GTM_TOM_3_Phase_Inverter_PWM_1 for KIT_AURIX_TC275_LK

GTM TOM 3 Phase Inverter using PWM

AURIX™ TC2xx Microcontroller Training
V1.0.0

Scope of work

The GTM TOM is configured to generate PWM signals for two-level three phase inverter.

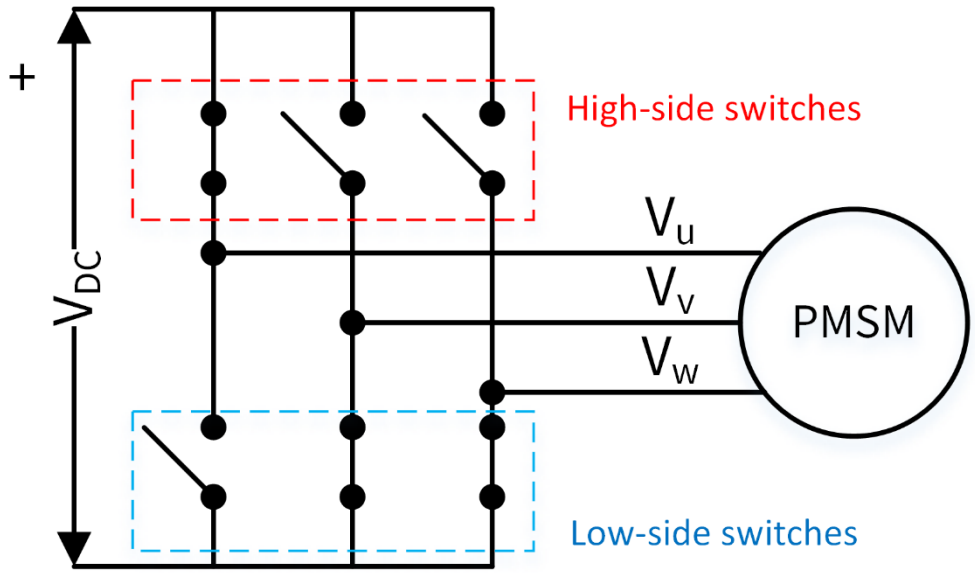
The states of 6 pins are controlled by the PWM signals generated by the Generic Timer Module (GTM) in-built Timer Output Module (TOM). All signals are synchronous to each other, center-aligned and with dead-times (positive/negative) for the complementary pairs.

Introduction

- The Generic Timer Module (GTM) is a modular timer unit designed to accommodate many timer applications
- It has an in-built Timer Output Module (TOM) that can offer up to 16 independent channels to generate output signals
- The Clock Management Unit (CMU) is responsible for clock generation of the GTM. The Fixed Clock Generation (FXU) is one of its subunits and it provides five predefined non-configurable clocks for GTM modules, including the TOM

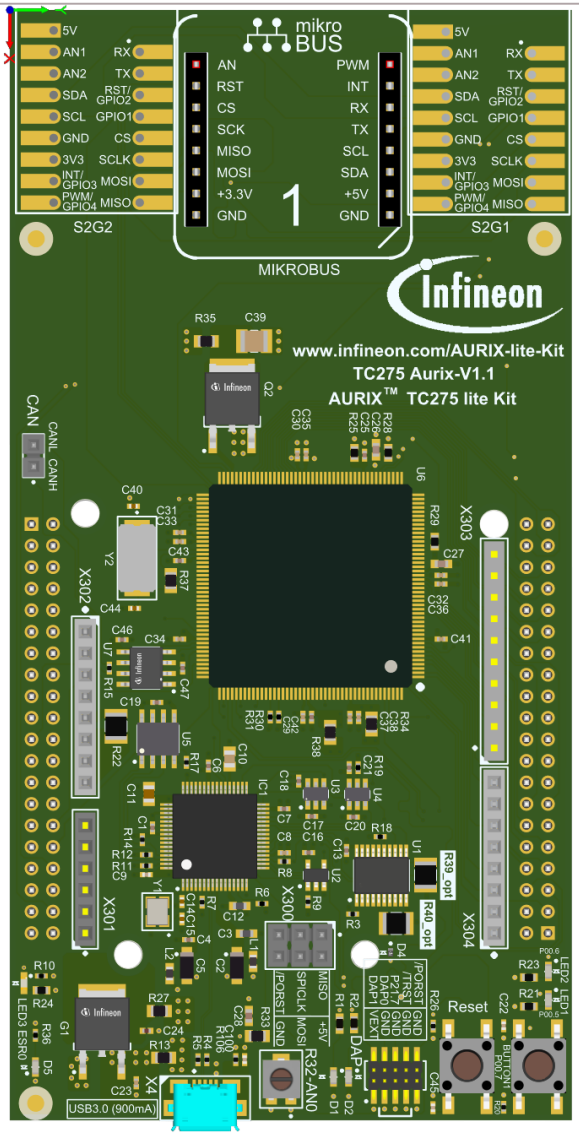
Introduction

- This example shows how to generate PWM signals to control a two-level three-phase inverter
- A simplified schema of the two-level three-phase inverter is shown in the image below



Hardware setup

This code example has been developed for the board KIT_AURIX_TC275_LITE.



Implementation

GTM configuration

- The ***IfxGtm_Tom_PwmHl.h*** iLLDs provide the GTM PWM driver to configure the required peripheral resources and drive them to produce the PWM waveform. PWM drivers are initialized and driven by the TriCore™ core

- The configuration of the TOM is done once in the setup phase by calling the initialization function ***initGtmTomPwm()***, which contains the following steps:

  - Configuration of the GTM frequencies
  - Configuration of the PWM master channel
  - Configuration of the PWM channels used to produce 3 complementary pair signals
  - Initialization and run of the PWM signals

Implementation

Configuration of the GTM frequencies

- First of all, the GTM module is enabled with the function ***IfxGtm_enable()***
- The GTM global clock frequency is then set with the function ***IfxGtm_Cmu_setGclkFrequency()***
- The GTM configurable clock frequency is set with the function ***IfxGtm_Cmu_setClkFrequency()***
- Finally, the FXU clocks are enabled by calling the function ***IfxGtm_Cmu_enableClocks()***

Implementation

Configuration of the PWM master channel

- To configure the PWM master channel the function ***IfxGtm_Tom_Timer_initConfig()*** initializes an instance of the structure ***IfxGtm_Tom_Timer_Config*** with its default values

- The elements of the ***IfxGtm_Tom_Timer_Config*** structure allows to set specific parameters:
  - ***base.frequency*** – Set the timer frequency
  - ***clock*** – Select the CMU-FXCLK0
  - ***tom*** – Select the timer to be used
  - ***timerChannel*** – Select the channel to be used

- Finally, the function ***IfxGtm_Tom_Timer_init()*** initializes the TOM with the user configuration

Implementation

Configuration of the PWM channels

- To configure the PWM channels to produce three complementary pair signals, an instance of the structure ***IfxGtm_Tom_PwmHl_Config*** is created and initialized with its default values by the function ***IfxGtm_Tom_PwmHl_initConfig()***

- The elements of the ***IfxGtm_Tom_PwmHl_Config*** structure allows to set specific parameters:
  - ***base.channelCount*** – Set the number of PWM channels
  - ***base.deadtime*** – Set the dead time between the top and bottom channels
  - ***base.minPulse*** – Set the minimum pulse time allowed as active state
  - ***base.outputMode*** – Set the port pin mode of the channels
  - ***base.outputDriver*** – Set the port pin strength and slew rate of the channels
  - ***base.ccxActiveState*** – Set the High-side PWM signals active state
  - ***base.coutxActiveState*** – Set the Low-side PWM signals active state
  - ***ccx*** – Select the channels used for the High-side PWM signals
  - ***coutx*** – Select the channels used for the Low-side PWM signals
  - ***timer*** – Select the timer to be used
  - ***tom*** – Select the module to be used

- The function ***IfxGtm_Tom_Timer_init()*** initializes the TOM with the user configuration

Implementation

Configuration of the PWM channels (cont.)

- The PWM mode is then configured to be center aligned with the function ***IfxGtm_Tom_PwmHl_setMode()***
- Finally, the input frequency of the TOM is updated by calling ***IfxGtm_Tom_Timer_updateInputFrequency()***

Initialization and run of the PWM signals

- The timer starts running after calling the function ***IfxGtm_Tom_Timer_run()***
- The initial values of the PWM signals are calculated and set calling:
  - ***IfxGtm_Tom_Timer_disableUpdate()*** – to stop the update of the TOM (In order to update all signals at the same time)
  - ***IfxGtm_Tom_PwmHl_setOnTime()*** – to set the calculated duty cycle
  - ***IfxGtm_Tom_Timer_applyUpdate()*** – to apply the changes by re-starting the update of the TOM channels

Implementation

The following PWM parameters are enabled/configured with this example:

PWM Type	Center Aligned
Frequency	20 kHz
Polarity	Duty-On High
Complementary Output	Enabled (opposite polarity)
Dead times	0,5 µS
Minimum pulse time	1 µS

› Initial values of PWM duty cycles:

PHASE_U	PHASE_V	PHASE_w
25%	50%	75%

Implementation

The below table provides the mapping between the PWM signal and the Port Pins:

PWM Signal	Pin Mapping
PHASE_U_HS	P00.11
PHASE_U_LS	P00.10
PHASE_V_HS	P33.0
PHASE_V_LS	P00.12
PHASE_W_HS	P33.2
PHASE_W_LS	P23.0

Implementation

GTM update

- Once the GTM is configured and started, a duty cycle update is performed every 10ms in the ***updateGtmTomPwmDutyCycles()*** function:
  - Each channel x is cyclically modified incrementing its duty cycle by 10%, from 10% to 90% using the variable ***g_pwm3PhaseOutput[x]***
  - The duty cycle of all channels is then updated using the iLLD functions:
    - ***IfxGtm_Tom_Timer_disableUpdate()***
    - ***IfxGtm_Tom_PwmHl_setOnTime()***
    - ***IfxGtm_Tom_Timer_applyUpdate()***

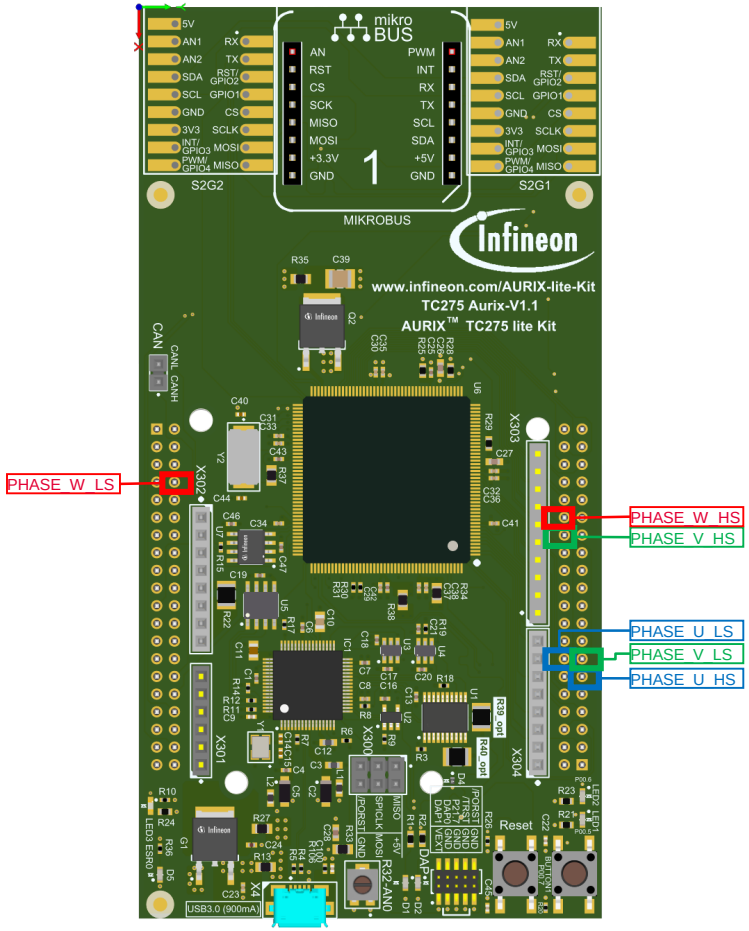
All the functions used for the configuration of the TOM are provided by the iLLD header ***IfxGtm_Tom_PwmHl.h***.

Run and Test

After code compilation and flashing the device, the PWM signals can be observed using a logic analyser or an oscilloscope connected to the following pins:

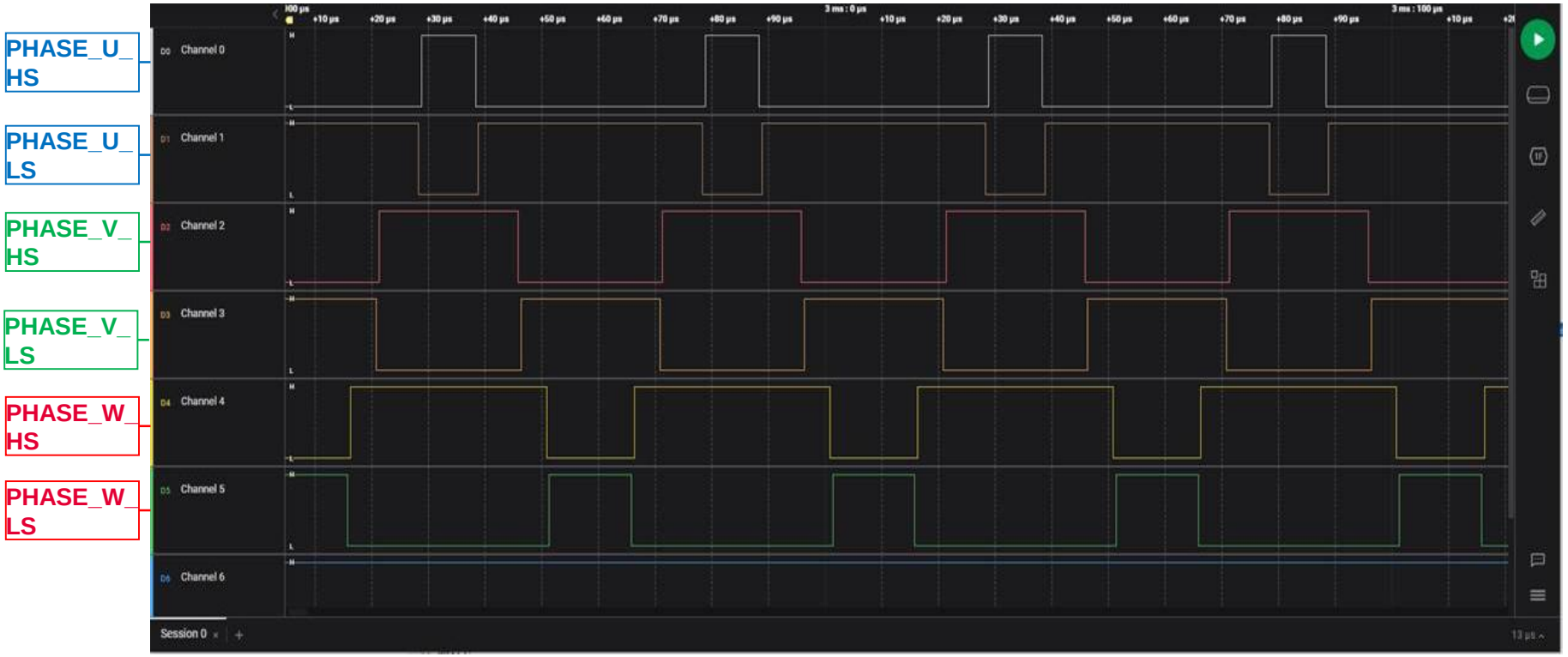
	X1			
	GND	1	2	VEXT	
	P33.11	3	4	P33.12	
	P33.13	5	6	P32.0	
	P23.1	7	8	P23.0	PHASE_W_LS
	P23.3	9	10	P23.2	
	P23.5	11	12	P23.4	
MRST -	P22.1	13	14	P22.0 - MTSR	
	P21.0	15	16	P22.2 - SS	
MDC -	P21.2	17	18	P22.3 - SCLK	
	P21.4	19	20	P21.3 - MDIO	
	P20.10	21	22	P21.5	
TXD2_S2G2 -	P20.0	23	24	P20.1	
RXD2_S2G2 -	P20.3	25	26	ESR1	
	ESR0	27	28	P20.14	
SDA0 -	P15.5	29	30	/PORST - Reset	
SCL0 -	P15.4	31	32	P11.12	
CRSDV -	P11.11	33	34	P11.10 - CS_S2G2	
RXD1 -	P11.9	35	36	P11.6	
TXD0 -	P11.3	37	38	P11.2 - CS_S2G1	
	VDD_USB	39	40	GND	

	X2			
	GND	40	39	VEXT	
	P33.10	38	37	P33.9 - TXD1_S2G1	
RXD1_S2G1 -	P33.8	36	35	P33.7	
	P33.6	34	33	P33.5	
	P33.4	32	31	P33.3	
PHASE_W_HS	P33.2	30	29	P33.1	
PHASE_V_HS	P33.0	28	27	AN0 - Potentiometer	
	AN1	26	25	AN2	
	AN3	24	23	AN4	
	AN5	22	21	AN6	
	AN7	20	19	AN44	
	AN45	18	17	AN46	
	AN47	16	15	VAREF1	
PHASE_U_HS	P00.11	14	13	P00.12	PHASE_V_LS
	P00.9	12	11	P00.10	PHASE_U_LS
Button1 -	P00.7	10	9	P00.8	
LED1 -	P00.5	8	7	P00.6 - LED2	
	P00.3	6	5	P00.2	
	P00.1	4	3	P00.0 - TXDCAN	
	VDD_USB	2	1	GND	



Run and Test

The following image shows the generated PWM signals:



References

- AURIX™ Development Studio is available online:
- https://www.infineon.com/aurixdevelopmentstudio
- Use the *„Import...“* function to get access to more code examples.

- More code examples can be found on the GIT repository:
- https://github.com/Infineon/AURIX_code_examples

- For additional trainings, visit our webpage:
- https://www.infineon.com/aurix-expert-training

- For questions and support, use the AURIX™ Forum:
- https://www.infineonforums.com/forums/13-Aurix-Forum

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2023-03
Published by
Infineon Technologies AG
81726 Munich, Germany

Do you have a question about this document?
Email: erratum@infineon.com

Document reference
GTM_TOM_PWM_1_KIT_TC275_LK

IMPORTANT NOTICE
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie") .

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (www.infineon.com).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.