PWM_Buzzer_1
for KIT_AURIX_TC297_TFT

PWM controlled melody on Buzzer

AURIX™ TC2xx Microcontroller Training
V1.0.0

Scope of work

The GTM unit is used to play a PWM-controlled song with a piezo buzzer.

The generated PWM (Pulse-Width Modulation) signal is controlled by timer TOM1 of the GTM block. The signal drives the piezo buzzer via the pin 0 of the port 33.

Introduction

› The Generic Timer Module (GTM) is a modular timer unit designed to accommodate many timer applications.

› The Timer Output Module (TOM), which is part of the GTM, offers sixteen independent channels to generate output signals.

› The Clock Management Unit (CMU) is responsible for clock generation of the GTM. The Fixed Clock Generation (FXU), subunit of the CMU, provides five predefined non-configurable clocks for the TOM submodules.

Hardware setup

This code example has been developed for the board KIT_AURIX_TC297_TFT_BC-Step.

The piezo buzzer (1) is used for this example.
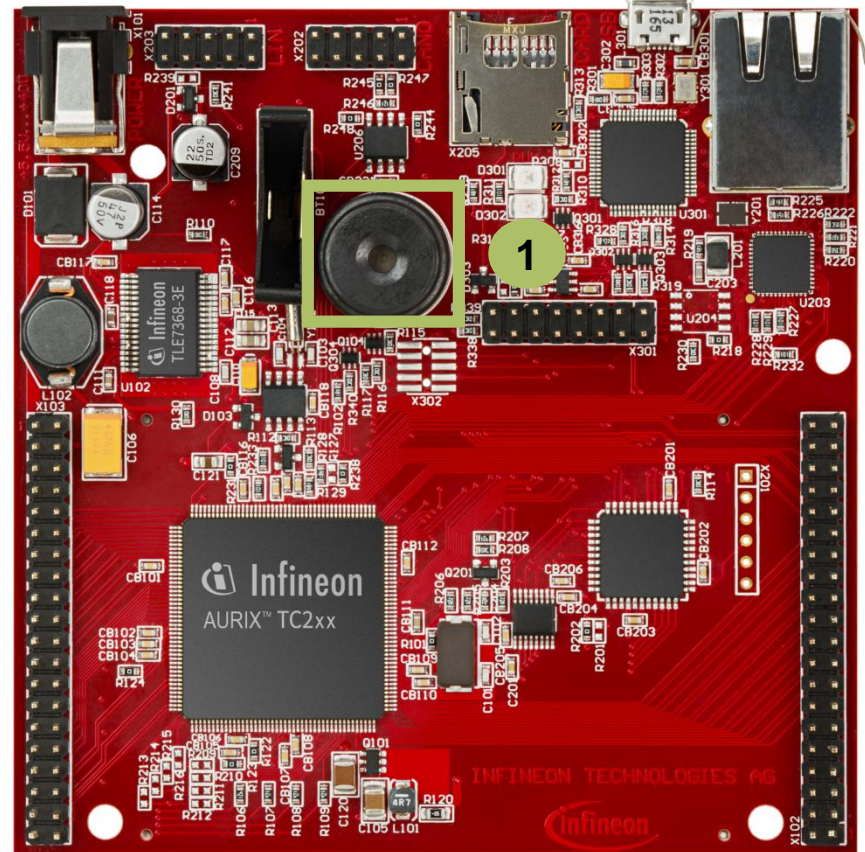
Implementation

Initialization of PWM Buzzer

The initialization of the PWM Buzzer is done once in the setup phase by calling the function *initPWMBuzzer()*, which contains the following steps:

› Initialize the time constants by calling the service function *initTime()* from iLLD header file *Bsp.h*
› Enable the GTM by calling the function *IfxGtm_enable()*
› Enable the FXU clocks by calling the function *IfxGtm_Cmu_enableClocks()*
› Initialize the timer configuration with default values by calling the function *IfxGtm_Tom_Timer_initConfig()*
› Set the timer configuration variable *timerCfg* with the user defined values
› Initialize the timer through the iLLD function *IfxGtm_Tom_Timer_init()*
› Enable the GTM TOM channel 4 by calling the iLLD function *IfxGtm_Tom_Tgc_enableChannelsUpdate()*

All the functions used for the configuration of the PWM Buzzer are provided by the iLLD header *IfxGtm_Tom_Pwm.h*.

Implementation

Configuration of the variable timerCfg

Setting of the *timerCfg* variable is done once in the setup phase in the function *initPWMBuzzer()* with the following steps:

› Set TOM1 as the timer for generating the PWM signal
   timerCfg.tom = IfxGtm_Tom_1
› Set the channel 4 of TOM1 as the channel used for the timer
   timerCfg.timerChannel = IfxGtm_Tom_Ch_4
› Select the clock source
   timerCfg.clock = IfxGtm_Tom_Ch_ClkSrc_cmuFxclk2
› Set port and pin for PWM output
   timerCfg.triggerOut = &IfxGtm_TOM1_4_TOUT22_P33_0_OUT
› Enable TOM channel
   timerCfg.base.trigger.enabled = TRUE
› Enable TOM channel output
   timerCfg.base.trigger.outputEnabled = TRUE
› Starting PWM with rising edge
   timerCfg.base.trigger.risingEdgeAtPeriod = TRUE

Implementation

Playing song

The song is played by repeatedly calling the function *playTone()* for each note, which contains the following steps:

1. Set the PWM frequency (in Hz) through the iLLD function *IfxGtm_Tom_Timer_setFrequency()*
2. Set the volume level by calling the function *setVolume()*. This is done by setting the PWM duty cycle with the function *IfxGtm_Tom_Timer_setTrigger()*
3. Start the timer with the iLLD function *IfxGtm_Tom_Timer_run()*
4. Wait until the end of the note through the service function *waitTime()*
5. Stop the timer with the iLLD function *IfxGtm_Tom_Timer_stop()*

PWM signal
Period of PWM
Duty cycle of the PWM
Sound duration
t
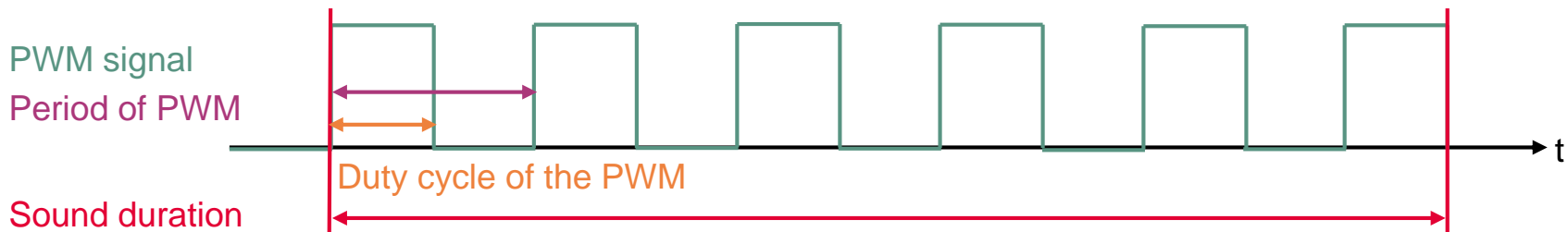
Run and Test

After code compilation and flashing the device, listen to the Buzzer (1) which should play the sound from the game Super Mario.
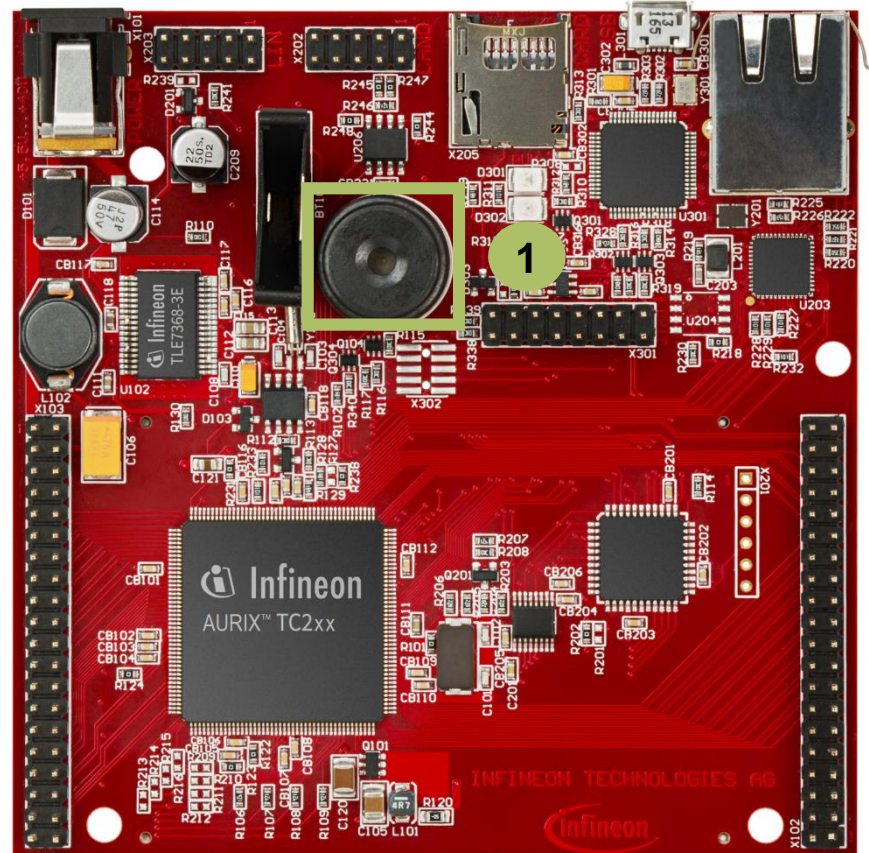
References

› 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

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.