Embedded Coder® Support Package for STMicroelectronics® STM32 Processors

Voltage Mode Control of DC-DC Buck Converter using STM32 BG474EDPOW1 Discovery Kit

MATLAB®

R2022a

MathWorks®

How to Contact MathWorks

Latest news:	`www.mathworks.com`
Sales and services:	`www.mathworks.com/sales_and_services`
User community:	`www.mathworks.com/matlabcentral`
Technical support:	`www.mathworks.com/support/contact_us`
Phone:	508-647-7000

The MathWorks, Inc.
1 Apple Hill Drive
Natick, MA 01760-2098

Embedded Coder® Support Package for STMicroelectronics® STM32 Processors (Nonrelease)

Revision History

April 2022	First printing	"Release for R2022a"

Contents

1 About This Example

2 Voltage Mode Control of DC-DC Buck Converter using STM32 BG474EDPOW1 Discovery Kit

About This Example

Introduction

This example implements a DC-DC buck converter control system using the B-G474E-DPOW1 Discovery kit which uses the STMicroelectronics Arm® Cortex®-M4 core-based STM32G474RET6 microcontroller. This kit showcases among other features the buck-boost converter with variable load. This example will focus to regulate the synchronous buck converter power stage as available on the onboard B-G474E-DPOW1 Discovery kit in voltage-mode control and its implementation using the MathWorks® Embedded Coder Support Package for STMicroelectronics STM32 Processors.

Using this example, you can:

1. Verify the closed-loop Proportional-Integral (PI) control algorithm can optimally regulate the buck converter output voltage onboard the B-G474E-DPOW1 Discovery kit (control algorithm executes at 200KHz sampling rate).
2. Activate/de-activate the onboard embedded loads to verify if the PI controller can compensate for any load variations and maintain the desired set output voltage.
3. Use custom code blocks to perform initialization and run-time update of the hardware peripheral registers that aren’t currently supported under Simulink Support Package.
4. Perform data logging at high sample rate using the two-model approach.

Hardware Specifications

Hardware tools required to run example:

- PC with Windows® 7 or later.
- B-G474E-DPOW1 Discovery kit.
- Micro-USB cable for ST-LINK in-circuit debugger and programmer.
- 5V buck converter input power via the USB-C cable.

Software Requirements

This section lists the software products from MathWorks and STMicroelectronics that you need to run the example models on the B-G474E-DPOW1 Discovery kit.

Required MathWorks Products

To generate code and deploy an example model:

- Simulink®
- Embedded Coder Support Package for STMicroelectronics STM32 Processors
- Embedded Coder
- Instrument Control Toolbox™

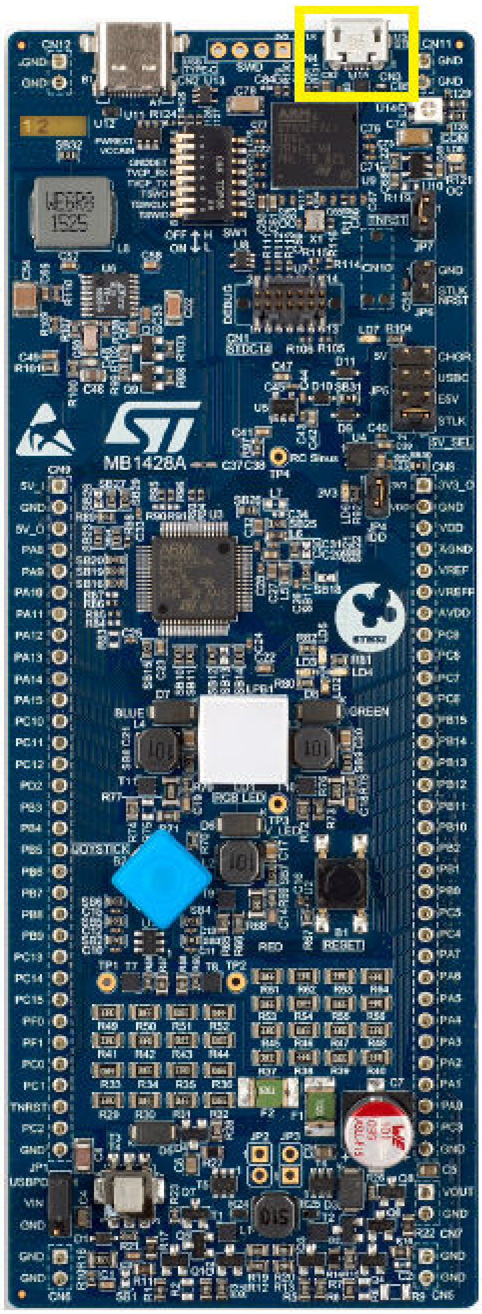
Required STMicroelectronics Products

- STM32CubeMX (v5.6.0 or later) together with STM32Cube firmware library for G4 (v1.2.0 or later) installed

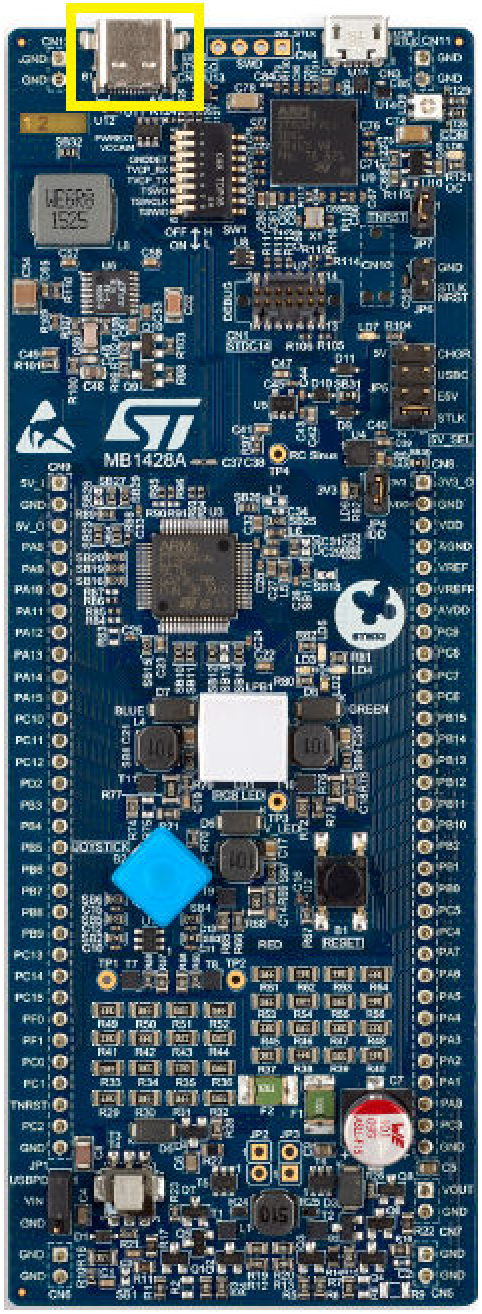
Hardware Setup

The hardware setup procedure is described below:

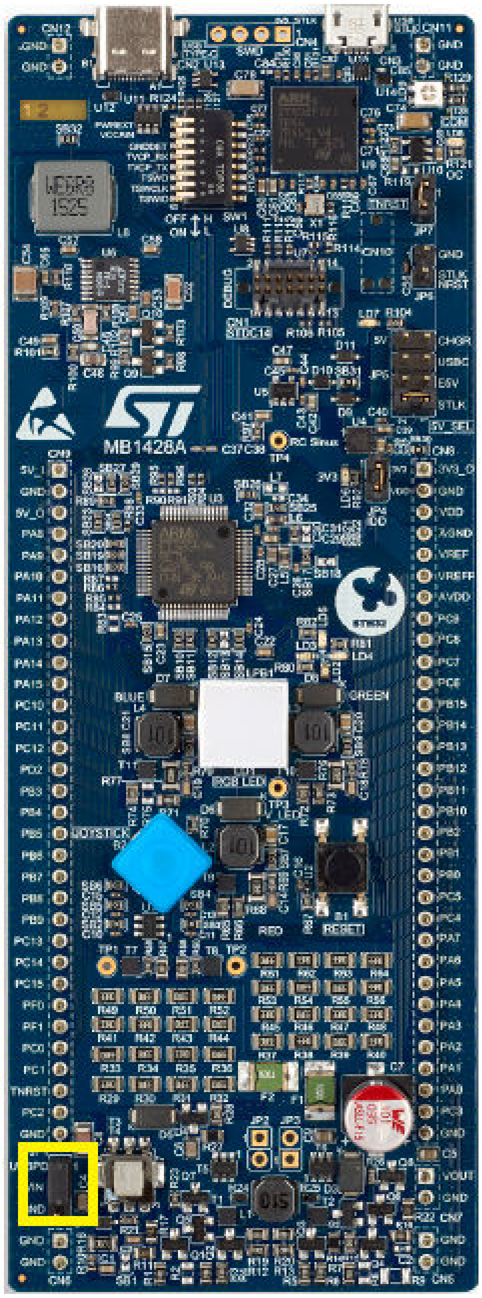
1 Connect the micro-USB cable from the PC to CN3 on the Discovery kit.



2 Apply power via the USB-C and connect this to CN2 on the Discovery kit.



3 Ensure that the jumper JP1 is in the USB PD-VIN position.



Contents of Downloaded ZIP Folder

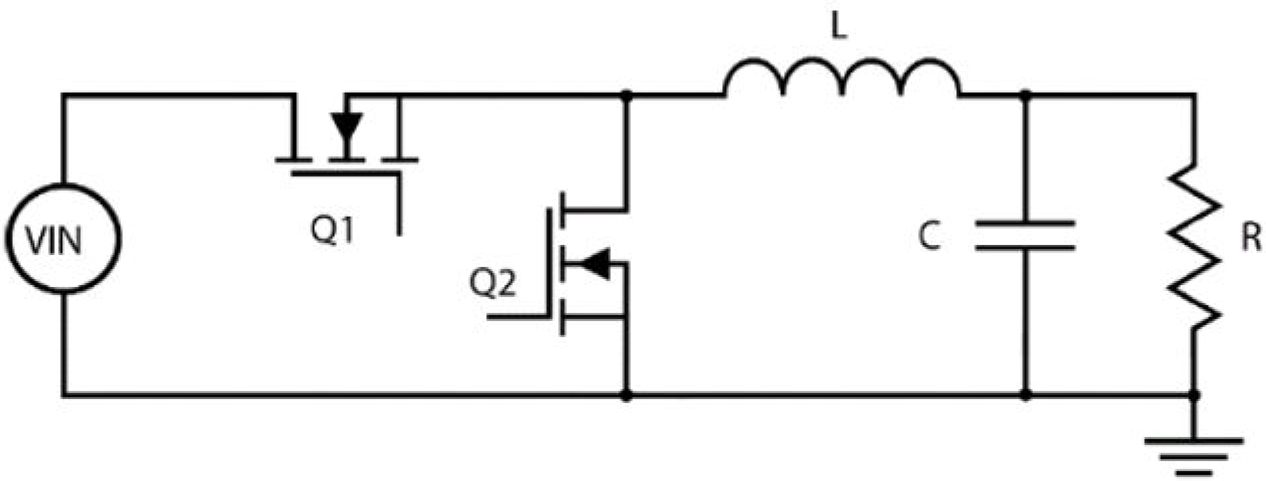
The ZIP folder that you downloaded from the GitHub® repository, includes:

File Name	Description
stm32g474_buck_vmc_discoverykit.slx	Target model
stm32_buck_vmc_host_read.slx	Host model
STM32_VMC_DC_DC_Buckconverter_Example.pdf	Documentation
STM32G474_Discovery_Buck_VMC.ioc	STM32CubeMX IOC file

Voltage Mode Control of DC-DC Buck Converter using STM32 BG474EDPOW1 Discovery Kit

Introduction

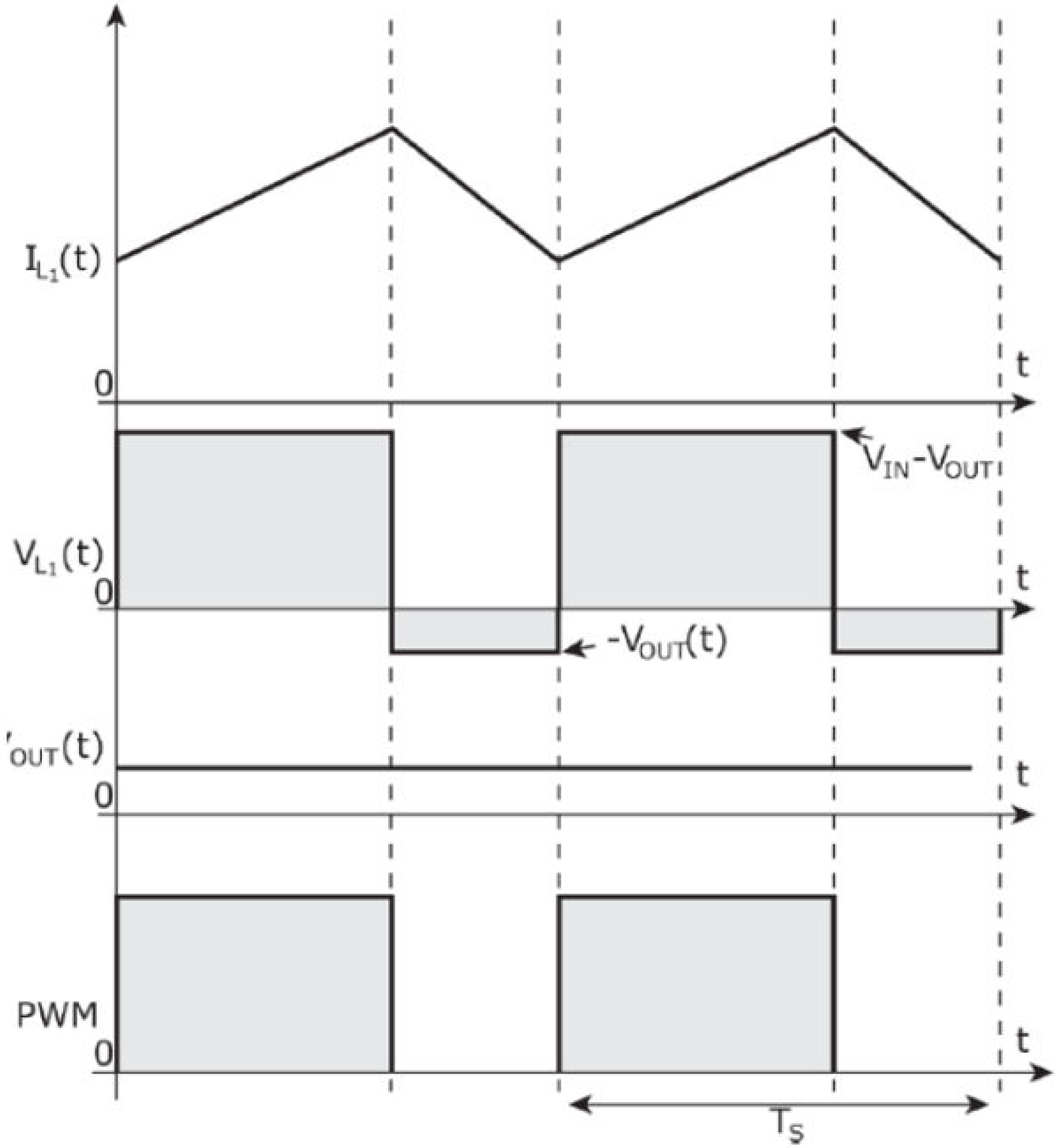
The B-G474E-DPOW1 Discovery kit contains a synchronous buck converter power stage. The simplified schematic of the power stage for a synchronous buck converter is shown as follows:



The operation of the synchronous buck is as follows. At the beginning of the switching period, the PWM of the top switch (Q1) is set HIGH and the bottom switch (Q2) is set LOW.

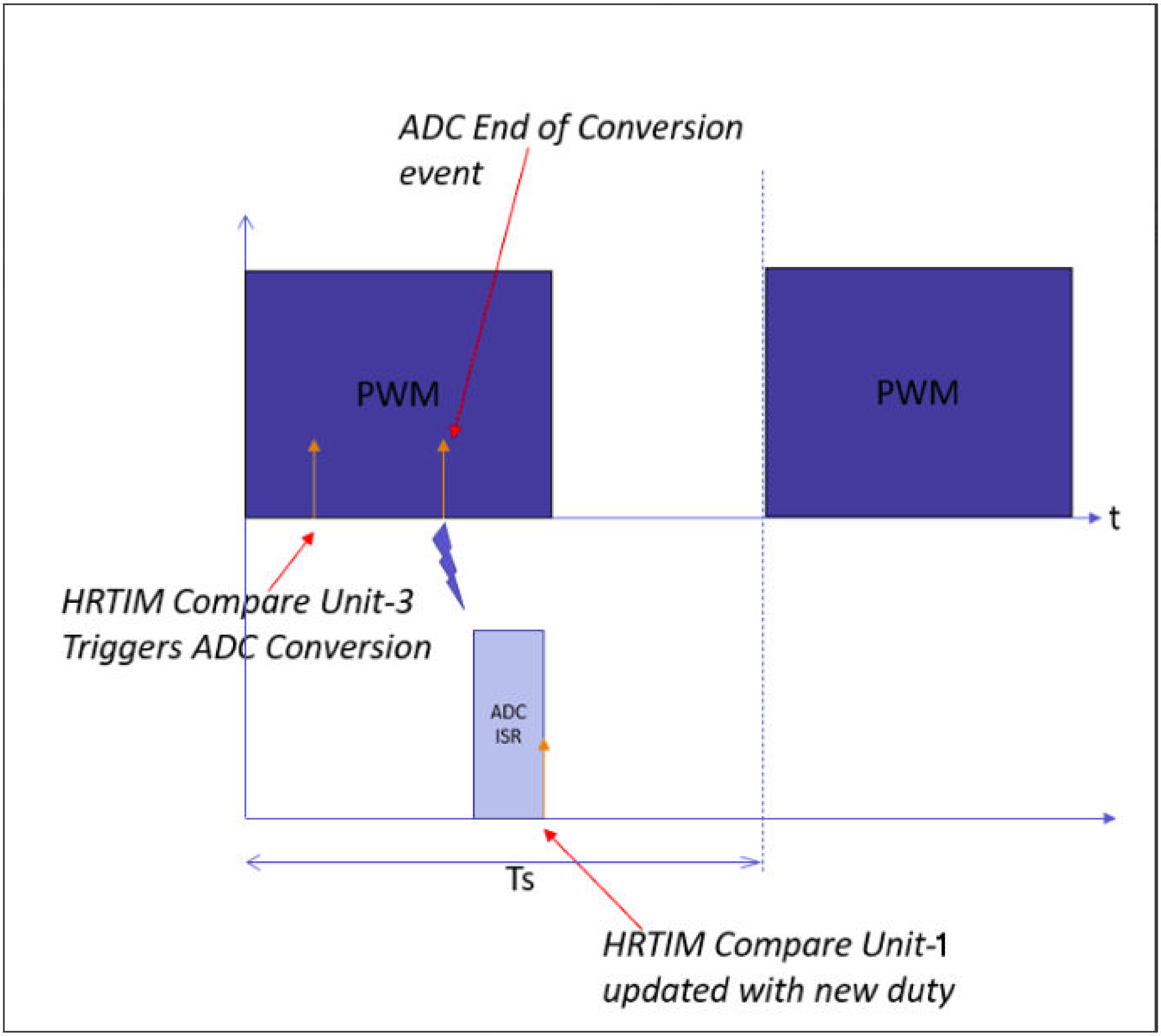
This turns on MOSFET-Q1 and turns off MOSFET Q2. With the Q1 switch conducting, the current through the inductor L1 begins increasing linearly. At the end of the high-side duty cycle, the switch Q1 is turned off.

A dead-time is inserted between the high-side and low-side PWM for switches Q1 and Q2 to prevent shoot-through, where both switches are partially on at the same time causing a large current to flow through Q1 and Q2 and can damage the MOSFETs. When this dead time has elapsed the low-side PWM for the switch Q2 goes HIGH which turns on the switch Q2. At this time, the inductor acts to continue the flow of current and the current now flows through switch Q2. The current through the inductor begins decreasing linearly. This switching action is described in the buck converter waveforms is as follows:



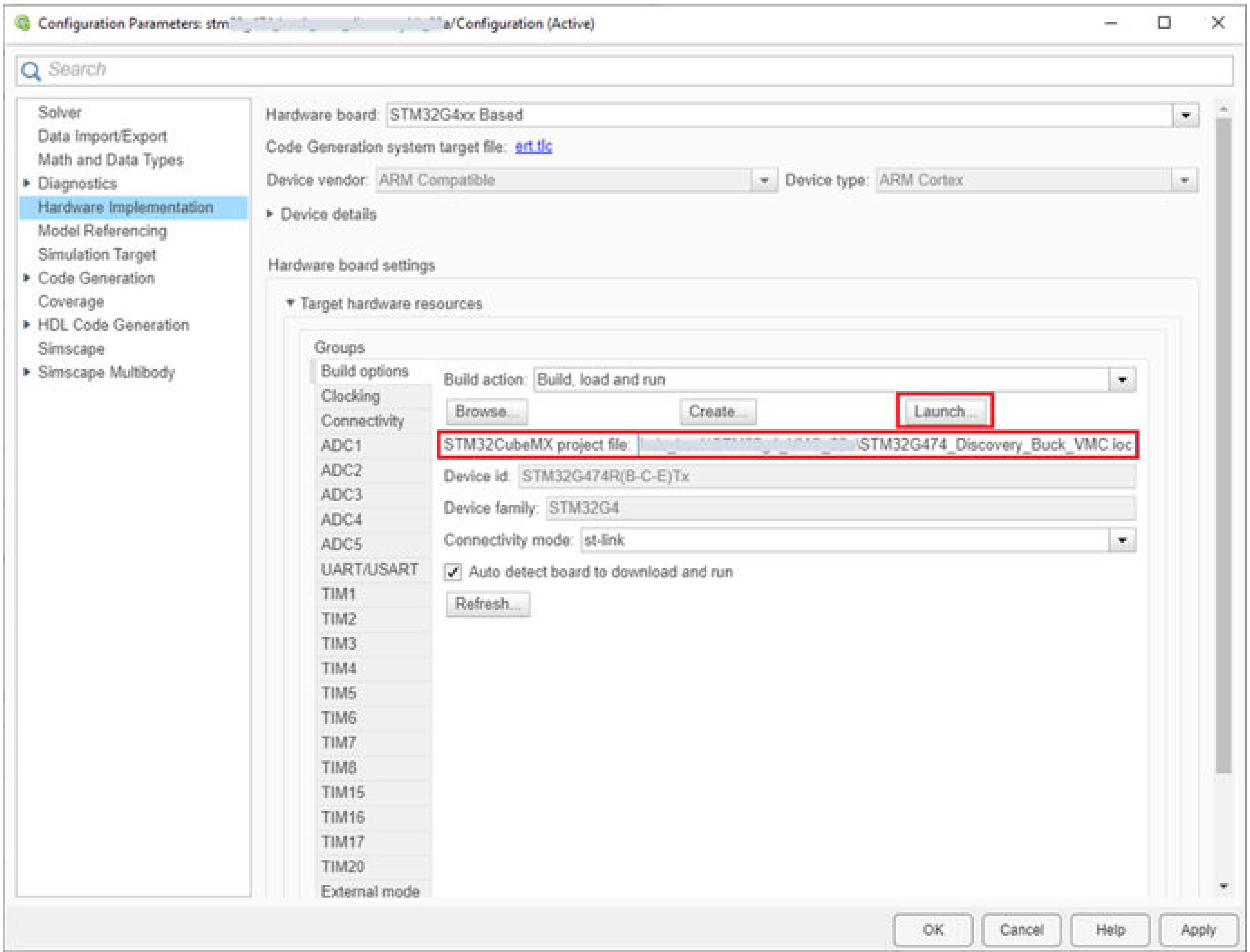
The following figure shows the overall design w.r.t peripherals to achieve the voltage mode control of the buck converter on the B-G474E-DPOW1 Discovery kit.

The sampling of the ADC module is triggered by the HRTIM module, in this case by comparing unit 3, which is set to a number of HRTIM ticks after the beginning of the switching period to avoid switching noise corrupting the sample. Once the sampling and conversion are complete, the ADC triggers End of Conversion interrupt to CPU. Within this ISR the PI controller would compute the correction duty and update the HRTIM compare unit 1. The new duty will apply from the next switching cycle of the PWM.



STM32CubeMX Configuration

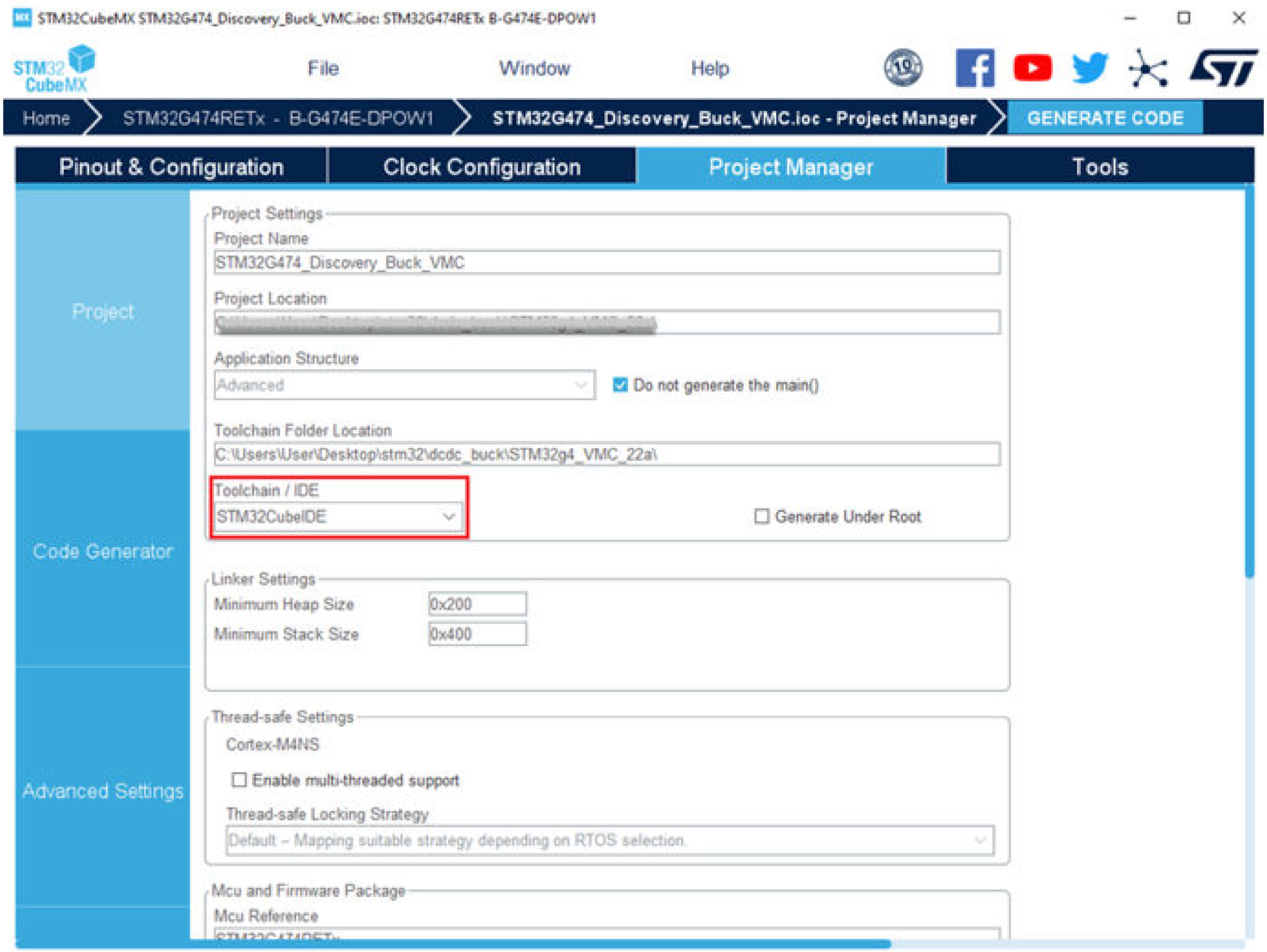
Save the `STM32G474_Discovery_Buck_VMC.ioc` provided in the ZIP folder to a known path in the PC and click **Launch** as shown.



Open STM32CubeMX project and perform the following configurations.

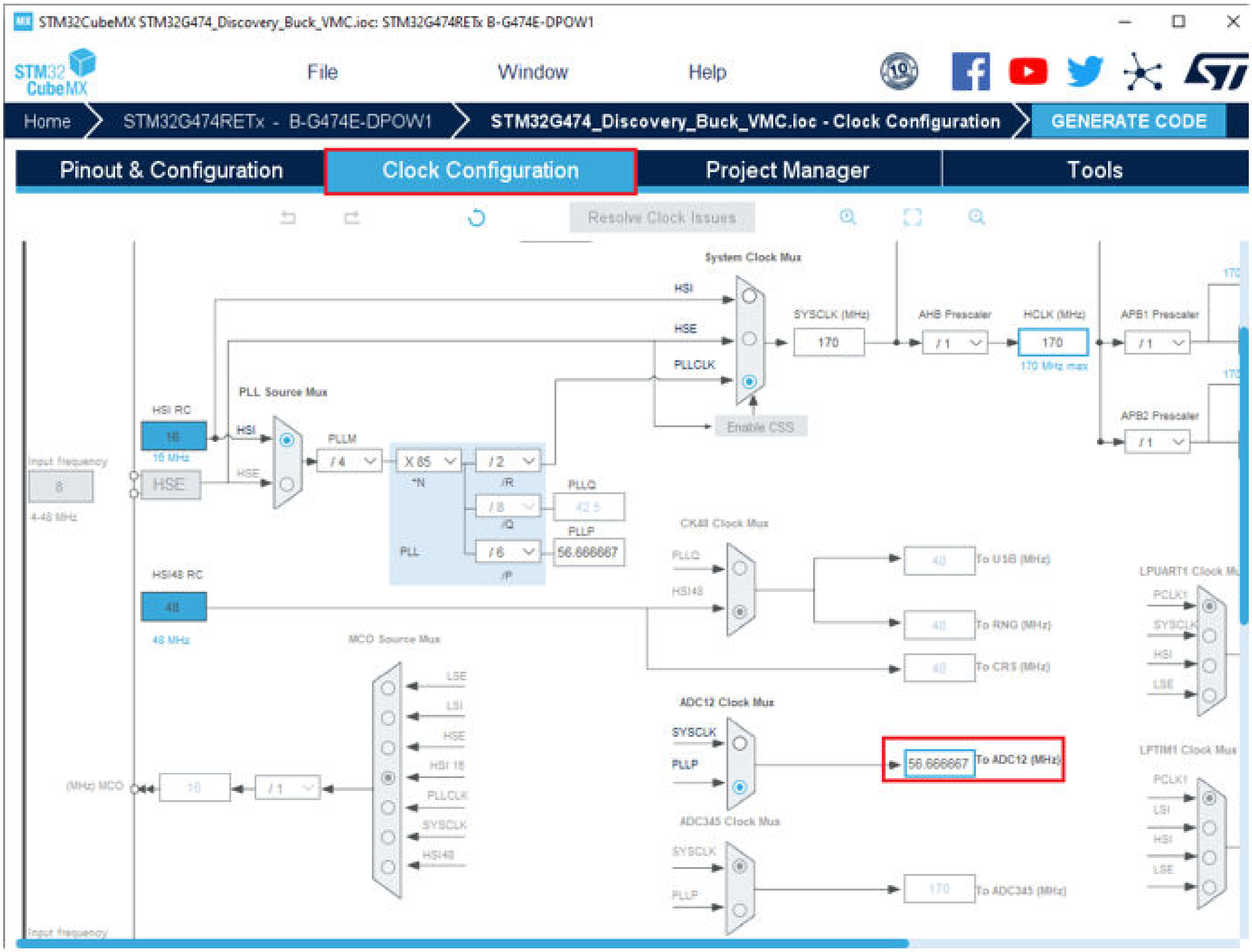
STM32CubeMX Project Manager Configuration

Select the preferred toolchain.



STM32CubeMX Clock Configuration

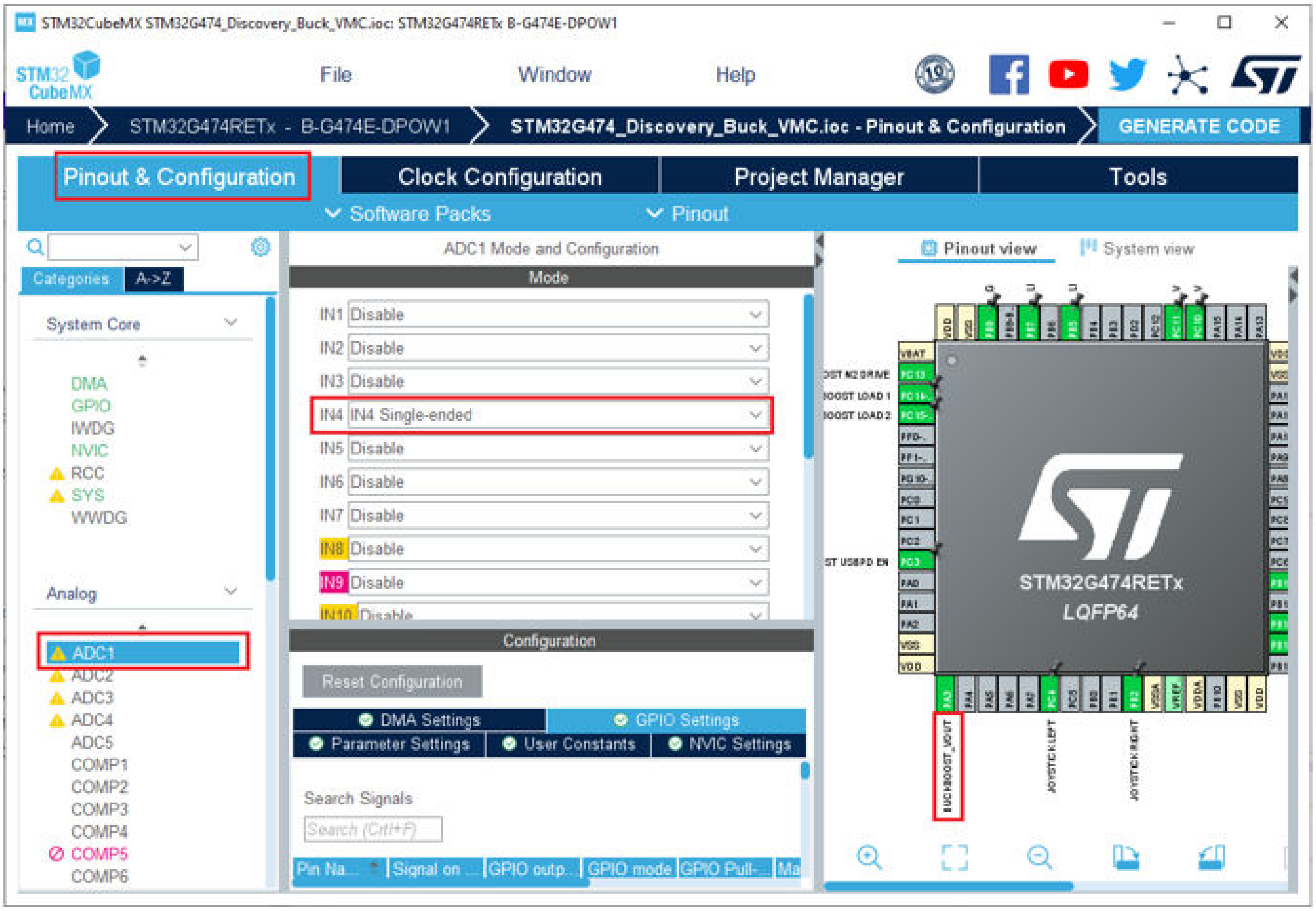
- The ADC12 clock is set to 56.66 MHz.
- HRTIM & USART3 clocks is set to 170MHz



STM32CubeMX Pin Configuration

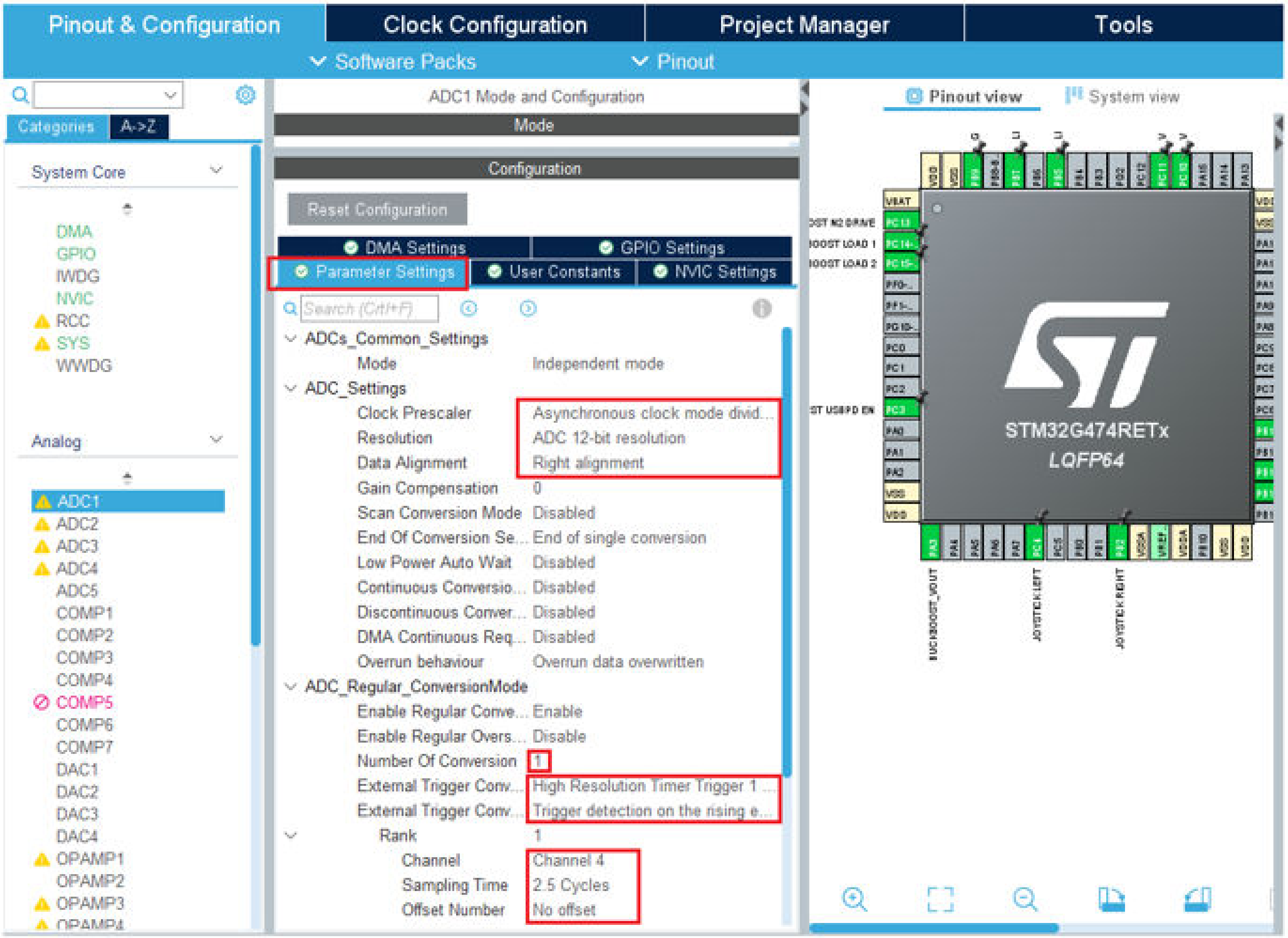
ADC Mode Settings

ADC Channel 4 (ADC_IN4) has the user label BUCKBOOST_VOUT and is connected to PA3. This is the output voltage of the buck converter that is regulated.



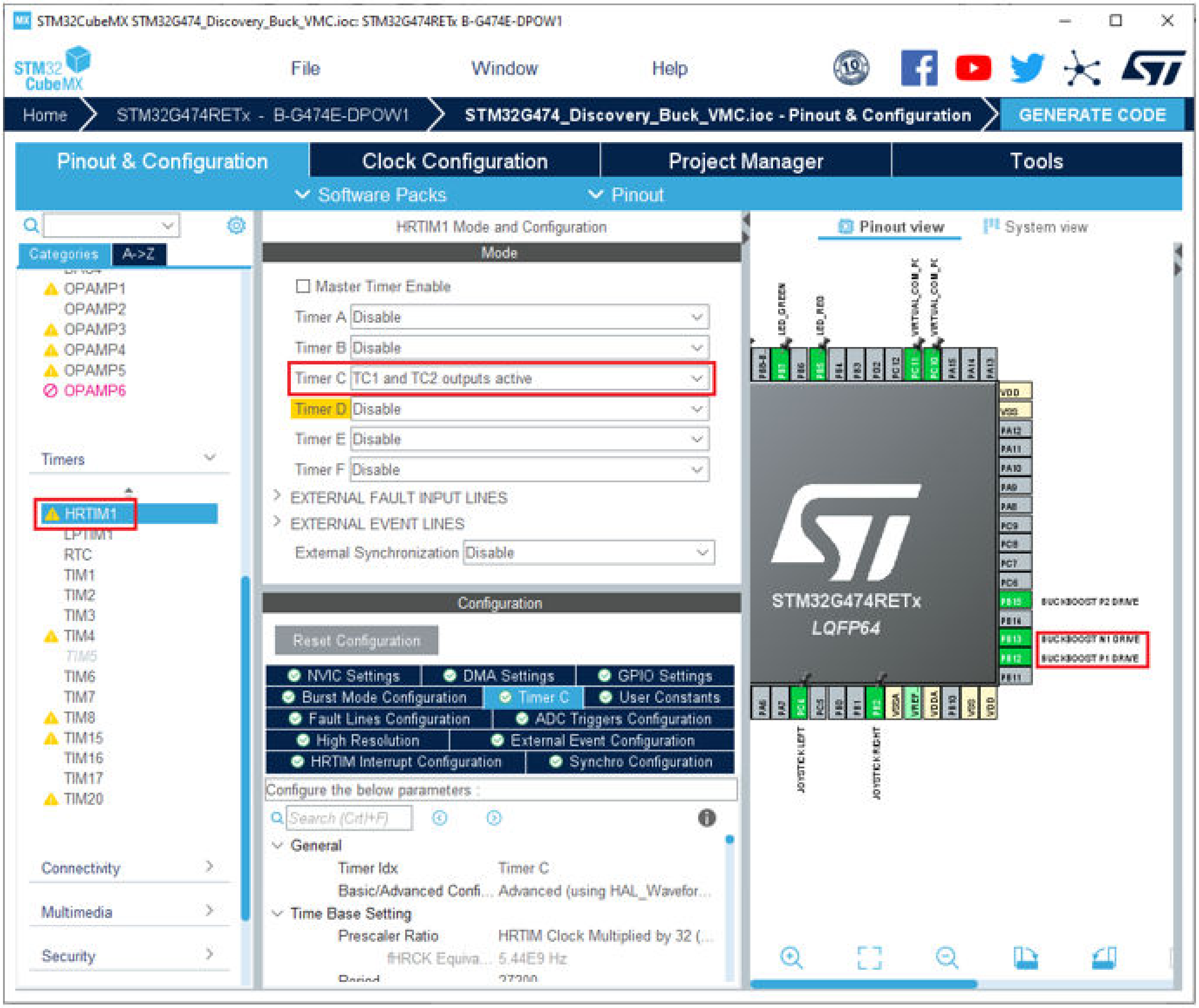
ADC Parameter Settings

- Data Alignment set to Right alignment. This stores the 12-bit result from the ADC in the 16-bit results register with the right alignment. That means the Lower12-bits (plus 1 sign bit) of the 16-bit register is used.
- Under the ADC_Regular_ConversionMode section, set the External Trigger Conversion Source to High-Resolution Trigger 1 event. Then set the External Trigger Conversion Edge to Trigger detection on the rising edge.
- For this voltage mode buck converter, only one ADC conversion is required. This is the output voltage rail that is regulated. Therefore, the number of conversions is currently set to one. This conversion can be configured by expanding the Rank subcategory.
- Set the Channel to Channel 4.



HRTIM Mode Settings

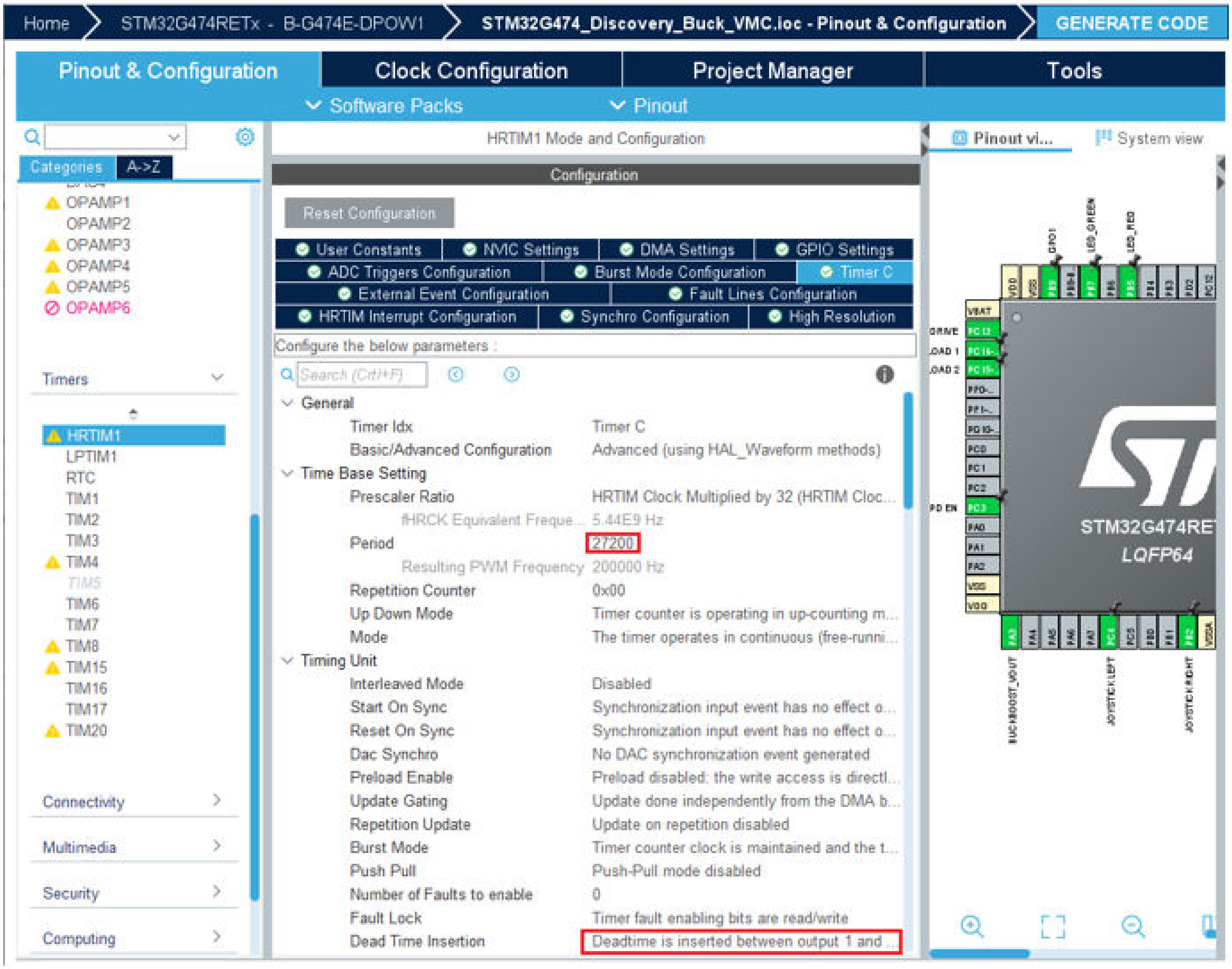
The hi-resolution timer module is configured to generate the PWM. The Buck power stage is driven by Timer C.



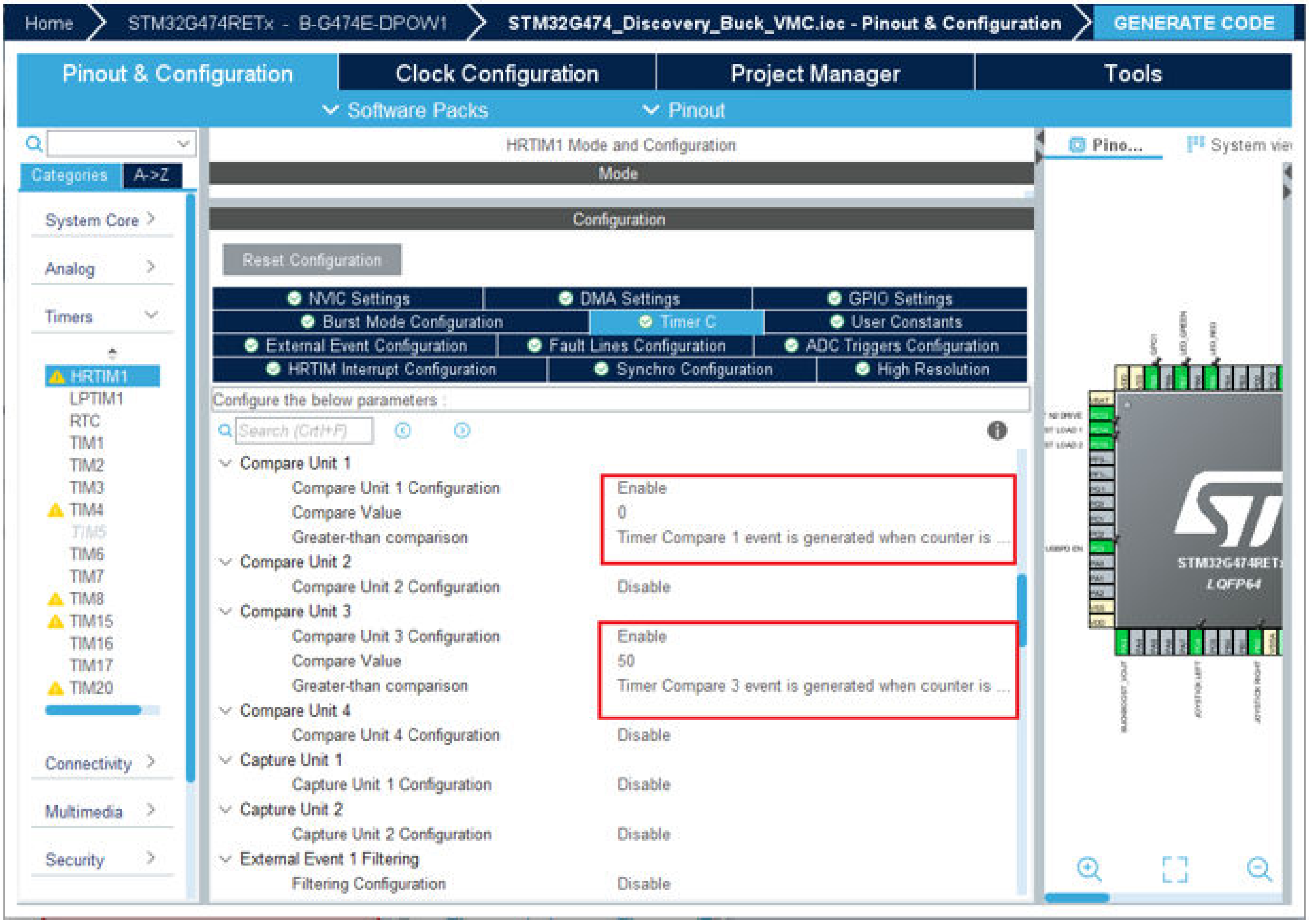
HRTIM Parameter Settings

- Set the period value to 27200 with the resulting PWM period now calculated as 200000 Hz.
- Set Dead Time insertion to Deadtime is inserted between output 1 and output 2. This enables the deadtime section which allows the outputs to be configured such that both high-side and low-side

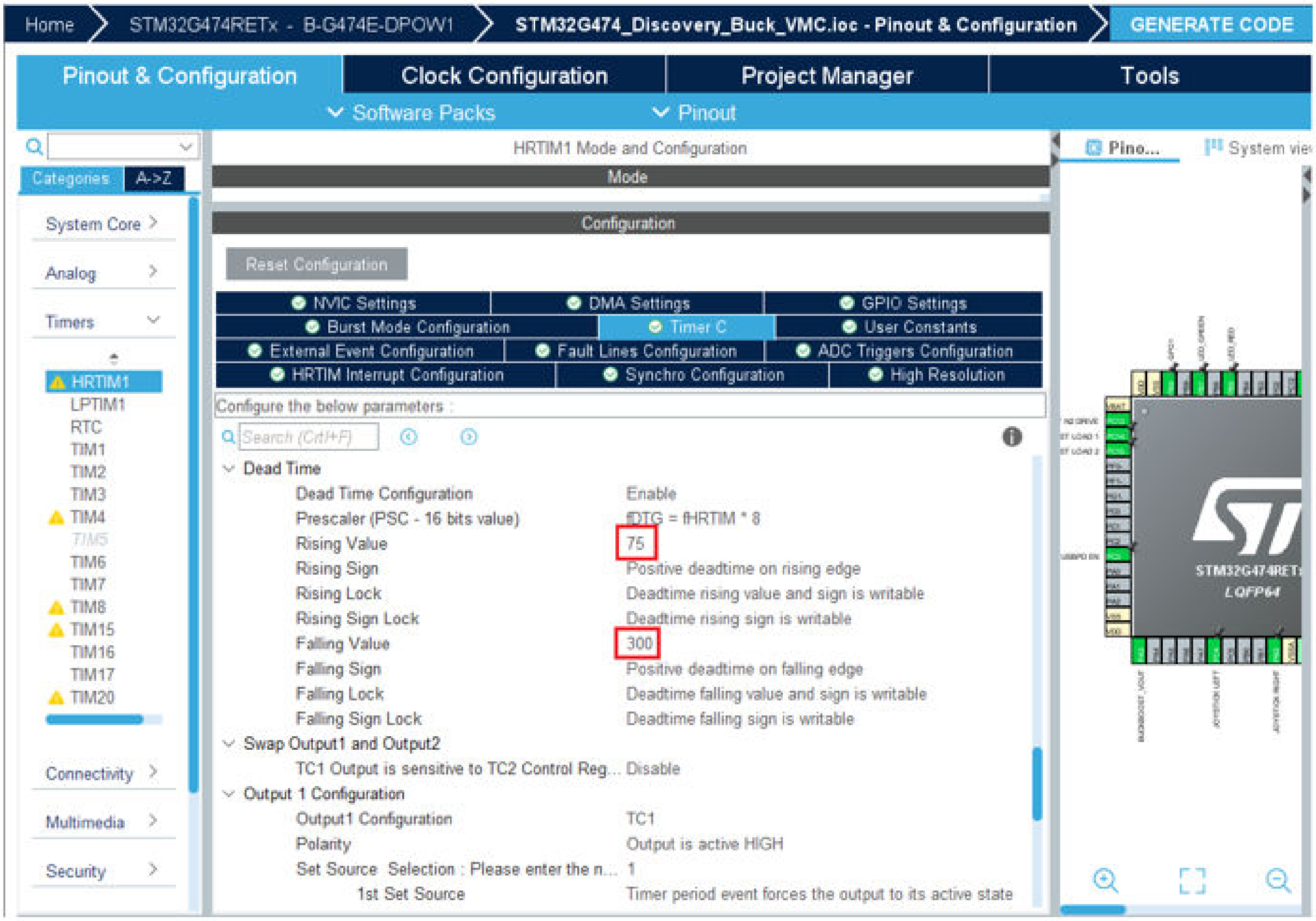
switches are never driven at the same time.



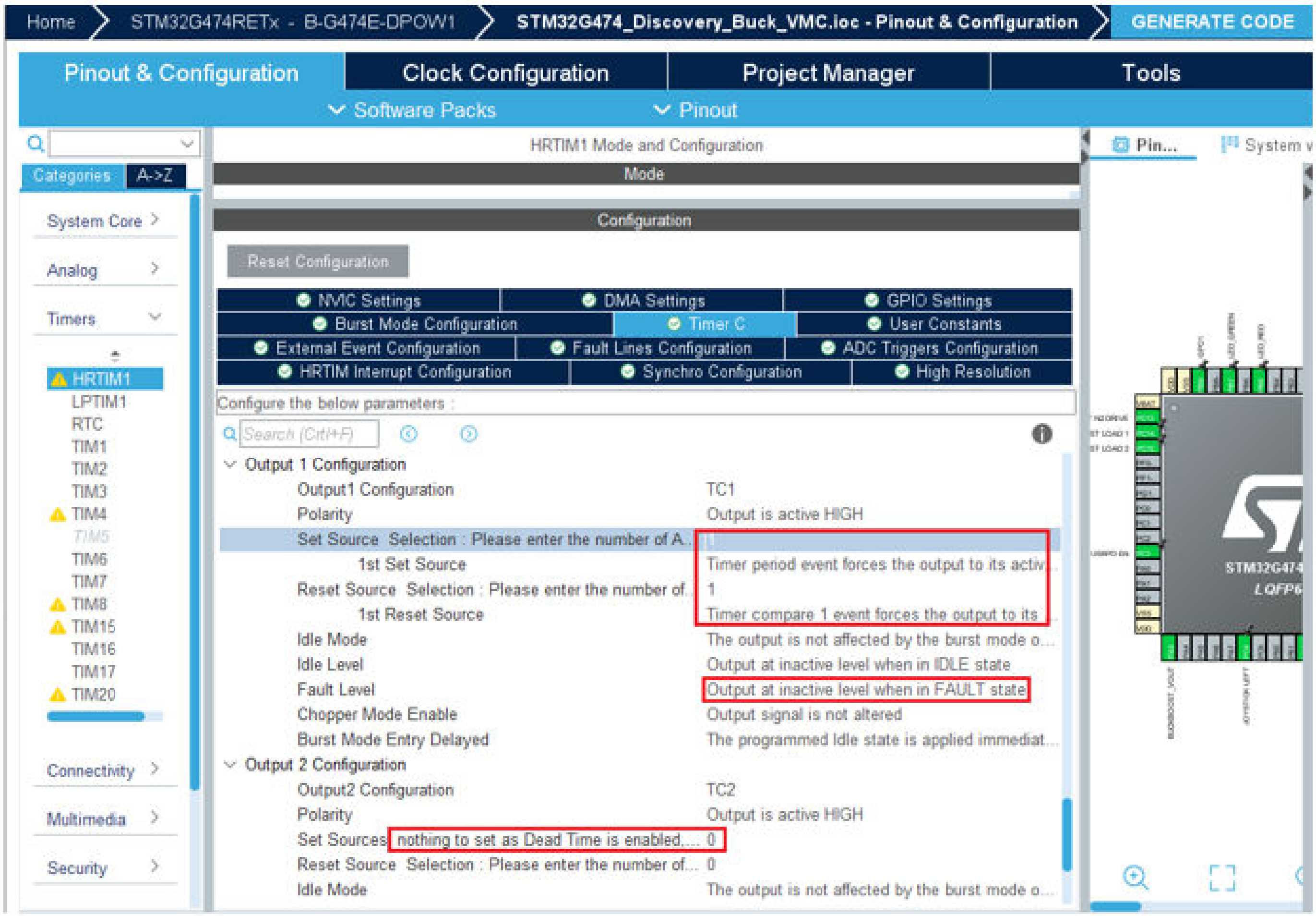
- For Compare Unit 1 section change the configuration setting to Enable. Enter a Compare Value of 0. This is the initial value of the duty cycle fixed to 0%.
- For Compare Unit 3 section change the configuration setting to Enable. Enter a Compare Value of 50. This is the event that is used to trigger the ADC sample. This can be moved to a clean point in the switching cycle, which means away from the turn-on or turn-off of the switch.



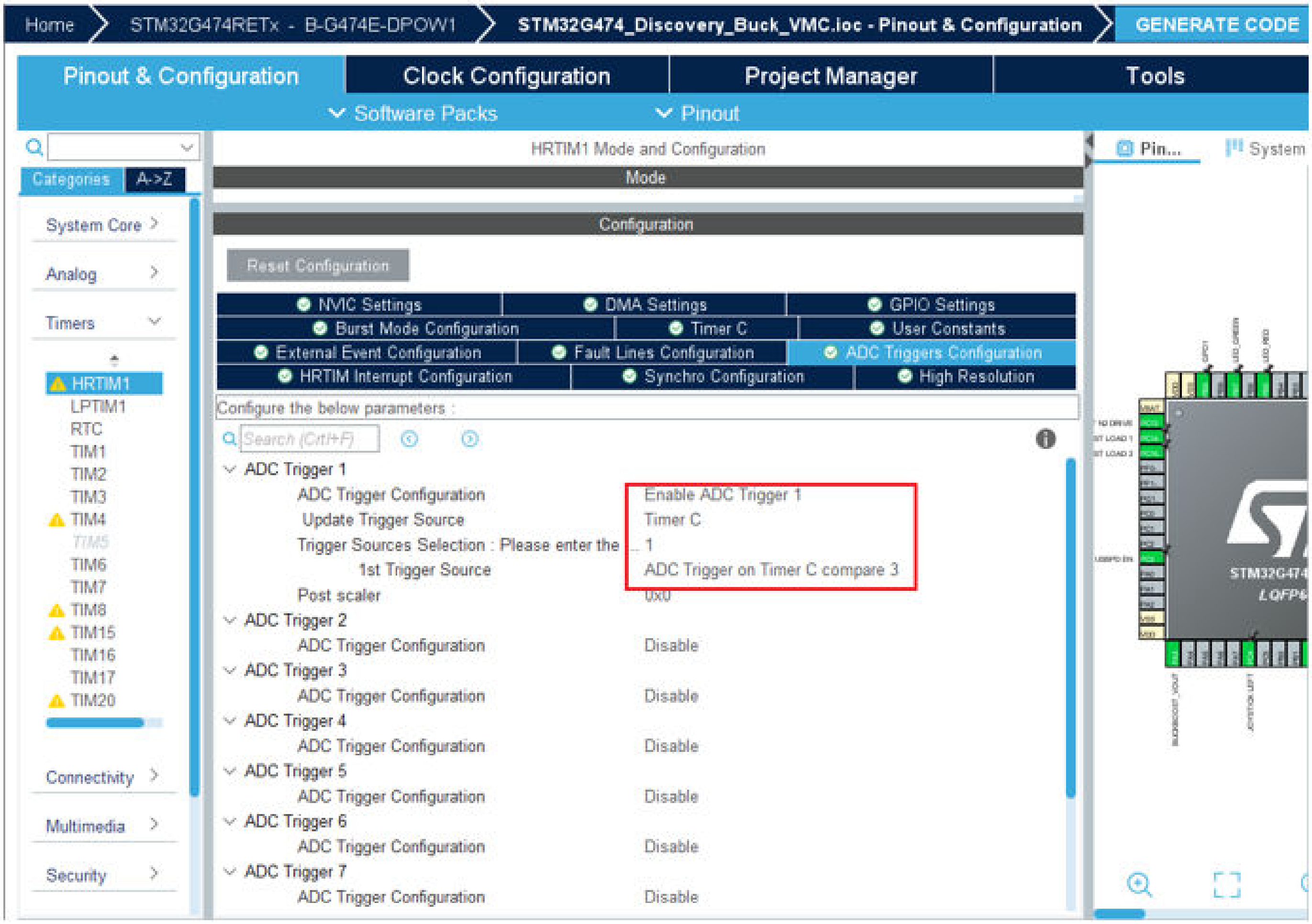
- Rising Value to 75 and Falling Value to 300.
- The dead-time ticks fDTG are generated from fHRTIM and a prescaler:
  - fDTG = fHRTIM * 8, and because fHRTIM = 170 MHz: fDTG = 170 MHz * 8 = 1360 MHz
  - therefore: 1 dead-time tick = 0.735 ns.
  - Hence: Rising value = 75 ticks = 55 ns Falling value = 300 ticks = 220 ns



- Under the Output 1 Configuration section, change the Set Source Selection number to 1. Change the 1st Set Source event to Timer period event forces the output to its active state.
- Change the Reset Source Selection number to 1. Change the 1st Reset Source event to Timer compare 1 event forces the output to its inactive state.
- Finally, set the Fault Level to Output at the inactive level when in FAULT state.
- The Output 2 Configuration section is left unchanged as this output is driven by the dead-time.

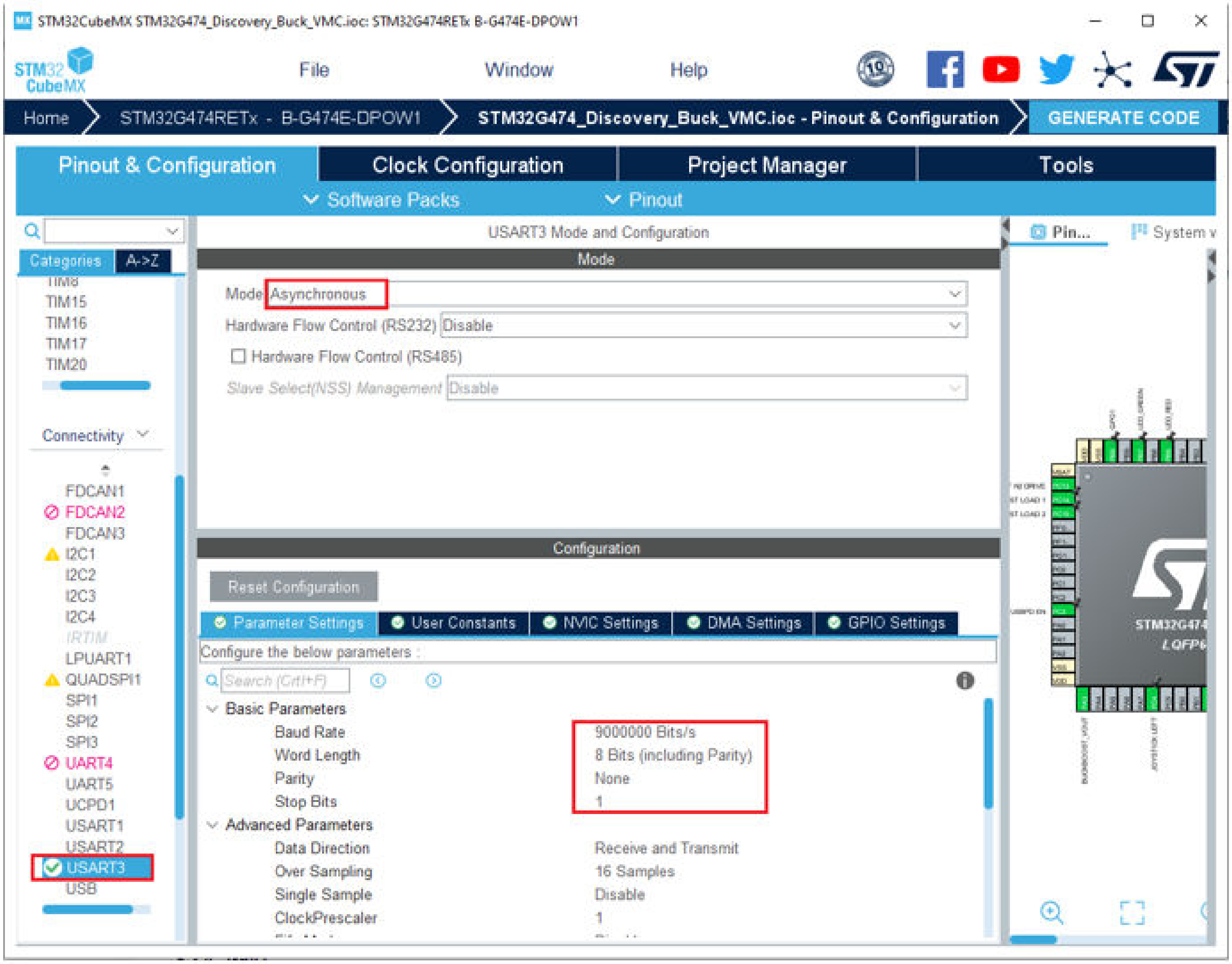


- Enable the ADC Trigger 1 and set the Update Trigger Source to Timer C, enter the number of Trigger Sources Selection to 1. From the 1st Trigger Source drop-down box, select ADC Trigger on Timer C compare 3.
- This is the event that triggers the ADC.


Home
STM32G474RETx - B-G474E-DPOW1
STM32G474_Discovery_Buck_VMC.ioc - Pinout & Configuration
GENERATE CODE
Pinout & Configuration
Clock Configuration
Project Manager
Tools
Software Packs
Pinout
Categories
A->Z
System Core
Analog
Timers
HRTIM1
LPTIM1
RTC
TIM1
TIM2
TIM3
TIM4
TIM5
TIM6
TIM7
TIM8
TIM15
TIM16
TIM17
TIM20
Connectivity
Multimedia
Security
HRTIM1 Mode and Configuration
Mode
Configuration
Reset Configuration
NVIC Settings
DMA Settings
GPIO Settings
Burst Mode Configuration
Timer C
User Constants
External Event Configuration
Fault Lines Configuration
ADC Triggers Configuration
HRTIM Interrupt Configuration
Synchro Configuration
High Resolution
Configure the below parameters :
Search (Crtl+F)
ADC Trigger 1
ADC Trigger Configuration
Enable ADC Trigger 1
Update Trigger Source
Timer C
Trigger Sources Selection : Please enter the 1st Trigger Source
1
ADC Trigger on Timer C compare 3
Post scaler
0x0
ADC Trigger 2
ADC Trigger Configuration
Disable
ADC Trigger 3
ADC Trigger Configuration
Disable
ADC Trigger 4
ADC Trigger Configuration
Disable
ADC Trigger 5
ADC Trigger Configuration
Disable
ADC Trigger 6
ADC Trigger Configuration
Disable
ADC Trigger 7
ADC Trigger Configuration
Disable
Pin...
System
STM32G474
LQFP6

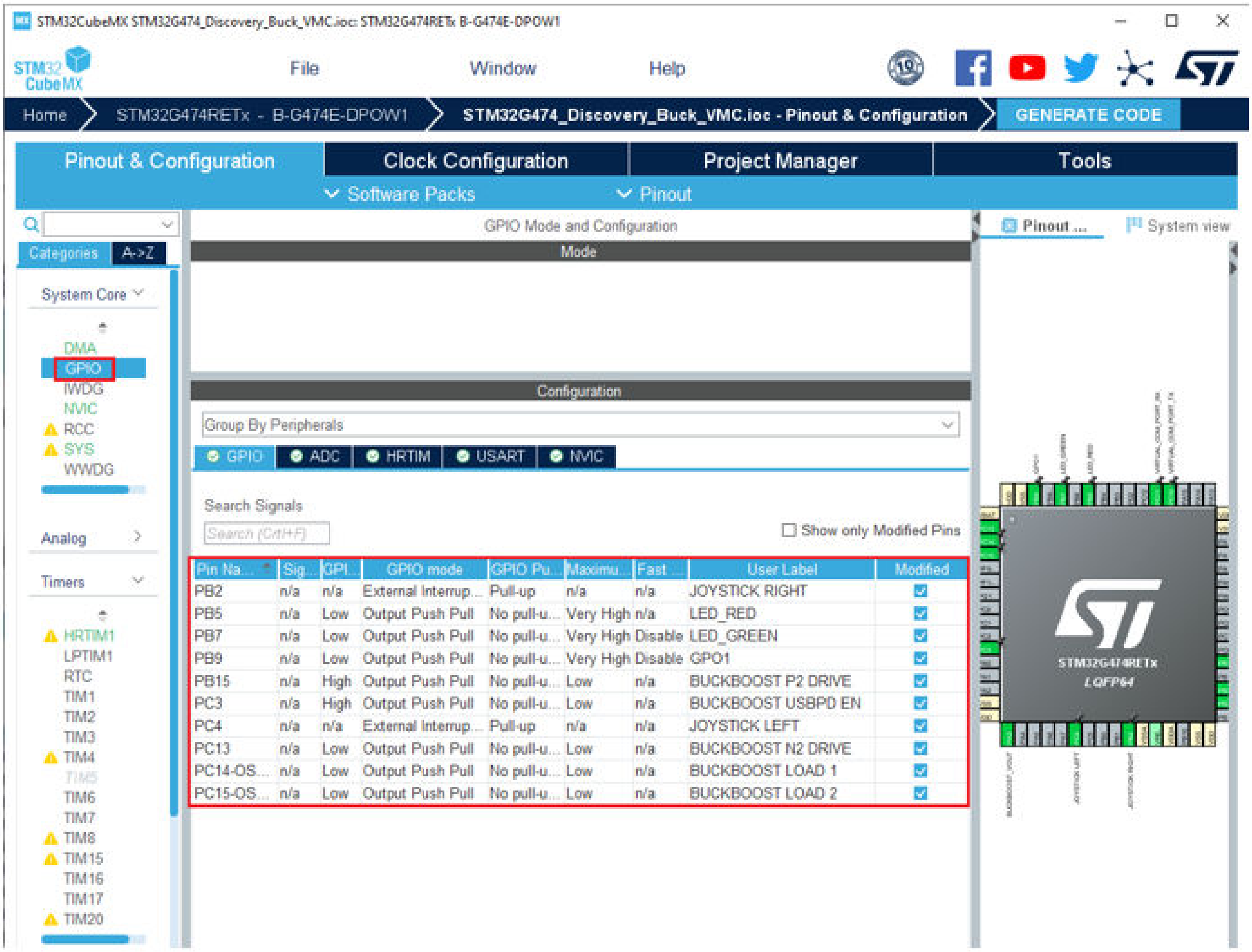
USART3 mode and configuration settings

Configure USART3 for Host-Target communication.



GPIO configuration settings

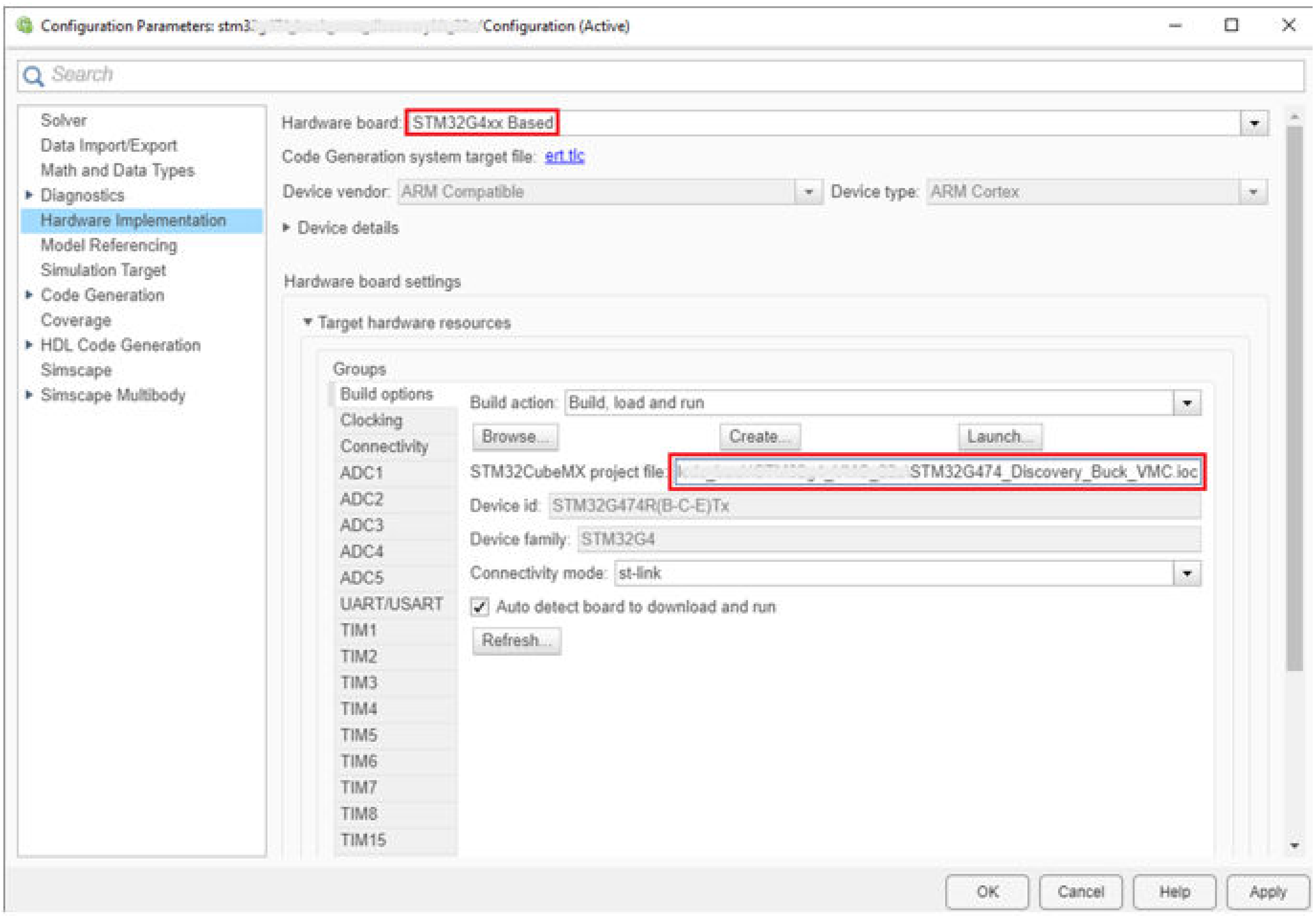
- BUCKBOOST P2 DRIVE and BUCKBOOST N2 DRIVE are not used in the Buck-only configuration.
- However, the BUCKBOOST P2 DRIVE must be High to support the Buck-only configuration.
- BUCKBOOST USBPD EN would enable the input 5V to the power converter stage.


STM32CubeMX STM32G474_Discovery_Buck_VMC.ioc: STM32G474RETx B-G474E-DPOW1
File
Window
Help
Home
STM32G474RETx - B-G474E-DPOW1
STM32G474_Discovery_Buck_VMC.ioc - Pinout & Configuration
GENERATE CODE
Pinout & Configuration
Clock Configuration
Project Manager
Tools
Software Packs
Pinout
Categories
A->Z
System Core
DMA
GPIO
IWDG
NVIC
RCC
SYS
WWDG
Analog
Timers
HRTIM1
LPTIM1
RTC
TIM1
TIM2
TIM3
TIM4
TIM5
TIM6
TIM7
TIM8
TIM15
TIM16
TIM17
TIM20
GPIO Mode and Configuration
Mode
Configuration
Group By Peripherals
GPIO
ADC
HRTIM
USART
NVIC
Search Signals
Search (Crtl+F)
Show only Modified Pins
Pin Na... | Sig... | GPI... | GPIO mode | GPIO Pu... | Maximu... | Fast ... | User Label | Modified
PB2 | n/a | n/a | External Interrup... | Pull-up | n/a | n/a | JOYSTICK RIGHT
PB5 | n/a | Low | Output Push Pull | No pull-u... | Very High | n/a | LED_RED
PB7 | n/a | Low | Output Push Pull | No pull-u... | Very High | Disable | LED_GREEN
PB9 | n/a | Low | Output Push Pull | No pull-u... | Very High | Disable | GPO1
PB15 | n/a | High | Output Push Pull | No pull-u... | Low | n/a | BUCKBOOST P2 DRIVE
PC3 | n/a | High | Output Push Pull | No pull-u... | Low | n/a | BUCKBOOST USBPD EN
PC4 | n/a | n/a | External Interrup... | Pull-up | n/a | n/a | JOYSTICK LEFT
PC13 | n/a | Low | Output Push Pull | No pull-u... | Low | n/a | BUCKBOOST N2 DRIVE
PC14-OS... | n/a | Low | Output Push Pull | No pull-u... | Low | n/a | BUCKBOOST LOAD 1
PC15-OS... | n/a | Low | Output Push Pull | No pull-u... | Low | n/a | BUCKBOOST LOAD 2
Pinout ...
System view
STM32G474RETx
LQFP64

Simulink Configuration

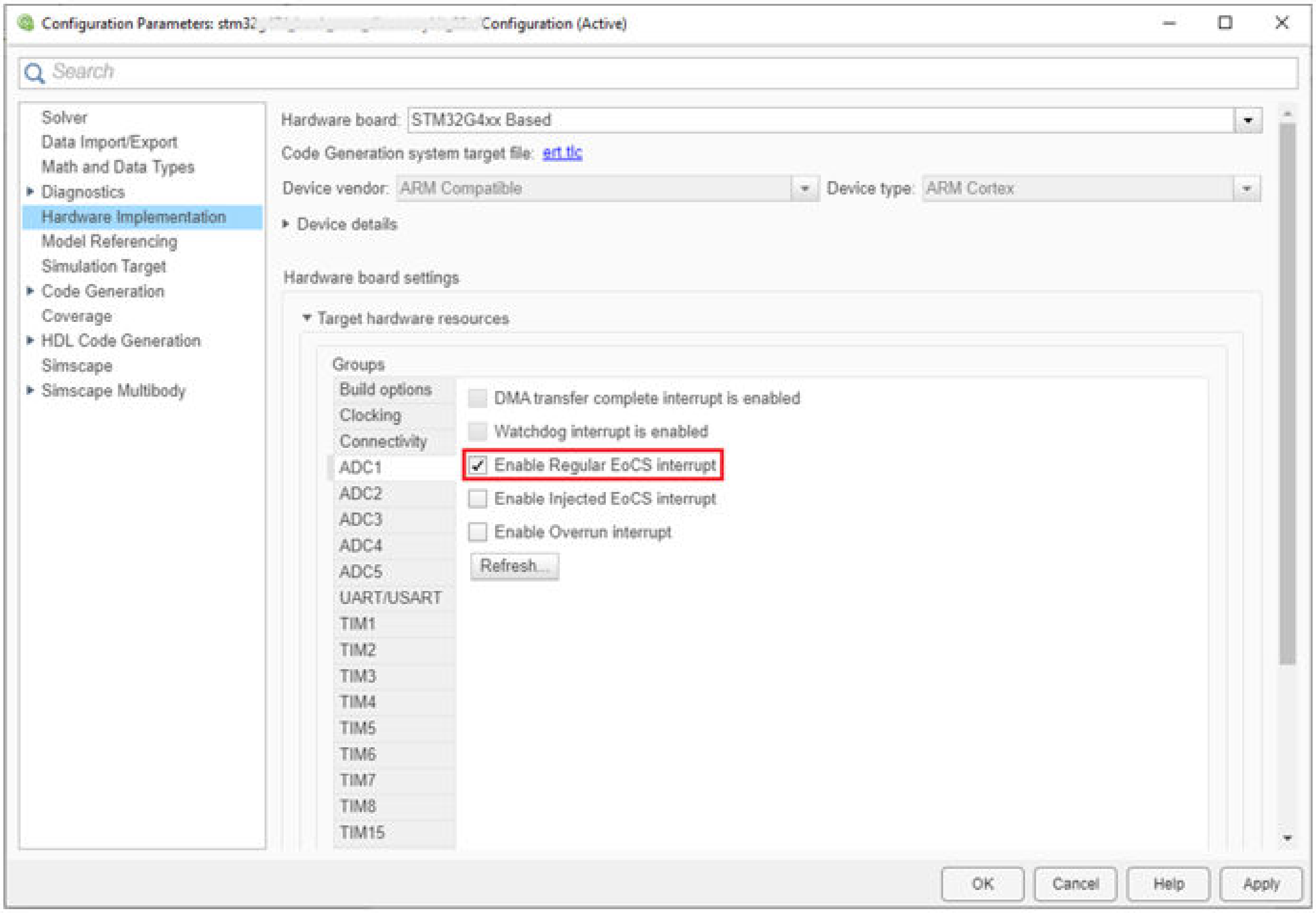
Hardware Implementation

- The example is based on B-G474E-DPOW1 Discovery kit. Set the hardware board to STM32G4xx Based.
- Notice the path to STM32CubeMX project `\..\STM32G474_Discovery_Buck_VMC.ioc` which contains the peripheral level configurations as discussed before.



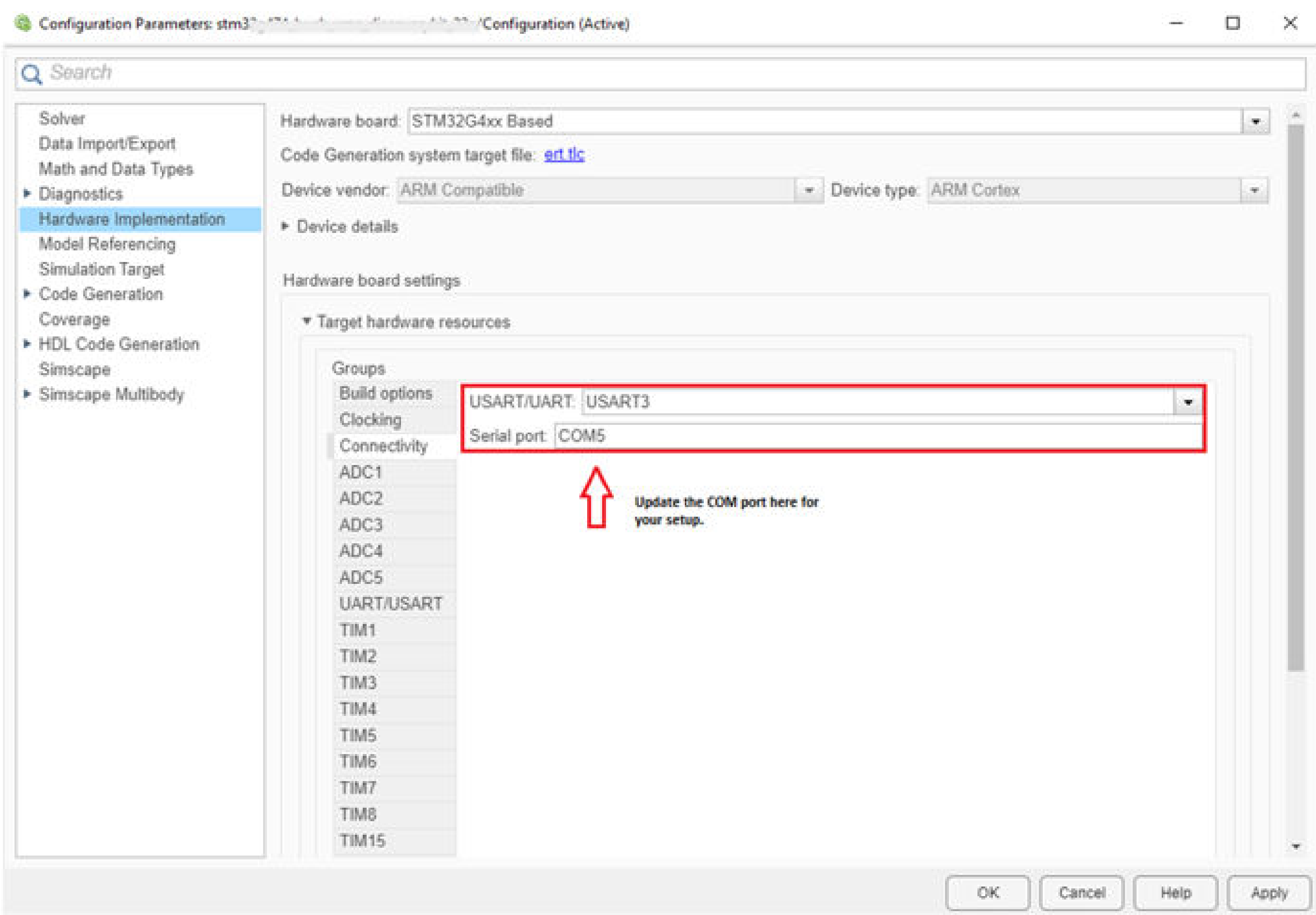
ADC Configuration

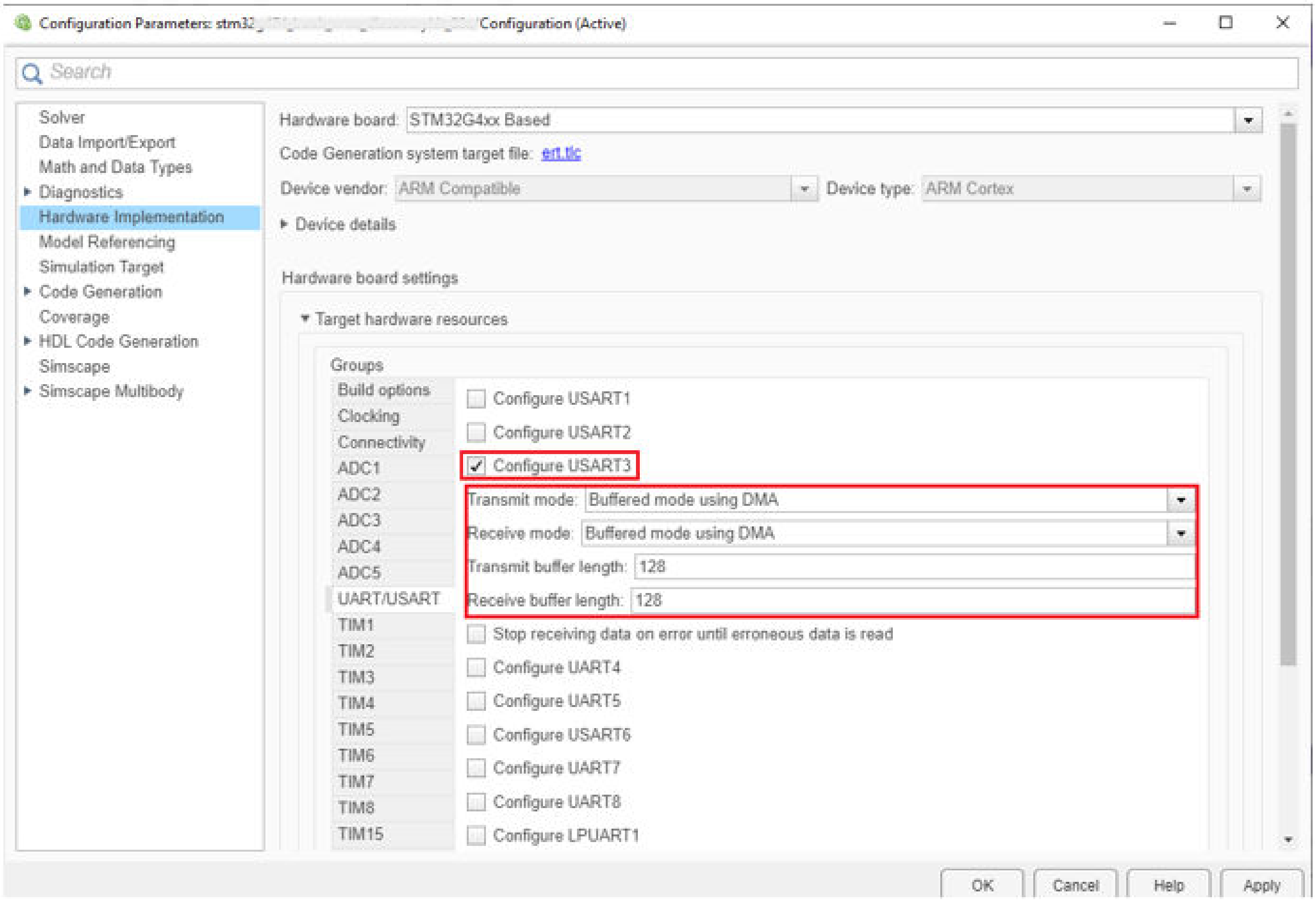
Enable ADC EoC (End of Conversion) interrupt.



USART Configuration

Configure USART3 for Host-Target communication.

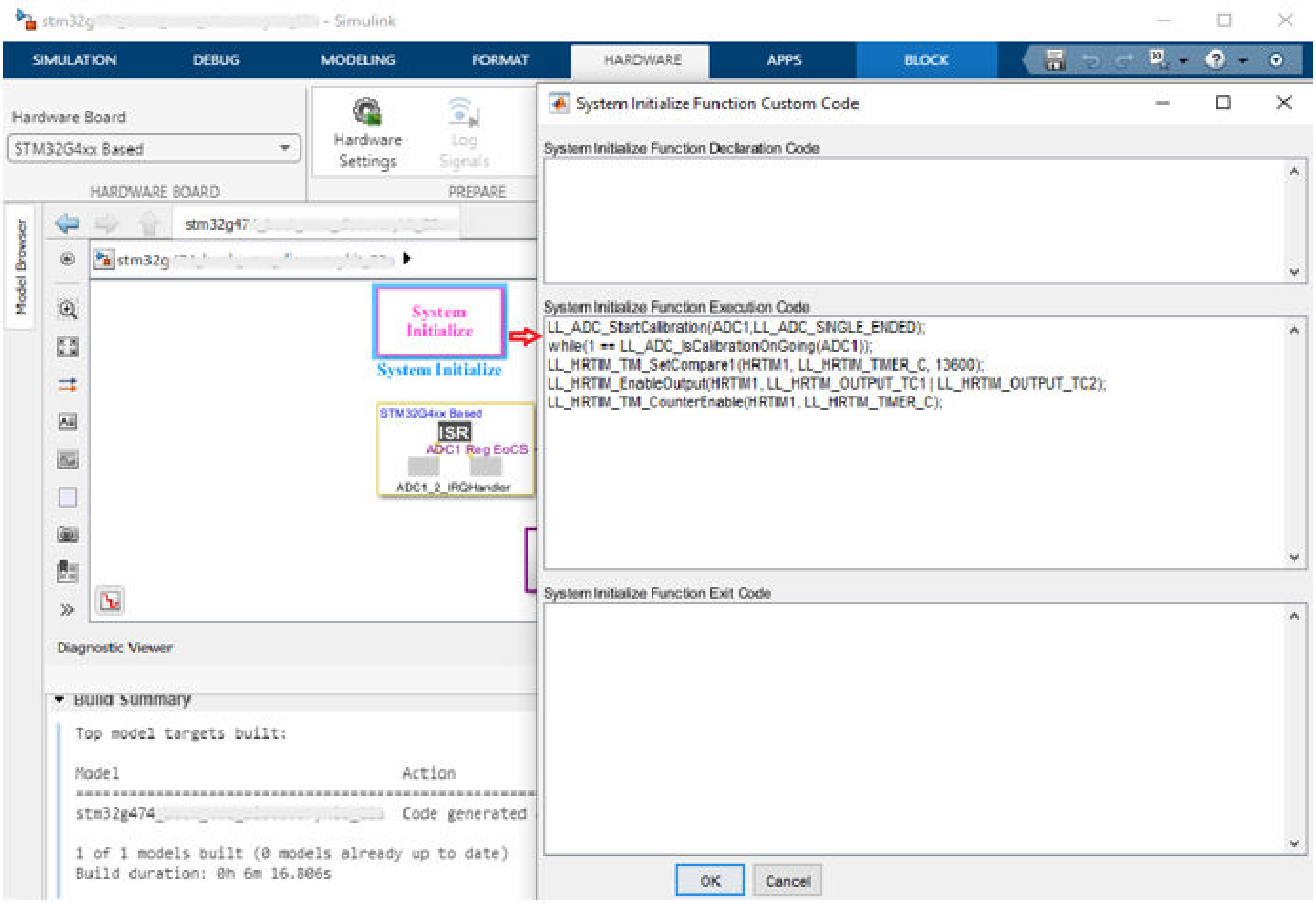

Configuration Parameters: stm32 /Configuration (Active)
Search
Solver
Data Import/Export
Math and Data Types
Diagnostics
Hardware Implementation
Model Referencing
Simulation Target
Code Generation
Coverage
HDL Code Generation
Simscape
Simscape Multibody
Hardware board: STM32G4xx Based
Code Generation system target file: ert.tlc
Device vendor: ARM Compatible
Device type: ARM Cortex
Device details
Hardware board settings
Target hardware resources
Groups
Build options
Clocking
Connectivity
ADC1
ADC2
ADC3
ADC4
ADC5
UART/USART
TIM1
TIM2
TIM3
TIM4
TIM5
TIM6
TIM7
TIM8
TIM15
USART/UART: USART3
Serial port: COM5
Update the COM port here for your setup.
OK
Cancel
Help
Apply


Configuration Parameters: stm32_ Configuration (Active)
Search
Solver
Data Import/Export
Math and Data Types
Diagnostics
Hardware Implementation
Model Referencing
Simulation Target
Code Generation
Coverage
HDL Code Generation
Simscape
Simscape Multibody
Hardware board: STM32G4xx Based
Code Generation system target file: ert.tlc
Device vendor: ARM Compatible
Device type: ARM Cortex
Device details
Hardware board settings
Target hardware resources
Groups
Build options
Clocking
Connectivity
ADC1
ADC2
ADC3
ADC4
ADC5
UART/USART
TIM1
TIM2
TIM3
TIM4
TIM5
TIM6
TIM7
TIM8
TIM15
Configure USART1
Configure USART2
Configure USART3
Transmit mode: Buffered mode using DMA
Receive mode: Buffered mode using DMA
Transmit buffer length: 128
Receive buffer length: 128
Stop receiving data on error until erroneous data is read
Configure UART4
Configure UART5
Configure USART6
Configure UART7
Configure UART8
Configure LPUART1
OK
Cancel
Help
Apply

Simulink Custom Code

1 System initialize function execution code for ADC calibration and HRTIM enable.

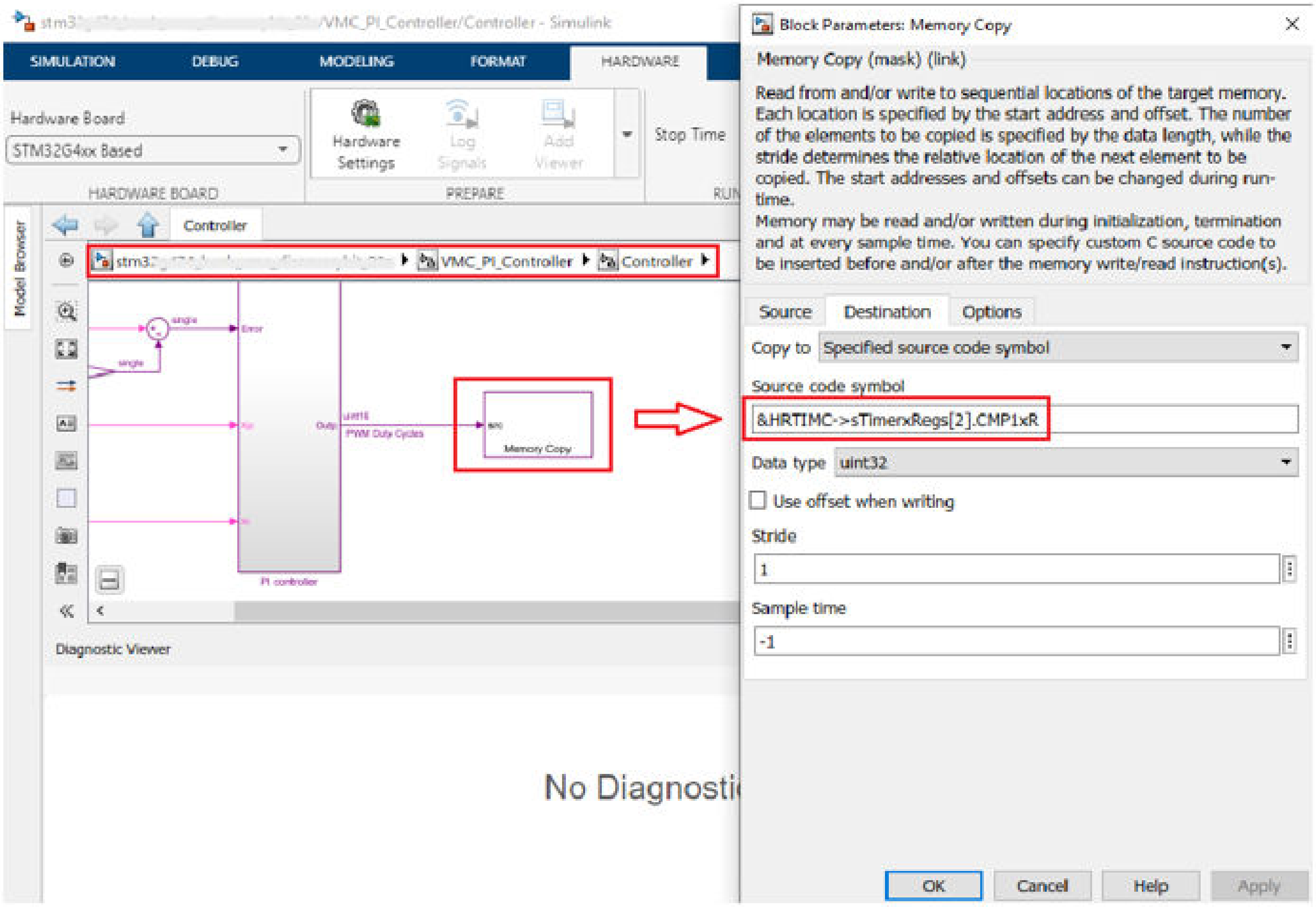
Note This feature is not supported in R2022a.



- HRTIM Custom Code configuration to ensure we enable the TIMER C to generate the complementary dead-band PWM outputs to drive the buck converter stage
- Default duty cycle set to 50%
- ADC auto-calibration is performed to compensate for any ADC bias voltage

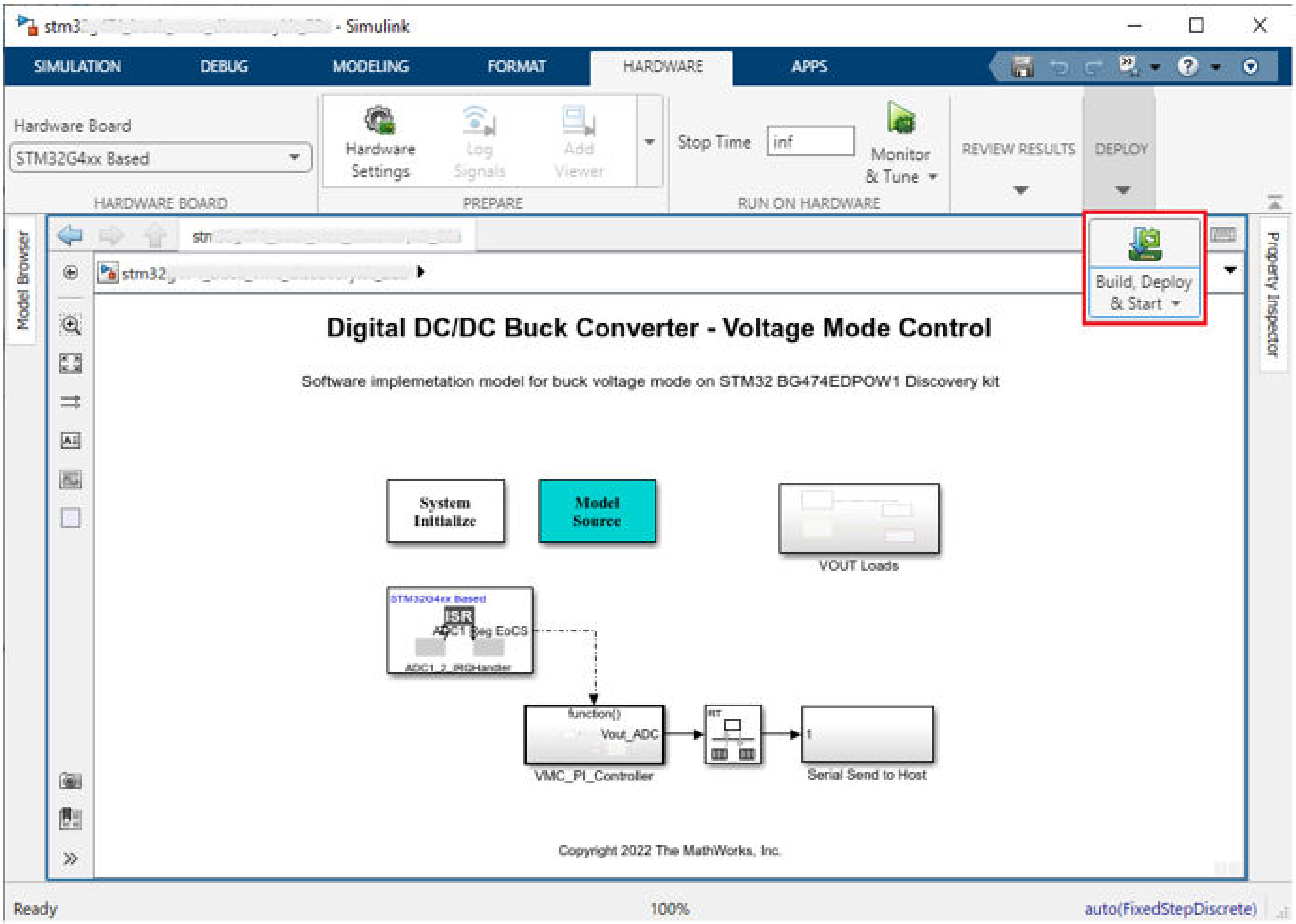
2 Dynamic update of peripheral registers.

Note HRTIM not supported in R2022a.



Duty cycle correction as obtained from PI controller is updated into the HRTIM TIMERC CMP1 register.

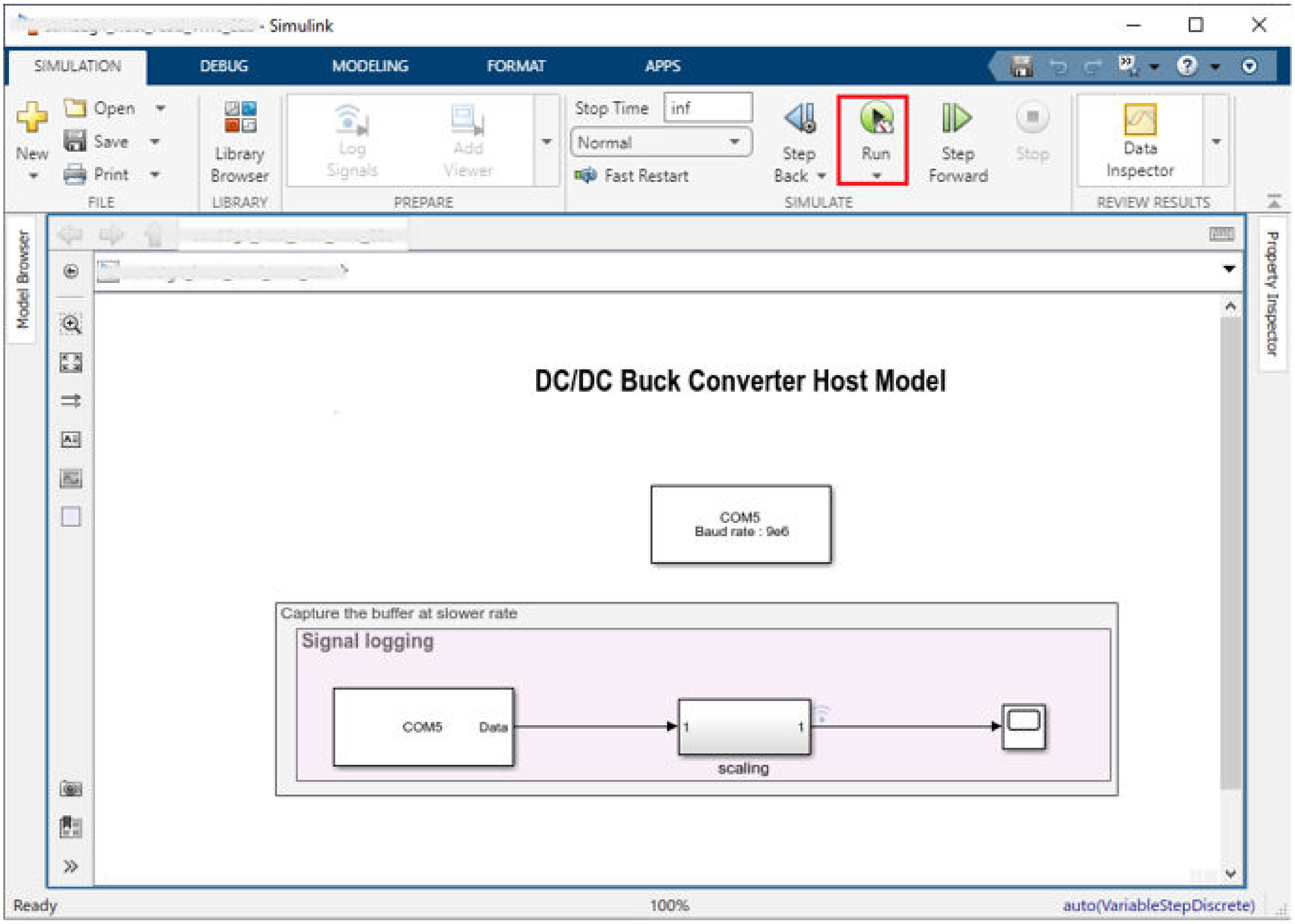
Deploy Model to Target Hardware



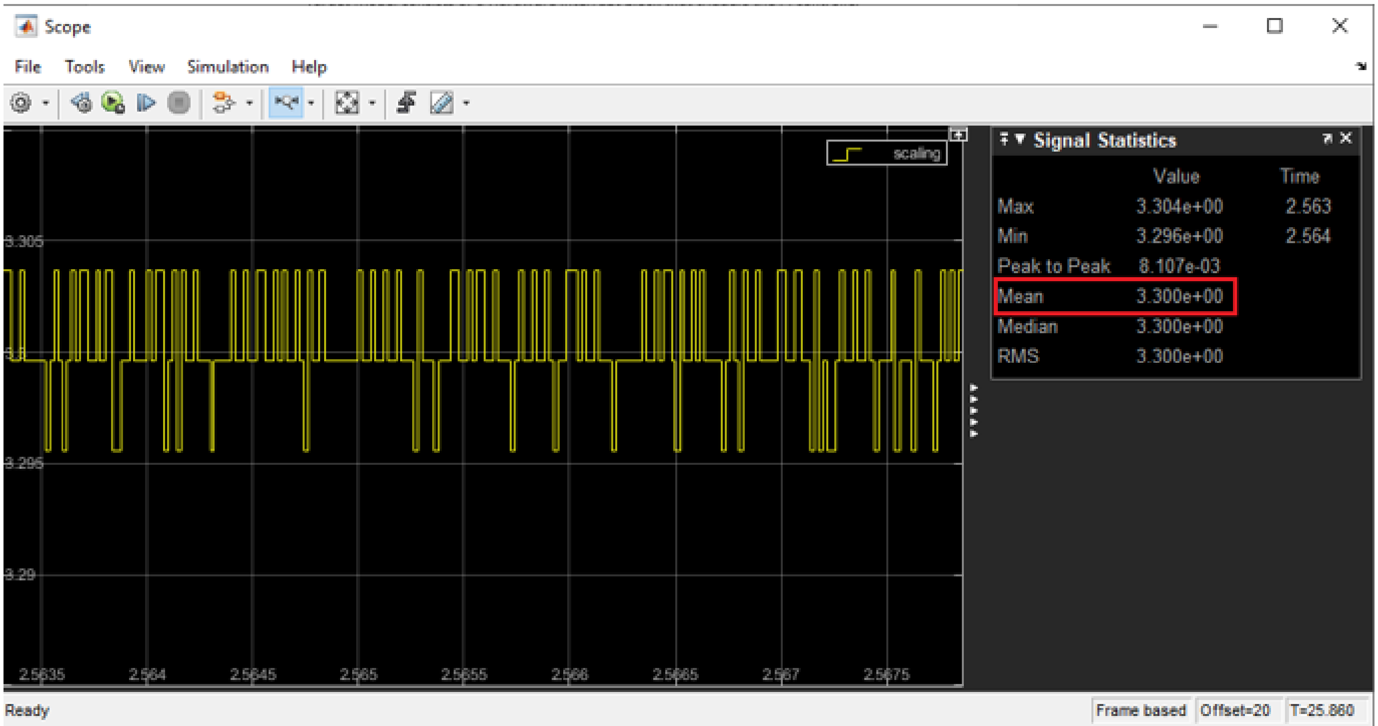
1. Unzip the contents of the project zip folder.
2. Target Model consists of a Hardware Interrupt block that triggers the PI controller Subsystem at 200KHz.
3. The VMC_PI_Controller would then read the current output voltage and compare it to the voltage setpoint.
4. The error is then used by PI controller to compute the correction signal.
5. The new computed duty is then programmed into the HRTIM compare register.
6. The Vout is also sent to Host model for signal logging.

Signal Logging on Host Model

1 Click **Run**.



2 Observe the signal logging in the Host model to read the Vout as transmitted from the target model.


Scope
File Tools View Simulation Help
scaling
Signal Statistics
Value Time
Max 3.304e+00 2.563
Min 3.296e+00 2.564
Peak to Peak 8.107e-03
Mean 3.300e+00
Median 3.300e+00
RMS 3.300e+00
3.305
3.295
3.29
2.5635 2.564 2.5645 2.565 2.5655 2.566 2.5665 2.567 2.5675
Ready
Frame based Offset=20 T=25.860

Enable Active Load

- Onboard joystick can be used to enable the active loads and monitor the impact on the voltage regulation.
- Press left part of the joystick for 50% load and right part of the joystick for 100% load as shown.