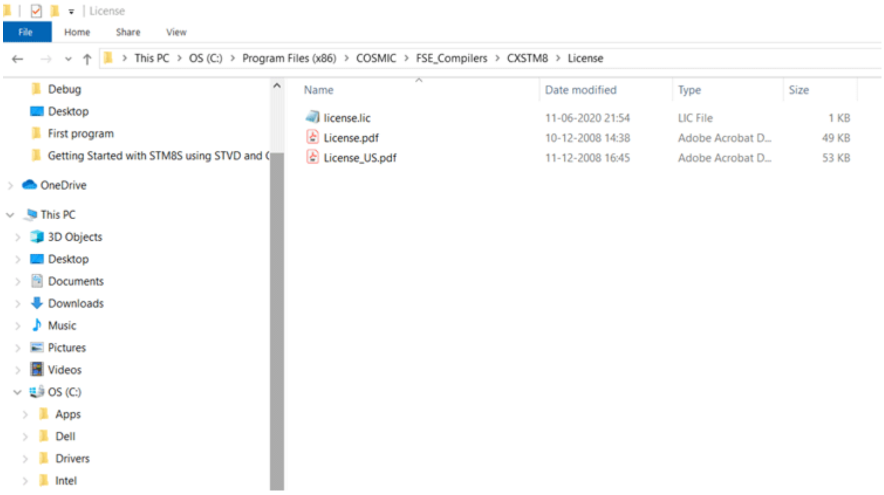
Support is provided via:

- support.stm8.free@cosmic.fr or
- the STM8 discussion forum on ST Community website

Regards,

The Cosmic Software Team.

As per the E-mail instruction, just copy the license.lic file and paste in the “license” sub-folder in your installation folder. For me the path was *“C:\Program Files (x86)\COSMIC\FSE_Compilers\CXSTM8\License”*. Just paste the file in location as shown below.



Name	Date modified	Type	Size
license.lic	11-06-2020 21:54	LIC File	1 KB
License.pdf	10-12-2008 14:38	Adobe Acrobat D...	49 KB
License_US.pdf	11-12-2008 16:45	Adobe Acrobat D...	53 KB

Standard Peripheral Library for STM8S103F3P6

As told earlier, ST Microelectronics provides Libraries called SPL which can be used for all 8-bit STM8S/A Microcontrollers. You can either download the original SPL from ST Microelectronics and make the required changes or download my STM8S103F3P6 SPL libraries and use them as such. I would recommend the later.

Original SPL from ST Microelectronics (https://www.st.com/en/embedded-software/stsw-stm8069.html)

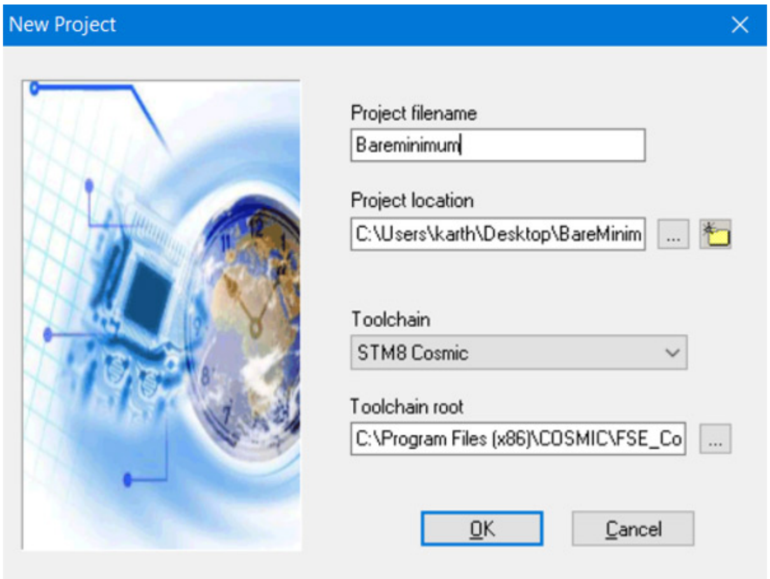
STM8S103F3P6 SPL (https://github.com/CircuitDigest/STM8S103F3_SPL)

While you are there also ensure to download the SPL User (https://github.com/CircuitDigest/STM8S103F3_SPL/blob/master/SPL_User_Manual.chm), whi be very handy when programming the controller.

Compiling the First Program

Now that everything is ready, let's try compiling our first program to check if the IDE, Compil Library are all working the way they should. You can also check the video at the bottom of the p detailed instructions.

Start by opening STVD and select File -> WorkSpace, in the pop-up, select "New Workspace" an the Project name and path where the program should be saved. I am namir program BareMinimum and saving it in a folder on the desktop. Click OK and you will get tl Project dialog box as shown below.



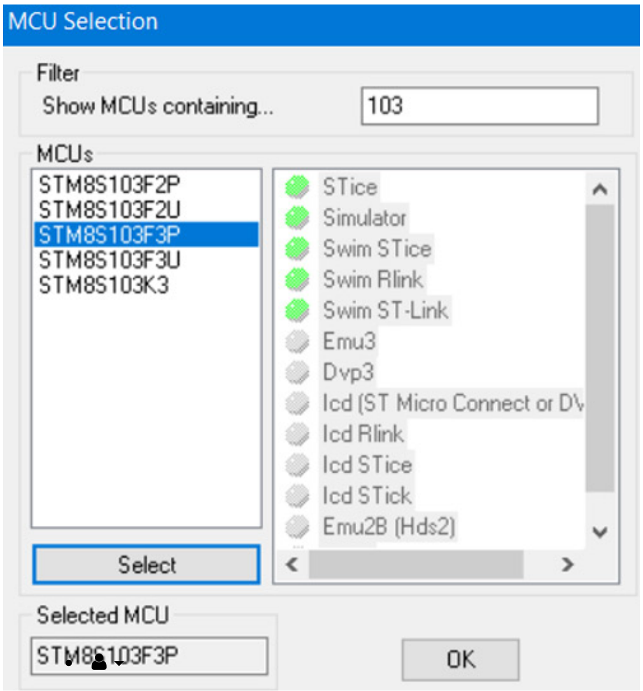
Project filename: Bareminimum

Project location: C:\Users\karth\Desktop\BareMinim

Toolchain: STM8 Cosmic

Toolchain root: C:\Program Files (x86)\COSMIC\FSE_Co

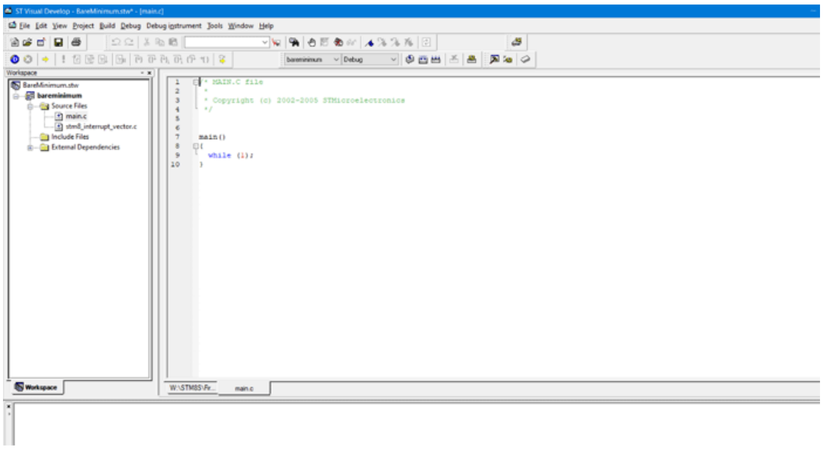
I am naming the project as bareminimum again and under toolchain we have to specify the toolchain root to the path where the STM8 Cosmic compiler is installed. The default path address is *"C:\Program Files (x86)\COSMIC\FSE_Compilers\CXSTM8"*. After that just click OK to get the "MCU Selection" window.



Show MCUs containing... 103

Selected MCU: STM8S103F3P

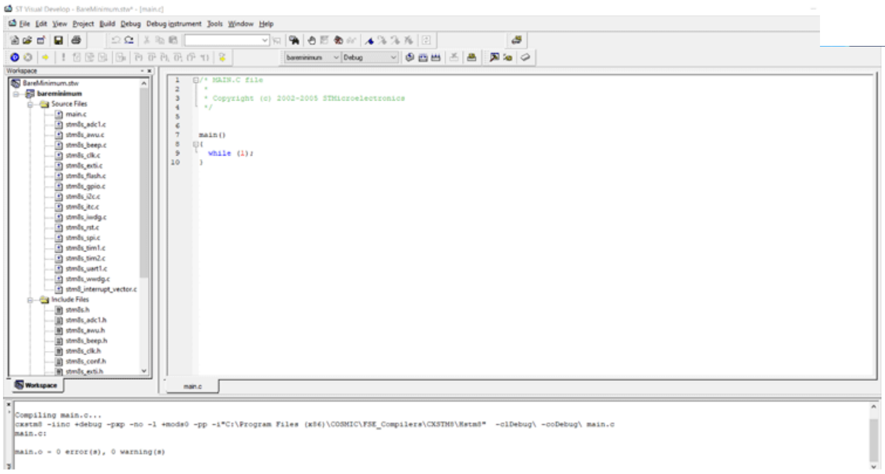
Search for STM8S103F3P and select it and click OK. This will open a new project for STM8S103F3P on STVD, the windows should look like this when done.



```
/* MAIN.C file
 *
 * Copyright (c) 2002-2005 STMicroelectronics
 */


main()
{
	while (1);
}
```

Right-click on “source file” and select “Add files to folder” to include all the c files from our SPL similarly right-click on Include files to include all the header files. If you have any questions, y refer to the video below. Once all the files are added, click *Build* -> *rebuil*d all and then Compile tc if the compiler and SPL are working as expected. If everything goes well, you should see the fo screen with build result 0 error and 0 warning.



```
/* MAIN.C file
 *
 * Copyright (c) 2002-2005 STMicroelectronics
 */


main()
{
	while (1);
}
```

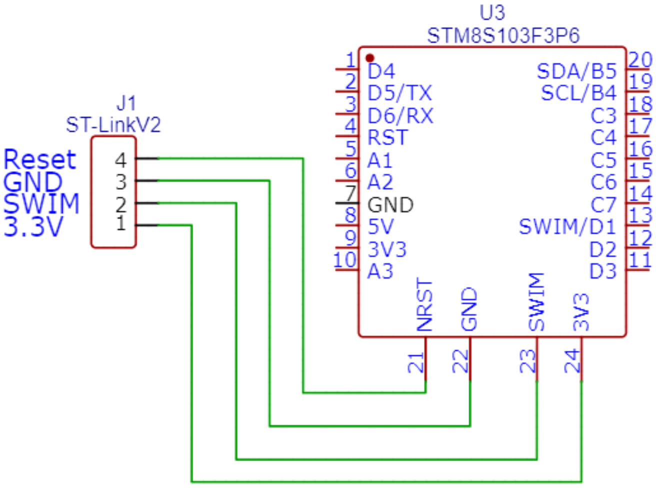
```
Compiling main.c...
cxstm8 -iinc +debug -pxp -no -l +mods0 -pp -i"C:\Program Files (x86)\COSMIC\FSE_Compilers\CXSTM8\Hstm8"  -clDebug\ -coDebug\ main.c
main.c:

main.o - 0 error(s), 0 warning(s)
```

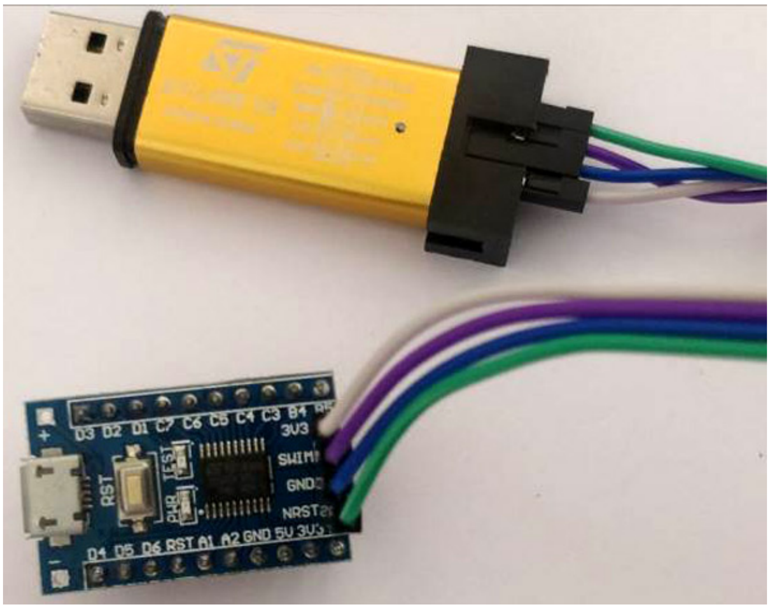
With this done, now we can be sure all our SPL libraries are working with Cosmic compiler and STVD. This is the standard procedure that we will follow for every new project. You can also include only the necessary header and source files that are needed for the project to reduce build time if required.

Uploading the Code from STVD to STM8S Development board using ST-Link V2

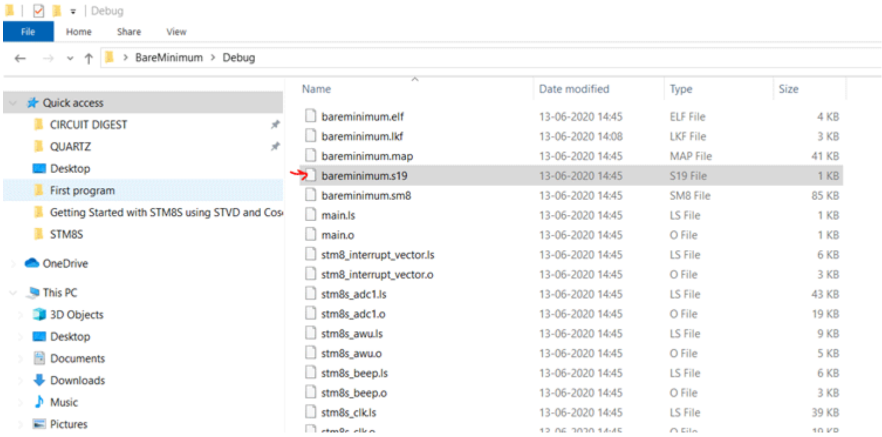
Connect the ST-Link V2 with the development board as shown below.



The connections are pretty straight forward and the best part is you need no external components. My hardware setup for programming is shown below, I have just used the female header wires to m connection. However, do note that the pinout of your ST-Link might differ from mine, make sure tc the pinout on the device before making the connections.

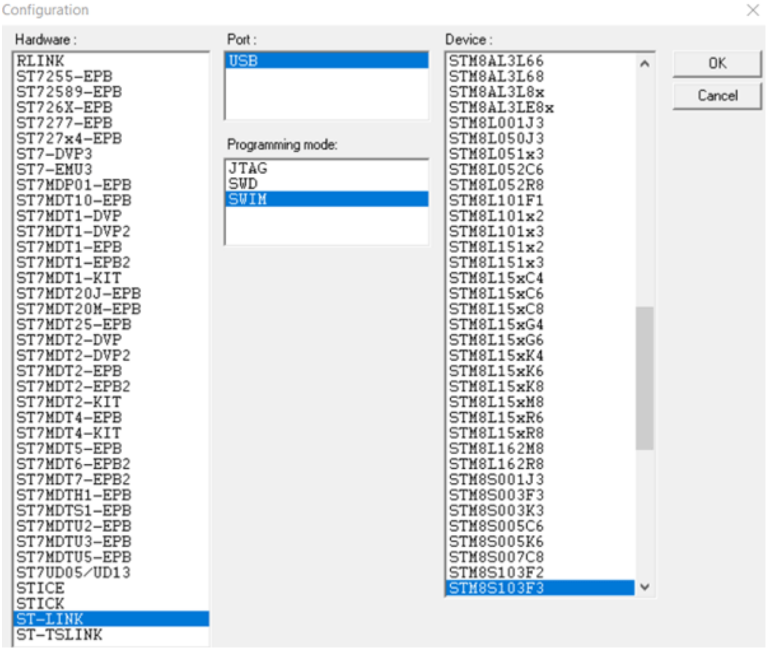


Make the connection and connect the device to your computer, the driver installation should begin automatically. You can use the device manager to ensure if your computer has discovered ST-LINK V2 correctly. You will also notice the test LED on the board blinking if this is the first time powering the board. As soon as we have successfully compiled the code on STVD, we will get an “S19” Extention file under the debug directory of the project folder. My S19 file is shown below.



Name	Date modified	Type	Size
bareminimum.elf	13-06-2020 14:45	ELF File	4 KB
bareminimum.lkf	13-06-2020 14:08	LKF File	3 KB
bareminimum.map	13-06-2020 14:45	MAP File	41 KB
bareminimum.s19	13-06-2020 14:45	S19 File	1 KB
bareminimum.sm8	13-06-2020 14:45	SM8 File	85 KB
main.ls	13-06-2020 14:45	LS File	1 KB
main.o	13-06-2020 14:45	O File	1 KB
stm8_interrupt_vector.ls	13-06-2020 14:45	LS File	6 KB
stm8_interrupt_vector.o	13-06-2020 14:45	O File	3 KB
stm8s_adc1.ls	13-06-2020 14:45	LS File	43 KB
stm8s_adc1.o	13-06-2020 14:45	O File	19 KB
stm8s_awu.ls	13-06-2020 14:45	LS File	9 KB
stm8s_awu.o	13-06-2020 14:45	O File	5 KB
stm8s_beep.ls	13-06-2020 14:45	LS File	6 KB
stm8s_beep.o	13-06-2020 14:45	O File	3 KB
stm8s_clk.ls	13-06-2020 14:45	LS File	39 KB

This S19 is like the Hex file which should be uploaded to the controller. To upload the program, open ST Visual Programmer (STVP) which should have installed along with STVD. Then in the configuration window, select as shown in the below image and click OK.



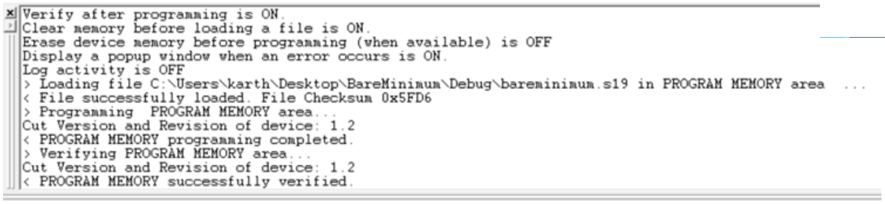
Hardware: RLINK, ST7255-EPB, ST72589-EPB, ST726X-EPB, ST7277-EPB, ST727x4-EPB, ST7-DVP3, ST7-EMU3, ST7MDP01-EPB, ST7MDT10-EPB, ST7MDT1-DVP, ST7MDT1-DVP2, ST7MDT1-EPB, ST7MDT1-EPB2, ST7MDT1-KIT, ST7MDT20J-EPB, ST7MDT20M-EPB, ST7MDT25-EPB, ST7MDT2-DVP, ST7MDT2-DVP2, ST7MDT2-EPB, ST7MDT2-EPB2, ST7MDT2-KIT, ST7MDT4-EPB, ST7MDT4-KIT, ST7MDT5-EPB, ST7MDT6-EPB2, ST7MDT7-EPB2, ST7MDTH1-EPB, ST7MDTS1-EPB, ST7MDTU2-EPB, ST7MDTU3-EPB, ST7MDTU5-EPB, ST7UD05/UD13, STICE, STICK, ST-LINK, ST-TSLINK

Port: USB

Programming mode: JTAG, SWD, SWIM

Device: STM8AL3L66, STM8AL3L68, STM8AL3L8x, STM8AL3LE8x, STM8L001J3, STM8L050J3, STM8L051x3, STM8L052C6, STM8L052R8, STM8L101F1, STM8L101x2, STM8L101x3, STM8L151x2, STM8L151x3, STM8L15xC4, STM8L15xC6, STM8L15xC8, STM8L15xG4, STM8L15xG6, STM8L15xK4, STM8L15xK6, STM8L15xK8, STM8L15xM8, STM8L15xR6, STM8L15xR8, STM8L162M8, STM8L162R8, STM8S001J3, STM8S003F3, STM8S003K3, STM8S005C6, STM8S005K6, STM8S007C8, STM8S103F2, STM8S103F3

Then click on *File->Open* and navigate to the S19 file that we showed earlier. Then to flash the follow *Program -> Current* Tab. If the flashing was successful, you should see the following output



```
Verify after programming is ON.
Clear memory before loading a file is ON.
Erase device memory before programming (when available) is OFF
Display a popup window when an error occurs is ON.
Log activity is OFF
> Loading file C:\Users\karth\Desktop\BareMinimum\Debug\bareminimum.s19 in PROGRAM MEMORY area  ...
< File successfully loaded. File Checksum 0x5FD6
> Programming  PROGRAM MEMORY area...
Cut Version and Revision of device: 1.2
< PROGRAM MEMORY programming completed.
> Verifying PROGRAM MEMORY area...
Cut Version and Revision of device: 1.2
< PROGRAM MEMORY successfully verified.
```

By default, when you purchase the STM8S, it will have a blin program that blinks the test LED. Now after uploading this blank code, the LED will no longer blink.

Important: I found that my ST-Link did not reset the board automatically after programming. I had to disconnect and reconnect it to check the output of my program. I am not sure if this is a problem for everyone, let me know in the comment section. Also, check out the video at the bottom which explains the solution to this.

With this, let's wind up this tutorial, we have learned the basics of hardware, have set up the development environment and learned how to compile and upload code. We are now ready to progress and we will be using this in all our upcoming tutorials. If you have any questions, please post them on our forums (https://circuitdigest.com/forums) and stay tuned for more!!

Code

```c
/* MAIN.C file
 *
 * Copyright (c) 2002-2005 STMicroelectronics
 */
#include "STM8s.h"
main()
{
GPIO_Init(GPIOB, GPIO_PIN_5, GPIO_MODE_OUT_PP_HIGH_SLOW);
while (1);
}
```

Video

TAGS STMICROELECTRONICS (/TAGS/STMICROELECTRONICS)

GETTING STARTED (/TAGS/GETTING-STARTED) LED BLINKING (/TAGS/LED-BLINKING)

STM8 (/TAGS/STM8)

RECOMMENDED POSTS

A.I. DAY
Assemble all the pieces needed for AI at the Edge of the IoT
September 9

(https://bit.ly/3serpko)

AI Day – September 9th 2021 (https://bit.ly/3serpko)

(https://bit.ly/2XmYEqp)

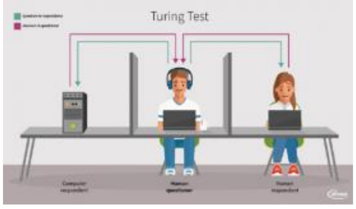
Register to Win an 11th Generation Intel® C i7-based NUC Topaz (https://bit.ly/2XmYEq

(https://bit.ly/3g2qYFa)

Hands-On Experience: Build Your First AI Solution in an Hour with the OpenVINO Toolkit (https://bit.ly/3g2qYFa)

(https://bit.ly/2VTK7SM)

Embedded Computing Design's Annual AI Survey (https://bit.ly/2VTK7SM)



(https://bit.ly/2VWV71N)

Know Thy AI Processor & Programming Options (https://bit.ly/2VWV71N)

(https://bit.ly/3g2zGDa)

How to Accelerate AI at the IoT Edge with Siemens (https://bit.ly/3g2zGDa)

(https://bit.ly/3xQmtmV)

Embedded Toolbox: Let's Build a Robot with VxWorks (https://bit.ly/3xQmtmV)

(https://bit.ly/2VT7njP)

Dev Kit Weekly: Xilinx Kria KV260 Vision AI Starter Kit (https://bit.ly/2VT7njP)

Get Our Weekly Newsletter!

Subscribe below to receive most popular news, articles and DIY projects from Circuit Digest

Email Address *

Name

Country

United States of America

Subscribe

RELATED CONTENT

STPOWER
STPOWER RF LDMOS transistors
low power consumption and high reliability

(/news/new-rf-ldmos-transistors-to-handle-high-rf-power-in-commercial-and-industrial-applications)

New RF LDMOS Transistors to Handle High RF Power in Commercial and Industrial Applications (/news/new-rf-ldmos-transistors-to-handle-high-rf-power-in-commercial-and-industrial-applications)

ST25
ST25R3918
Multi-functional NFC transceiver for
cost-conscious contactless applications

(/news/new-multi-purpose-nfc-transceiver-f cost-conscious-contactless-applications)

New Multi-Purpose NFC Transceiver for Co Conscious Contactless Applications (/news/new-multi-purpose-nfc-transceiver-f cost-conscious-contactless-applications)

STM32G0x1
STM32G0 boosted with 512KB flash
FDCAN & USB-C FS Dual-Role port

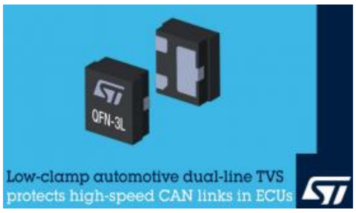
(/news/new-stm32g0-microcontrollers-with-usb-c-full-speed-dual-roll-ports-can-fd-and-high-capacity-memory)

New STM32G0 Microcontrollers with USB-C Full-Speed Dual-Roll Ports, CAN FD and High-Capacity Memory (/news/new-stm32g0-microcontrollers-with-usb-c-full-speed-dual-roll-ports-can-fd-and-high-capacity-memory)

Flexible 3A automotive LED driver
for precise interior and exterior lighting

(/news/single-channel-automotive-led-driver-with-integrated-dcdc-converter-and-dimming-capability)

Single Channel Automotive LED Driver with Integrated DC/DC Converter and Dimming Capability (/news/single-channel-automotive-led-driver-with-integrated-dcdc-converter-and-dimming-capability)


QFN-3L
Low-clamp automotive dual-line TVS
protects high-speed CAN links in ECUs

(/news/low-capacitance-automotive-transient-voltage-suppressor-for-protection-in-high-speed-interface-designs)

Low-Capacitance Automotive Transient Voltage Suppressor for Protection in High-Speed Interface Designs (/news/low-capacitance-automotive-transient-voltage-suppressor-for-protection-in-high-speed-interface-designs)

STISO621
High-performance digital isolator
boosts reliability in industrial systems

(/news/new-digital-isolator-with-thick-oxide-galvanic-isolation-technology-boosts-performance-and-reliability)

(/news/new-family-of-gan-solutions-combines-power-and-intelligence-for-automotive-applications)

New Digital Isolator with Thick-Oxide Galvanic Isolation Technology Boosts Performance and Reliability (/news/new-digital-isolator-with-thick-oxide-galvanic-isolation-technology-boosts-performance-and-reliability)

AIS2IH
Next-gen automotive MEMS accelerometer
for telematics and emerging applications
ST

(/news/low-power-three-axis-linear-accelerometer-for-high-performance-automotive-applications)

Low Power, Three Axis Linear MEMS Accelerometer for High-Performance Automotive Applications (/news/low-power-three-axis-linear-accelerometer-for-high-performance-automotive-applications)

New Family of GaN Solutions Combines Power and Intelligence for Automotive Application (/news/new-family-of-gan-solutions-combir power-and-intelligence-for-automotive-applications)

‹ PREVIOUS POST

Interfacing 5 inch HDMI Touchscreen Display with Raspberry Pi (https://circuitdigest.com/microcontroller-projects/interfacing-5-inch-hdmi-touchscreen-display-with-raspberry-pi)

NEX

ESP32 PWM Tutorial: Controlling the Bri(

(https://circuitdigest.com/microco projects/esp32-pwm-tutorial-con brightness

COMMENTS

(/users/joseph-tannenbaum)

Joseph Tannenbaum (/users/joseph-tannenbaum)
Jun 25, 2020

Hello,

I get this error:

```
Compiling main.c...
rcst7 main.c OBJECT(Debug\main.o) PIN("C:\Program Files (x86)\COSMIC\FSE_Compilers\CXSTM8\INC\STM8") PIN("C:\Program Files (x86)\COSMIC\FSE_Compilers\CXSTM8\INC\ST7") PIN("C:\Program Files (x86)\COSMIC\FSE_Compilers\CXSTM8\INC")  WRV(0) STM8(SMALL) DEBUG DGC(page0) AUTO  OT(0) PR(Debug\main.lst) CD CO SB LAOB PIN(..\..\downloads\downloaded\st8\stm8s103f3_spl-master\inc)

Error creating process for executable rcst7
The system cannot find the file specified.
```

and I don't see and INC directory under CXSTM8.

Looks like compiler did not install correctly, but it didn't get any errors.

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register?destination=node/6442%23comment-form) to post comments

(/users/joseph-tannenbaum)

Joseph Tannenbaum (/users/joseph-tannenbaum)
Jun 25, 2020

Rebooted and tried again and get error that Access to C:\Program Files (86)\STMicroelectronics\st_toolset\stvd\default.wed was denied. I ran the installer as Aministrator by right click.??

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register?destination=node/6442%23comment-form) to post comments

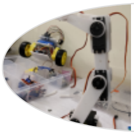
(/users/joseph-tannenbaum)

Joseph Tannenbaum (/users/joseph-tannenbaum)
Jun 25, 2020

Didn't change toolchain to Cosmic. It compiled.

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register?destination=node/6442%23comment-form) to post comments

- **Aswinth Raj (/users/aswinth-raj)**
  Jul 15, 2020

 (/users/aswinth-raj)

glad you got it working.

And yes you have to run them as admin

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/regis destination=node/6442%23comment-form) to post comments

(/users/octarudin)

octarudin (/users/octarudin)
Oct 26, 2020

Dear Aswinth Raj,

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register? destination=node/6442%23comment-form) to post comments

I get error when I click the "Rebuild All" button.

```
The command: "cxstm8 -istm8s_stdperiph_driver\inc -
i..\..\stm8s_stdperiph_lib\libraries\stm8s_stdperiph_driver\inc +debug -pxp ·
+mods0 -pp -iD:\Hstm8  -clDebug\ -coDebug\
stm8s_stdperiph_driver\src\stm8s_adc1.c " has failed, the returned value is:

exit code=1.

project2.elf - 181 error(s), 0 warning(s)
```

Please help me to solve this problem.

Thank you & Best Regards,

Octarudin

(/users/octarudin)

octarudin (/users/octarudin)
Oct 26, 2020

Dear Aswinth Raj,

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register? destination=node/6442%23comment-form) to post comments

Fortunately, I solve the error by doing this.

1. Click "Project" on menu bar
2. Choose "Settings" (or SHIFT+F7)
3. Open "C Compiler" tab
4. Change the category from "General" to "Preprocessor"
5. Under "Preprocessor Definitions" write your device in the textbox. (ie. STM8S103)
6. Click "OK" button then problem solved.

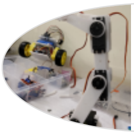
But, I got new error

```
#error cpstm8 stm8s_stdperiph_driver\inc\stm8s.h:2708 can't open stm8s_conf.h
```

I am still find out the problem.

Thank you & Best Regards,

Octarudin

 (/users/aswinth-raj)

Aswinth Raj (/users/aswinth-raj)
Oct 27, 2020

Hey both the errors are because of improper

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register?destination=node/6442%23comment-form) to post comments

workspace settings. Please follow the first tutorial https://circuitdigest.com/microcontroller-projects/getting-started-with-... (https://circuitdigest.com/microcontroller-projects/getting-started-with-stm8s-using-stvd-and-cosmic-c-compiler)

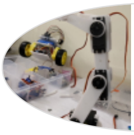
Make sure you have added all the needed src and header files before you compile.

(/users/octarudin)

octarudin (/users/octarudin)
Oct 28, 2020

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register?destination=node/6442%23comment-form) to post comments

Why in my libraries does not contain stm8_conf.h file ? I download officially from ST website.



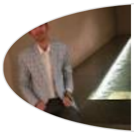
(/users/aswinth-raj)

Aswinth Raj (/users/aswinth-raj)
Oct 29, 2020

Not sure what you mean by official ST

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register?destination=node/6442%23comment-form) to post commer

I have already provided the link in the article from where you ca download



(/users/kunal-gupta)

Kunal Gupta (/users/kunal-gupta)
Dec 31, 2020

Bro as you showed in end of video in which code is n

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register? destination=node/6442%23comment-form) to post comments

actually uploaded ob the stm8s ....
I am getting that issue , i have seen your next tutorial also but still when i uploa normal blink code it compiles, shows sucessful uploading of code into prograr ,data, option memory via STVP. But on board leds does not link.
Stlink programmer turns out red at first then i reset it and then it led turns out t what i suppose no led blinks on onboard stm8.
Will be waiting for your reply if u want can send u video also via drive link

(/users/flexo)

flexo (/users/flexo)
Jan 18, 2021

for those of you having compile errors you have to add the

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register? destination=node/6442%23comment-form) to post comments

stm8s_config.h in you project directory else you'll keep getting compile errors you can find it in the standard peripheral library zip you downloaded open project folder you'll see STM8S_StdPheriph_Template you'll find the stm8s_config.h file there just copy that file to you project directory you don't have to include it in you code you only need stm8s.h included and it compiles successfully

(/users/nikhil-koul)

nikhil koul (/users/nikhil-koul)
Mar 11, 2021

While making the programming connections the LED does not

Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register? destination=node/6442%23comment-form) to post comments

turn on.

However when i use the USB micro connection , LED starts blinking.

Can you please help with the issue?

(/users/madhusudan-b)

Madhusudan B. (/users/madhusudan-b)
Apr 08, 2021

while trying to program, its giving an error message

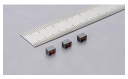
Log in (/user/login?destination=node/6442%23comment-form) or register (/user/register? destination=node/6442%23comment-form) to post comments

Error : The device is protected
Operation aborted
Error : < PROGRAM MEMORY verifying failed.

whaat can be done for this problem

LOG IN (/USER/LOGIN?DESTINATION=NODE/6442%23COMMENT-FORM) OR REGISTER (/USER/REGISTER?DESTINATION=NODE/6442%23COMMENT-FORM) TO POST COMMENT

NEWS ARTICLES PROJECTS



Small-Sized Broadband Inductors for High-Impedance Performance in Vehicle-Mounted PoC (/news/small-sized-broadband-inductors-for-high-impedance-performance-in-vehicle-mounted-poc)

(/news/small-sized-broadband-inductors-for-high-impedance-performance-in-vehicle-mounted-poc)

OceanMedallion Wearable Devices Integrated with Dialog Semiconductor's WiRa Techı for Enhanced User Experience (/news/oceanmedallion-wearable-devices-integrated-wi dialog-semiconductor%E2%80%99s-wira-technology-for-enhanced-user-experience)

(/news/oceanmedallion-wearable-devices-integrated-with-dialog-semiconductor%E2%80%99s-wira-technology-for-enhanced-user-experience)

New Prototype "Robocar" with Second-Generation AI Chip designed for Autonomous D (/news/new-prototype-robocar-with-second-generation-ai-chip-designed-for-autonomc driving)

(/news/new-prototype-robocar-with-second-generation-ai-chip-designed-for-autonomous-driving)

IIT Jodhpur Researchers develops new blockchain technology to secure IoT networks (/news/iit-jodhpur-researchers-develops-new-blockchain-technology-to-secure-iot-networks)

(/news/iit-jodhpur-researchers-develops-new-blockchain-technology-to-secure-iot-networks)

Asia's most valuable firm is now TSMC; overtakes Tencent and Alibaba (/news/asia%E2%80%99s-most-valuable-firm-is-now-tsmc-overtakes-tencent-and-alibaba)

(/news/asia%E2%80%99s-most-valuable-firm-is-now-tsmc-overtakes-tencent-and-alibaba)

Connect with us on social media and stay updated with latest news, articles and projects!

(https://www.linkedin.com/company/circuit-

(https://www.facebook.com/circuitdigest/)(https://twitter.com/CircuitDigest)(https://www.youtube.com/channel/UCy3CLdAYgZdAOG9k3IPdLmw)