Community

1

Search all content... Ctrl + K

STM32 MCUs



B.Montanari
ST Technical Moderator · 1 year ago

How to use the STM32N6 in overdrive mode

2 replies

Summary

This article provides a quick guide on how to enable overdrive mode on the STM32N6 Discovery kit and configure your project accordingly. It includes instructions and explanation on configuring overdrive mode, responsible for changing the VDD_{CORE}'s voltage to 0.89 V instead of the typical 0.80 V using STM32Cube's ecosystem. This allows the STM32N6570-DK to achieve its highest CPU and Neural-ART accelerator (NPU) frequencies up to 800 MHz and 1 GHz, respectively.

Introduction

The overdrive mode on the STM32N6 board is a power and performance configuration that allows the MCU to run at higher frequencies with optimized voltage scaling. This operation mode allows the Cortex®-M55 core to run at up to 800 MHz by increasing the core voltage (VDD_{CORE}) to 0.89 V through the external SMPS (switched-mode power supply). This mode balances power and performance using dynamic voltage and frequency scaling and is supported by the board's power management design.

By default, the STM32N6 uses the typical core voltage (VDD_{CORE}= 0.8 V). To use its high-frequency features, overdrive mode is a must. We use the STM32N6570-DK for this tutorial.

1. Hardware and OTP fuse

The STM32N6570-DK uses external memories at 1.8 V voltage and relevant I/Os should be configured from the 3.3 V default using firmware or software and OTP to get nominal speed.

Be aware that setting HSLV_VDDIOx and VDDIOxVRSEL bits while VDDIOx is in the 3.3V range damages the device.

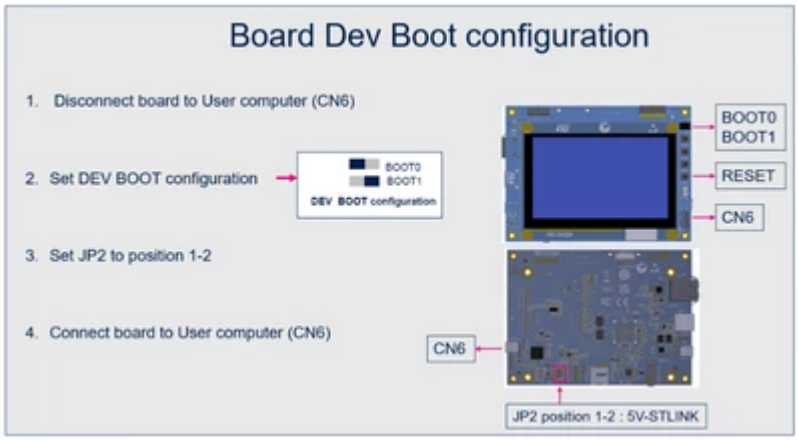
This article provides a quick guide on what the STM32N6's OTP bits are and how to accordingly.

2. Software setup

Feedback

Ask AI

This article assumes you have installed STM32CubeMX (6.13 or later), the latest version of the STM32N6 HAL driver, STM32CubeProgrammer (2.18 or later), and STM32CubeIDE (1.18.0 or later). The hardware used to showcase is the STM32N6570-DK and make sure you have it in DEV boot mode to program the code:



Be sure to add the postbuild step to your STM32N6 project when using STM32CubeIDE. This step can be done by following the tutorial in this article.

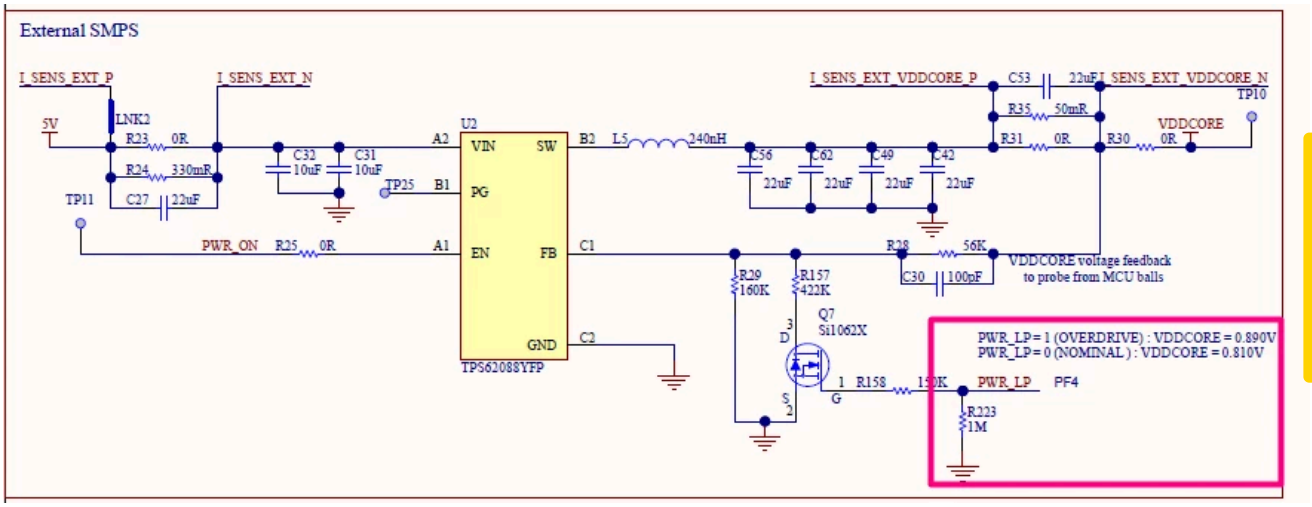
3. STM32CubeMX configuration

Custom configurations in STM32CubeMX are required to set up the overdrive correctly.

Create a new project using the STM32CubeMX and select the **[STM32N657X0H3Q]**. Select the option to use the **[Secure Domain only]**.

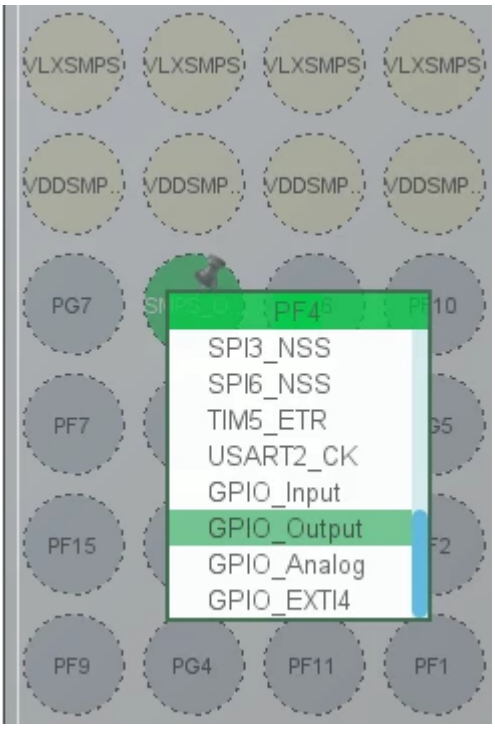
3.1 Configure the GPIO pins

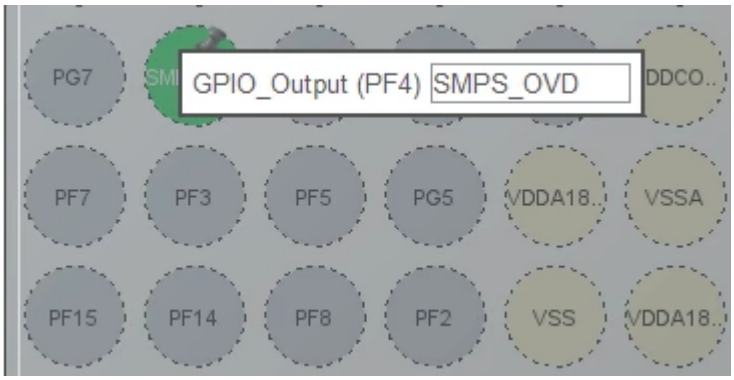
On STM32CubeMX, you need to turn on PF4 (usually turned off by default) to get the VDD_{CORE} to 0.89 V (OVERDRIVE). This is shown in the Discovery kit schematic, on page 5.



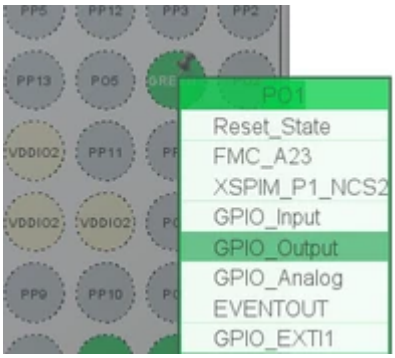
Locate the PF4 and configure it as GPIO_Output and use the label to name it **SMPS**

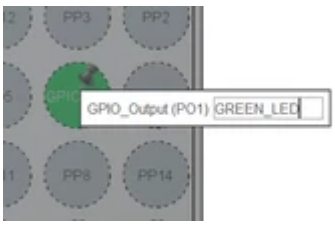
Ask AI





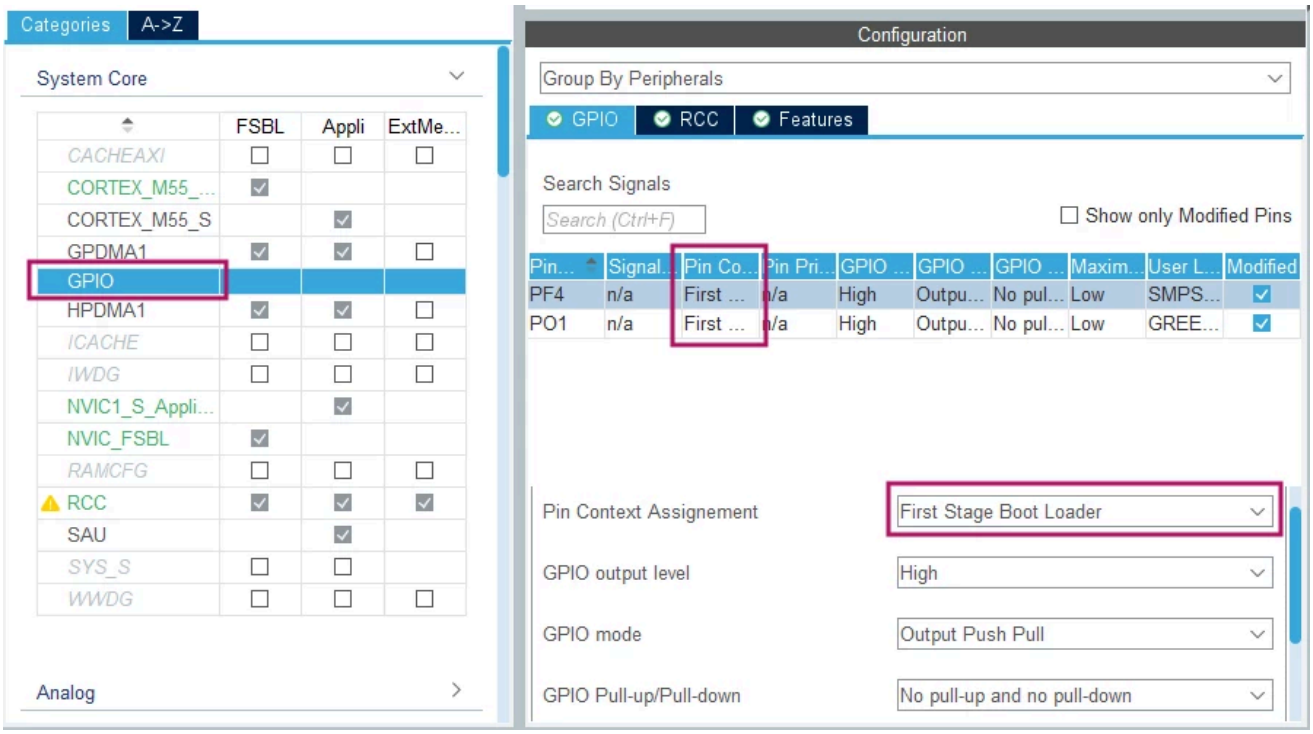
Also, locate the PO1 and configure it as GPIO_Output and use the label to name it **GREEN_LED**.





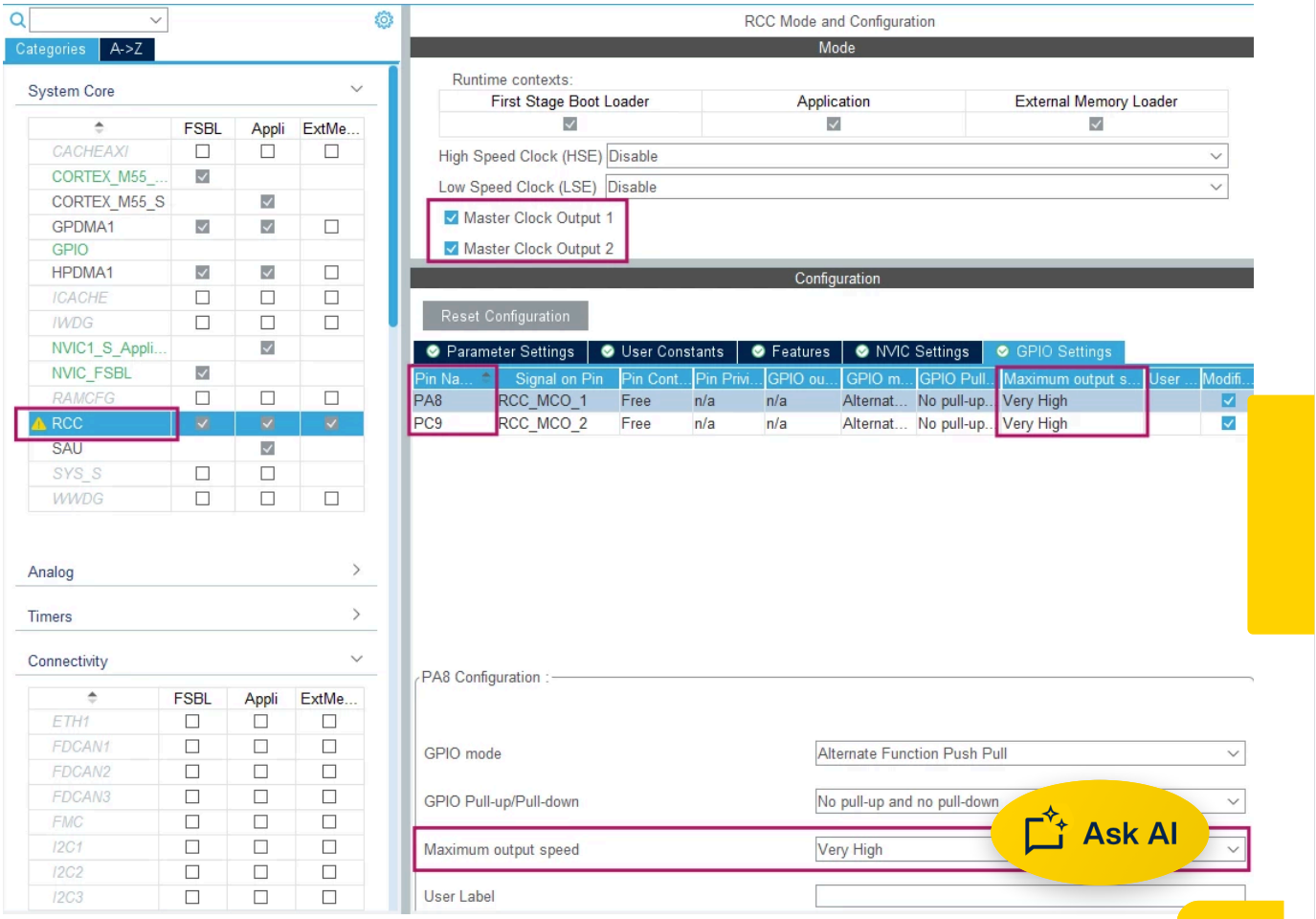
We need to assign both GPIOs to be used by the FSBL.

Ask AI

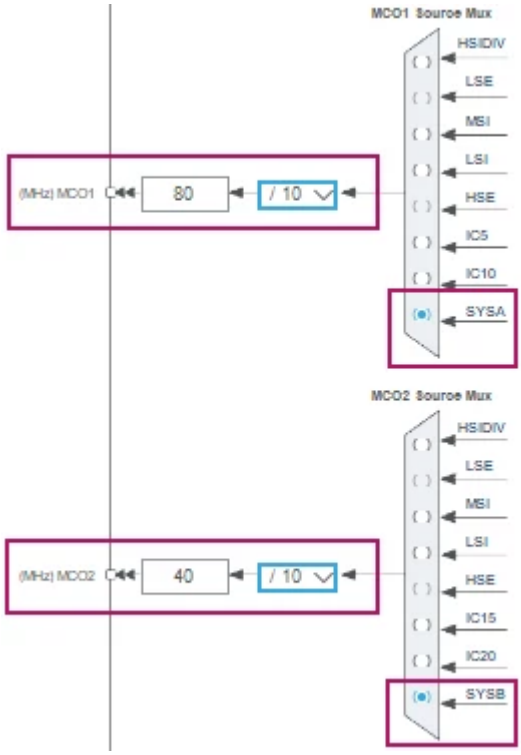


3.2 Configure the RCC and ICACHE

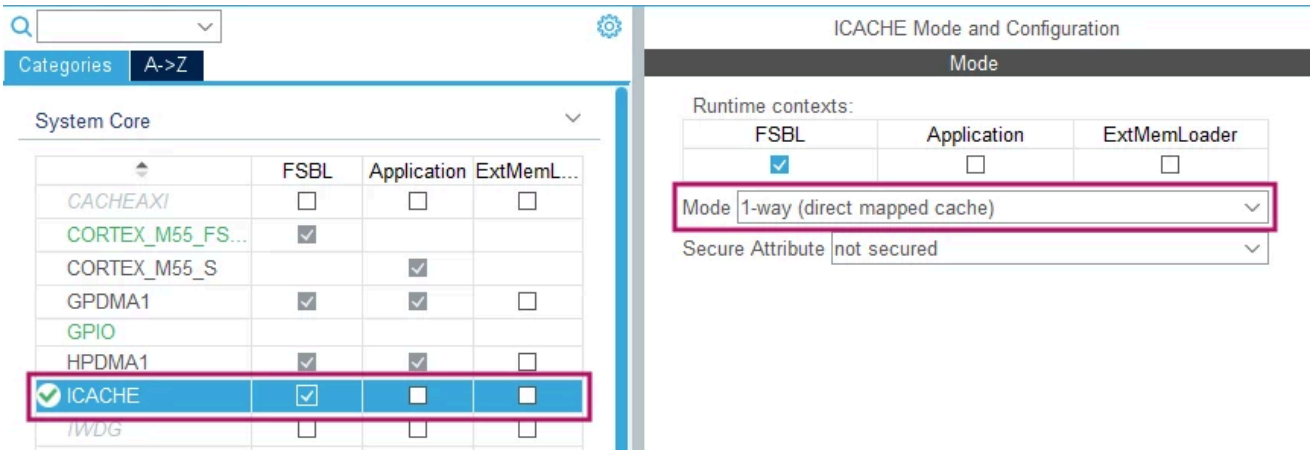
The next step is to configure the RCC by enabling the MCO1 and MCO2 (automatically assigned to PA8 and PC9, respectively). By reproducing this step, we can later confirm the desired frequency output value using an oscilloscope.



Along with that, on the “Clock Configuration” tab, activate the following:



Back to the “Pinout & Configuration” tab, you need to turn on the ICACHE feature to make the most out of the overdrive mode.

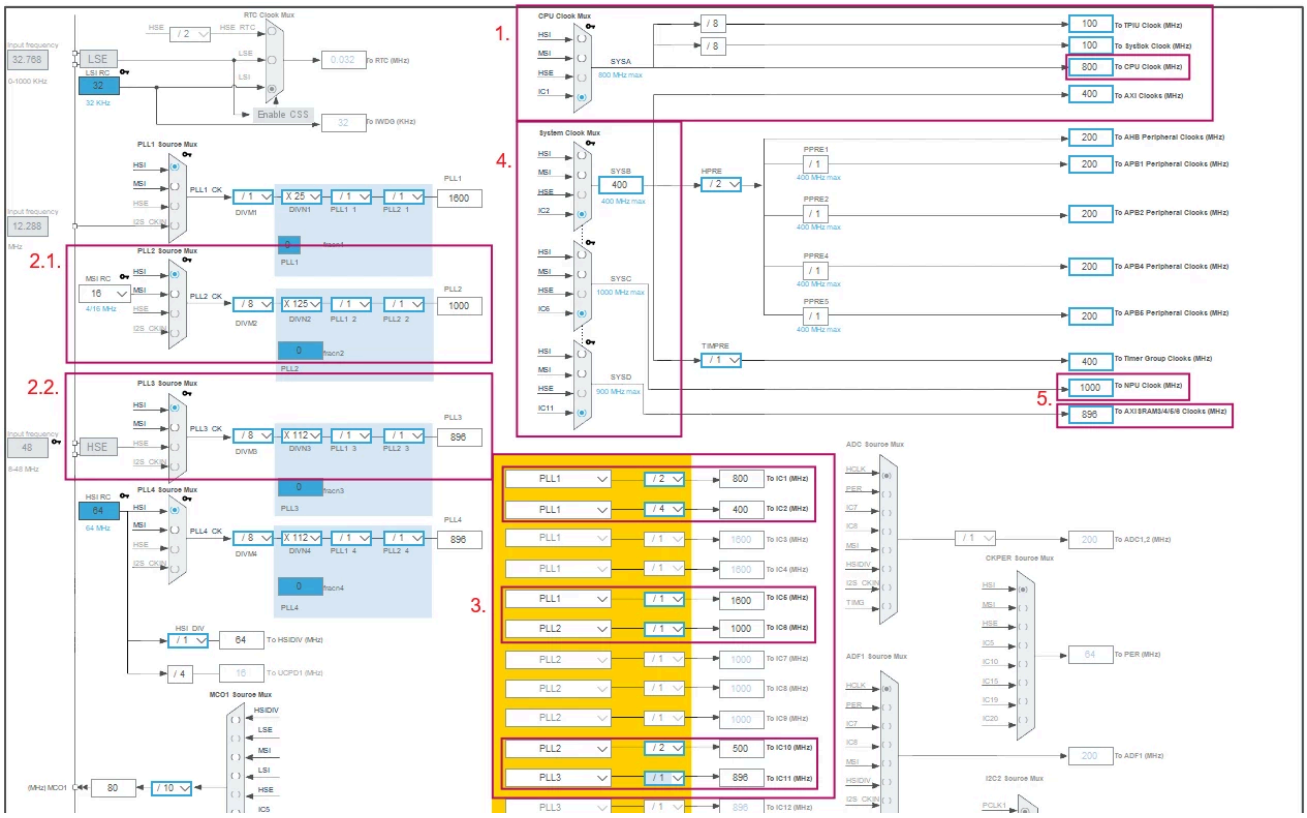


3.3 Clock configuration

Since we already set the VDD_{CORE} to 0.89 V, the next step shows the correct clock tree configuration to have the maximum frequency output on the CPU, NPU, and AXISRAM3/4/5/6.

In STM32CubeMX’s “Clock Configuration” tab, follow according to the image below:

Ask AI



4. Code configuration

In the **main.c** file, you need to add a **MX_GPIO_Init()** function before the **SystemClock_Config()**. We need to configure the correct tension on VDD_{CORE} first, so that the clock value can be properly set up. This way, we ensure the CPU's correct operation.

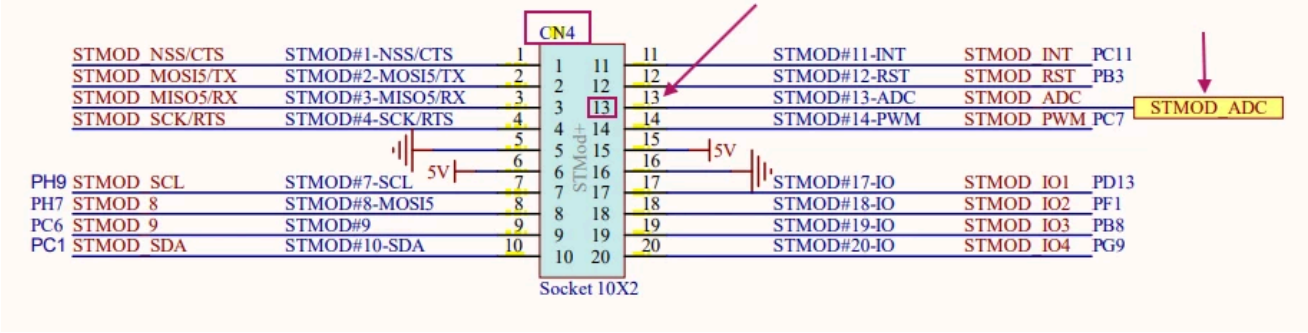
Note: The duplicate MX_GPIO_Init() should not cause any problems.

```
/* USER CODE END 1 */
 /* Enable the CPU Cache */
 /* Enable I-Cache---------------------------------------------------------*/
 SCB_EnableICache();
 /* Enable D-Cache---------------------------------------------------------*/
 SCB_EnableDCache();
 /* MCU Configuration--------------------------------------------------------*/
 HAL_Init();
 /* USER CODE BEGIN Init */
 /* Initialize all configured peripherals before the SysClock */
 MX_GPIO_Init();
 /* USER CODE END Init */
 /* Configure the system clock */
 SystemClock_Config();
 /* USER CODE BEGIN SysInit */
 /* USER CODE END SysInit */
 /* Initialize all configured peripherals */
 MX_GPIO_Init();
 MX_ICACHE_Init();
/* USER CODE BEGIN 2 */
```

5. Validation

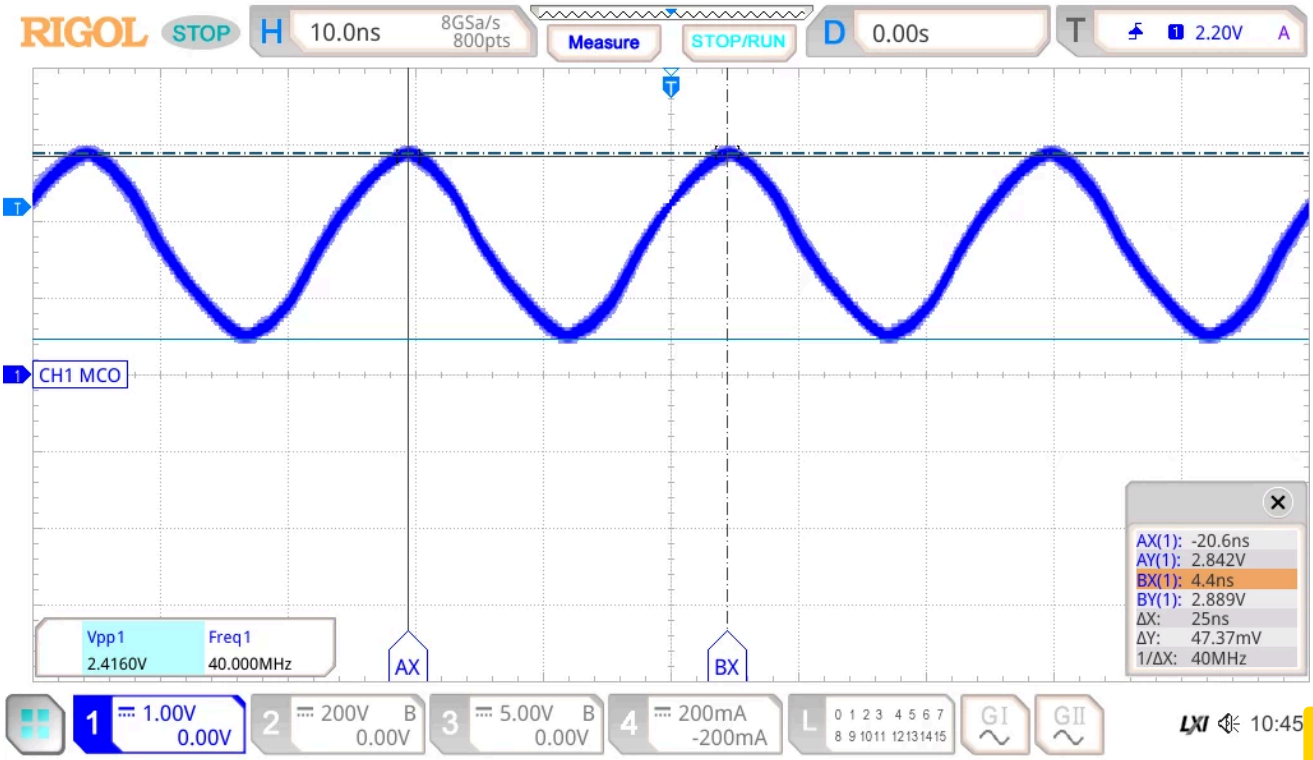
Ask AI

To see your application running, you can use the MCO2 pin configured before. This pin can be found in the CN4 socket, as shown in the Discovery kit schematic (page 10), or at the TP2 and TP3 test points.



Note: During the “STM32CubeMX Configuration” portion of this article, we configured the MCO2 frequency with a /10 divider. This modification allows us to measure the CPU frequency with most scopes precisely.

The final wave should be something like the following:



Conclusion

In this article, we explored how to configure STM32N6 microcontrollers’ Overdrive mode, as well as how to validate this change using simple methods and configurations. By adjusting the hardware to the specific voltage settings and constraints, it is possible to ensure maximum performance and reliability of your embedded applications.

Related links

- Product page: Discovery kit with STM32N657X0 MCU
- Reference manual 0486: STM32N647/657xx Arm®-based 32-bit MCUs
- Data brief 5351: STM32N6570-DK - Discovery kit with STM32N657X0 MCU
- User manual 3249: Getting started with STM32CubeN6 for STM32N6 series

Ask AI

- Board schematic: MB1939-N6570-C02
- How to debug STM32N6 using STM32CubeIDE - STMicroelectronics Community

STM32N6 series

4

Leave a reply...

2 replies

Sort by: **Oldest first**

Lennart_Heske
Explorer · 8 months ago

Hello @B.Montanari ,

is it correct that on the NUCLEO board of the same MCU you'd have to use the internal SMPS through the ACTVOS bit of the PWR_VOSC register instead of the BF4 GPIO config to select the overdrive power mode? Are there any other substantial differences I should know of before trying to apply this strategy to the NUCLEO version?

Kind regards,

Lennart Heske

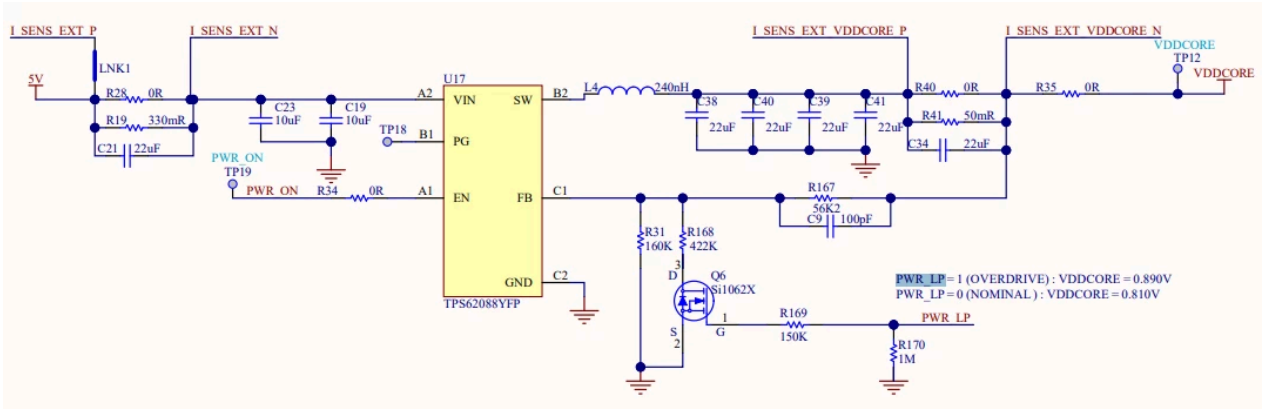
Engineering student and hobbyist

1 reply

GoranDragovic
Associate II · 18 hours ago

Have you managed to get overdrive to work with Nucleo? How did you achieve it?
In Nucleo Board schematic its not noted that the PWR_LP is tied to any pin ?

Ask AI



Nucelo board external SMPS

ST Community highlights – April to June 2026



Related topics

How to build an AI application from scratch on the NUCLEO-N657X0-Q using STM32CubeMX

STM32 MCUs

How to build an AI application from scratch on the STM32N6570-DK using STM32CubeMX

STM32 MCUs

How to create an STM32N6 FSBL load and run

STM32 MCUs

How to measure power on the STM32N6

STM32 MCUs

STM32N6 USBX Standalone Guide

STM32 MCUs Embedded software

Ask AI

Popular tags

STM32H7 series | STM32F4 series | STM32CubeMX | STM32CubeIDE | TouchGF

STM32L4 series STM32F7 series USB UART-USART

STM32Cube MCUs packages



About STMicroelectronics

Who we are

Investor relations

Sustainability

Innovation & technology

Careers

Suppliers to ST

Blog

General terms and conditions

Connect with us

Contact ST offices

Find sales offices & distributors

Community

Newsroom

Events & trainings

AI at ST

Browse

Shortcuts

Sitemap

Compliance, Ethics, Security & Privacy

Ethics and compliance

ST Ethics Hotline

ST's Code of Conduct

Security & privacy portal

Ask AI

Follow us

All rights reserved © 2026 STMicroelectronics | Terms of use | Sales terms & conditions | Trademarks | Privacy portal | Manage cookies

Ask AI